User's Manual

32

SH7724

User's Manual: Hardware

Renesas 32-Bit RISC Microcomputer
SH7780 Series

R8A7724

[Portions omitted in accordance with NDA]

The contents of the Video Processing Unit (VPU), 2D Graphic Accelerator (2DG), and Sound Process Unit 2 (SPU2) sections of this document are covered by a non-disclosure agreement. Please contact a Renesas Electronics sales representative for details.

Renesas Electronics
www.renesas.com

Rev.1.00 May 2010

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

General Precautions on Handling of Product

1. Treatment of NC Pins

Note: Do not connect anything to the NC pins.
The NC (not connected) pins are either not connected to any of the internal circuitry or are they are used as test pins or to reduce noise. If something is connected to the NC pins, the operation of the LSI is not guaranteed.

2. Treatment of Unused Input Pins

Note: Fix all unused input pins to high or low level.
Generally, the input pins of CMOS products are high-impedance input pins. If unused pins are in their open states, intermediate levels are induced by noise in the vicinity, a pass-through current flows internally, and a malfunction may occur.

3. Processing before Initialization

Note: When power is first supplied, the product's state is undefined.
The states of internal circuits are undefined until full power is supplied throughout the chip and a low level is input on the reset pin. During the period where the states are undefined, the register settings and the output state of each pin are also undefined. Design your system so that it does not malfunction because of processing while it is in this undefined state. For those products which have a reset function, reset the LSI immediately after the power supply has been turned on.

4. Prohibition of Access to Undefined or Reserved Addresses

Note: Access to undefined or reserved addresses is prohibited.
The undefined or reserved addresses may be used to expand functions, or test registers may have been be allocated to these addresses. Do not access these registers; the system's operation is not guaranteed if they are accessed.

5. Reading from/Writing to Reserved Bit of Each Register

Note: Treat the reserved bit of register used in each module as follows except in cases where the specifications for values which are read from or written to the bit are provided in the description.
The bit is always read as 0. The write value should be 0 or one, which has been read immediately before writing.
Writing the value, which has been read immediately before writing has the advantage of preventing the bit from being affected on its extended function when the function is assigned.

Configuration of This Manual

This manual comprises the following items:

1. General Precautions on Handling of Product
2. Configuration of This Manual
3. Preface
4. Contents
5. Overview
6. Description of Functional Modules
    - CPU and System-Control Modules
    - On-Chip Peripheral Modules

    The configuration of the functional description of each module differs according to the module. However, the generic style includes the following items:

    i) Feature
    ii) Input/Output Pin
    iii) Register Description
    iv) Operation
    v) Usage Note

When designing an application system that includes this LSI, take notes into account. Each section includes notes in relation to the descriptions given, and usage notes are given, as required, as the final part of each section.

7. Electrical Characteristics
8. Appendix
9. Index

Preface

This LSI is a RISC (Reduced Instruction Set Computer) microcomputer which includes a Renesas original RISC CPU as its core, and the peripheral functions required to configure a system.

Target Users: This manual was written for users who will be using this LSI in the design of application systems. Users of this manual are expected to understand the fundamentals of electrical circuits, logical circuits, and microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of this LSI to the above users.

Notes on reading this manual:

- In order to understand the overall functions of the chip
  Read the manual according to the contents. This manual can be roughly categorized into parts on the CPU, system control functions, peripheral functions and electrical characteristics.
- In order to understand the details on FPU functions and each instructions
  Read the additional volume, SH-4A Extended Functions Software Manual.

Rules:	Bit order:	The MSB is on the left and the LSB is on the right.
	Number notation:	Binary is B'xxxx, hexadecimal is H'xxxx, decimal is xxxx.
	Signal notation:	An overbar is added to a low-active signal: $\overline{\text{xxxx}}$

Abbreviations

ATAPI	ATAPI Controller
CPG	Clock Pulse Generator
DMAC	Direct Memory Access Controller
E-DMAC	Ethernet Controller Direct Memory Access Controller
EtherC	Ethernet Controller
FLCTL	NAND Flash Memory Controller
G2D	2D Graphics Engine
GPIO	General Purpose I/O
H-UDI	User Debugging Interface
IIC	I2C Bus Interface
INTC	Interrupt Controller
MCU	Memory Controller Unit
MMU	Memory Management Unit
SCIF	Serial Communication Interface with FIFO
TMU	Timer Unit
UBC	User Break Controller
USB	USB Host/Function Interface
VDC2	Video Display Controller 2
WDT	Watchdog Timer and Reset
SSI	Serial Sound Interface
LCDC	LCD Controller
SRC	Sampling Rate Converter

bps	bits per second
CRC	Cyclic Redundancy Check
DMA	Direct Memory Access
DMAC	Direct Memory Access Controller
Hi-Z	High Impedance
I/O	Input/Output
LSB	Least Significant Bit
MSB	Most Significant Bit
NC	Non-Connection
PLL	Phase Locked Loop

Contents

Section 1 Overview

1.1 SH7724 Features

The SH7724 (SH-MobileR2R) is a system LSI that incorporates a Renesas original SH-4A processor core along with peripheral functions required for multimedia applications such as mobile devices, car navigation systems, and digital consumer products. The features of this LSI are listed in table 1.1.

Table 1.1 SH7724 Features

Item	Feature
Maximum operating frequency	• CPU core: 500 MHz • Internal bus: 166.7 MHz • Peripheral bus: 41.7 MHz
CPU performance	• 900 MIPS, 3.5 GFLOPS (at 500 MHz)
Cache	• Primary cache: 32 Kbytes for instructions and 32 Kbytes for data • Secondary cache: 256 Kbytes (mixed instructions and data)
Memory interface	• Common interface for DDR2 and Mobile-DDR: DDR333, 32-bit bus width, up to 512 Mbytes connectable • SRAM, NOR flash, and PCMCIA interface: 32-bit bus width, up to 83.3 MHz
On-chip peripheral functions	• On-chip multi-codec (the VPU5F) handles various video formats (H.264, MPEG4, VC-1), for recording and playback of video at high-definition (1280 x 720 pixel) resolution and 30 fps. • 2D graphics accelerator achieves fast drawing of high-quality maps. • Includes an LCD controller for 24-bit color TFT LCD panels and on-chip digital video (ITU-R BT.601/656) output functionality; handles simultaneous output for two displays. • Dedicated DSP for 24-bit audio achieves low power consumption for all kinds of audio processing. • Two USB interfaces (host and function) for USB 2.0 high-speed mode • MMC 4.2 interface allows connection with and booting up from NAND flash memory. • On-chip MAC for Ethernet (10 or 100 Mbps)
Power-down function	• Separate power-supply areas and support of various standby modes

Item	Feature
Package	• 449-pin BGA (0.8-mm-pitch, 21 mm × 21 mm) • 441-pin POP (0.5-mm-pitch, 14 mm × 14 mm) Note: Refer to the separate documentation for package information on the POP.
Power-supply voltage	• Core: 1.15 V to 1.30 V (with the VPU running at less than 125 MHz), 1.25 V to 1.35 V (with the VPU running at 125 to 166.7 MHz) • I/O: 1.65 V to 1.95 V or 2.7 V to 3.6 V selectable
Operating temperature	• -40°C to 85°C (449-pin BGA product) • -20°C to 70°C (441-pin POP product)
Fabrication process	• 65-nm CMOS process

1.2 Block Diagram

Figure 1.1 shows a block diagram of the SH7724.

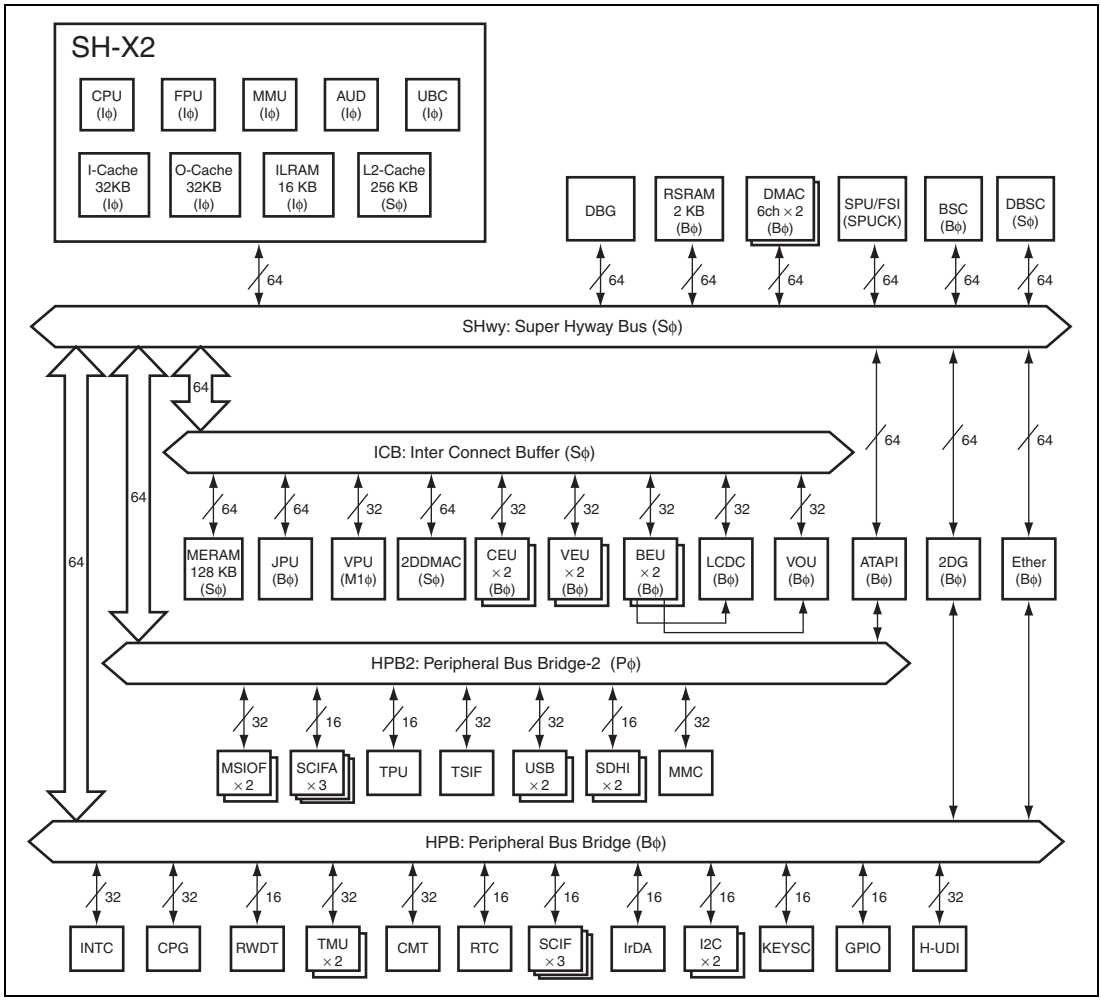


Figure 1.1 SH7724 Block Diagram

1.3 Overview of Module Specifications

Table 1.2 gives an overview of the specifications of each module. For detailed descriptions, refer to the corresponding section.

Table 1.2 Overview of Module Specifications

Item	Description
CPU	• Renesas original architecture (SH-4A) • 32-bit internal data bus • General register file • RISC-type instruction set (upward compatible with the SH-1, SH-2, SH-3, and SH-4) • Superscalar architecture (providing simultaneous execution of two instructions) including FPU • Instruction execution time: Up to two instructions/cycle • Virtual address space: 4 Gbytes • Space identifier ASIDs: 8 bits, 256 virtual address spaces • Internal multiplier • Eight-stage pipeline
Floating-point unit (FPU)	• On-chip floating-point coprocessor • Supports single precision (32 bits) and double precision (64 bits) • Supports IEEE754-compliant data types and exceptions • Rounding modes: Round to Nearest and Round to Zero • Handling of denormalized numbers: Truncation to zero or interrupt generation for compliance with IEEE754 • Floating-point registers: 32 bits × 16 words × 2 banks (single-precision × 16 registers or double-precision × 8 registers) × 2 banks • 32-bit CPU-FPU floating-point communication register (FPUL) • Supports FMAC (multiply-and-accumulate) instruction • Supports FDIV (divide) and FSQRT (square root) instructions • Supports FLDI0/FLDI1 (load constant 0/1) instructions • 3D graphics instructions (single-precision only) • 10-stage pipeline

Item	Description
Memory management unit (MMU)	• 4-Gbyte address space, 256 address areas (8-bit ASIDs) • Single-virtual-memory mode and multiple-virtual-memory mode • Supports multiple page sizes: 1 Kbyte, 4 Kbytes, 8 Kbytes, 64 Kbytes, 256 Kbytes, 1 Mbyte, 4 Mbytes, and 64 Mbytes • 4-entry fully-associative TLB for instructions • 64-entry fully-associative TLB for instructions and operands • Supports software-controlled replacement and random-counter replacement algorithm • TLB contents can be accessed directly by address mapping • Supports 32-bit physical address expansion mode Note: The 29-bit physical address mode is selected in the initial state.
Cache memory	• Instruction cache (IC) — 32-Kbyte, 4-way set associative — 32-byte block length • Operand cache (OC) — 32-Kbyte, 4-way set associative — 32-byte block length — Selectable write mode (copy-back or write-through) • Store queue (32 bytes × 2 entries)
Secondary cache (L2C)	• Mixed 256-Kbyte cache for instructions and data • 32-byte block length • Write-through mode
IL memory (ILRAM)	• Three independent read/write ports — Instruction fetch access by the CPU — 8-/16-/32-bit operand access by the CPU — 8-/16-/32-/64-bit and 16-/32-byte access by the SuperHyway bus master • 16K-byte capacity
SuperHyway bus (SHwy)	• High-performance on-chip 64-bit-width system bus accessed with 32-bit addresses • Packet router (GPR) controls transfer between the initiator and target • Initiator modules: SH-X2, DBG, DMAC, ICB, 2DG, ATAPI, EtherMAC, and SPU • Dynamically controls the priority among initiators Normal transfer: Complete LRU control Urgent transfer: Desired priority level can be set for the target initiator

Item	Description
RS memory (RSRAM)	• On-chip RAM accessible from the CPU and SuperHyway bus master • Data retention in R-standby mode • 2-Kbyte capacity • Can be used to store the program for returning from R-standby mode
Interrupt controller (INTC)	• Nine independent external interrupts (NMI and IRQ7 to IRQ0) — NMI: Falling or rising edge selectable — IRQ: Falling or rising edge or high or low level selectable • On-chip module interrupts: Priority can be set for each module The following modules can issue on-chip module interrupts DMAC, ATAPI, TPU, TMU, CMT, MSIOF, SCIF, SCIFA, RTC, IrDA, KEYSC, USB, IIC, MMCIF, VPU, VIO5 (CEU, VEU, BEU), 2DG, LCDC, VOU, JPU, ICB, 2DDMAC, TSIF, FSI, SPU, EtherMAC, and SDHI
Bus state controller (BSC)	• Provides SRAM, burst ROM, and PCMCIA interface functions • Supports external address space up to 256 Mbytes in total. The space can be divided into areas in either of the following two ways. — Address map 1: Six areas (areas 0, 4, 5A, 5B, 6A, and 6B) — Address map 2: Four areas (areas 0, 4, 5, and 6) • The following parameters can be specified for each area — Memory type: SRAM, NOR-flash memory, burst ROM, PCMCIA — Data bus width: 8, 16, or 32 bits selectable (only 16 or 32 bits for area 0) — Number of wait cycles
DDR SDRAM bus state controller (DBSC)	• DDR2-SDRAM or Mobile DDR-SDRAM can be directly connected • Up to 512-Mbyte physical address space • 32- or 16-bit data bus width • Supports auto-refresh and self-refresh modes • Number of banks: 4 or 8 banks (DDR2) or 4 banks (Mobile-DDR) Note that up to four banks can be opened together in 8-bank mode. • Burst length: Fixed at 4 • Burst type: Sequential • CAS latency: Fixed at 3 • Power-down mode • Deep power-down mode (only for Mobile-DDR) • Partial self-refresh mode (only for Mobile-DDR) • Auto-precharge mode or bank active mode

Item	Description
Direct memory access controller (DMAC)	• Twelve channels; two channels accept external requests • Address space: 4 Gbytes on architecture • Data transfer length: Bytes, words (2 bytes), longwords (4 bytes), 16 bytes, and 32 bytes • Maximum number of transfer times: 16,777,216 times • Addressing mode: Dual addressing mode Transfer request selectable from three types: External request, on-chip peripheral module request, and auto request • On-chip peripheral module requests can be issued from the following: SCIF, SCIFA, MSIOF, SDHI, TSIF, IrDA, USB, and MMCIF • Bus mode: Cycle steal mode (normal mode and intermittent mode) or burst mode is selectable • Priority: Fixed channel priority mode or round-robin mode is selectable • Interrupt request: Supports the interrupt request to CPU at the end of data transfer • Repeat function: Automatically resets the transfer source, destination, and count at the end of DMA transfer • Reload function: Automatically resets the transfer source and destination at the end of the specified number of DMA transfers

Item	Description
Clock pulse generator (CPG)	• Three types of clock source selectable — EXTAL pin input: 10 MHz to 66.7 MHz — RTC_CLK pin input: 32.768 KHz. RTC operating clock. This clock can be multiplied up to the order of MHz through the FLL circuit as the `system clock source. — Crystal resonator: Connect to the EXTAL and XTAL pins • Generates four types of system clocks — CPU clock (Iϕ): 500 MHz max. — SuperHyway/DDR clock (Sϕ): 166.7 MHz max. (DDR333 is supported) — Bus clock (Bϕ): 83.4 MHz max. — Peripheral clock (Pϕ): 41.7 MHz max. • Generates specialized clocks for peripheral modules — M1ϕ: Operating clock for the VPU. 166.7 MHz max. — RCLK: RWDT and CMT operating clock — FSICK: FSI interface clock • Dynamic control of the system clock frequency through modification of PLL multiplication or division ratio by software • Supports power-down modes — Module standby mode (stops the clock in module units) — Sleep mode (stops the clock for the CPU core) — Software standby mode (stops the clock inside the LSI except the I/O block and RCLK operation area) — R-standby mode (stops the power supply inside the LSI except RCLK operation area, RS memory, and some registers) — U-standby mode (stops the power supply inside the LSI except the I/O block and RCLK operation area)
R watchdog timer (RWDT)	• One-channel watchdog timer operating on RCLK • Can operate in power-down modes • Generates a system reset at a counter overflow
Timer unit (TMU)	• On-chip 6-channel 32-bit timer • Auto-reload-type 32-bit down counter • Internal prescaler for Pϕ

Item	Description
Timer pulse unit (TPU)	• On-chip 4-channel 16-bit timer • Four pulse signals can be output • Up to 4-phase PWM output is available
Compare match timer (CMT)	• On-chip 1-channel 32-bit timer (16 bits or 32 bits selectable) • Source clock: RCLK • Compare match function
Realtime clock (RTC)	• On-chip clock and calendar functions operating on RTC_CLK • Generates alarm and peripheral interrupts
Serial interface with FIFO (MSIOF)	• Two channels • Separate internal 64-byte FIFOs for transmission and reception • Clocked synchronous serial master and slave modes. Full-duplex communication supported. — Supports 8-/16-bit data and 16-bit stereo audio input/output — Supports 24-bit stereo audio input/output — Sampling rate clock input selectable from Bϕ and external pin — Internal prescaler for Bϕ • SPI master and slave modes. Full-duplex communication supported. — Serial clock (SCK) falling or rising edge selectable for data sampling timing — SCK clock phase selectable for transmit timing. Three slave devices selectable. — Transmit/receive data length selectable from 8 bits, 16 bits, and 32 bits
Serial communication interface with FIFO (SCIF)	• Three channels (SCIF0 to SCIF2) • Separate internal 16-byte FIFOs (8 bits × 16 FIFOs) for transmission and reception • Supports asynchronous mode and clocked synchronous mode (master/slave) • High-speed UART for Bluetooth • Internal prescaler for Pϕ
Serial communication interface with FIFO (SCIFA)	• Three channels (SCIFA3 to SCIFA5) • Separate internal 64-byte FIFOs (8 bits × 64 FIFOs) for transmission and reception • Supports asynchronous mode and clocked synchronous mode (master/slave) • Modem control function (RTS, CTS) (only in SCIF3) • High-speed UART for Bluetooth • Internal prescaler for Bϕ
IrDA interface (IrDA)	• Conforms to version 1.2a • CRC generation function

Item	Description
Key scan interface (KEYSC)	• Key scan: Chattering elimination in key input interrupt detection • Number of input or output bits is selectable (5 inputs/6 outputs, 6 inputs/5 outputs, or 7 inputs/4 outputs) • Software standby, R-standby, and U-standby modes can be canceled by a key input
ATAPI interface (ATAPI)	• Supports primary channel • Master and slave functions • Supports PIO modes 0 to 4, multiword DMA modes 0 to 2, and Ultra DMA modes 0 to 4 • Supports descriptor mode • 3.3-V I/O interface
USB2.0 host and function module (USB)	• On-chip USB2.0 host controller and function controller • Two channels of USB host controller and function controller can be switched through register settings • Supports high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) transfer • Internal USB transceiver • Supports all USB transfer types • Control, bulk, interrupt (high bandwidth not supported), and isochronous (high bandwidth not supported) transfer • Up to ten pipes selectable (including the default control pipe) • Desired endpoint numbers can be assigned to pipes 1 to 9 • Transfer type selectable for each pipe • Pipe 0: Control transfer • Pipes 1 and 2: Bulk or isochronous transfer • Pipes 3 to 5: Bulk transfer • Pipes 6 to 9: Interrupt transfer • Input EXTALUSB clock: 48 MHz
I^2C bus interface (IIC)	• Two channels (IIC0 and IIC1) • Supports single master transmission and reception • Supports standard mode (100 kHz) and fast mode (400 kHz)

Item	Description
Video processing unit (VPU)	• The incorporated multi-codec (VPU5F) handles various moving-picture formats. • MPEG-4 single video object plane (VPO) encoding and decoding • Applicable standard MPEG-4 Simple Profile MPEG-4 H.264 (Baseline) *1 JPEG Baseline (VLC requires software processing) WMV Simple Profile MainProfile*2 Notes: 1. Some Baseline tools are not supported. 2. Some tools (Dynamic Resolution Change, B-Frame, and Range Reduction) are not supported • Image size: Sub-QCIF to XGA and HD (1280 × 720) supported • Bit rate: 8 Mbps max. • Motion detection: Layer tracking (Renesas original method) • Rate control: Control with quantizing amount predicted (Renesas original method), both VOP and MB supported
Video I/O module (VIO5)	Consists of the following three modules that provide the interface with the camera module and perform image processing • CEU (capturing engine: image capturing from camera module) — Two channels (supports two cameras) — Camera module interface YCbCr data (8 or 16 bits: YCbCr 4:2:2), horizontal sync signal (HD), vertical sync signal (VD), and binary data (such as RGB565) — Size of captured image: 5-M pixels, 3-M pixels, 2-M pixels, UXGA, SXGA, XGA, SVGA, VGA, CIF, QVGA, QCIF, QQVGA, Sub-QCIF — Output image format: YCbCr (4:2:2/4:2:0) — Image format conversion function Reduced image generating prefilter function YCbCr 4:2:2 → YCbCr 4:2:2, YCbCr 4:2:0 YCbCr format (Y: 8 bits; CbCr: 16 bits)

Item	Description
Video I/O module (VIO5)	• VEU3F (video engine: image processing in memory) — Two channels — Video image processing function Input image format: YCbCr image (Y/CbCr plane image), RGB image (packed RGB image) Output image format: YCbCr image (Y/CbCr plane image), RGB image (packed RGB image) — Image processing function Scaled image generating filter function YCbCr → RGB and RGB → YCbCr conversion function Dithering function (in RGB color subtraction) — Filter processing function Mirroring, vertical inversion, point symmetry, and ±90-degree image conversion functions Deblocking filter Median filter — Video image processing and filter processing combined operation • BEU2G (blend engine: image blending) — Two channels (supports simultaneous two-plane output) — PinP function Input image format: YCbCr image (Y/CbCr plane image), RGB image (packed RGB image) Output image format: YCbCr image (Y/CbCr plane image), RGB image (packed RGB image) — Graphic processing function Input graphic format: YCbCr/RGB image Output graphic format: YCbCr/RGB image — PinP and graphic combined operation Two PinP planes and one graphic plane can be blended simultaneously — Results of processing are written back to memory — Frame drop function (1/2, 1/3, 1/4, 1/5, or 1/6 drop)

Item	Description
2D graphics accelerator (2DG)	• Drawing functions Four-vertex drawing, polygon drawing, line drawing, high-functional bold line drawing, antialiasing, raster operation/BitBLT with alpha blending • Color representation — Source: 1, 8, or 16 bits/pixel — Drawing: 8 or 16 bits/pixel — Work: Binary • Screen coordinates — X direction: 0 to 4095 — Y direction: 0 to 4095
LCD controller (LCDC)	• LCD panel: TFT color LCD, up to XGA and HD (1280 × 720) supported • Input data format: 12, 16, 18, or 24 bpp • LCD driver interface — Specialized LCD bus, independent of memory bus — RGB interface or 80-series CPU bus interface selectable — Bus width: 8, 9, 12, 16, 18, and 24 bits supported — One-pixel one-time, two-time, and three-time transfer modes supported — Signal polarity and SYNC output position and width programmable in RGB interface — Access cycle programmable in 80-series CPU bus interface • Dot clock: Bus clock, peripheral clock, or external clock selectable as the source clock • Display data fetch: Continuous mode (according to the refresh rate of the LCD panel) and one-shot mode (according to the frame rate of the movie) are supported. Image data only for updated sections can be fetched selectively. • Internal 256-entry, 24-bit-input/output color palette • An interrupt can be generated in frame units or at the user-specified line.
Video output unit (VOU)	• Output format: Conforms to ITU-R BT.601 and ITU-R BT.656 • Output interface: Supports 16-bit Y/C separate interface and 8-bit Y/C composite interface • Pixel frequency: 13.5 MHz (for 16-bit interface) or 27 MHz (for 8-bit interface) • Partial image display: Any background color (specified through register) + display image • Supported image size: Sub-QCIF, QVGA, VGA, etc.

Item	Description
JPEG processing unit (JPU)	• Applicable standard: JPEG baseline • Operating precision: Conforms to JPEG Part 2, ISO-IEC10918-2 • Color format: YCbCr (4:2:2/4:2:0) • Quantization tables: Four tables provided • Huffman tables: Four tables provided (two AC tables and two DC tables) • Target markers: SOI, SOF0, SOS, DQT, DHT, DRI, RSTm, EOI
Media RAM (MERAM)	• 128 Kbytes • Line buffer control function of maximum 32 planes • Can be used as the read fill buffer or write back buffer of the corresponding IP • Can be used as the middle buffer between corresponding IPs • Part of the frame buffer data for the LCDC is cacheable
2D DMAC (2DDMAC)	• Two YCbCr planes and four RGB planes can be processed • Supports image clipping in 1-pixel units for RGB and 2-pixel units for YCbCr • Conversion between various RGB formats • Images can be scaled up twice respectively in the X direction and the Y direction • Vertical and horizontal inversion and 90° and 270° rotation
TS interface (TSIF)	• Serial TS data input • Filters 38 types of PIDs in total (Note that the PID values of PAT and CAT packets are fixed. For PCR, video, and audio packets, the PID values are predefined.)
Sound interface with FIFO (FSI)	• 24-bit stereo • Supports PCM and I2S formats • Two sound input systems and two sound output systems • Serial I/O can be directly connected to external A/D and D/A converters • Can be directly connected to the SPU
Sound processing unit (SPU)	• Two internal audio DSPs (24-bit dual MAC) • DMA function available • Internal 160-Kbyte RAM for programs and 264-Kbyte RAM for data

Item	Description
Ethernet controller (EtherMAC)	• Ethernet controller conforming to the Ethernet media access control (MAC) layer standard — Transmission and reception of Ethernet/IEEE802.3 frames — 10-Mbps and 100-Mbps transfer — Full-duplex and half-duplex modes — Reduced media independent interface (RMII) — Flow control conforming to the IEEE802.3x standard • Specialized DMA controller — Reduces the load on the CPU by means of a descriptor management system — Reflects transmit/receive frame status on the descriptor — Achieves efficient system bus utilization through block (32-byte) transfer — Supports single-frame multi-buffer transfer — Improves the software processing performance through padding in received data
MMC interface (MMCIF)	• Provides Multi Media Card (MMC) and CE-ATA device control functions • Data bus: 1-bit, 4-bit, and 8-bit MMC modes supported (SPI mode not supported) • Supports block transfer (stream transfer not supported) • Block size for multi-block transfer: 512 bytes • Supports the command completion signal (CE-ATA) • NAND boot function conforming to MMC4.2
SD card host interface (SDHI)	• Two channels • Supports SD memory and SDIO card interface • Card detecting function • Maximum operating frequency: 50 MHz (high speed supported)
I/O ports	• Input or output can be selected for each input/output port independently
User break controller (UBC)	• Provides user-break interrupts for debugging • Two break channels • Addresses, data values, access types, and data sizes are all specifiable as break conditions • Provides sequential break functions
User debugging interface (H-UDI)	• Supports E10A emulator • Realtime branch trace functions

1.4 Pin Assignment

Figure 1.2 shows the BGA449 pin arrangement, and figure 1.3 and table 1.3 show the pin assignment.

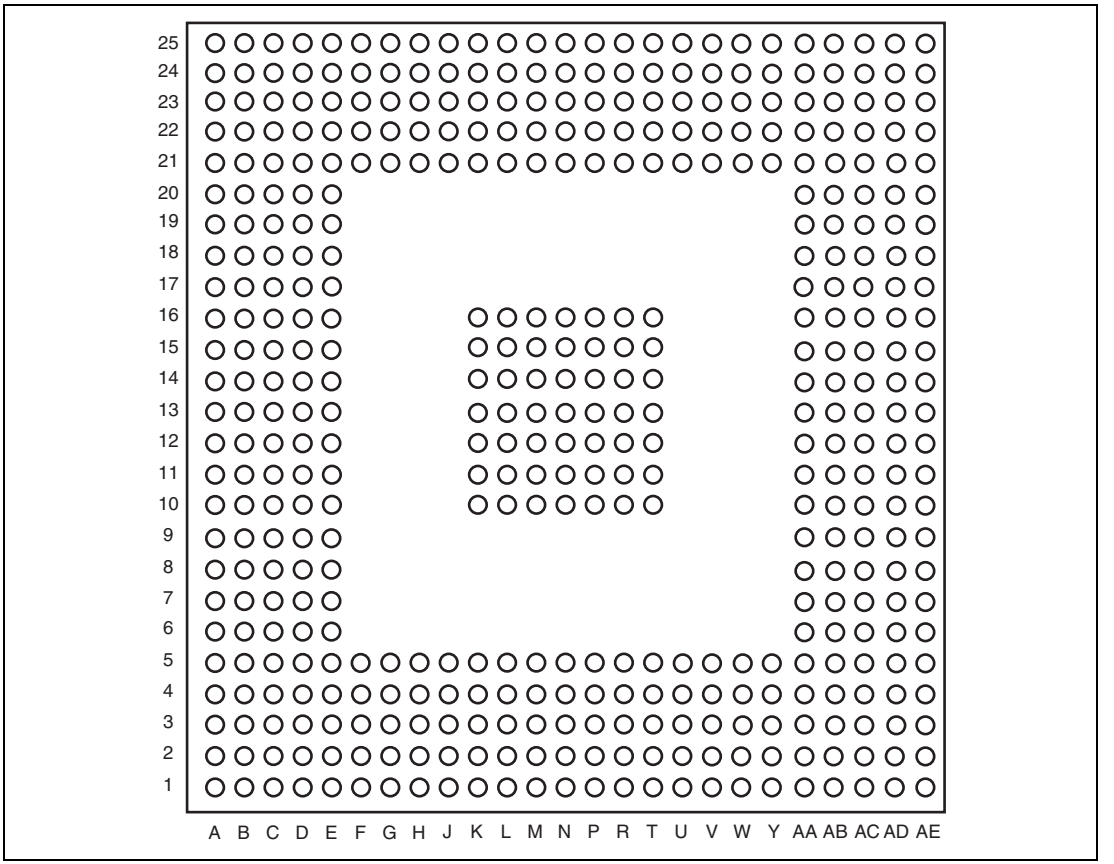


Figure 1.2 BGA449 Pin Arrangement (Top View)

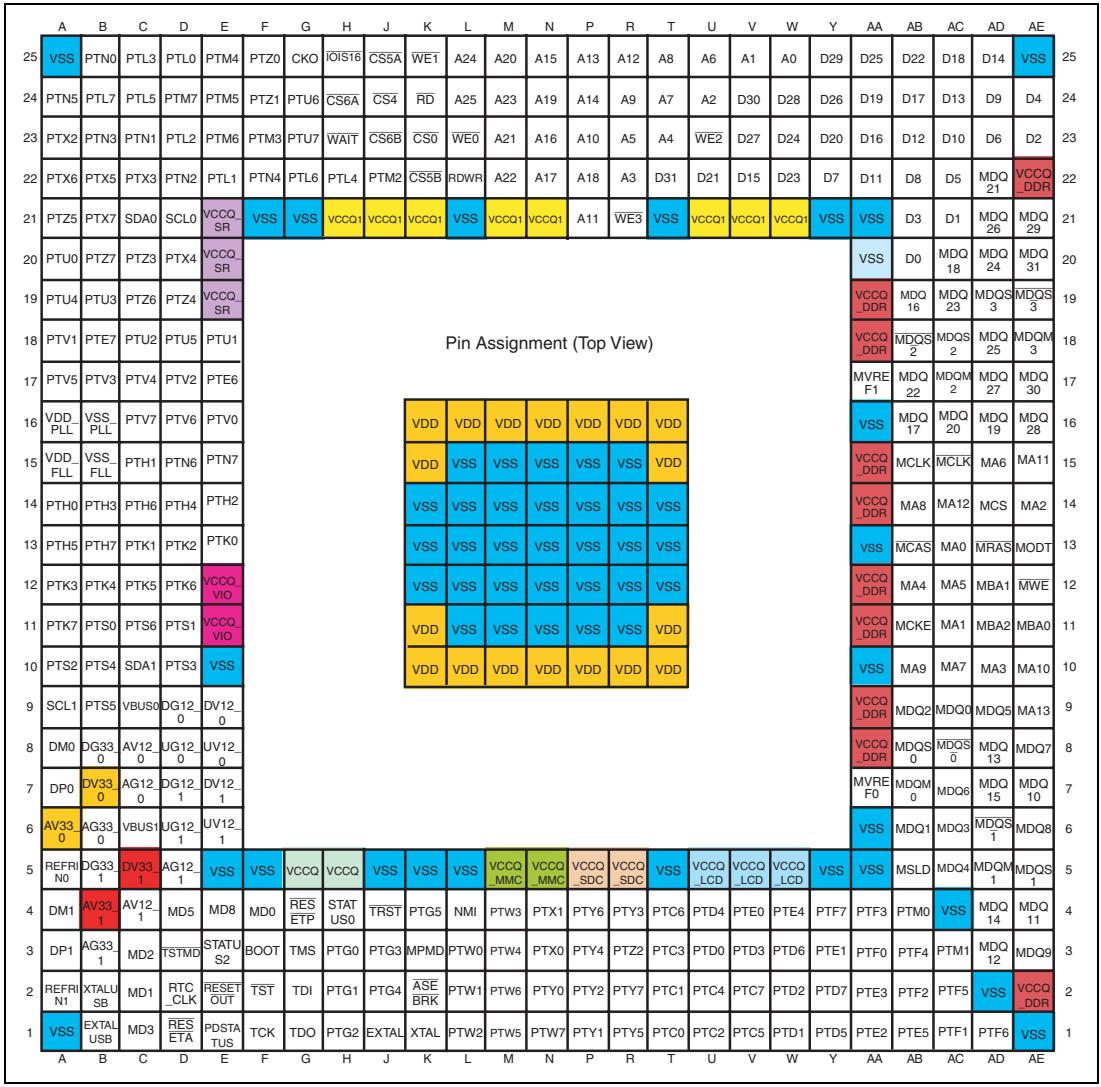


Figure 1.3 BGA449 Pin Assignment (Top View)

1.5 Pin Assignment (BGA449)

Table 1.3 gives the pin assignment of the BGA449 package, including multiplexed functions, and the specifications of the I/O buffers. For details on the pin functions, refer to section 1.6, Pin Functions or sections on the relevant modules. For the setting of multiplexed functions, refer to section 48, Pin Function Controller (PFC). For pin states in the respective operating modes, see appendix D, Pin States after Reset and in Power-down Modes.

- No.
  Pin Number
- Ball arrangement
  The arrangement of balls on the BGA449 is shown in figure 1.4.
- Pin Name
- Multiplexed function
  Multiplexed functions are listed for each pin.
- I/O
  IO: Input output (bi-directional)
- I/O buffer types
  Analog, Schmitt-trigger input, open-drain output, pull-up, and pull-down
- I/O buffer power supplies
  This product uses multiple I/O power supplies.
- Handling when unused
  Handling when pins are not in use is specified. If the entry in this column is "-", the corresponding pin must always be used.
  PFC (pin function controller) register settings that relate to unused pins are prohibited.

Table 1.3 Pin Assignment

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
1	A1	VSS	VSS	—	—	—	—
2	A2	REFRIN1	REFRIN1	I	analog	AV33_1	pull-down
3	A3	DP1	DP1	IO	analog	DV33_1	open
4	A4	DM1	DM1	IO	analog	DV33_1	open
5	A5	REFRIN0	REFRIN0	I	analog	AV33_0	pull-down
6	A6	AV33_0	AV33	—	—	—	—
7	A7	DP0	DP0	IO	analog	DV33_0	open
8	A8	DM0	DM0	IO	analog	DV33_0	open
9	A9	SCL1	SCL1	IO	open-drain	VCCQ_VIO	open
10	A10	PTS2	PTS2 / VIO1_CLK / SCIF5_TXD	IO / I / O	pull-up	VCCQ_VIO	open
11	A11	PTK7	PTK7 / VIO1_D5 / VIO0_D13 / IDED5	IO / I / I / IO	pull-up	VCCQ_VIO	open
12	A12	PTK3	PTK3 / VIO1_D1 / VIO0_D9 / IDED1	IO / I / I / IO	pull-up	VCCQ_VIO	open
13	A13	PTH5	PTH5 / VIO0_D7	IO / I	pull-up	VCCQ_VIO	open
14	A14	PTH0	PTH0 / VIO0_D2	IO / I	pull-up	VCCQ_VIO	open
15	A15	VDD_FLL	VDD_FLL	—	—	—	—
16	A16	VDD_PLL	VDD_PLL	—	—	—	—
17	A17	PTV5	PTV5 / FSIIBBCK / MSIOF1_RXD	IO / I / I	pull-up	VCCQ_SR	open
18	A18	PTV1	PTV1 / CLKAUDIOBO / MSIOF1_MCK	IO / O / I	pull-up	VCCQ_SR	open
19	A19	PTU4	PTU4 / FSIIABCK	IO / I	pull-up	VCCQ_SR	open
20	A20	PTU0	PTU0 / CLKAUDIOAO	IO / O	pull-up	VCCQ_SR	open
21	A21	PTZ5	PTZ5 / IRQ5 / SCIF3_SCK	IO / I / IO	pull-up	VCCQ_SR	open
22	A22	PTX6	PTX6 / DREQ1 / $\overline{\text{IRDA\_IN}}$	IO / I / I	pull-up	VCCQ_SR	open
23	A23	PTX2	PTX2 / TS_SPSYNC	IO / I	pull-down	VCCQ_SR	open
24	A24	PTN5	PTN5 / DV_CLKI	IO / I	pull-down	VCCQ_SR	open
25	A25	VSS	VSS	—	—	—	—
26	B1	EXTALUSB	EXTALUSB	I		VCCQ	pull-down
27	B2	XTALUSB	XTALUSB	O		VCCQ	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
28	B3	AG33_1	AG33	—	—	—	—
29	B4	AV33_1	AV33	—	—	—	—
30	B5	DG33_1	DG33	—	—	—	—
31	B6	AG33_0	AG33	—	—	—	—
32	B7	DV33_0	DV33	—	—	—	—
33	B8	DG33_0	DG33	—	—	—	—
34	B9	PTS5	PTS5 / VIO1_FLD / TPUTI2 / IDEIORDY	IO / I / I / I	pull-up	VCCQ_VIO	open
35	B10	PTS4	PTS4 / VIO1_HD / SCIF5_SCK	IO / I / IO	pull-up	VCCQ_VIO	open
36	B11	PTS0	PTS0 / VIO1_D6 / VIO0_D14 / IDED6	IO / I / I / IO	pull-up	VCCQ_VIO	open
37	B12	PTK4	PTK4 / VIO1_D2 / VIO0_D10 / IDED2	IO / I / I / IO	pull-up	VCCQ_VIO	open
38	B13	PTH7	PTH7 / VIO0_VD	IO / I	pull-up	VCCQ_VIO	open
39	B14	PTH3	PTH3 / VIO0_D5	IO / I	pull-up	VCCQ_VIO	open
40	B15	VSS_FLL	VSS_FLL	—	—	—	—
41	B16	VSS_PLL	VSS_PLL	—	—	—	—
42	B17	PTV3	PTV3 / FSIOBBCK / MSIOF1_TSCK	IO / O / IO	pull-up	VCCQ_SR	open
43	B18	PTE7	PTE7 / FSIMCKB	IO / I	pull-up	VCCQ_SR	open
44	B19	PTU3	PTU3 / FSIIALRCK	IO / I	pull-up	VCCQ_SR	open
45	B20	PTZ7	PTZ7 / IRQ7 / $\overline{\text{SCIF3\_CTS}}$	IO / I / I	pull-up	VCCQ_SR	open
46	B21	PTX7	PTX7 / DACK1 / IRDA_OUT	IO / O / O	pull-up	VCCQ_SR	open
47	B22	PTX5	PTX5 / TS_SDAT / LNKSTA	IO / I / I	pull-down	VCCQ_SR	open
48	B23	PTN3	PTN3 / DV_VSYNC / SCIF2_RXD	IO / O / I	pull-up	VCCQ_SR	open
49	B24	PTL7	PTL7 / DV_D5 / SCIF3_SCK / RMII_RXD0	IO / O / IO / I	pull-up	VCCQ_SR	open
50	B25	PTN0	PTN0 / DV_D6 / $\overline{\text{SCIF3\_RTS}}$ / RMII_CRS_DV	IO / O / O / I	pull-up	VCCQ_SR	open
51	C1	MD3	MD3	I	schmitt	VCCQ	—
52	C2	MD1	MD1	I	schmitt	VCCQ	—
53	C3	MD2	MD2	I	schmitt	VCCQ	—
54	C4	AV12_1	AV12	—	—	—	—

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
55	C5	DV33_1	DV33	—	—	—	—
56	C6	VBUS1	VBUS1	I		DV33_1	pull-down
57	C7	AG12_0	AG12	—	—	—	—
58	C8	AV12_0	AV12	—	—	—	—
59	C9	VBUS0	VBUS0	I		DV33_0	pull-down
60	C10	SDA1	SDA1	IO	open-drain	VCCQ_VIO	open
61	C11	PTS6	PTS6 / VIO_CKO	IO / O	pull-up	VCCQ_VIO	open
62	C12	PTK5	PTK5 / VIO1_D3 / VIO0_D11 / IDED3	IO / I / I / IO	pull-up	VCCQ_VIO	open
63	C13	PTK1	PTK1 / VIO0_FLD	IO / I	pull-up	VCCQ_VIO	open
64	C14	PTH6	PTH6 / VIO0_CLK	IO / I	pull-up	VCCQ_VIO	open
65	C15	PTH1	PTH1 / VIO0_D3	IO / I	pull-up	VCCQ_VIO	open
66	C16	PTV7	PTV7 / FSIIBSD / $\overline{\text{MSIOF1\_SS2}}$ / MSIOF1_RSYNC	IO / I / IO	pull-up	VCCQ_SR	open
67	C17	PTV4	PTV4 / FSIIBLRCK / MSIOF1_TSYNC	IO / I / O	pull-up	VCCQ_SR	open
68	C18	PTU2	PTU2 / FSIOABCK	IO / O	pull-up	VCCQ_SR	open
69	C19	PTZ6	PTZ6 / IRQ6 / $\overline{\text{SCIF3\_RTS}}$	IO / I / O	pull-up	VCCQ_SR	open
70	C20	PTZ3	PTZ3 / IRQ3 / SCIF3_TXD	IO / I / O	pull-up	VCCQ_SR	open
71	C21	SDA0	SDA0	IO	open-drain	VCCQ_SR	open
72	C22	PTX3	PTX3 / TS_SDEN / MDC	IO / I / O	pull-down	VCCQ_SR	open
73	C23	PTN1	PTN1 / DV_D7 / $\overline{\text{SCIF3\_CTS}}$ / RMII_RX_ER	IO / O / I / I	pull-up	VCCQ_SR	open
74	C24	PTL5	PTL5 / DV_D3 / SCIF3_TXD / RMII_REF_CLK	IO / O / O / I	pull-up	VCCQ_SR	open
75	C25	PTL3	PTL3 / DV_D1 / SCIF1_RXD / RMII_TXD0	IO / O / I / O	pull-up	VCCQ_SR	open
76	D1	$\overline{\text{RESETA}}$	$\overline{\text{RESETA}}$	I	schmitt	VCCQ	pull up
77	D2	RTC_CLK	RTC_CLK	I	schmitt	VCCQ	pull up or Vss
78	D3	$\overline{\text{TSTMD}}$	$\overline{\text{TSTMD}}$	I	schmitt	VCCQ	pull up
79	D4	MD5	MD5	I	schmitt	VCCQ	—
80	D5	AG12_1	AG12	—	—	—	—
81	D6	UG12_1	UG12	—	—	—	—

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
82	D7	DG12_1	DG12	—	—	—	—
83	D8	UG12_0	UG12	—	—	—	—
84	D9	DG12_0	DG12	—	—	—	—
85	D10	PTS3	PTS3 / VIO1_VD / SCIF5_RXD	IO / I / I	pull-up	VCCQ_VIO	open
86	D11	PTS1	PTS1 / VIO1_D7 / VIO0_D15 / IDED7	IO / I / I / IO	pull-up	VCCQ_VIO	open
87	D12	PTK6	PTK6 / VIO1_D4 / VIO0_D12 / IDED4	IO / I / I / IO	pull-up	VCCQ_VIO	open
88	D13	PTK2	PTK2 / VIO1_D0 / VIO0_D8 / IDED0	IO / I / I / IO	pull-up	VCCQ_VIO	open
89	D14	PTH4	PTH4 / VIO0_D6	IO / I	pull-up	VCCQ_VIO	open
90	D15	PTN6	PTN6 / VIO0_D0	IO / I	pull-up	VCCQ_VIO	open
91	D16	PTV6	PTV6 / FSIOBSD / $\overline{\text{MSIOF1\_SS1}}$ / MSIOF1_RSCK	IO / O / IO	pull-up	VCCQ_SR	open
92	D17	PTV2	PTV2 / FSIOBLRCK / MSIOF1_TXD	IO / O / O	pull-up	VCCQ_SR	open
93	D18	PTU5	PTU5 / FSIOASD	IO / O	pull-up	VCCQ_SR	open
94	D19	PTZ4	PTZ4 / IRQ4 / SCIF3_RXD	IO / I / I	pull-up	VCCQ_SR	open
95	D20	PTX4	PTX4 / TS_SCK / MDIO	IO / I / IO	schmitt, pull-down	VCCQ_SR	open
96	D21	SCL0	SCL0	IO	open-drain	VCCQ_SR	open
97	D22	PTN2	PTN2 / DV_HSYNC / SCIF2_TXD	IO / O / O	pull-up	VCCQ_SR	open
98	D23	PTL2	PTL2 / DV_D0 / SCIF1_TXD / RMII_TXD1	IO / O / O / O	pull-up	VCCQ_SR	open
99	D24	PTM7	PTM7 / DV_D13 / MSIOF0_TSCK	IO / O / IO	pull-up	VCCQ_SR	open
100	D25	PTL0	PTL0 / DV_D14 / MSIOF0_MCK	IO / O / IO	pull-up	VCCQ_SR	open
101	E1	PDSTATUS	PTJ7 / PDSTATUS	O / O		VCCQ	open
102	E2	$\overline{\text{RESETOUT}}$	$\overline{\text{RESETOUT}}$	O		VCCQ	open
103	E3	STATUS2	PTJ6 / STATUS2	O / O		VCCQ	open
104	E4	MD8	MD8	I	schmitt	VCCQ	—
105	E5	VSS	VSS	—	—	—	—

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
106	E6	UV12_1	UV12	—	—	—	—
107	E7	DV12_1	DV12	—	—	—	—
108	E8	UV12_0	UV12	—	—	—	—
109	E9	DV12_0	DV12	—	—	—	—
110	E10	VSS	VSS	—	—	—	—
111	E11	VCCQ_VIO	VCCQ_VIO	—	—	—	—
112	E12	VCCQ_VIO	VCCQ_VIO	—	—	—	—
113	E13	PTK0	PTK0 / VIO0_HD	IO / I	pull-up	VCCQ_VIO	open
114	E14	PTH2	PTH2 / VIO0_D4	IO / I	pull-up	VCCQ_VIO	open
115	E15	PTN7	PTN7 / VIO0_D1	IO / I	pull-up	VCCQ_VIO	open
116	E16	PTV0	PTV0 / FSIIASD	IO / I	pull-up	VCCQ_SR	open
117	E17	PTE6	PTE6 / FSIMCKA	IO / I	pull-up	VCCQ_SR	open
118	E18	PTU1	PTU1 / FSIOALRCK	IO / O	pull-up	VCCQ_SR	open
119	E19	VCCQ_SR	VCCQ_SR	—	—	—	—
120	E20	VCCQ_SR	VCCQ_SR	—	—	—	—
121	E21	VCCQ_SR	VCCQ_SR	—	—	—	—
122	E22	PTL1	PTL1 / DV_D15	IO / O	pull-up	VCCQ_SR	open
123	E23	PTM6	PTM6 / DV_D12 / MSIOF0_RXD	IO / O / I	pull-up	VCCQ_SR	open
124	E24	PTM5	PTM5 / DV_D11 / MSIOF0_TXD	IO / O / O	pull-up	VCCQ_SR	open
125	E25	PTM4	PTM4 / DV_D10 / MSIOF0_TSYNC	IO / O / IO	pull-up	VCCQ_SR	open
126	F1	TCK	TCK	I	pull-up	VCCQ	open
127	F2	$\overline{\text{TST}}$	$\overline{\text{TST}}$	I		VCCQ	pull-up
128	F3	BOOT	BOOT	I	schmitt	VCCQ	—
129	F4	MD0	MD0	I	schmitt	VCCQ	—
130	F5	VSS	VSS	—	—	—	—
131	F21	VSS	VSS	—	—	—	—
132	F22	PTN4	PTN4 / DV_CLK / SCIF2_SCK	IO / O / IO	pull-up	VCCQ_SR	open
133	F23	PTM3	PTM3 / DV_D9 / $\overline{\text{MSIOF0\_SS1}}$ / MSIOF0_RSCK	IO / O / IO	pull-up	VCCQ_SR	open
134	F24	PTZ1	PTZ1 / IRQ1	IO / I	pull-up	VCCQ1	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
135	F25	PTZ0	PTZ0 / IRQ0	IO / I	pull-up	VCCQ1	open
136	G1	TDO	TDO	O		VCCQ	open
137	G2	TDI	TDI	I	pull-up	VCCQ	open
138	G3	TMS	TMS	I	pull-up	VCCQ	open
139	G4	$\overline{\text{RESETP}}$	$\overline{\text{RESETP}}$	I	schmitt	VCCQ	—
140	G5	VCCQ	VCCQ	—	—	—	—
141	G21	VSS	VSS	—	—	—	—
142	G22	PTL6	PTL6 / DV_D4 / SCIF3_RXD / RMII_RXD1	IO / O / I / I	pull-up	VCCQ_SR	open
143	G23	PTU7	PTU7 / DACK0	IO / O	pull-up	VCCQ1	open
144	G24	PTU6	PTU6 / DREQ0	IO / I	pull-up	VCCQ1	open
145	G25	CKO	CKO	O		VCCQ1	open
146	H1	PTG2	PTG2 / AUDATA2	O / O		VCCQ	open
147	H2	PTG1	PTG1 / AUDATA1	O / O		VCCQ	open
148	H3	PTG0	PTG0 / AUDATA0	O / O		VCCQ	open
149	H4	STATUS0	PTJ5 / STATUS0	O / O		VCCQ	open
150	H5	VCCQ	VCCQ	—	—	—	—
151	H21	VCCQ1	VCCQ1	—	—	—	—
152	H22	PTL4	PTL4 / DV_D2 / SCIF1_SCK / RMII_TX_EN	IO / O / IO / O	pull-up	VCCQ_SR	open
153	H23	$\overline{\text{WAIT}}$	PTR2 / $\overline{\text{WAIT}}$	I / I	pull-up	VCCQ1	open
154	H24	$\overline{\text{CS6A}}$	PTR6 / $\overline{\text{CS6A}}$/$\overline{\text{CE2B}}$	IO / O	pull-up	VCCQ1	open
155	H25	$\overline{\text{IOIS16}}$	PTR3 / $\overline{\text{IOIS16}}$ / LCDLCLK	I / I / I	—	VCCQ1	open
156	J1	EXTAL	EXTAL	I		VCCQ	pull-down
157	J2	PTG4	PTG4 / AUDSYNC	O / O		VCCQ	open
158	J3	PTG3	PTG3 / AUDATA3	O / O		VCCQ	open
159	J4	$\overline{\text{TRST}}$	$\overline{\text{TRST}}$	I	schmitt	VCCQ	pull-down
160	J5	VSS	VSS	—	—	—	—
161	J21	VCCQ1	VCCQ1	—	—	—	—
162	J22	PTM2	PTM2 / DV_D8 / $\overline{\text{MSIOF0_SS2}}$ / MSIOF0_RSYNC	IO / O / IO	pull-up	VCCQ_SR	open
163	J23	$\overline{\text{CS6B}}$	PTR7 / $\overline{\text{CS6B}}$/$\overline{\text{CE1B}}$	IO / O	pull-up	VCCQ1	open
164	J24	$\overline{\text{CS4}}$	$\overline{\text{CS4}}$	O		VCCQ1	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
165	J25	$\overline{\text{CS5A}}$	PTR4 / $\overline{\text{CS5A}}$/$\overline{\text{CE2A}}$	IO / O	pull-up	VCCQ1	open
166	K1	XTAL	XTAL	O		VCCQ	open
167	K2	ASEBRKBRKAK	$\overline{\text{ASEBRK}}$/BRKAK	IO		VCCQ	open
168	K3	MPMD	MPMD	I	schmitt	VCCQ	—
169	K4	PTG5	PTG5 / AUDCK	O / O		VCCQ	open
170	K5	VSS	VSS	—	—	—	—
171	K10	VDD	VDD	—	—	—	—
172	K11	VDD	VDD	—	—	—	—
173	K12	VSS	VSS	—	—	—	—
174	K13	VSS	VSS	—	—	—	—
175	K14	VSS	VSS	—	—	—	—
176	K15	VDD	VDD	—	—	—	—
177	K16	VDD	VDD	—	—	—	—
178	K21	VCCQ1	VCCQ1	—	—	—	—
179	K22	$\overline{\text{CS5B}}$	PTR5 / $\overline{\text{CS5B}}$/$\overline{\text{CE1A}}$	IO / O	pull-up	VCCQ1	open
180	K23	$\overline{\text{CS0}}$	$\overline{\text{CS0}}$	O		VCCQ1	open
181	K24	$\overline{\text{RD}}$	$\overline{\text{RD}}$	O		VCCQ1	open
182	K25	$\overline{\text{WE1}}$	$\overline{\text{WE1}}$	O		VCCQ1	open
183	L1	PTW2	PTW2 / MMC_D2 / SDHI1D0	IO / IO / IO	pull-up	VCCQ_MMC	open
184	L2	PTW1	PTW1 / MMC_D1 / SDHI1CMD	IO / IO / IO	pull-up	VCCQ_MMC	open
185	L3	PTW0	PTW0 / MMC_D0 / SDHI1CLK	IO / IO / O	pull-up	VCCQ_MMC	open
186	L4	NMI	NMI	I	schmitt	VCCQ	pull-up
187	L5	VSS	VSS	—	—	—	—
188	L10	VDD	VDD	—	—	—	—
189	L11	VSS	VSS	—	—	—	—
190	L12	VSS	VSS	—	—	—	—
191	L13	VSS	VSS	—	—	—	—
192	L14	VSS	VSS	—	—	—	—
193	L15	VSS	VSS	—	—	—	—
194	L16	VDD	VDD	—	—	—	—
195	L21	VSS	VSS	—	—	—	—
196	L22	RDWR	RDWR	O		VCCQ1	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
197	L23	$\overline{\text{WE0}}$	$\overline{\text{WE0}}$	O		VCCQ1	open
198	L24	A25	PTJ3 / A25 / $\overline{\text{BS}}$	IO / O / O	pull-up	VCCQ1	open
199	L25	A24	PTJ2 / A24	IO / O	pull-up	VCCQ1	open
200	M1	PTW5	PTW5 / MMC_D5 / SDHI1D3 / $\overline{\text{EXBUF\_ENB}}$	IO / IO / IO / O	pull-up	VCCQ_MMC	open
201	M2	PTW6	PTW6 / MMC_D6 / SDHI1WP / $\overline{\text{IDERST}}$	IO / IO / I / O	pull-up	VCCQ_MMC	open
202	M3	PTW4	PTW4 / MMC_D4 / SDHI1D2 / DIRECTION	IO / IO / IO / O	pull-up	VCCQ_MMC	open
203	M4	PTW3	PTW3 / MMC_D3 / SDHI1D1	IO / IO / IO	pull-up	VCCQ_MMC	open
204	M5	VCCQ_MMC	VCCQ_MMC	—	—	—	—
205	M10	VDD	VDD	—	—	—	—
206	M11	VSS	VSS	—	—	—	—
207	M12	VSS	VSS	—	—	—	—
208	M13	VSS	VSS	—	—	—	—
209	M14	VSS	VSS	—	—	—	—
210	M15	VSS	VSS	—	—	—	—
211	M16	VDD	VDD	—	—	—	—
212	M21	VCCQ1	VCCQ1	—	—	—	—
213	M22	A22	PTJ0 / A22	IO / O	pull-up	VCCQ1	open
214	M23	A21	A21	O		VCCQ1	open
215	M24	A23	PTJ1 / A23	IO / O	pull-up	VCCQ1	open
216	M25	A20	A20	O		VCCQ1	open
217	N1	PTW7	PTW7 / MMC_D7 / $\overline{\text{SDHI1CD}}$ / $\overline{\text{IODACK}}$	IO / IO / I / O	pull-up	VCCQ_MMC	open
218	N2	PTY0	PTY0 / SDHI0CLK	IO / O	pull-up	VCCQ_SDC	open
219	N3	PTX0	PTX0 / MMC_CMD	IO / IO	pull-up	VCCQ_MMC	open
220	N4	PTX1	PTX1 / MMC_CLK	IO / O	pull-up	VCCQ_MMC	open
221	N5	VCCQ_MMC	VCCQ_MMC	—	—	—	—
222	N10	VDD	VDD	—	—	—	—
223	N11	VSS	VSS	—	—	—	—
224	N12	VSS	VSS	—	—	—	—
225	N13	VSS	VSS	—	—	—	—

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
226	N14	VSS	VSS	—	—	—	—
227	N15	VSS	VSS	—	—	—	—
228	N16	VDD	VDD	—	—	—	—
229	N21	VCCQ1	VCCQ1	—	—	—	—
230	N22	A17	A17	O		VCCQ1	open
231	N23	A16	A16	O		VCCQ1	open
232	N24	A19	A19	O		VCCQ1	open
233	N25	A15	A15	O		VCCQ1	open
234	P1	PTY1	PTY1 / SDHI0CMD	IO / IO	pull-up	VCCQ_SDC	open
235	P2	PTY2	PTY2 / SDHI0D0	IO / IO	pull-up	VCCQ_SDC	open
236	P3	PTY4	PTY4 / SDHI0D2	IO / IO	pull-up	VCCQ_SDC	open
237	P4	PTY6	PTY6 / SDHI0WP	IO / I	pull-up	VCCQ_SDC	open
238	P5	VCCQ_SDC	VCCQ_SDC	—	—	—	—
239	P10	VDD	VDD	—	—	—	—
240	P11	VSS	VSS	—	—	—	—
241	P12	VSS	VSS	—	—	—	—
242	P13	VSS	VSS	—	—	—	—
243	P14	VSS	VSS	—	—	—	—
244	P15	VSS	VSS	—	—	—	—
245	P16	VDD	VDD	—	—	—	—
246	P21	A11	A11	O		VCCQ1	open
247	P22	A18	A18	O		VCCQ1	open
248	P23	A10	A10	O		VCCQ1	open
249	P24	A14	A14	O		VCCQ1	open
250	P25	A13	A13	O		VCCQ1	open
251	R1	PTY5	PTY5 / SDHI0D3	IO / IO	pull-up	VCCQ_SDC	open
252	R2	PTY7	PTY7 / $\overline{\text{SDHI0CD}}$	IO / I	pull-up	VCCQ_SDC	open
253	R3	PTZ2	PTZ2 / IRQ2	IO / I	pull-up	VCCQ_SDC	open
254	R4	PTY3	PTY3 / SDHI0D1	IO / IO	pull-up	VCCQ_SDC	open
255	R5	VCCQ_SDC	VCCQ_SDC	—	—	—	—
256	R10	VDD	VDD	—	—	—	—
257	R11	VSS	VSS	—	—	—	—

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
258	R12	VSS	VSS	—	—	—	—
259	R13	VSS	VSS	—	—	—	—
260	R14	VSS	VSS	—	—	—	—
261	R15	VSS	VSS	—	—	—	—
262	R16	VDD	VDD	—	—	—	—
263	R21	$\overline{WE3}$	PTR1 / $\overline{WE3}$/ICIOWR / TPUTO3 / TPUTI3	IO / O / O / I	pull-up	VCCQ1	open
264	R22	A3	A3	O		VCCQ1	open
265	R23	A5	A5	O		VCCQ1	open
266	R24	A9	A9	O		VCCQ1	open
267	R25	A12	A12	O		VCCQ1	open
268	T1	PTC0	PTC0 / LCDD0	IO / IO	pull-up	VCCQ_LCD	open
269	T2	PTC1	PTC1 / LCDD1	IO / IO	pull-up	VCCQ_LCD	open
270	T3	PTC3	PTC3 / LCDD3	IO / IO	pull-up	VCCQ_LCD	open
271	T4	PTC6	PTC6 / LCDD6	IO / IO	pull-up	VCCQ_LCD	open
272	T5	VSS	VSS	—	—	—	—
273	T10	VDD	VDD	—	—	—	—
274	T11	VDD	VDD	—	—	—	—
275	T12	VSS	VSS	—	—	—	—
276	T13	VSS	VSS	—	—	—	—
277	T14	VSS	VSS	—	—	—	—
278	T15	VDD	VDD	—	—	—	—
279	T16	VDD	VDD	—	—	—	—
280	T21	VSS	VSS	—	—	—	—
281	T22	D31	PTB7 / D31 / TPUTO1 / IDEA1	IO / IO / O / O	pull-up	VCCQ1	open
282	T23	A4	A4	O		VCCQ1	open
283	T24	A7	A7	O		VCCQ1	open
284	T25	A8	A8	O		VCCQ1	open
285	U1	PTC2	PTC2 / LCDD2	IO / IO	pull-up	VCCQ_LCD	open
286	U2	PTC4	PTC4 / LCDD4	IO / IO	pull-up	VCCQ_LCD	open
287	U3	PTD0	PTD0 / LCDD8	IO / IO	pull-up	VCCQ_LCD	open
288	U4	PTD4	PTD4 / LCDD12	IO / IO	pull-up	VCCQ_LCD	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
289	U5	VCCQ_LCD	VCCQ_LCD	—	—	—	—
290	U21	VCCQ1	VCCQ1	—	—	—	—
291	U22	D21	PTA5 / D21 / KEYOUT0 / IDED13	IO / IO / O / IO	pull-up	VCCQ1	open
292	U23	$\overline{\text{WE2}}$	PTR0 / $\overline{\text{WE2}}$/$\overline{\text{ICIORD}}$ / TPUTO2 / IDEA2	IO / O / O / O	pull-up	VCCQ1	open
293	U24	A2	A2	O		VCCQ1	open
294	U25	A6	A6	O		VCCQ1	open
295	V1	PTC5	PTC5 / LCDD5	IO / IO	pull-up	VCCQ_LCD	open
296	V2	PTC7	PTC7 / LCDD7	IO / IO	pull-up	VCCQ_LCD	open
297	V3	PTD3	PTD3 / LCDD11	IO / IO	pull-up	VCCQ_LCD	open
298	V4	PTE0	PTE0 / LCDD16	IO / IO	pull-up	VCCQ_LCD	open
299	V5	VCCQ_LCD	VCCQ_LCD	—	—	—	—
300	V21	VCCQ1	VCCQ1	—	—	—	—
301	V22	D15	PTT7 / D15	IO / IO	pull-down	VCCQ1	open
302	V23	D27	PTB3 / D27 / $\overline{\text{IDECS1}}$	IO / IO / O	pull-up	VCCQ1	open
303	V24	D30	PTB6 / D30 / TPUTO0 / IDEA0	IO / IO / O / O	pull-up	VCCQ1	open
304	V25	A1	A1	O		VCCQ1	open
305	W1	PTD1	PTD1 / LCDD9	IO / IO	pull-up	VCCQ_LCD	open
306	W2	PTD2	PTD2 / LCDD10	IO / IO	pull-up	VCCQ_LCD	open
307	W3	PTD6	PTD6 / LCDD14	IO / IO	pull-up	VCCQ_LCD	open
308	W4	PTE4	PTE4 / LCDD20 / SCIF4_SCK	IO / IO / IO	pull-down	VCCQ_LCD	open
309	W5	VCCQ_LCD	VCCQ_LCD	—	—	—	—
310	W21	VCCQ1	VCCQ1	—	—	—	—
311	W22	D23	PTA7 / D23 / KEYOUT2 / IDED15	IO / IO / O / IO	pull-up	VCCQ1	open
312	W23	D24	PTB0 / D24 / KEYOUT3 / IDEINT	IO / IO / O / I	pull-up	VCCQ1	open
313	W24	D28	PTB4 / D28 / $\overline{\text{IDECS0}}$	IO / IO / O	pull-up	VCCQ1	open
314	W25	A0	A0	O		VCCQ1	open
315	Y1	PTD5	PTD5 / LCDD13	IO / IO	pull-up	VCCQ_LCD	open
316	Y2	PTD7	PTD7 / LCDD15	IO / IO	pull-up	VCCQ_LCD	open
317	Y3	PTE1	PTE1 / LCDD17	IO / IO	pull-up	VCCQ_LCD	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
318	Y4	PTF7	PTF7 / LCDVSYN	IO / IO	pull-up	VCCQ_LCD	open
319	Y5	VSS	VSS	—	—	—	—
320	Y21	VSS	VSS	—	—	—	—
321	Y22	D7	PTQ7 / D7	IO / IO	pull-down	VCCQ1	open
322	Y23	D20	PTA4 / D20 / KEYIN4 / IDED12	IO / IO / I / IO	pull-up	VCCQ1	open
323	Y24	D26	PTB2 / D26 / KEYOUT5/IN5 / $\overline{\text{IDEIORD}}$	IO / IO / IO / O	pull-up	VCCQ1	open
324	Y25	D29	PTB5 / D29 / IODREQ	IO / IO / I	pull-up	VCCQ1	open
325	AA1	PTE2	PTE2 / LCDD18 / SCIF4_TXD	IO / IO / O	pull-down	VCCQ_LCD	open
326	AA2	PTE3	PTE3 / LCDD19 / SCIF4_RXD	IO / IO / I	pull-down	VCCQ_LCD	open
327	AA3	PTF0	PTF0 / LCDD22 / SCIF2_RXD	IO / IO / I	pull-down	VCCQ_LCD	open
328	AA4	PTF3	PTF3 / LCDDCK / $\overline{\text{LCDWR}}$	IO / O / O	pull-up	VCCQ_LCD	open
329	AA5	VSS	VSS	—	—	—	—
330	AA6	VSS	VSS	—	—	—	—
331	AA7	MVREF0	MVREF0	I	analog	VCCQ_DDR	open
332	AA8	VCCQ_DDR	VCCQ_DDR	—	—	—	—
333	AA9	VCCQ_DDR	VCCQ_DDR	—	—	—	—
334	AA10	VSS	VSS	—	—	—	—
335	AA11	VCCQ_DDR	VCCQ_DDR	—	—	—	—
336	AA12	VCCQ_DDR	VCCQ_DDR	—	—	—	—
337	AA13	VSS	VSS	—	—	—	—
338	AA14	VCCQ_DDR	VCCQ_DDR	—	—	—	—
339	AA15	VCCQ_DDR	VCCQ_DDR	—	—	—	—
340	AA16	VSS	VSS	—	—	—	—
341	AA17	MVREF1	MVREF1	I	analog	VCCQ_DDR	open
342	AA18	VCCQ_DDR	VCCQ_DDR	—	—	—	—
343	AA19	VCCQ_DDR	VCCQ_DDR	—	—	—	—
344	AA20	VSS	VSS	—	—	—	—
345	AA21	VSS	VSS	—	—	—	—
346	AA22	D11	PTT3 / D11	IO / IO	pull-down	VCCQ1	open
347	AA23	D16	PTA0 / D16 / KEYIN0 / IDED8	IO / IO / I / IO	pull-up	VCCQ1	open
348	AA24	D19	PTA3 / D19 / KEYIN3 / IDED11	IO / IO / I / IO	pull-up	VCCQ1	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
349	AA25	D25	PTB1 / D25 / KEYOUT4/IN6 / $\overline{\text{IDEIOWR}}$	IO / IO / IO / O	pull-up	VCCQ1	open
350	AB1	PTE5	PTE5 / LCDD21 / SCIF2_TXD	IO / IO / O	pull-down	VCCQ_LCD	open
351	AB2	PTF2	PTF2 / LCDVEPWC / SCIF0_TXD	IO / O / O	pull-up	VCCQ_LCD	open
352	AB3	PTF4	PTF4 / LCDDON	IO / O	pull-up	VCCQ_LCD	open
353	AB4	PTM0	PTM0 / $\overline{\text{LCDRD}}$ / SCIF0_SCK	IO / O / IO	pull-up	VCCQ_LCD	open
354	AB5	MSLD	MSLD	I		VCCQ_DDR	—
355	AB6	MDQ1	MDQ1	IO		VCCQ_DDR	open
356	AB7	MDQM0	MDQM0	O		VCCQ_DDR	open
357	AB8	MDQS0	MDQS0	IO		VCCQ_DDR	open
358	AB9	MDQ2	MDQ2	IO		VCCQ_DDR	open
359	AB10	MA9	MA9	O		VCCQ_DDR	open
360	AB11	MCKE	MCKE	O		VCCQ_DDR	open
361	AB12	MA4	MA4	O		VCCQ_DDR	open
362	AB13	$\overline{\text{MCAS}}$	$\overline{\text{MCAS}}$	O		VCCQ_DDR	open
363	AB14	MA8	MA8	O		VCCQ_DDR	open
364	AB15	MCLK	MCLK	O		VCCQ_DDR	open
365	AB16	MDQ17	MDQ17	IO		VCCQ_DDR	open
366	AB17	MDQ22	MDQ22	IO		VCCQ_DDR	open
367	AB18	$\overline{\text{MDQS2}}$	$\overline{\text{MDQS2}}$	IO		VCCQ_DDR	open
368	AB19	MDQ16	MDQ16	IO		VCCQ_DDR	open
369	AB20	D0	PTQ0 / D0	IO / IO	pull-down	VCCQ1	open
370	AB21	D3	PTQ3 / D3	IO / IO	pull-down	VCCQ1	open
371	AB22	D8	PTT0 / D8	IO / IO	pull-down	VCCQ1	open
372	AB23	D12	PTT4 / D12	IO / IO	pull-down	VCCQ1	open
373	AB24	D17	PTA1 / D17 / KEYIN1 / IDED9	IO / IO / I / IO	pull-up	VCCQ1	open
374	AB25	D22	PTA6 / D22 / KEYOUT1 / IDED14	IO / IO / O / IO	pull-up	VCCQ1	open
375	AC1	PTF1	PTF1 / LCDD23 / SCIF2_SCK	IO / IO / IO	pull-down	VCCQ_LCD	open
376	AC2	PTF5	PTF5 / LCDHSYN / $\overline{\text{LCDCS}}$	IO / O / O	pull-up	VCCQ_LCD	open
377	AC3	PTM1	PTM1 / LCDVCPWC / SCIF0_RXD	IO / O / I	pull-up	VCCQ_LCD	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
378	AC4	VSS	VSS	—	—	—	—
379	AC5	MDQ4	MDQ4	IO		VCCQ_DDR	open
380	AC6	MDQ3	MDQ3	IO		VCCQ_DDR	open
381	AC7	MDQ6	MDQ6	IO		VCCQ_DDR	open
382	AC8	$\overline{\text{MDQS0}}$	$\overline{\text{MDQS0}}$	IO		VCCQ_DDR	open
383	AC9	MDQ0	MDQ0	IO		VCCQ_DDR	open
384	AC10	MA7	MA7	O		VCCQ_DDR	open
385	AC11	MA1	MA1	O		VCCQ_DDR	open
386	AC12	MA5	MA5	O		VCCQ_DDR	open
387	AC13	MA0	MA0	O		VCCQ_DDR	open
388	AC14	MA12	MA12	O		VCCQ_DDR	open
389	AC15	$\overline{\text{MCLK}}$	$\overline{\text{MCLK}}$	O		VCCQ_DDR	open
390	AC16	MDQ20	MDQ20	IO		VCCQ_DDR	open
391	AC17	MDQM2	MDQM2	O		VCCQ_DDR	open
392	AC18	MDQS2	MDQS2	IO		VCCQ_DDR	open
393	AC19	MDQ23	MDQ23	IO		VCCQ_DDR	open
394	AC20	MDQ18	MDQ18	IO		VCCQ_DDR	open
395	AC21	D1	PTQ1 / D1	IO / IO	pull-down	VCCQ1	open
396	AC22	D5	PTQ5 / D5	IO / IO	pull-down	VCCQ1	open
397	AC23	D10	PTT2 / D10	IO / IO	pull-down	VCCQ1	open
398	AC24	D13	PTT5 / D13	IO / IO	pull-down	VCCQ1	open
399	AC25	D18	PTA2 / D18 / KEYIN2 / IDED10	IO / IO / I / IO	pull-up	VCCQ1	open
400	AD1	PTF6	PTF6 / LCDDISP / LCDRS	IO / O / O	pull-up	VCCQ_LCD	open
401	AD2	VSS	VSS	—	—	—	—
402	AD3	MDQ12	MDQ12	IO		VCCQ_DDR	open
403	AD4	MDQ14	MDQ14	IO		VCCQ_DDR	open
404	AD5	MDQM1	MDQM1	O		VCCQ_DDR	open
405	AD6	$\overline{\text{MDQS1}}$	$\overline{\text{MDQS1}}$	IO		VCCQ_DDR	open
406	AD7	MDQ15	MDQ15	IO		VCCQ_DDR	open
407	AD8	MDQ13	MDQ13	IO		VCCQ_DDR	open
408	AD9	MDQ5	MDQ5	IO		VCCQ_DDR	open
409	AD10	MA3	MA3	O		VCCQ_DDR	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
410	AD11	MBA2	MBA2	O		VCCQ_DDR	open
411	AD12	MBA1	MBA1	O		VCCQ_DDR	open
412	AD13	$\overline{\text{MRAS}}$	$\overline{\text{MRAS}}$	O		VCCQ_DDR	open
413	AD14	MCS	MCS	O		VCCQ_DDR	open
414	AD15	MA6	MA6	O		VCCQ_DDR	open
415	AD16	MDQ19	MDQ19	IO		VCCQ_DDR	open
416	AD17	MDQ27	MDQ27	IO		VCCQ_DDR	open
417	AD18	MDQ25	MDQ25	IO		VCCQ_DDR	open
418	AD19	MDQS3	MDQS3	IO		VCCQ_DDR	open
419	AD20	MDQ24	MDQ24	IO		VCCQ_DDR	open
420	AD21	MDQ26	MDQ26	IO		VCCQ_DDR	open
421	AD22	MDQ21	MDQ21	IO		VCCQ_DDR	open
422	AD23	D6	PTQ6 / D6	IO / IO	pull-down	VCCQ1	open
423	AD24	D9	PTT1 / D9	IO / IO	pull-down	VCCQ1	open
424	AD25	D14	PTT6 / D14	IO / IO	pull-down	VCCQ1	open
425	AE1	VSS	VSS	—	—	—	—
426	AE2	VCCQ_DDR	VCCQ_DDR	—	—	—	—
427	AE3	MDQ9	MDQ9	IO		VCCQ_DDR	open
428	AE4	MDQ11	MDQ11	IO		VCCQ_DDR	open
429	AE5	MDQS1	MDQS1	IO		VCCQ_DDR	open
430	AE6	MDQ8	MDQ8	IO		VCCQ_DDR	open
431	AE7	MDQ10	MDQ10	IO		VCCQ_DDR	open
432	AE8	MDQ7	MDQ7	IO		VCCQ_DDR	open
433	AE9	MA13	MA13	O		VCCQ_DDR	open
434	AE10	MA10	MA10	O		VCCQ_DDR	open
435	AE11	MBA0	MBA0	O		VCCQ_DDR	open
436	AE12	$\overline{\text{MWE}}$	$\overline{\text{MWE}}$	O		VCCQ_DDR	open
437	AE13	MODT	MODT	O		VCCQ_DDR	open
438	AE14	MA2	MA2	O		VCCQ_DDR	open
439	AE15	MA11	MA11	O		VCCQ_DDR	open
440	AE16	MDQ28	MDQ28	IO		VCCQ_DDR	open
441	AE17	MDQ30	MDQ30	IO		VCCQ_DDR	open

No.	Location	Pin Name	Function (Multiplexed Functions Where Applicable)	I/O	I/O Buffer Type	I/O Buffer Power Supply	Handling when unused
442	AE18	MDQM3	MDQM3	O		VCCQ_DDR	open
443	AE19	$\overline{\text{MDQS3}}$	$\overline{\text{MDQS3}}$	IO		VCCQ_DDR	open
444	AE20	MDQ31	MDQ31	IO		VCCQ_DDR	open
445	AE21	MDQ29	MDQ29	IO		VCCQ_DDR	open
446	AE22	VCCQ_DDR	VCCQ_DDR	—	—	—	—
447	AE23	D2	PTQ2 / D2	IO / IO	pull-down	VCCQ1	open
448	AE24	D4	PTQ4 / D4	IO / IO	pull-down	VCCQ1	open
449	AE25	VSS	VSS	—	—	—	—

1.6 Pin Functions

Table 1.4 lists the pin functions. For details of pin multiplexing, refer to section 48, Pin Function Controller (PFC). For pin states and handling of unused pins, refer to Appendix, Pin States at Reset and in Power-Down Modes.

Table 1.4 Pin Functions

Classification	Symbol	I/O	Function	Description
Power supply	VDD	I	Core power supply	Core power supply: 1.15 V to 1.3 V
	VSS	I	Ground	Ground pin: 0 V
	VCCQ	I	I/O power supply	Power supply for I/O pins: 2.7 V to 3.6 V
	VCCQ1	I	I/O power supply	Power supply for BSC and DMAC0 I/O pins: 1.8 V or 3.3 V selectable
	VCCQ_LCD	I	I/O power supply	Power supply for LCDC I/O pins: 1.8 V or 3.3 V selectable
	VCCQ_MMC	I	I/O power supply	Power supply for MMC I/O pins: 1.8 V or 3.3 V selectable
	VCCQ_SDC	I	I/O power supply	Power supply for SDHI0 I/O pins: 3.3 V
	VCCQ_SR	I	I/O power supply	Power supply for serial I/O pins: 1.8 V or 3.3 V selectable
	VCCQ_VIO	I	I/O power supply	Power supply for VIO and I2C_1 I/O pins: 1.8 V or 3.3 V selectable
	DV33, DV12, AV33, AV12, UV12	I	USB power supply	DV33: USB digital 3.3-V power supply DV12: USB digital 1.2-V power supply AV33: USB analog 3.3-V power supply AV12: USB analog 1.2-V power supply UV12: USB digital 1.2-V power supply
	DG33, DG12, AG33, AG12, UG12	I	USB ground	DG33: DV33 ground DG12: DV12 ground AG33: AV33 ground AG12: AV12 ground UG12: DV12 ground
	VCCQ_DDR	I	DDR I/O power supply	1.8-V power supply for DDR-IO
	VDD_PLL	I	PLL power supply	1.2-V power supply for on-chip PLL
	VSS_PLL	I	PLL ground	Ground pin for on-chip PLL
	VDD_FLL	I	FLL power supply	1.2-V power supply for on-chip FLL
	VSS_FLL	I	FLL ground	Ground pin for on-chip FLL

Classification	Symbol	I/O	Function	Description
Clock	EXTAL	I	External clock	Connect the external clock or a crystal resonator. When this pin is not used, connect it to Vss.
	XTAL	O	Clock output	Connect a crystal resonator. When no crystal resonator is used, leave this pin open.
	RTC_CLK	I	RTC clock	Input the 32.768-kHz RTC clock. When this pin is not used, pull it up or connect it to Vss.
	FSIMCKA FSIMCKB	I	External clock for FSI	External clock input for the FSI module interface
	EXTALUSB	I	USB clock	48-MHz clock pin for the USB. Connect a crystal resonator between EXTALUSB and XTALUSB. When inputting an external clock, connect it to EXTALUSB and leave XTALUSB open.
	XTALUSB	O	USB clock	
Operating mode control	MD8, MD5, TSTMD, MD3, MD2, MD1, MD0	I	Mode setting	These pins set operating mode. MD2 to MD0 are for setting clock mode; MD3 for setting bus width; MD5 for setting endian. For details of settings, refer to the descriptions of the BSC and CPG. MD8 and $\overline{\text{TSTMD}}$ are for testing; fix MD8 at VccQ or GND and fix $\overline{\text{TSTMD}}$ at VccQ. Do not change any of these pins during operation.
System control	$\overline{\text{RESETP}}$	I	Power-on reset	Power-on reset pin
	$\overline{\text{RESETOUT}}$	O	Reset output	This pin goes low while the LSI is in power-on reset or system reset state.
	$\overline{\text{RESETA}}$	I	Reset input	System reset pin
	STATUS0	O	Status output	This pin goes high while the LSI is in software standby state.
	STATUS2	O	Status output	This pin goes high while the LSI is in R-standby state.
	PDSTATUS	O	Power-down status output	This pin goes high while the internal power switch is off in software standby or R-standby state.

Classification	Symbol	I/O	Function	Description
System control	BOOT	I	Boot mode input	Specifies boot from the NAND flash connected through the MMCIF. Fix this pin at a high level for NAND boot and fix it at a low level for boot from area 0.
	$\overline{\text{TST}}$	I	Test pin	Pin for testing the LSI before shipment. Fix it at VccQ.
Interrupt	NMI	I	Non-maskable interrupt	NMI request pin. When this pin is not used, fix it at a high level.
	IRQ7 to IRQ0	I	Interrupt requests 7 to 0	IRQ request pins. Either level input or edge input is selectable.
BSC (external memory bus)	A25 to A0	O	Address bus	Address output
	D31 to D0	I/O	Data bus	Bidirectional data bus
	$\overline{\text{CS0}}$, $\overline{\text{CS4}}$, $\overline{\text{CS5A}}$, $\overline{\text{CS5B}}$, $\overline{\text{CS6A}}$, $\overline{\text{CS6B}}$	O	Chip select	Chip select
	CKO	O	System clock	System clock output
	$\overline{\text{RD}}$	O	Read strobe	Read strobe
	RDWR	O	Read/write	Read/write
	$\overline{\text{WE3}}$ to $\overline{\text{WE0}}$	O	Write enable 3 to 0	Write enable
	$\overline{\text{WAIT}}$	I	Wait	External wait cycle request input
	$\overline{\text{BS}}$	O	Bus start	Indicates the start of a bus cycle. It is asserted when the normal space, burst ROM, or PCMCIA is accessed.
	$\overline{\text{CE1A}}$, $\overline{\text{CE2A}}$ $\overline{\text{CE1B}}$, $\overline{\text{CE2B}}$	O	PCMCIA card select	PCMCIA card select signals
	$\overline{\text{ICIOWR}}$	O	PCMCIA IO write	Strobe signal indicating I/O write
	$\overline{\text{ICIORD}}$	O	PCMCIA IO read	Strobe signal indicating I/O read
	$\overline{\text{WE}}$	O	PCMCIA memory write	Strobe signal indicating a memory write cycle
	$\overline{\text{IOIS16}}$	I	PCMCIA 16-bit I/O	Indicates the 16-bit I/O in PCMCIA. It is valid only in little endian; fix it at a low level in big endian.

Classification	Symbol	I/O	Function	Description
DBSC (DDR bus)	MA13 to MA0	O	Address bus	Address output
	MBA2 to MBA0	O	Bank address	Bank address output
	MDQ31 to MDQ0	I/O	Data bus	Bidirectional data bus
	$\overline{\text{MCS}}$	O	Chip select	Chip select output signal
	MCLK, $\overline{\text{MCLK}}$	O	Synchronizing clock	DDR-SDRAM clock output
	$\overline{\text{MWE}}$	O	Write enable	Write enable output signal
	MDQM3 to MDQM0	O	Data mask 3 to 0	Data mask output signals
	MDQS3 to MDQS0	I/O	Data strobe	Data strobe input/output signals
	$\overline{\text{MDQS3}}$ to $\overline{\text{MDQS0}}$	I/O	Data strobe	Data strobe input/output signals
	$\overline{\text{MCAS}}$	O	Column address strobe	Column address strobe output signal
	$\overline{\text{MRAS}}$	O	Row address strobe	Row address strobe output signal
	MCKE	O	Clock enable	Clock enable output signal
	MSLD	I	Memory select	Memory type select input signal. Fix it at a low level when DDR2-SDRAM is used; fix it at a high level when mobile-DDR-SDRAM is used.
	MODT	O	ODT enable	ODT enable output signal from DDR2-SDRAM
	MVREF1, MVREF0	I	Reference voltage input	Reference voltage input These pins should be open when mobile-DDR-SDRAM is used.
Direct memory access controller (DMAC)	DREQ0, DREQ1	I	DMA transfer request	External DMA transfer request input
	DACK0, DACK1	O	DMA transfer request acknowledge	Output pin for request acknowledge in response to DREQ

Classification	Symbol	I/O	Function	Description
Clocked synchronous serial I/O with FIFO (MSIOF0/ MSIOF1)	MSIOF0_MCK, MSIOF1_MCK	I	Master clock input	Master clock input pins
	MSIOF0_TXD, MSIOF1_TXD	O	Transmit data	Transmit data pins
	MSIOF0_RXD, MSIOF1_RXD	I	Receive data	Receive data pins
	MSIOF0_TSCK, MSIOF1_TSCK	I/O	Transmit serial clock	Serial clock pins for transmission. When a common clock is used for transmission and reception, these pins are used as SCK.
	MSIOF0_TSYNC, MSIOF1_TSYNC	I/O	Transmit frame synchronizing signal	Transmit frame synchronizing signal input/output pins for channel 0. When a common synchronizing signal is used for transmission and reception, these pins are used as SYNC.
	$\overline{\text{MSIOF0\_SS1}}$, $\overline{\text{MSIOF1\_SS1}}$	O	Transmit frame synchronizing signal	Transmit frame synchronizing signal output pins for channel 1. Selectable only in the slave device.
	$\overline{\text{MSIOF0\_SS2}}$, $\overline{\text{MSIOF1\_SS2}}$	O	Transmit frame synchronizing signal	Transmit frame synchronizing signal output pins for channel 2. Selectable only in the slave device.
	MSIOF0_RSCK, MSIOF1_RSCK	I/O	Receive serial clock	Receive serial clock
	MSIOF0_RSYNC, MSIOF1_RSYNC	I/O	Receive frame synchronizing signal	Receive frame synchronizing signals
Serial communication interface with 16 FIFOs (SCIF)	SCIF0_TXD, SCIF1_TXD, SCIF2_TXD	O	Transmit data	Transmit data pins
	SCIF0_RXD, SCIF1_RXD, SCIF2_RXD	I	Receive data	Receive data pins
	SCIF0_SCK, SCIF1_SCK, SCIF2_SCK	I/O	Serial clock	Clock input/output pins

Classification	Symbol	I/O	Function	Description
Serial communication interface with 64 FIFOs (SCIFA)	SCIF3_TXD, SCIF4_TXD, SCIF5_TXD	O	Transmit data	Transmit data pins
	SCIF3_RXD, SCIF4_RXD, SCIF5_RXD	I	Receive data	Receive data pins
	SCIF3_SCK, SCIF4_SCK, SCIF5_SCK	I/O	Serial clock	Clock input/output pins
	$\overline{\text{SCIF3\_RTS}}$	O	$\overline{\text{RTS}}$ signal	$\overline{\text{RTS}}$ output pin
	$\overline{\text{SCIF3\_CTS}}$	I	$\overline{\text{CTS}}$ signal	$\overline{\text{CTS}}$ input pin
Timer pulse unit (TPU)	TPUTO4 to 0	O	Output signal	Pulse output from the TPU
	TPUTI2 to 3	I	External clock input signal	External count clock input pins
IrDA interface (IrDA)	$\overline{\text{IrDA\_IN}}$	I	Receive data input	Receive data input
	IrDA_OUT	O	Transmit data output	Transmit data output
I^2C bus interface (IIC)	SCL1, SCL0	I/O	I^2C clock input/output	I^2C bus clock input/output pins. These pins have the bus drive function. Output type is NMOS open drain.
	SDA1, SDA0	I/O	I^2C data input/output	I^2C bus data input/output pins. These pins have the bus drive function. Output type is NMOS open drain.
Ethernet controller (EtherMAC)	LNKSTA	I	Link status	Pin for LINK signal input from PHY
	MDC	O	Management data clock	Output of clock for RMII management data
	MDIO	I/O	Management data input/output	RMII management data input/output
	RMII_CRS_DV	I	RMII carrier sense	RMII carrier sense input signal
	RMII_REF_CLK	I	RMII reference clock	RMII reference clock (50 MHz)
	RMII_RX_ER	I	RMII receive error	RMII receive error input
	RMII_RXD0	I	RMII receive data	2-bit RMII receive data
	RMII_RXD1	I	RMII receive data	2-bit RMII receive data
	RMII_TX_EN	O	RMII transmit enable	RMII transmit enable signal
	RMII_TXD0	O	RMII transmit data	2-bit RMII transmit data
	RMII_TXD1	O	RMII transmit data	2-bit RMII transmit data

Classification	Symbol	I/O	Function	Description
MMC I/F (MMC)	MMC_CLK	O	Clock output	Clock output pin
	MMC_CMD	I/O	Command output/response input	Command output/response input pin (in MMC mode)
	MMC_D7 to 0	I/O	Data output/response input	Data output/response input pin (in MMC mode)
Video I/O (VIO)	VIO0_D15 to VIO0_D0, VIO1_D7 to VIO1_D0	I	VIO data bus	VIO camera image data input
	VIO0_CLK, VIO1_CLK,	I	VIO clock	VIO camera clock input
	VIO0_VD, VIO1_VD	I	VIO vertical sync	VIO camera vertical sync signals
	VIO0_HD, VIO1_HD	I	VIO horizontal sync	VIO camera horizontal sync signals
	VIO0_FLD, VIO1_FLD	I	Field signal	Field identification signals
	VIO_CKO	O	Camera clock output	Clock output to camera
LCD controller (RGB interface)	LCDD23 to LCDD0	O	LCD data bus	24-bit data for LCD panel
	LCDDON	O	Display ON/OFF signal	Display ON/OFF signal
	LCDHSYN	O	Horizontal sync signal	Horizontal sync signal
	LCDDISP	O	Display enable signal	Display enable signal
	LCDVSYN	O	Vertical sync signal	Vertical sync signal
	LCDVCPWC	O	Power supply control	LCD module power supply control signal
	LCDVEPWC	O	Power supply control	LCD module power supply control signal
	LCDDCK	O	Dot clock signal	Data synchronizing signal
	LCDLCLK	I	Input clock	Input clock signal

Classification	Symbol	I/O	Function	Description
LCD controller (SYS interface)	LCDD23 to LCDD0	I/O	Data	Data input/output
	LCDDON	O	Display ON/OFF signal	Display ON/OFF signal
	$\overline{\text{LCDCS}}$	O	Chip select	Chip select signal
	$\overline{\text{LCDRD}}$	O	Read strobe	Read strobe signal
	LCDRS	O	Register select	Register select signal
	LCDVSYN	I/O	Vertical sync signal	Vertical sync signal
	LCDVCPWC	O	Power supply control	LCD module power supply control signal
	LCDVEPWC	O	Power supply control	LCD module power supply control signal
	$\overline{\text{LCDWR}}$	O	Write strobe	Write strobe signal
	LCDLCLK	I	Input clock	Input clock signal
Video output unit (VOU)	DV_D15 to 0	O	Data output	Data output
	DV_CLK	O	Clock output	Pixel clock output
	DV_VSYNC	O	Vertical sync signal output	Vertical sync signal output from VOU
	DV_HSYNC	O	Horizontal sync signal output	Horizontal sync signal output from VOU
	DV_CLKI	I	Video clock input	Video clock input pin
TS interface (TSIF)	TS_SCK	I	Clock	TS input clock
	TS_SDAT	I	Receive data	TS serial data
	TS_SDEN	I	Data enable	TS data enable signal
	TS_SPSYNC	I	Data sync signal	TS data sync signal
USB2.0 host and function module (USB)	VBUS1, VBUS0	I	USB power sense	USB cable connection monitoring pins
	DP1, DP0	I/O	D+ input/output	USB internal transceiver D+ input/output
	DM1, DM0	I/O	D- input/output	USB internal transceiver D- input/output
	REFRIN1, REFRIN0	I	Reference input	Pins for connecting reference resistors for constant-current circuits. Pull down to AG33.

Classification	Symbol	I/O	Function	Description
Sound interface unit (FSI)	FSIOALRCK, FSIOBLRCK	I/O	Sound output L/R clock	Sound output L/R clock pins (master or slave)
	FSIOABCK, FSIOBBCK	I/O	Sound output bit clock	Sound output bit clock pins (master or slave)
	FSIOASD, FSIOBSD	O	Sound output serial data	Sound output serial data pins
	FSIIALRCK, FSIIBLRCK	I/O	Sound input L/R clock	Sound input L/R clock pins (master or slave)
	FSIIABCK, FSIIBBCK	I/O	Sound input bit clock	Sound input bit clock pins (master or slave)
	FSIIASD, FSIIBSD	I	Sound input	Sound input serial data pins
	CLKAUDIOAO, CLKAUDIOBO	O	Audio clock	FSI basic operating clock
ATAPI interface (ATAPI)	IDED15 to 0	I/O	Data bus	16-bit bidirectional bus
	IDEA2 to 0	O	Address bus	Address bus
	IDEINT	I	Interrupt request	Primary channel interrupt request
	$\overline{\text{IDEIOWR}}$	O	WR enable	Primary channel write
	$\overline{\text{IDEIORD}}$	O	RD enable	Primary channel read
	$\overline{\text{IDECS0}}$, $\overline{\text{IDECS1}}$	O	Chip select	Primary channel chip select
	$\overline{\text{IODACK}}$	O	DMA acknowledge	Primary channel DMA acknowledge
	IODREQ	I	DMA request	Primary channel DMA request
	IDEIORDY	I	Ready signal	Primary channel ready signal
	$\overline{\text{IDERST}}$	O	Reset	Primary channel ATAPI device reset
	$\overline{\text{EXBUF\_ENB}}$	O	External data enable	External level shifter enable signal
	DIRECTION	O	External data direction	External level shifter direction signal
Key scan interface (KEYSC)	KEYIN6 to 0	I	Key input	Key input
	KEYOUT5 to 0	O	Key output	Key output
I/O port	PTA to PTZ	I/O I O	General I/O port	General I/O port pins

Classification	Symbol	I/O	Function	Description
SD host interface (SDHI0/ SDHI1)	$\overline{\text{SDHI0CD}}$, $\overline{\text{SDHI1CD}}$	I	Card detection	SD card detection signals
	SDHI0WP, SDHI1WP	I	Write protection	SD write-protection signals
	SDHI0D3 to 0, SDHI1D3 to 0	I/O	Data bus	SD data bus signals
	SDHI0CMD, SDHI1CMD	I/O	Command output and response input	SD command output and response input signals
	SDHI0CLK, SDHI1CLK	O	Clock	SD clock output pins
User debugging interface (H-UDI) *	TCK	I	Test clock	Test clock input pin
	TMS	I	Test mode select	Test mode select signal input pin
	TDI	I	Test data input	Serial instruction and data input pin
	TDO	O	Test data output	Serial instruction and data output pin
	$\overline{\text{TRST}}$	I	Test reset	H-UDI reset pin. When the emulator is not used, fix it at a low level.
	$\overline{\text{ASEBRK}}$/BRKACK	I/O	Break input/acknowledge	Break signal input from emulator/break acknowledge output signal
	MPMD	I	ASE mode	ASE mode (emulation support mode) setting pin
Advanced user debugger (AUD)	AUDATA3 to AUDATA0	O	AUD data	Branch destination address output pin in branch trace mode
	AUDCK	O	AUD clock	Sync clock output pin in branch trace mode
	AUDSYNC	O	AUD sync signal	Data start position recognition signal output pin in branch trace mode

Note: * For the pin functions when using an emulator, refer to the emulator manual.

1.7 Internal Bus Configuration

This LSI has adopted the SuperHyway bus (hereafter referred to as SHwy bus) as the internal system bus. The SHwy bus is a system bus that consists of a 32-bit address line, a 64-bit data line, and a bus clock (Sϕ) with a maximum frequency of 166.7 MHz. Data can be transferred at high speed between the modules connected to the SHwy bus. Data transfer is controlled by the SuperHyway packet router.

The SHwy bus and modules are connected with the following bus bridges.

- ICB (Inter Connect Buffer)

  This bridge connects the SHwy bus to a group of image modules. This bridge has 128-Kbyte SRAM (MERAM: Media RAM), and can be used as the external memory access buffer (ICB: Inter Connect Buffer) to reduce the external memory accesses and improve its efficiency. For details on MERAM, refer to section 41, Media RAM (MERAM).
- HPB (Peripheral bus bridge)

  This bridge connects the SHwy bus to a group of peripheral modules that operate on Pϕ.
- HPB2 (Peripheral bus bridge -2)

  This bridge connects the SHwy bus to a group of peripheral modules that operate on Bϕ.

In data transfer, each module operates as an initiator (SHwy bus master) that sends requests or a target that returns responses. The modules that can operate as an initiator are the SH-X2, DBG, DMAC, ICB, 2DG, ATAPI, EtherMAC, and SPU. The LRU (Least Recently Used) method is used to determine the priority of the requests. The initial state of priority is ICB > DBG > SH-X2 > 2DG > LMB. However, the priority setting needs to be changed according to the modules used and their usage methods. For details, refer to section 12, SuperHyway Packet Router.

1.8 Memory Address Map

The CPU incorporated in this LSI supports a 32-bit virtual address space. Virtual addresses can be allocated to any physical addresses by the MMU. For details, refer to section 7, Memory Management Unit (MMU).

Figure 1.4 shows the physical address space of this LSI. This LSI supports a 29-bit address mode (normal mode) and a 32-bit address mode (expansion mode).

At a power-on reset or a system reset, the 29-bit address mode is selected. Use the SE bit in PASCR to switch to the 32-bit address mode. For the setting method, refer to section 7, Memory Management Unit (MMU). When using the 32-bit address mode, the module stop bit MSTP031 for the TLB must be cleared to 0 (TLB operates).

Areas 2 and 3 in the physical address space are dedicated to the DBSC. The DBSC and BSC can be selected as the bus state controller for areas 4 and 5 by the AREASEL bits in MMSELR. Figure 1.5 shows the correspondence between settings of the AREASEL bits and the address map. For details on the settings, refer to section 14, Bus State Controller (BSC). In the 32-bit address mode, physical addresses from H'40000000 to H'5FFFFFFF can be used as a DRAM expansion space.

Note that access that is not from the SH-4A core does not pass through the MMU and so will be access to physical addresses. Physical addresses should be set as the addresses for the registers of each module unless stated otherwise. Specify 29-bit physical addresses in the 29-bit address mode.

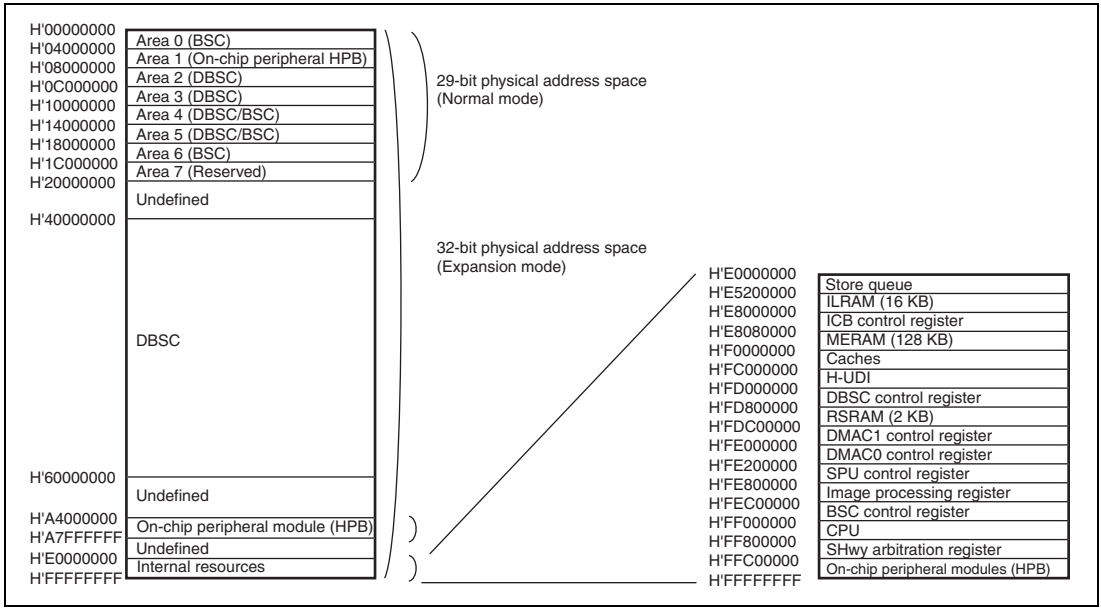


Figure 1.4 Physical Address Space

	MMSELR.AREASEL	0	1
H'00000000 to H'03FFFFFF	Area 0	BSC	BSC
H'04000000 to H'07FFFFFF	Area 1 (HPB, On-chip peripheral module)	HPB	HPB
H'08000000 to H'0BFFFFFF	Area 2	DBSC-2	DBSC-2
H'0C000000 to H'0FFFFFFF	Area 3	DBSC-3	DBSC-3
H'10000000 to H'13FFFFFF	Area 4	DBSC-4	BSC
H'14000000 to H'17FFFFFF	Area 5	DBSC-5	BSC
H'18000000 to H'1BFFFFFF	Area 6	BSC	BSC
H'1C000000 to H'1FFFFFFF	Area 7 (Reserved)		
H'20000000 to H'3FFFFFFF	Undefined		
H'40000000 to H'43FFFFFF	DBSC (DDR2-SDRAM / Mobile DDR)	DBSC-0	DBSC-0
H'44000000 to H'47FFFFFF		DBSC-1	DBSC-1
H'48000000 to H'4BFFFFFF		DBSC-2	DBSC-2
H'4C000000 to H'4FFFFFFF		DBSC-3	DBSC-3
H'50000000 to H'53FFFFFF		DBSC-4	DBSC-4
H'54000000 to H'57FFFFFF		DBSC-5	DBSC-5
H'58000000 to H'5BFFFFFF		DBSC-6	DBSC-6
H'5C000000 to H'5FFFFFFF		DBSC-7	DBSC-7
H'60000000 to H'7FFFFFFF	Undefined		
H'80000000 to H'A3FFFFFF	Undefined		
H'A4000000 to H'A7FFFFFF	HPB (On-chip peripheral module)	HPB	HPB
H'A8000000 to H'BFFFFFFF	Undefined		
H'C0000000 to H'DFFFFFFF	Undefined		
H'E0000000 to H'FFFFFFFF	Internal resources		

29-bit physical address space (Normal mode)

32-bit physical address space (Expansion mode)

[Legend]

: Shadow space

Figure 1.5 Memory Map Selection by AREASEL Bit

1.9 Clocks

An overview of the SH7724 clocks is given blow. For details, refer to section 17, Clock Pulse Generator (CPG). Table 1.5 lists the clocks available in the SH7724.

Table 1.5 Clocks

Classification	Clock Name	Use Range	Generated by	Clock Source
System clock	Iϕ	SH-X2	CPG (PLL)	External input (EXTAL), crystal oscillator (EXTAL, XTAL), external input (RTC_CLK) Note: The frequency must be multiplied by the FLL circuit to generate the system clock from RTC_CLK.
	Sϕ	L2 cache, SHwy, DBSC	CPG (PLL)	
	Bϕ	BSC, modules connected to SHwy, HPB2, modules connected to HPB2	CPG (PLL)	
	Pϕ	HPB, modules connected to HPB	CPG (PLL)	
Individual module clock	M1ϕ	VPU	CPG (PLL)	
	SPUCK	SPU	CPG (PLL)	
	CKO	Clock output for BSC interface	CPG (PLL)	
	MCLK	Clock output for DBSC interface	CPG (PLL)	
	VIO_CKO	Clock output for camera	CPG (PLL)	
	IrDACK	Clock for IrDA interface	CPG (PLL)	
	RCLK	RWDT, CMT counting clock	CPG	
	FSICKA, FSICKB	Clock for FSI interface	CPG (PLL)	Above three clock sources, external input (FSIMCKA, FSIMCKB)
	RTC_CLK	RTC counting clock	External input	External input (RTC_CLK)
	LCDDCK	LCDC dot clock output	LCDC	External input (LCDLCLK), Bϕ, Pϕ
	DV_CLK	Clock output for VOU interface	VOU	External input (DV_CLKI)
	FSIOABCK, FSIOALRCK, FSIOBBCK, FSIOBLRCK	Clock output for FSI interface	FSI	FSICKA, FSICKB
	CLKAUDIOAO, CLKAUDIOBO	FSI basic operating clock output	FSI	FSICKA, FSICKB

Classification	Clock Name	Use Range	Generated by	Clock Source
Individual module clock	MSIOF0_TSCK, MSIOF1_TSCK	Serial clock output for MSIOF transmission	MSIOF	Bϕ, external input (MSIOF0_MCK, MSIOF1_MCK)
	SCIF0_SCK, SCIF1_SCK, SCIF2_SCK	SCIF serial clock output	SCIF	Pϕ
	SCIF3_SCK, SCIF4_SCK, SCIF5_SCK	SCIFA serial clock output	SCIFA	Bϕ
	MMC_CLK	Clock output for MMC interface	MMC	Bϕ
	SDHI0CLK, SDHI1CLK	Clock output for SDHI interface	SDHI	Bϕ
	USB_CLK	USB operating clock	USB (PLL)	External input (EXTAL_USB), crystal oscillator (EXTAL_USB, XTAL_USB)

1.10 Operating Modes

The SH7724 has three processing states: the reset state, program execution state, and power-down state. The reset states and power-down states are outlined below.

1.10.1 Reset

The SH7724 can be reset by a power-on reset, system reset, manual reset, and software reset.

(1) Power-On Reset

Inputting a low-level signal to the $\overline{\text{RESETP}}$ pin halts all processings currently being executed and starts the LSI using the power-on sequence. A power-on reset should be used for resetting the LSI when the power is turned on.

(2) System Reset

A system reset occurs when a low-level signal is input to the $\overline{\text{RESETA}}$ pin or the RWDT overflows.

(3) Manual Reset

A manual reset occurs if an exception other than a user break is generated when the BL bit in SR is 1. For details, refer to section 5, Exception Handling.

(4) Software Reset

Some modules can be reset individually by modifying the respective register setting. For details, refer to the section for each module.

1.10.2 Power-Down Modes

The power-down states available in this LSI are sleep mode, software standby mode, R-standby mode, U-standby mode, and module standby state. For details, refer to section 18, Reset and Power-Down Modes.

(1) Sleep Mode

In sleep mode, the clock provided to the SH-4A core is halted. The on-chip peripheral modules connected to the SHwy bus continue to operate and external memory accesses are possible. The program execution state can be returned to by an interrupt.

(2) Software Standby Mode

In software standby mode, clocks except for RCLK are halted. The program execution state can be returned to by an interrupt.

(3) R-Standby Mode

In R-standby mode, clocks except for RCLK are halted and the power supply is cut off for most of the areas except for the I/O areas. The contents of some registers and memory (RS memory) are held. By storing an interrupt processing program in the RS memory, the program execution state can be returned to by an interrupt.

(4) U-Standby Mode

In U-standby mode, the power supply is cut off for the modules that operate on RCLK and the areas except for the I/O areas. The current consumption can be reduced the most in this mode. Though the pin states are held, the internal state is not. Therefore, to cancel this mode, this LSI has to be restarted by a reset.

(5) Module Standby State

In the module standby state, the clock supply to modules can be individually halted with the CPU continuing to operate. The current consumption can be reduced by stopping the clocks of modules not being used.

1.11 Power Supply Sequence

The sequences for turning on and off the I/O power supplies (VccQ, VccQ1, VccQ_LCD, VccQ_MMC, VccQ_SDC, VccQ_SR, VccQ_VIO, DV33, AV33, and VccQ_DDR) and 1.2 V-power supplies (VDD, VDD_PLL, VDD_FLL, DV12, AV12, and UV12) are described. For details, refer to section 52, Electrical Characteristics.

(1) Sequence for Turning on Power

Set the power supply voltage provided to VccQ so as to satisfy the following condition:

VccQ + 0.3 V ≥ VccQ1, VccQ_LCD, VccQ_MMC, VccQ_SDC, VccQ_SR, VccQ_VIO, DV33, AV33

Follow the power-on sequence below to start this LSI.

1. First, turn on the VccQ power supply. At power-on, the $\overline{\text{RESETP}}$ pin must be driven low to be V_{IL} or less.
2. Wait until the VccQ voltage becomes equal to or greater than the minimum value specified in section 52, Electrical Characteristics.
3. Turn on the I/O power supplies except for VccQ. They can be turned on in any order (random order).
4. Turn on the 1.2 V-power supplies.
5. After all power supplies have been turned on, clear the $\overline{\text{RESETP}}$ pin.

(2) Sequence for Turning off Power

Follow the sequence below to turn off the power supplies.

1. Turn off the 1.2 V-power supplies.
2. Turn off the I/O power supplies except for VccQ. They can be turned off in any order (random order).
3. Finally, turn off the VccQ power supply.

1.12 Notes on Board Design

(1) Note on Bypass Capacitor

A multilayer ceramic capacitor should be inserted for each pair of power supply and ground as a bypass capacitor.

The bypass capacitor must be inserted close to the power supply pins of the LSI. Note that the capacitance and frequency characteristics of the bypass capacitor must be appropriate for the operating frequency of the LSI.

(2) Note on Power Supply Wiring of PLL and FLL

Since the power supply of the PLL and FLL circuits are sensitive to noise, wiring patterns should be designed in accordance with the notes in section 17, Clock Pulse Generator (CPG).

(3) Note on DDR Interface Board Design

A DDR installation guide is available. Use it as reference material when designing your board.

Section 2 Programming Model

The programming model of this LSI is explained in this section. This LSI has registers and data formats as shown below.

2.1 Data Formats

The data formats supported in this LSI are shown in figure 2.1.

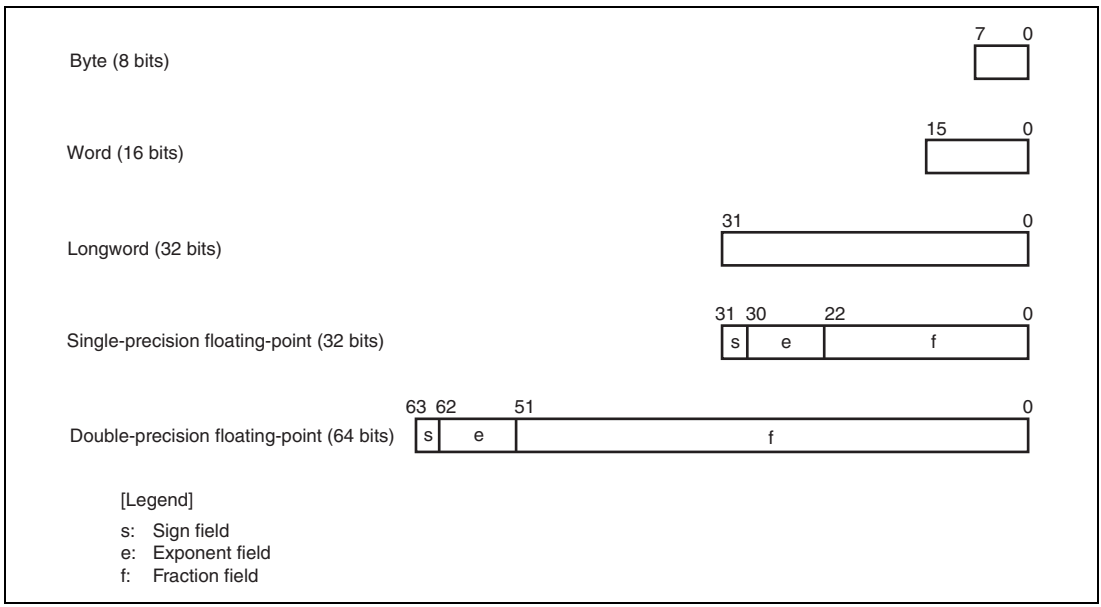


Figure 2.1 Data Formats

2.2 Register Descriptions

2.2.1 Privileged Mode and Banks

(1) Processing Modes

This LSI has two processing modes, user mode and privileged mode. This LSI normally operates in user mode, and switches to privileged mode when an exception occurs or an interrupt is accepted. There are four kinds of registers—general registers, system registers, control registers, and floating-point registers—and the registers that can be accessed differ in the two processing modes.

(2) General Registers

There are 16 general registers, designated R0 to R15. General registers R0 to R7 are banked registers which are switched by a processing mode change.

- Privileged mode

  In privileged mode, the register bank bit (RB) in the status register (SR) defines which banked register set is accessed as general registers, and which set is accessed only through the load control register (LDC) and store control register (STC) instructions.

  When the RB bit is 1 (that is, when bank 1 is selected), the 16 registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. In this case, the eight registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 are accessed by the LDC/STC instructions. When the RB bit is 0 (that is, when bank 0 is selected), the 16 registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. In this case, the eight registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 are accessed by the LDC/STC instructions.
- User mode

  In user mode, the 16 registers comprising bank 0 general registers R0_BANK0 to R7_BANK0 and non-banked general registers R8 to R15 can be accessed as general registers R0 to R15. The eight registers comprising bank 1 general registers R0_BANK1 to R7_BANK1 cannot be accessed.

(3) Control Registers

Control registers comprise the global base register (GBR) and status register (SR), which can be accessed in both processing modes, and the saved status register (SSR), saved program counter (SPC), vector base register (VBR), saved general register 15 (SGR), and debug base register (DBR), which can only be accessed in privileged mode. Some bits of the status register (such as the RB bit) can only be accessed in privileged mode.

(4) System Registers

System registers comprise the multiply-and-accumulate registers (MACH/MACL), the procedure register (PR), and the program counter (PC). Access to these registers does not depend on the processing mode.

(5) Floating-Point Registers and System Registers Related to FPU

There are thirty-two floating-point registers, FR0–FR15 and XF0–XF15. FR0–FR15 and XF0–XF15 can be assigned to either of two banks (FPR0_BANK0–FPR15_BANK0 or FPR0_BANK1–FPR15_BANK1).

FR0–FR15 can be used as the eight registers DR0/2/4/6/8/10/12/14 (double-precision floating-point registers, or pair registers) or the four registers FV0/4/8/12 (register vectors), while XF0–XF15 can be used as the eight registers XD0/2/4/6/8/10/12/14 (register pairs) or register matrix XMTRX.

System registers related to the FPU comprise the floating-point communication register (FPUL) and the floating-point status/control register (FPSCR). These registers are used for communication between the FPU and the CPU, and the exception handling setting.

Register values after a reset are shown in table 2.1.

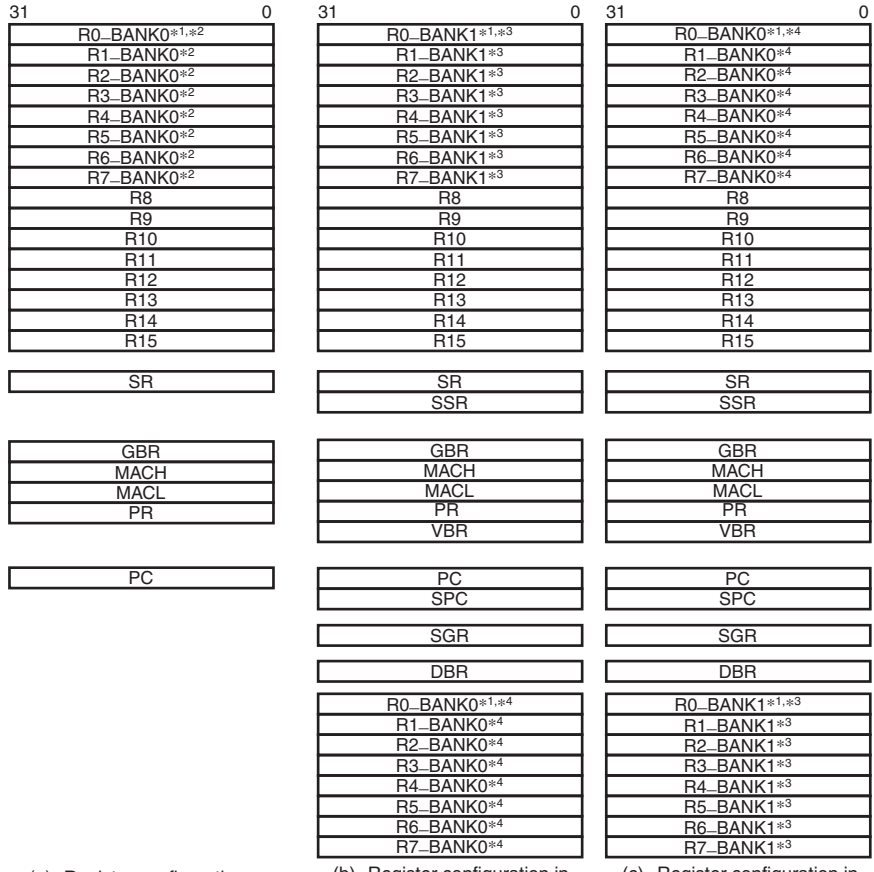
Table 2.1 Initial Register Values

Type	Registers	Initial Value*
General registers	R0_BANK0 to R7_BANK0, R0_BANK1 to R7_BANK1, R8 to R15	Undefined
Control registers	SR	MD bit = 1, RB bit = 1, BL bit = 1, FD bit = 0, IMASK = B'1111, reserved bits = 0, others = undefined
	GBR, SSR, SPC, SGR, DBR	Undefined
	VBR	H'00000000
System registers	MACH, MACL, PR	Undefined
	PC	H'A0000000
Floating-point registers	FR0 to FR15, XF0 to XF15, FPUL	Undefined
	FPSCR	H'00040001

Note: * Initialized by a power-on reset and manual reset.

The CPU register configuration in each processing mode is shown in figure 2.2.

User mode and privileged mode are switched by the processing mode bit (MD) in the status register.



(a) Register configuration in user mode

(b) Register configuration in privileged mode (RB = 1)

(c) Register configuration in privileged mode (RB = 0)

Notes: 1. R0 is used as the index register in indexed register-indirect addressing mode and indexed GBR indirect addressing mode.
2. Banked registers
3. Banked registers
Accessed as general registers when the RB bit is set to 1 in SR. Accessed only by LDC/STC instructions when the RB bit is cleared to 0.
4. Banked registers
Accessed as general registers when the RB bit is cleared to 0 in SR. Accessed only by LDC/STC instructions when the RB bit is set to 1.

Figure 2.2 CPU Register Configuration in Each Processing Mode

2.2.2 General Registers

Figure 2.3 shows the relationship between the processing modes and general registers. This LSI has twenty-four 32-bit general registers (R0_BANK0 to R7_BANK0, R0_BANK1 to R7_BANK1, and R8 to R15). However, only 16 of these can be accessed as general registers R0 to R15 in one processing mode. This LSI has two processing modes, user mode and privileged mode.

- R0_BANK0 to R7_BANK0
  Allocated to R0 to R7 in user mode (SR.MD = 0)
  Allocated to R0 to R7 when SR.RB = 0 in privileged mode (SR.MD = 1).
- R0_BANK1 to R7_BANK1
  Cannot be accessed in user mode.
  Allocated to R0 to R7 when SR.RB = 1 in privileged mode.

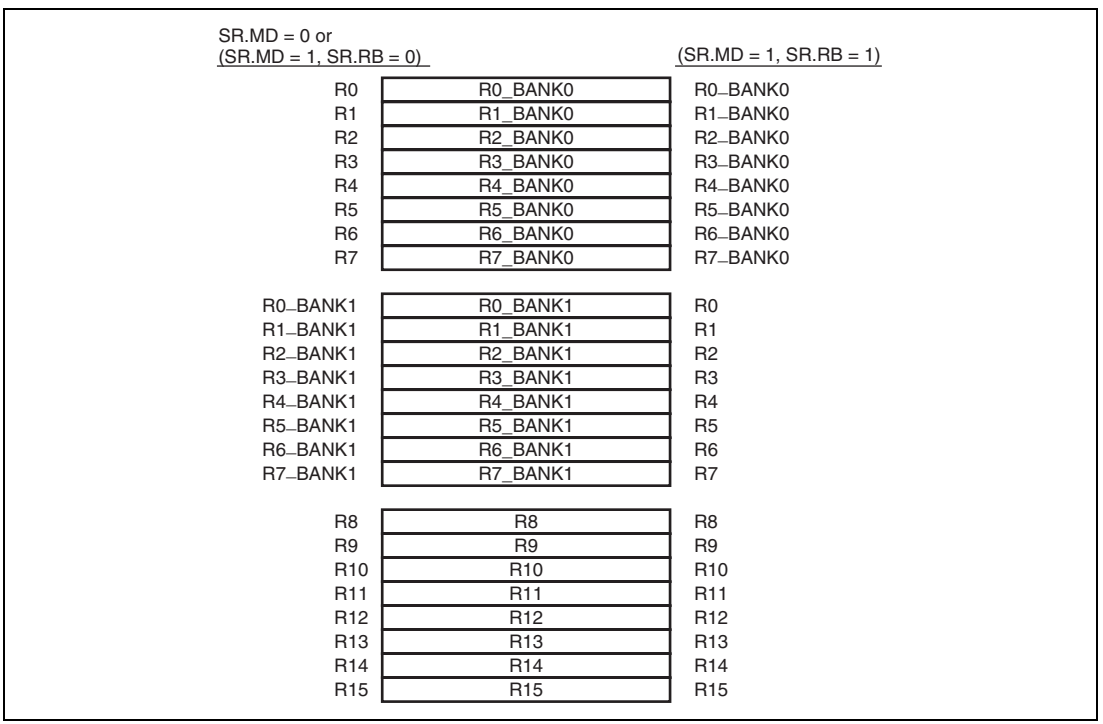


Figure 2.3 General Registers

Note on Programming: As the user's R0 to R7 are assigned to R0_BANK0 to R7_BANK0, and after an exception or interrupt R0 to R7 are assigned to R0_BANK1 to R7_BANK1, it is not necessary for the interrupt handler to save and restore the user's R0 to R7 (R0_BANK0 to R7_BANK0).

2.2.3 Floating-Point Registers

Figure 2.4 shows the floating-point register configuration. There are thirty-two 32-bit floating-point registers, FPR0_BANK0 to FPR15_BANK0, AND FPR0_BANK1 to FPR15_BANK1, comprising two banks. These registers are referenced as FR0 to FR15, DR0/2/4/6/8/10/12/14, FV0/4/8/12, XF0 to XF15, XD0/2/4/6/8/10/12/14, or XMTRX. Reference names of each register are defined depending on the state of the FR bit in FPSCR (see figure 2.4).

1. Floating-point registers, FPRn_BANKj (32 registers)
   FPR0_BANK0 to FPR15_BANK0
   FPR0_BANK1 to FPR15_BANK1

2. Single-precision floating-point registers, FRi (16 registers)
   When FPSCR.FR = 0, FR0 to FR15 are assigned to FPR0_BANK0 to FPR15_BANK0;
   when FPSCR.FR = 1, FR0 to FR15 are assigned to FPR0_BANK1 to FPR15_BANK1.

3. Double-precision floating-point registers or single-precision floating-point registers, DRi (8 registers): A DR register comprises two FR registers.
   DR0 = {FR0, FR1}, DR2 = {FR2, FR3}, DR4 = {FR4, FR5}, DR6 = {FR6, FR7},
   DR8 = {FR8, FR9}, DR10 = {FR10, FR11}, DR12 = {FR12, FR13}, DR14 = {FR14, FR15}

4. Single-precision floating-point vector registers, FVi (4 registers): An FV register comprises four FR registers.
   FV0 = {FR0, FR1, FR2, FR3}, FV4 = {FR4, FR5, FR6, FR7},
   FV8 = {FR8, FR9, FR10, FR11}, FV12 = {FR12, FR13, FR14, FR15}

5. Single-precision floating-point extended registers, XFi (16 registers)
   When FPSCR.FR = 0, XF0 to XF15 are assigned to FPR0_BANK1 to FPR15_BANK1;
   when FPSCR.FR = 1, XF0 to XF15 are assigned to FPR0_BANK0 to FPR15_BANK0.

6. Double-precision floating-point extended registers, XDi (8 registers): An XD register comprises two XF registers.
   XD0 = {XF0, XF1}, XD2 = {XF2, XF3}, XD4 = {XF4, XF5}, XD6 = {XF6, XF7},
   XD8 = {XF8, XF9}, XD10 = {XF10, XF11}, XD12 = {XF12, XF13}, XD14 = {XF14, XF15}

7. Single-precision floating-point extended register matrix, XMTRX: XMTRX comprises all 16 XF registers.

$$XMTRX = \begin{bmatrix} XF0 & XF4 & XF8 & XF12 \\ XF1 & XF5 & XF9 & XF13 \\ XF2 & XF6 & XF10 & XF14 \\ XF3 & XF7 & XF11 & XF15 \end{bmatrix}$$

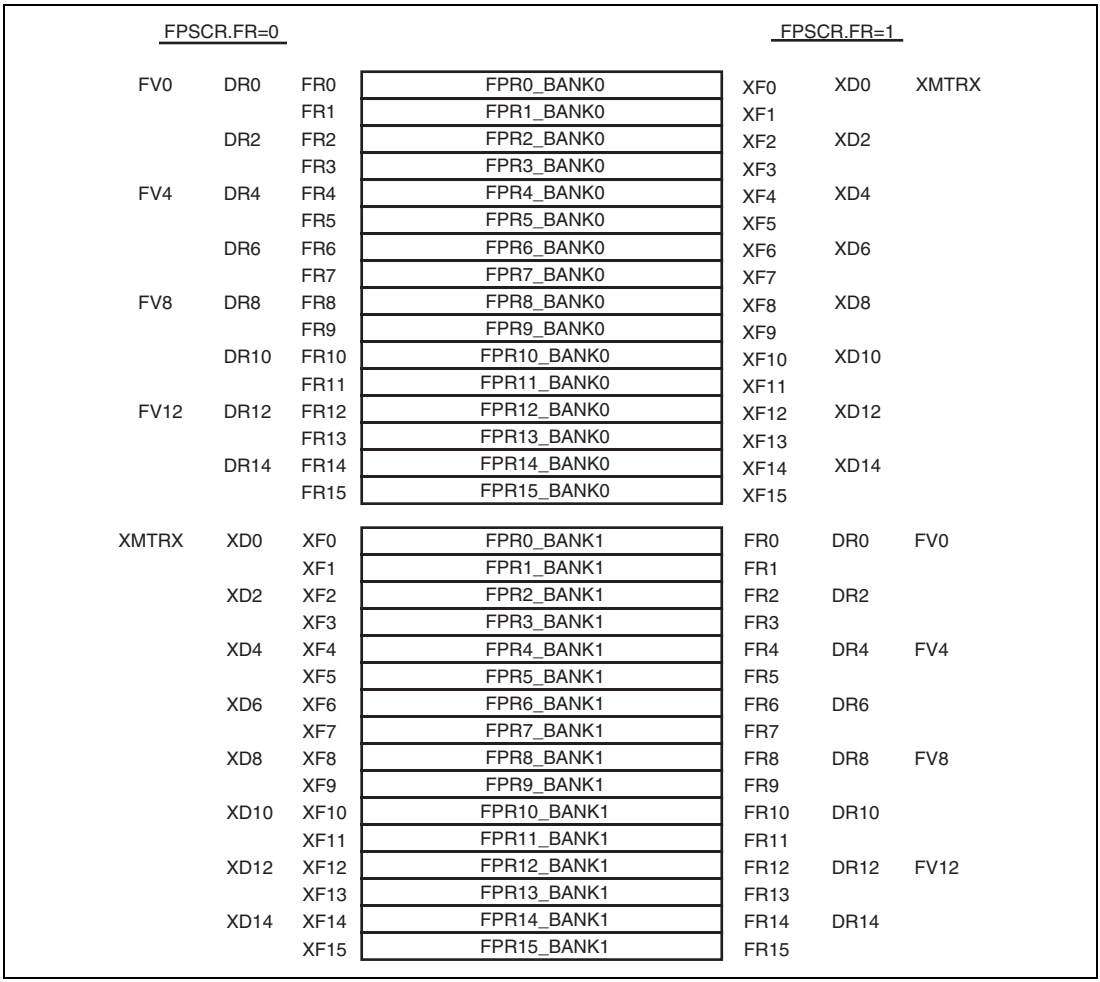


Figure 2.4 Floating-Point Registers

2.2.4 Control Registers

(1) Status Register (SR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MD	RB	BL	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FD	—	—	—	—	—	M	Q	IMASK				—	—	S	T
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
30	MD	1	R/W	Processing Mode Selects the processing mode. 0: User mode (Some instructions cannot be executed and some resources cannot be accessed.) 1: Privileged mode This bit is set to 1 by an exception or interrupt.
29	RB	1	R/W	Privileged Mode General Register Bank Specification Bit 0: R0_BANK0 to R7_BANK0 are accessed as general registers R0 to R7 and R0_BANK1 to R7_BANK1 can be accessed using LDC/STC instructions 1: R0_BANK1 to R7_BANK1 are accessed as general registers R0 to R7 and R0_BANK0–R7_BANK0 can be accessed using LDC/STC instructions This bit is set to 1 by an exception or interrupt.
28	BL	1	R/W	Exception/Interrupt Block Bit This bit is set to 1 by a reset, a general exception, or an interrupt. While this bit is set to 1, an interrupt request is masked. In this case, this processor enters the reset state when a general exception other than a user break occurs.
27 to 16	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
15	FD	0	R/W	FPU Disable Bit When this bit is set to 1 and an FPU instruction is not in a delay slot, a general FPU disable exception occurs. When this bit is set to 1 and an FPU instruction is in a delay slot, a slot FPU disable exception occurs. (FPU instructions: H'F*** instructions and LDS (.L)/STS(.L) instructions using FPUL/FPSCR)
14 to 10	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
9	M	0	R/W	M Bit Used by the DIV0S, DIV0U, and DIV1 instructions.
8	Q	0	R/W	Q Bit Used by the DIV0S, DIV0U, and DIV1 instructions.
7 to 4	IMASK	1111	R/W	Interrupt Mask Level Bits An interrupt whose priority is equal to or less than the value of the IMASK bits is masked. It can be chosen by CPU operation mode register (CPUOPM) whether the level of IMASK is changed to accept an interrupt or not when an interrupt is occurred. For details, see appendix A, CPU Operation Mode Register (CPUOPM).
3, 2	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
1	S	0	R/W	S Bit Used by the MAC instruction.
0	T	0	R/W	T Bit Indicates true/false condition, carry/borrow, or overflow/underflow. For details, see section 3, Instruction Set.

(2) Saved Status Register (SSR) (32 bits, Privileged Mode, Initial Value = Undefined)

The contents of SR are saved to SSR in the event of an exception or interrupt.

(3) Saved Program Counter (SPC) (32 bits, Privileged Mode, Initial Value = Undefined)

The address of an instruction at which an interrupt or exception occurs is saved to SPC.

(4) Global Base Register (GBR) (32 bits, Initial Value = Undefined)

GBR is referenced as the base address of addressing @(disp,GBR) and @(R0,GBR).

(5) Vector Base Register (VBR) (32 bits, Privileged Mode, Initial Value = H'00000000)

VBR is referenced as the branch destination base address in the event of an exception or interrupt. For details, see section 5, Exception Handling.

(6) Saved General Register 15 (SGR) (32 bits, Privileged Mode, Initial Value = Undefined)

The contents of R15 are saved to SGR in the event of an exception or interrupt.

(7) Debug Base Register (DBR) (32 bits, Privileged Mode, Initial Value = Undefined)

When the user break debugging function is enabled (CBCR.UBDE = 1), DBR is referenced as the branch destination address of the user break handler instead of VBR.

2.2.5 System Registers

(1) Multiply-and-Accumulate Registers (MACH and MACL) (32 bits, Initial Value = Undefined)

MACH and MACL are used for the added value in a MAC instruction, and to store the operation result of a MAC or MUL instruction.

(2) Procedure Register (PR) (32 bits, Initial Value = Undefined)

The return address is stored in PR in a subroutine call using a BSR, BSRF, or JSR instruction. PR is referenced by the subroutine return instruction (RTS).

(3) Program Counter (PC) (32 bits, Initial Value = H'A0000000)

PC indicates the address of the instruction currently being executed.

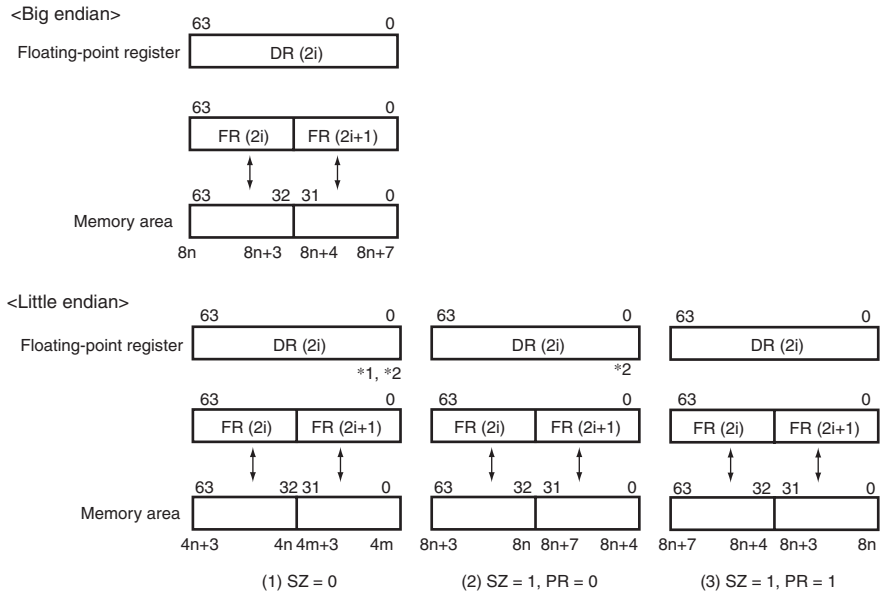
(4) Floating-Point Status/Control Register (FPSCR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FR	SZ	PR	DN	Cause	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable (EN)					Flag					RM	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
21	FR	0	R/W	Floating-Point Register Bank 0: FPR0_BANK0 to FPR15_BANK0 are assigned to FR0 to FR15 and FPR0_BANK1 to FPR15_BANK1 are assigned to XF0 to XF15 1: FPR0_BANK0 to FPR15_BANK0 are assigned to XF0 to XF15 and FPR0_BANK1 to FPR15_BANK1 are assigned to FR0 to FR15
20	SZ	0	R/W	Transfer Size Mode 0: Data size of FMOV instruction is 32-bits 1: Data size of FMOV instruction is a 32-bit register pair (64 bits) For relationship between the SZ bit, PR bit, and endian, see figure 2.5.
19	PR	0	R/W	Precision Mode 0: Floating-point instructions are executed as single-precision operations 1: Floating-point instructions are executed as double-precision operations (graphics support instructions are undefined) For relationship between the SZ bit, PR bit, and endian, see figure 2.5
18	DN	1	R/W	Denormalization Mode 0: Denormalized number is treated as such 1: Denormalized number is treated as zero

Bit	Bit Name	Initial Value	R/W	Description
17 to 12	Cause	000000	R/W	FPU Exception Cause Field FPU Exception Enable Field FPU Exception Flag Field Each time an FPU operation instruction is executed, the FPU exception cause field is cleared to 0. When an FPU exception occurs, the bits corresponding to FPU exception cause field and flag field are set to 1. The FPU exception flag field remains set to 1 until it is cleared to 0 by software. For bit allocations of each field, see table 2.2.
11 to 7	Enable (EN)	00000	R/W	
6 to 2	Flag	00000	R/W	
1, 0	RM	01	R/W	Rounding Mode These bits select the rounding mode. 00: Round to Nearest 01: Round to Zero 10: Reserved 11: Reserved



Notes: 1. In the case of SZ = 0 and PR = 0, DR register can not be used.

2. The bit-location of DR register is used for double precision format when PR = 1.
(In the case of (2), it is used when PR is changed from 0 to 1.)

Figure 2.5 Relationship between SZ bit and Endian

Table 2.2 Bit Allocation for FPU Exception Handling

Field Name		FPU Error (E)	Invalid Operation (V)	Division by Zero (Z)	Overflow (O)	Underflow (U)	Inexact (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

(5) Floating-Point Communication Register (FPUL) (32 bits, Initial Value = Undefined)

Information is transferred between the FPU and CPU via FPUL.

2.3 Memory-Mapped Registers

Some control registers are mapped to the following memory areas. Each of the mapped registers has two addresses.

H'1C00 0000 to H'1FFF FFFF
H'FC00 0000 to H'FFFF FFFF

These two areas are used as follows.

- H'1C00 0000 to H'1FFF FFFF

  This area must be accessed using the address translation function of the MMU.

  Setting the page number of this area to the corresponding field of the TLB enables access to a memory-mapped register.

  The operation of an access to this area without using the address translation function of the MMU is not guaranteed.
- H'FC00 0000 to H'FFFF FFFF

  Access to area H'FC00 0000 to H'FFFF FFFF in user mode will cause an address error. Memory-mapped registers can be referenced in user mode by means of access that involves address translation.

Note: Do not access addresses to which registers are not mapped in either area. The operation of an access to an address with no register mapped is undefined. Also, memory-mapped registers must be accessed using a fixed data size. The operation of an access using an invalid data size is undefined.

2.4 Data Formats in Registers

Register operands are always longwords (32 bits). When a memory operand is only a byte (8 bits) or a word (16 bits), it is sign-extended into a longword when loaded into a register.

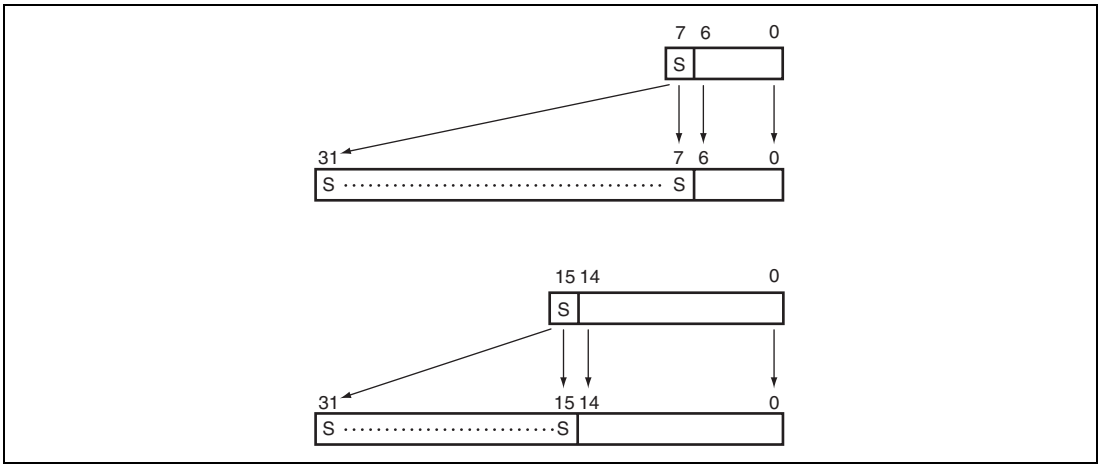


Figure 2.6 Formats of Byte Data and Word Data in Register

2.5 Data Formats in Memory

Memory data formats are classified into bytes, words, and longwords. Memory can be accessed in an 8-bit byte, 16-bit word, or 32-bit longword form. A memory operand less than 32 bits in length is sign-extended before being loaded into a register.

A word operand must be accessed starting from a word boundary (even address of a 2-byte unit: address 2n), and a longword operand starting from a longword boundary (even address of a 4-byte unit: address 4n). An address error will result if this rule is not observed. A byte operand can be accessed from any address.

Big endian or little endian byte order can be selected for the data format. The endian should be set with the external pin after a power-on reset. The endian cannot be changed dynamically. Bit positions are numbered left to right from most-significant to least-significant. Thus, in a 32-bit longword, the leftmost bit, bit 31, is the most significant bit and the rightmost bit, bit 0, is the least significant bit.

The data format in memory is shown in figure 2.7.

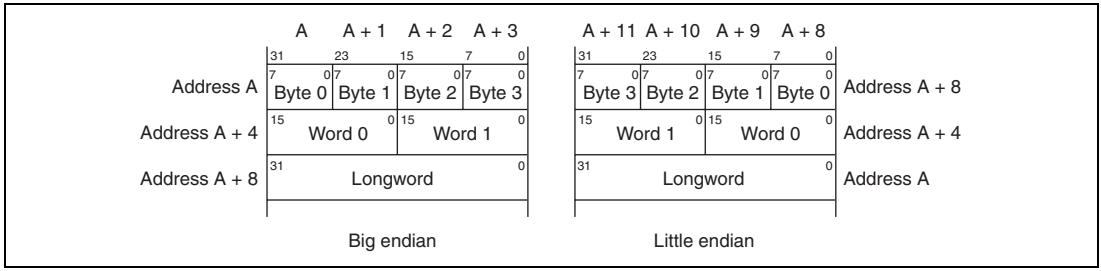


Figure 2.7 Data Formats in Memory

For the 64-bit data format, see figure 2.5.

2.6 Processing States

This LSI has major three processing states: the reset state, instruction execution state, and power-down state.

(1) Reset State

In this state the CPU is reset. The reset state is divided into the power-on reset state and the manual reset.

In the power-on reset state, the internal state of the CPU and the on-chip peripheral module registers are initialized. In the manual reset state, the internal state of the CPU and some registers of on-chip peripheral modules are initialized. For details, see register descriptions for each section of the hardware manual of the product.

(2) Instruction Execution State

In this state, the CPU executes program instructions in sequence. The Instruction execution state has the normal program execution state and the exception handling state.

(3) Power-Down State

In a power-down state, CPU halts operation and power consumption is reduced. The power-down state is entered by executing a SLEEP instruction. There are two modes in the power-down state: sleep mode and standby mode. For details, see section 18, Reset and Power-Down Modes.

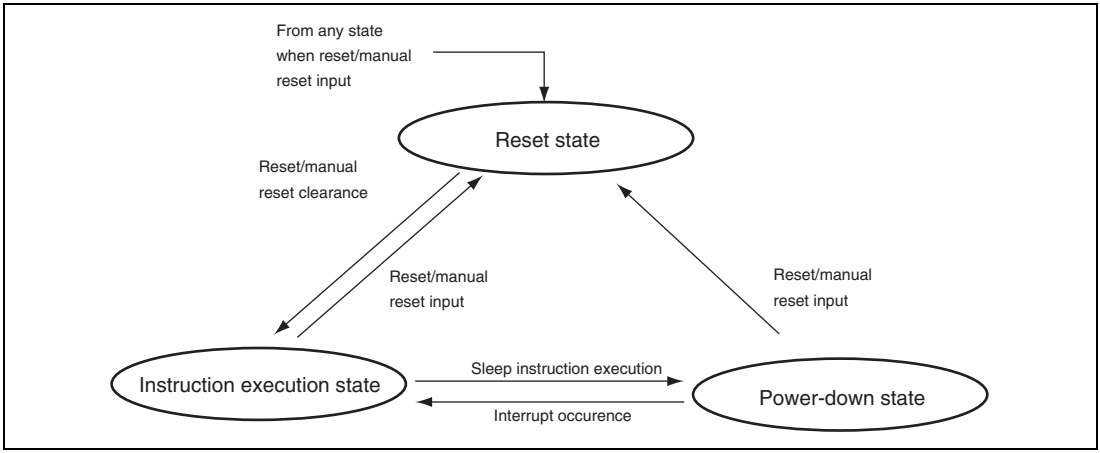


Figure 2.8 Processing State Transitions

2.7 Usage Notes

2.7.1 Notes on Self-Modifying Code

To accelerate the processing speed, this LSI performs the instruction prefetching. Therefore, in the case when a code in memory is rewritten and attempted to be executed immediately, there is increased possibility that the code before being modified, which has already been prefetched, is executed.

To ensure execution of the modified code, one of the following sequence of instructions should be executed between the code rewriting instruction and execution of the modified code.

(1) When the Codes to be Modified are in Non-Cacheable Area

```
SYNCO
ICBI @Rn
```

The target for the ICBI instruction can be any address within the range where no address error exception occurs.

(2) When the Codes to be Modified are in Cacheable Area (Write-Through)

```
SYNCO
ICBI @Rn
```

All instruction cache areas corresponding to the modified codes should be invalidated by the ICBI instruction. The ICBI instruction should be issued to each cache line. One cache line is 32 bytes.

(3) When the Codes to be Modified are in Cacheable Area (Copy-Back)

```
OCBP @Rm or OCBWB @Rm
SYNCO
ICBI @Rn
```

All operand cache areas corresponding to the modified codes should be written back to the main memory by the OCBP or OCBWB instruction. Then all instruction cache areas corresponding to the modified codes should be invalidated by the ICBI instruction. The OCBP, OCBWB, and ICBI instruction should be issued to each cache line. One cache line is 32 bytes.

Note: Self-modifying is the processing which executes instructions while dynamically rewriting the codes in memory.

Section 3 Instruction Set

The instruction set of this LSI is implemented with 16-bit fixed-length instructions. This LSI can use byte (8-bit), word (16-bit), longword (32-bit), and quadword (64-bit) data sizes for memory access. Single-precision floating-point data (32 bits) can be moved to and from memory using longword or quadword size. Double-precision floating-point data (64 bits) can be moved to and from memory using longword size. When this LSI moves byte-size or word-size data from memory to a register, the data is sign-extended.

3.1 Execution Environment

(1) PC

At the start of instruction execution, the PC indicates the address of the instruction itself.

(2) Load-Store Architecture

This LSI has a load-store architecture in which operations are basically executed using registers. Except for bit-manipulation operations such as logical AND that are executed directly in memory, operands in an operation that requires memory access are loaded into registers and the operation is executed between the registers.

(3) Delayed Branches

Except for the two branch instructions BF and BT, the branch instructions and RTE of this LSI are delayed branches. In a delayed branch, the instruction following the branch is executed before the branch destination instruction.

(4) Delay Slot

This execution slot following a delayed branch is called a delay slot. For example, the BRA execution sequence is as follows:w

Table 3.1 Execution Order of Delayed Branch Instructions

	Instructions			Execution Order
	BRA	TARGET	(Delayed branch instruction)	BRA
	ADD		(Delay slot)	↓
	:			ADD
	:			↓
TARGET	target-inst		(Branch destination instruction)	target-inst

A slot illegal instruction exception may occur when a specific instruction is executed in a delay slot. For details, see section 5, Exception Handling. The instruction following BF/S or BT/S for which the branch is not taken is also a delay slot instruction.

(5) T Bit

The T bit in SR is used to show the result of a compare operation, and is referenced by a conditional branch instruction. An example of the use of a conditional branch instruction is shown below.

```
ADD     #1, R0      ; T bit is not changed by ADD operation
CMP/EQ  R1, R0      ; If R0 = R1, T bit is set to 1
BT      TARGET      ; Branches to TARGET if T bit = 1 (R0 = R1)
```

In an RTE delay slot, the SR bits are referenced as follows. In instruction access, the MD bit is used before modification, and in data access, the MD bit is accessed after modification. The other bits—S, T, M, Q, FD, BL, and RB—after modification are used for delay slot instruction execution. The STC and STC.L SR instructions access all SR bits after modification.

(6) Constant Values

An 8-bit constant value can be specified by the instruction code and an immediate value. 16-bit and 32-bit constant values can be defined as literal constant values in memory, and can be referenced by a PC-relative load instruction.

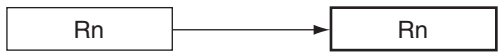
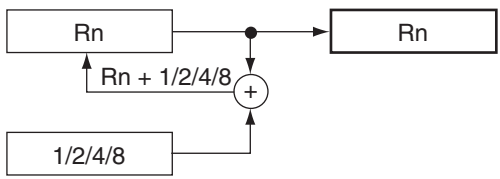
```
MOV.W   @(disp, PC), Rn
MOV.L   @(disp, PC), Rn
```

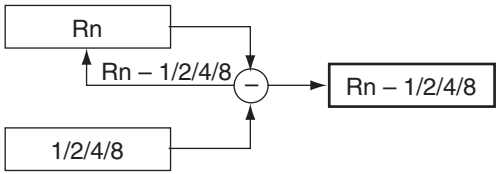
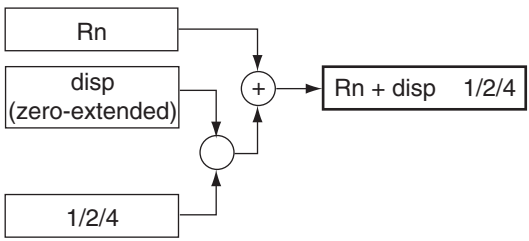
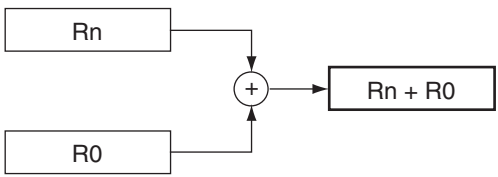
There are no PC-relative load instructions for floating-point operations. However, it is possible to set 0.0 or 1.0 by using the FLDI0 or FLDI1 instruction on a single-precision floating-point register.

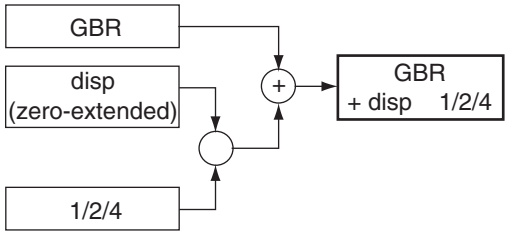
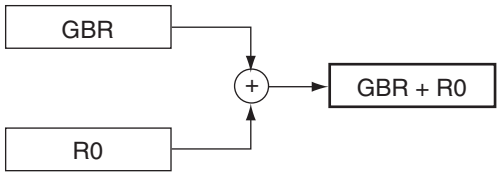
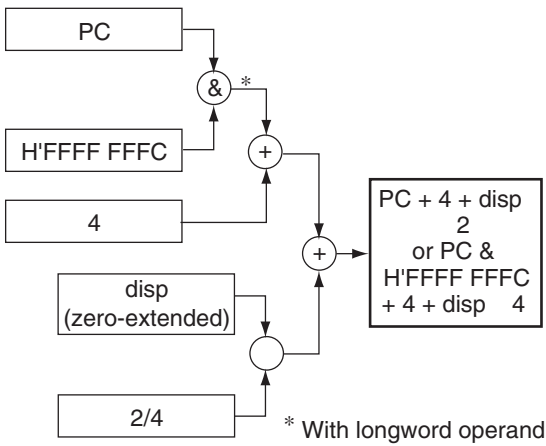
3.2 Addressing Modes

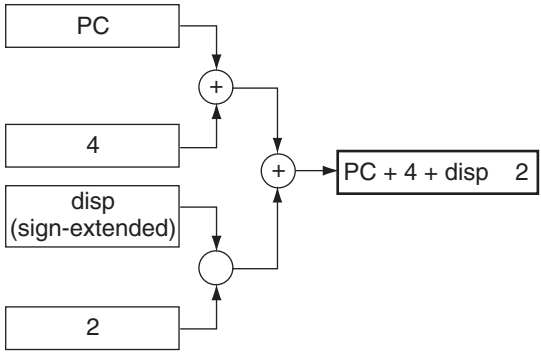
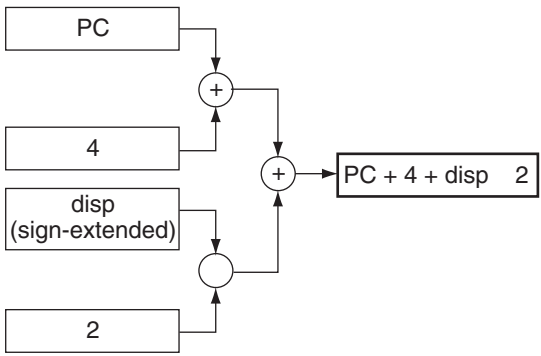
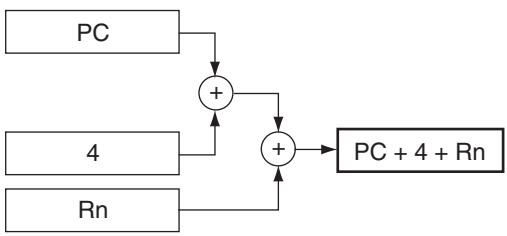
Addressing modes and effective address calculation methods are shown in table 3.2. When a location in virtual memory space is accessed (AT in MMUCR = 1), the effective address is translated into a physical memory address. If multiple virtual memory space systems are selected (SV in MMUCR = 0), the least significant bit of PTEH is also referenced as the access ASID. For details, see section 7, Memory Management Unit (MMU).

Table 3.2 Addressing Modes and Effective Addresses

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register direct	Rn	Effective address is register Rn. (Operand is register Rn contents.)	—
Register indirect	@Rn	Effective address is register Rn contents. 	Rn → EA (EA: effective address)
Register indirect with post-increment	@Rn+	Effective address is register Rn contents. A constant is added to Rn after instruction execution: 1 for a byte operand, 2 for a word operand, 4 for a longword operand, 8 for a quadword operand. 	Rn → EA After instruction execution Byte: Rn + 1 → Rn Word: Rn + 2 → Rn Longword: Rn + 4 → Rn Quadword: Rn + 8 → Rn

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Register indirect with pre-decrement	@–Rn	Effective address is register Rn contents, decremented by a constant beforehand: 1 for a byte operand, 2 for a word operand, 4 for a longword operand, 8 for a quadword operand. 	Byte: Rn – 1 → Rn Word: Rn – 2 → Rn Longword: Rn – 4 → Rn Quadword: Rn – 8 → Rn Rn → EA (Instruction executed with Rn after calculation)
Register indirect with displacement	@(disp:4, Rn)	Effective address is register Rn contents with 4-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size. 	Byte: Rn + disp → EA Word: Rn + disp × 2 → EA Longword: Rn + disp × 4 → EA
Indexed register indirect	@(R0, Rn)	Effective address is sum of register Rn and R0 contents. 	Rn + R0 → EA

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
GBR indirect with displacement	@(disp:8, GBR)	Effective address is register GBR contents with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 1 (byte), 2 (word), or 4 (longword), according to the operand size. 	Byte: GBR + disp → EA Word: GBR + disp × 2 → EA Longword: GBR + disp × 4 → EA
Indexed GBR indirect	@(R0, GBR)	Effective address is sum of register GBR and R0 contents. 	GBR + R0 → EA
PC-relative with displacement	@(disp:8, PC)	Effective address is PC + 4 with 8-bit displacement disp added. After disp is zero-extended, it is multiplied by 2 (word), or 4 (longword), according to the operand size. With a longword operand, the lower 2 bits of PC are masked. 	Word: PC + 4 + disp × 2 → EA Longword: PC & H'FFFF FFFC + 4 + disp × 4 → EA

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
PC-relative	disp:8	Effective address is PC + 4 with 8-bit displacement disp added after being sign-extended and multiplied by 2. 	PC + 4 + disp × 2 → Branch-Target
PC-relative	disp:12	Effective address is PC + 4 with 12-bit displacement disp added after being sign-extended and multiplied by 2. 	PC + 4 + disp × 2 → Branch-Target
	Rn	Effective address is sum of PC + 4 and Rn. 	PC + 4 + Rn → Branch-Target

Addressing Mode	Instruction Format	Effective Address Calculation Method	Calculation Formula
Immediate	#imm:8	8-bit immediate data imm of TST, AND, OR, or XOR instruction is zero-extended.	—
	#imm:8	8-bit immediate data imm of MOV, ADD, or CMP/EQ instruction is sign-extended.	—
	#imm:8	8-bit immediate data imm of TRAPA instruction is zero-extended and multiplied by 4.	—

Note: For the addressing modes below that use a displacement (disp), the assembler descriptions in this manual show the value before scaling (×1, ×2, or ×4) is performed according to the operand size. This is done to clarify the operation of the LSI. Refer to the relevant assembler notation rules for the actual assembler descriptions.

@ (disp:4, Rn) ; Register indirect with displacement
@ (disp:8, GBR) ; GBR indirect with displacement
@ (disp:8, PC) ; PC-relative with displacement
disp:8, disp:12 ; PC-relative

3.3 Instruction Set

Table 3.3 shows the notation used in the SH instruction lists shown in tables 3.4 to 3.13.

Table 3.3 Notation Used in Instruction List

Item	Format	Description	
Instruction mnemonic	OP.Sz SRC, DEST	OP:	Operation code
		Sz:	Size
		SRC:	Source operand
		DEST:	Source and/or destination operand
		Rm:	Source register
		Rn:	Destination register
		imm:	Immediate data
		disp:	Displacement
Operation notation		→, ←	Transfer direction
		(xx)	Memory operand
		M/Q/T	SR flag bits
		&	Logical AND of individual bits
		|	Logical OR of individual bits
		^	Logical exclusive-OR of individual bits
		~	Logical NOT of individual bits
		<<n, >>n	n-bit shift
Instruction code	MSB ↔ LSB	mmmm:	Register number (Rm, FRm)
		nnnn:	Register number (Rn, FRn)
		0000:	R0, FR0
		0001:	R1, FR1
		:	
		1111:	R15, FR15
		mmm:	Register number (DRm, XDm, Rm_BANK)
		nnn:	Register number (DRn, XDn, Rn_BANK)
		000:	DR0, XD0, R0_BANK
		001:	DR2, XD2, R1_BANK
		:	
		111:	DR14, XD14, R7_BANK
		mm:	Register number (FVm)
		nn:	Register number (FVn)
		00:	FV0
		01:	FV4
		10:	FV8
		11:	FV12
		iiii:	Immediate data
		dddd:	Displacement

Item	Format	Description
Privileged mode		"Privileged" means the instruction can only be executed in privileged mode.
T bit	Value of T bit after instruction execution	—: No change
New	—	"New" means the instruction which has been newly added in the SH-4A with H'20-valued VER bits in the processor version register (PVR).

Note: Scaling (×1, ×2, ×4, or ×8) is executed according to the size of the instruction operand.

Table 3.4 Fixed-Point Transfer Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOV	#imm, Rn	imm → sign extension → Rn	1110nnnniiiiiiii	—	—	—
MOV.W	@(disp*,PC),Rn	(disp × 2 + PC + 4) → sign extension → Rn	1001nnnndddddddd	—	—	—
MOV.L	@(disp*,PC),Rn	(disp × 4 + PC & H'FFFF FFFC + 4) → Rn	1101nnnndddddddd	—	—	—
MOV	Rm,Rn	Rm → Rn	0110nnnnmmmm0011	—	—	—
MOV.B	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0000	—	—	—
MOV.W	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0001	—	—	—
MOV.L	Rm,@Rn	Rm → (Rn)	0010nnnnmmmm0010	—	—	—
MOV.B	@Rm,Rn	(Rm) → sign extension → Rn	0110nnnnmmmm0000	—	—	—
MOV.W	@Rm,Rn	(Rm) → sign extension → Rn	0110nnnnmmmm0001	—	—	—
MOV.L	@Rm,Rn	(Rm) → Rn	0110nnnnmmmm0010	—	—	—
MOV.B	Rm,@-Rn	Rn-1 → Rn, Rm → (Rn)	0010nnnnmmmm0100	—	—	—
MOV.W	Rm,@-Rn	Rn-2 → Rn, Rm → (Rn)	0010nnnnmmmm0101	—	—	—
MOV.L	Rm,@-Rn	Rn-4 → Rn, Rm → (Rn)	0010nnnnmmmm0110	—	—	—
MOV.B	@Rm+,Rn	(Rm)→ sign extension → Rn, Rm + 1 → Rm	0110nnnnmmmm0100	—	—	—
MOV.W	@Rm+,Rn	(Rm) → sign extension → Rn, Rm + 2 → Rm	0110nnnnmmmm0101	—	—	—
MOV.L	@Rm+,Rn	(Rm) → Rn, Rm + 4 → Rm	0110nnnnmmmm0110	—	—	—
MOV.B	R0,@(disp*,Rn)	R0 → (disp + Rn)	10000000nnnndddd	—	—	—
MOV.W	R0,@(disp*,Rn)	R0 → (disp × 2 + Rn)	10000001nnnndddd	—	—	—
MOV.L	Rm,@(disp*,Rn)	Rm → (disp × 4 + Rn)	0001nnnnmmmmdddd	—	—	—
MOV.B	@(disp*, Rm),R0	(disp + Rm) → sign extension → R0	10000100mmmmdddd	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
MOV.W	@(disp*, Rm),R0	(disp × 2 + Rm) → sign extension → R0	10000101mmmmdddd	—	—	—
MOV.L	@(disp*, Rm),Rn	(disp × 4 + Rm) → Rn	0101nnnnmmmmdddd	—	—	—
MOV.B	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0100	—	—	—
MOV.W	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0101	—	—	—
MOV.L	Rm,@(R0,Rn)	Rm → (R0 + Rn)	0000nnnnmmmm0110	—	—	—
MOV.B	@(R0, Rm),Rn	(R0 + Rm) → sign extension → Rn	0000nnnnmmmm1100	—	—	—
MOV.W	@(R0, Rm),Rn	(R0 + Rm) → sign extension → Rn	0000nnnnmmmm1101	—	—	—
MOV.L	@(R0, Rm),Rn	(R0 + Rm) → Rn	0000nnnnmmmm1110	—	—	—
MOV.B	R0,@(disp*,GBR)	R0 → (disp + GBR)	11000000dddddddd	—	—	—
MOV.W	R0,@(disp*,GBR)	R0 → (disp × 2 + GBR)	11000001dddddddd	—	—	—
MOV.L	R0,@(disp*,GBR)	R0 → (disp × 4 + GBR)	11000010dddddddd	—	—	—
MOV.B	@(disp*,GBR),R0	(disp + GBR) → sign extension → R0	11000100dddddddd	—	—	—
MOV.W	@(disp*,GBR),R0	(disp × 2 + GBR) → sign extension → R0	11000101dddddddd	—	—	—
MOV.L	@(disp*,GBR),R0	(disp × 4 + GBR) → R0	11000110dddddddd	—	—	—
MOVA	@(disp*,PC),R0	disp × 4 + PC & H'FFFF FFFC + 4 → R0	11000111dddddddd	—	—	—
MOVCO.L	R0,@Rn	LDST → T If (T == 1) R0 → (Rn) 0 → LDST	0000nnnn01110011	—	LDST	New
MOVLI.L	@Rm,R0	1 → LDST (Rm) → R0 When interrupt/exception occurred 0 → LDST	0000mmmm01100011	—	—	New
MOVUA.L	@Rm,R0	(Rm) → R0 Load non-boundary alignment data	0100mmmm10101001	—	—	New
MOVUA.L	@Rm+,R0	(Rm) → R0, Rm + 4 → Rm Load non-boundary alignment data	0100mmmm11101001	—	—	New
MOVT	Rn	T → Rn	0000nnnn00101001	—	—	—
SWAP.B	Rm,Rn	Rm → swap lower 2 bytes → Rn	0110nnnnmmmm1000	—	—	—
SWAP.W	Rm,Rn	Rm → swap upper/lower words → Rn	0110nnnnmmmm1001	—	—	—
XTRCT	Rm,Rn	Rm:Rn middle 32 bits → Rn	0010nnnnmmmm1101	—	—	—

Note: * The assembler of Renesas uses the value after scaling (×1, ×2, or ×4) as the displacement (disp).

Table 3.5 Arithmetic Operation Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
ADD	Rm,Rn	Rn + Rm → Rn	0011nnnnmmmm1100	—	—	—
ADD	#imm,Rn	Rn + imm → Rn	0111nnnniiiiiiii	—	—	—
ADDC	Rm,Rn	Rn + Rm + T → Rn, carry → T	0011nnnnmmmm1110	—	Carry	—
ADDV	Rm,Rn	Rn + Rm → Rn, overflow → T	0011nnnnmmmm1111	—	Overflow	—
CMP/EQ	#imm,R0	When R0 = imm, 1 → T Otherwise, 0 → T	10001000iiiiiiii	—	Comparison result	—
CMP/EQ	Rm,Rn	When Rn = Rm, 1 → T Otherwise, 0 → T	0011nnnnmmmm0000	—	Comparison result	—
CMP/HS	Rm,Rn	When Rn ≥ Rm (unsigned), 1 → T Otherwise, 0 → T	0011nnnnmmmm0010	—	Comparison result	—
CMP/GE	Rm,Rn	When Rn ≥ Rm (signed), 1 → T Otherwise, 0 → T	0011nnnnmmmm0011	—	Comparison result	—
CMP/HI	Rm,Rn	When Rn > Rm (unsigned), 1 → T Otherwise, 0 → T	0011nnnnmmmm0110	—	Comparison result	—
CMP/GT	Rm,Rn	When Rn > Rm (signed), 1 → T Otherwise, 0 → T	0011nnnnmmmm0111	—	Comparison result	—
CMP/PZ	Rn	When Rn ≥ 0, 1 → T Otherwise, 0 → T	0100nnnn00010001	—	Comparison result	—
CMP/PL	Rn	When Rn > 0, 1 → T Otherwise, 0 → T	0100nnnn00010101	—	Comparison result	—
CMP/STR	Rm,Rn	When any bytes are equal, 1 → T Otherwise, 0 → T	0010nnnnmmmm1100	—	Comparison result	—
DIV1	Rm,Rn	1-step division (Rn ÷ Rm)	0011nnnnmmmm0100	—	Calculation result	—
DIV0S	Rm,Rn	MSB of Rn → Q, MSB of Rm → M, M^Q → T	0010nnnnmmmm0111	—	Calculation result	—
DIV0U		0 → M/Q/T	0000000000011001	—	0	—
DMULS.L	Rm,Rn	Signed, Rn × Rm → MAC, 32 × 32 → 64 bits	0011nnnnmmmm1101	—	—	—
DMULU.L	Rm,Rn	Unsigned, Rn × Rm → MAC, 32 × 32 → 64 bits	0011nnnnmmmm0101	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
DT	Rn	Rn – 1 → Rn; when Rn = 0, 1 → T When Rn ≠ 0, 0 → T	0100nnnn00010000	—	Comparison result	—
EXTS.B	Rm,Rn	Rm sign-extended from byte → Rn	0110nnnnmmmm1110	—	—	—
EXTS.W	Rm,Rn	Rm sign-extended from word → Rn	0110nnnnmmmm1111	—	—	—
EXTU.B	Rm,Rn	Rm zero-extended from byte → Rn	0110nnnnmmmm1100	—	—	—
EXTU.W	Rm,Rn	Rm zero-extended from word → Rn	0110nnnnmmmm1101	—	—	—
MAC.L	@Rm+,@Rn+	Signed, (Rn) × (Rm) + MAC → MAC Rn + 4 → Rn, Rm + 4 → Rm 32 × 32 + 64 → 64 bits	0000nnnnmmmm1111	—	—	—
MAC.W	@Rm+,@Rn+	Signed, (Rn) × (Rm) + MAC → MAC Rn + 2 → Rn, Rm + 2 → Rm 16 × 16 + 64 → 64 bits	0100nnnnmmmm1111	—	—	—
MUL.L	Rm,Rn	Rn × Rm → MACL 32 × 32 → 32 bits	0000nnnnmmmm0111	—	—	—
MULS.W	Rm,Rn	Signed, Rn × Rm → MACL 16 × 16 → 32 bits	0010nnnnmmmm1111	—	—	—
MULU.W	Rm,Rn	Unsigned, Rn × Rm → MACL 16 × 16 → 32 bits	0010nnnnmmmm1110	—	—	—
NEG	Rm,Rn	0 – Rm → Rn	0110nnnnmmmm1011	—	—	—
NEGC	Rm,Rn	0 – Rm – T → Rn, borrow → T	0110nnnnmmmm1010	—	Borrow	—
SUB	Rm,Rn	Rn – Rm → Rn	0011nnnnmmmm1000	—	—	—
SUBC	Rm,Rn	Rn – Rm – T → Rn, borrow → T	0011nnnnmmmm1010	—	Borrow	—
SUBV	Rm,Rn	Rn – Rm → Rn, underflow → T	0011nnnnmmmm1011	—	Underflow	—

Table 3.6 Logic Operation Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
AND	Rm,Rn	Rn & Rm → Rn	0010nnnnmmmm1001	—	—	—
AND	#imm,R0	R0 & imm → R0	11001001iiiiiiii	—	—	—
AND.B	#imm,@(R0,GBR)	(R0 + GBR) & imm → (R0 + GBR)	11001101iiiiiiii	—	—	—
NOT	Rm,Rn	~Rm → Rn	0110nnnnmmmm0111	—	—	—
OR	Rm,Rn	Rn | Rm → Rn	0010nnnnmmmm1011	—	—	—
OR	#imm,R0	R0 | imm → R0	11001011iiiiiiii	—	—	—
OR.B	#imm,@(R0,GBR)	(R0 + GBR) | imm → (R0 + GBR)	11001111iiiiiiii	—	—	—
TAS.B	@Rn	When (Rn) = 0, 1 → T Otherwise, 0 → T In both cases, 1 → MSB of (Rn)	0100nnnn00011011	—	Test result	—
TST	Rm,Rn	Rn & Rm; when result = 0, 1 → T Otherwise, 0 → T	0010nnnnmmmm1000	—	Test result	—
TST	#imm,R0	R0 & imm; when result = 0, 1 → T Otherwise, 0 → T	11001000iiiiiiii	—	Test result	—
TST.B	#imm,@(R0,GBR)	(R0 + GBR) & imm; when result = 0, 1 → T Otherwise, 0 → T	11001100iiiiiiii	—	Test result	—
XOR	Rm,Rn	Rn ∧ Rm → Rn	0010nnnnmmmm1010	—	—	—
XOR	#imm,R0	R0 ∧ imm → R0	11001010iiiiiiii	—	—	—
XOR.B	#imm,@(R0,GBR)	(R0 + GBR) ∧ imm → (R0 + GBR)	11001110iiiiiiii	—	—	—

Table 3.7 Shift Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
ROTL	Rn	T ← Rn ← MSB	0100nnnn00000100	—	MSB	—
ROTR	Rn	LSB → Rn → T	0100nnnn00000101	—	LSB	—
ROTCL	Rn	T ← Rn ← T	0100nnnn00100100	—	MSB	—
ROTCR	Rn	T → Rn → T	0100nnnn00100101	—	LSB	—
SHAD	Rm,Rn	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> Rm → [MSB → Rn]	0100nnnnmmmm1100	—	—	—
SHAL	Rn	T ← Rn ← 0	0100nnnn00100000	—	MSB	—
SHAR	Rn	MSB → Rn → T	0100nnnn00100001	—	LSB	—
SHLD	Rm,Rn	When Rm ≥ 0, Rn << Rm → Rn When Rm < 0, Rn >> Rm → [0 → Rn]	0100nnnnmmmm1101	—	—	—
SHLL	Rn	T ← Rn ← 0	0100nnnn00000000	—	MSB	—
SHLR	Rn	0 → Rn → T	0100nnnn00000001	—	LSB	—
SHLL2	Rn	Rn << 2 → Rn	0100nnnn00001000	—	—	—
SHLR2	Rn	Rn >> 2 → Rn	0100nnnn00001001	—	—	—
SHLL8	Rn	Rn << 8 → Rn	0100nnnn00011000	—	—	—
SHLR8	Rn	Rn >> 8 → Rn	0100nnnn00011001	—	—	—
SHLL16	Rn	Rn << 16 → Rn	0100nnnn00101000	—	—	—
SHLR16	Rn	Rn >> 16 → Rn	0100nnnn00101001	—	—	—

Table 3.8 Branch Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
BF	label	When T = 0, disp × 2 + PC + 4 → PC When T = 1, nop	10001011dddddddd	—	—	—
BF/S	label	Delayed branch; when T = 0, disp × 2 + PC + 4 → PC When T = 1, nop	10001111dddddddd	—	—	—
BT	label	When T = 1, disp × 2 + PC + 4 → PC When T = 0, nop	10001001dddddddd	—	—	—
BT/S	label	Delayed branch; when T = 1, disp × 2 + PC + 4 → PC When T = 0, nop	10001101dddddddd	—	—	—
BRA	label	Delayed branch, disp × 2 + PC + 4 → PC	1010dddddddddddd	—	—	—
BRAF	Rn	Delayed branch, Rn + PC + 4 → PC	0000nnnn00100011	—	—	—
BSR	label	Delayed branch, PC + 4 → PR, disp × 2 + PC + 4 → PC	1011dddddddddddd	—	—	—
BSRF	Rn	Delayed branch, PC + 4 → PR, Rn + PC + 4 → PC	0000nnnn00000011	—	—	—
JMP	@Rn	Delayed branch, Rn → PC	0100nnnn00101011	—	—	—
JSR	@Rn	Delayed branch, PC + 4 → PR, Rn → PC	0100nnnn00001011	—	—	—
RTS		Delayed branch, PR → PC	0000000000001011	—	—	—

Table 3.9 System Control Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
CLRMAC		0 → MACH, MACL	0000000000101000	—	—	—
CLRS		0 → S	0000000001001000	—	—	—
CLRT		0 → T	0000000000001000	—	0	—
ICBI	@Rn	Invalidates instruction cache block	0000nnnn11100011	—	—	New
LDC	Rm,SR	Rm → SR	0100mmmm00001110	Privileged	LSB	—
LDC	Rm,GBR	Rm → GBR	0100mmmm00011110	—	—	—
LDC	Rm,VBR	Rm → VBR	0100mmmm00101110	Privileged	—	—
LDC	Rm,SGR	Rm → SGR	0100mmmm00111010	Privileged	—	—
LDC	Rm,SSR	Rm → SSR	0100mmmm00111110	Privileged	—	—
LDC	Rm,SPC	Rm → SPC	0100mmmm01001110	Privileged	—	—
LDC	Rm,DBR	Rm → DBR	0100mmmm11111010	Privileged	—	—
LDC	Rm,Rn_BANK	Rm → Rn_BANK (n = 0 to 7)	0100mmmm1nnn1110	Privileged	—	—
LDC.L	@Rm+,SR	(Rm) → SR, Rm + 4 → Rm	0100mmmm00000111	Privileged	LSB	—
LDC.L	@Rm+,GBR	(Rm) → GBR, Rm + 4 → Rm	0100mmmm00010111	—	—	—
LDC.L	@Rm+,VBR	(Rm) → VBR, Rm + 4 → Rm	0100mmmm00100111	Privileged	—	—
LDC.L	@Rm+,SGR	(Rm) → SGR, Rm + 4 → Rm	0100mmmm00110110	Privileged	—	—
LDC.L	@Rm+,SSR	(Rm) → SSR, Rm + 4 → Rm	0100mmmm00110111	Privileged	—	—
LDC.L	@Rm+,SPC	(Rm) → SPC, Rm + 4 → Rm	0100mmmm01000111	Privileged	—	—
LDC.L	@Rm+,DBR	(Rm) → DBR, Rm + 4 → Rm	0100mmmm11110110	Privileged	—	—
LDC.L	@Rm+, Rn_BANK	(Rm) → Rn_BANK, Rm + 4 → Rm	0100mmmm1nnn0111	Privileged	—	—
LDS	Rm,MACH	Rm → MACH	0100mmmm00001010	—	—	—
LDS	Rm,MACL	Rm → MACL	0100mmmm00011010	—	—	—
LDS	Rm,PR	Rm → PR	0100mmmm00101010	—	—	—
LDS.L	@Rm+,MACH	(Rm) → MACH, Rm + 4 → Rm	0100mmmm00000110	—	—	—
LDS.L	@Rm+,MACL	(Rm) → MACL, Rm + 4 → Rm	0100mmmm00010110	—	—	—
LDS.L	@Rm+,PR	(Rm) → PR, Rm + 4 → Rm	0100mmmm00100110	—	—	—
LDTLB		PTEH/PTEL (/PTEA) → TLB	0000000000111000	Privileged	—	—
MOVCA.L	R0,@Rn	R0 → (Rn) (without fetching cache block)	0000nnnn11000011	—	—	—
NOP		No operation	0000000000001001	—	—	—
OCBI	@Rn	Invalidates operand cache block	0000nnnn10010011	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
OCBP	@Rn	Writes back and invalidates operand cache block	0000nnnn10100011	—	—	—
OCBWB	@Rn	Writes back operand cache block	0000nnnn10110011	—	—	—
PREF	@Rn	(Rn) → operand cache	0000nnnn10000011	—	—	—
PREFI	@Rn	Reads 32-byte instruction block into instruction cache	0000nnnn11010011	—	—	New
RTE		Delayed branch, SSR/SPC → SR/PC	0000000000101011	Privileged	—	—
SETS		1 → S	0000000001011000	—	—	—
SETT		1 → T	0000000000011000	—	1	—
SLEEP		Sleep or standby	0000000000011011	Privileged	—	—
STC	SR,Rn	SR → Rn	0000nnnn00000010	Privileged	—	—
STC	GBR,Rn	GBR → Rn	0000nnnn00010010	—	—	—
STC	VBR,Rn	VBR → Rn	0000nnnn00100010	Privileged	—	—
STC	SSR,Rn	SSR → Rn	0000nnnn00110010	Privileged	—	—
STC	SPC,Rn	SPC → Rn	0000nnnn01000010	Privileged	—	—
STC	SGR,Rn	SGR → Rn	0000nnnn00111010	Privileged	—	—
STC	DBR,Rn	DBR → Rn	0000nnnn11111010	Privileged	—	—
STC	Rm_BANK,Rn	Rm_BANK → Rn (m = 0 to 7)	0000nnnn1mmm0010	Privileged	—	—
STC.L	SR,@-Rn	Rn – 4 → Rn, SR → (Rn)	0100nnnn00000011	Privileged	—	—
STC.L	GBR,@-Rn	Rn – 4 → Rn, GBR → (Rn)	0100nnnn00010011	—	—	—
STC.L	VBR,@-Rn	Rn – 4 → Rn, VBR → (Rn)	0100nnnn00100011	Privileged	—	—
STC.L	SSR,@-Rn	Rn – 4 → Rn, SSR → (Rn)	0100nnnn00110011	Privileged	—	—
STC.L	SPC,@-Rn	Rn – 4 → Rn, SPC → (Rn)	0100nnnn01000011	Privileged	—	—
STC.L	SGR,@-Rn	Rn – 4 → Rn, SGR → (Rn)	0100nnnn00110010	Privileged	—	—
STC.L	DBR,@-Rn	Rn – 4 → Rn, DBR → (Rn)	0100nnnn11110010	Privileged	—	—
STC.L	Rm_BANK, @-Rn	Rn – 4 → Rn, Rm_BANK → (Rn) (m = 0 to 7)	0100nnnn1mmm0011	Privileged	—	—
STS	MACH,Rn	MACH → Rn	0000nnnn00001010	—	—	—
STS	MACL,Rn	MACL → Rn	0000nnnn00011010	—	—	—
STS	PR,Rn	PR → Rn	0000nnnn00101010	—	—	—
STS.L	MACH,@-Rn	Rn – 4 → Rn, MACH → (Rn)	0100nnnn00000010	—	—	—
STS.L	MACL,@-Rn	Rn – 4 → Rn, MACL → (Rn)	0100nnnn00010010	—	—	—
STS.L	PR,@-Rn	Rn – 4 → Rn, PR → (Rn)	0100nnnn00100010	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
SYNCO		Data accesses invoked by the following instructions are not executed until execution of data accesses which precede this instruction has been completed.	0000000010101011	—	—	New
TRAPA	#imm	PC + 2 → SPC, SR → SSR, R15 → SGR, 1 → SR.MD/BL/RB, #imm << 2 → TRA, H'160 → EXPEVT, VBR + H'0100 → PC	11000011iiiiiiii	—	—	—

Table 3.10 Floating-Point Single-Precision Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FLDI0	FRn	H'0000 0000 → FRn	1111nnnn10001101	—	—	—
FLDI1	FRn	H'3F80 0000 → FRn	1111nnnn10011101	—	—	—
FMOV	FRm,FRn	FRm → FRn	1111nnnnmmmm1100	—	—	—
FMOV.S	@Rm,FRn	(Rm) → FRn	1111nnnnmmmm1000	—	—	—
FMOV.S	@(R0,Rm),FRn	(R0 + Rm) → FRn	1111nnnnmmmm0110	—	—	—
FMOV.S	@Rm+,FRn	(Rm) → FRn, Rm + 4 → Rm	1111nnnnmmmm1001	—	—	—
FMOV.S	FRm,@Rn	FRm → (Rn)	1111nnnnmmmm1010	—	—	—
FMOV.S	FRm,@-Rn	Rn-4 → Rn, FRm → (Rn)	1111nnnnmmmm1011	—	—	—
FMOV.S	FRm,@(R0,Rn)	FRm → (R0 + Rn)	1111nnnnmmmm0111	—	—	—
FMOV	DRm,DRn	DRm → DRn	1111nnn0mmm01100	—	—	—
FMOV	@Rm,DRn	(Rm) → DRn	1111nnn0mmmm1000	—	—	—
FMOV	@(R0,Rm),DRn	(R0 + Rm) → DRn	1111nnn0mmmm0110	—	—	—
FMOV	@Rm+,DRn	(Rm) → DRn, Rm + 8 → Rm	1111nnn0mmmm1001	—	—	—
FMOV	DRm,@Rn	DRm → (Rn)	1111nnnnmmm01010	—	—	—
FMOV	DRm,@-Rn	Rn-8 → Rn, DRm → (Rn)	1111nnnnmmm01011	—	—	—
FMOV	DRm,@(R0,Rn)	DRm → (R0 + Rn)	1111nnnnmmm00111	—	—	—
FLDS	FRm,FPUL	FRm → FPUL	1111mmmm00011101	—	—	—
FSTS	FPUL,FRn	FPUL → FRn	1111nnnn00001101	—	—	—
FABS	FRn	FRn & H'7FFF FFFF → FRn	1111nnnn01011101	—	—	—
FADD	FRm,FRn	FRn + FRm → FRn	1111nnnnmmmm0000	—	—	—

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FCMP/EQ	FRm,FRn	When FRn = FRm, 1 → T Otherwise, 0 → T	1111nnnnmmmm0100	—	Comparison result	—
FCMP/GT	FRm,FRn	When FRn > FRm, 1 → T Otherwise, 0 → T	1111nnnnmmmm0101	—	Comparison result	—
FDIV	FRm,FRn	FRn/FRm → FRn	1111nnnnmmmm0011	—	—	—
FLOAT	FPUL,FRn	(float) FPUL → FRn	1111nnnn00101101	—	—	—
FMAC	FR0,FRm,FRn	FR0*FRm + FRn → FRn	1111nnnnmmmm1110	—	—	—
FMUL	FRm,FRn	FRn*FRm → FRn	1111nnnnmmmm0010	—	—	—
FNEG	FRn	FRn ∧ H'8000 0000 → FRn	1111nnnn01001101	—	—	—
FSQRT	FRn	√FRn → FRn	1111nnnn01101101	—	—	—
FSUB	FRm,FRn	FRn – FRm → FRn	1111nnnnmmmm0001	—	—	—
FTRC	FRm,FPUL	(long) FRm → FPUL	1111mmmm00111101	—	—	—

Table 3.11 Floating-Point Double-Precision Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FABS	DRn	DRn & H'7FFF FFFF FFFF FFFF → DRn	1111nnn001011101	—	—	—
FADD	DRm,DRn	DRn + DRm → DRn	1111nnn0mmm00000	—	—	—
FCMP/EQ	DRm,DRn	When DRn = DRm, 1 → T Otherwise, 0 → T	1111nnn0mmm00100	—	Comparison result	—
FCMP/GT	DRm,DRn	When DRn > DRm, 1 → T Otherwise, 0 → T	1111nnn0mmm00101	—	Comparison result	—
FDIV	DRm,DRn	DRn /DRm → DRn	1111nnn0mmm00011	—	—	—
FCNVDS	DRm,FPUL	double_to_ float(DRm) → FPUL	1111mmm010111101	—	—	—
FCNVSD	FPUL,DRn	float_to_ double (FPUL) → DRn	1111nnn010101101	—	—	—
FLOAT	FPUL,DRn	(float)FPUL → DRn	1111nnn000101101	—	—	—
FMUL	DRm,DRn	DRn *DRm → DRn	1111nnn0mmm00010	—	—	—
FNEG	DRn	DRn ^ H'8000 0000 0000 0000 → DRn	1111nnn001001101	—	—	—
FSQRT	DRn	√DRn → DRn	1111nnn001101101	—	—	—
FSUB	DRm,DRn	DRn – DRm → DRn	1111nnn0mmm00001	—	—	—
FTRC	DRm,FPUL	(long) DRm → FPUL	1111mmm000111101	—	—	—

Table 3.12 Floating-Point Control Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
LDS	Rm,FPSCR	Rm → FPSCR	0100mmmm01101010	—	—	—
LDS	Rm,FPUL	Rm → FPUL	0100mmmm01011010	—	—	—
LDS.L	@Rm+,FPSCR	(Rm) → FPSCR, Rm+4 → Rm	0100mmmm01100110	—	—	—
LDS.L	@Rm+,FPUL	(Rm) → FPUL, Rm+4 → Rm	0100mmmm01010110	—	—	—
STS	FPSCR,Rn	FPSCR → Rn	0000nnnn01101010	—	—	—
STS	FPUL,Rn	FPUL → Rn	0000nnnn01011010	—	—	—
STS.L	FPSCR,@-Rn	Rn – 4 → Rn, FPSCR → (Rn)	0100nnnn01100010	—	—	—
STS.L	FPUL,@-Rn	Rn – 4 → Rn, FPUL → (Rn)	0100nnnn01010010	—	—	—

Table 3.13 Floating-Point Graphics Acceleration Instructions

Instruction		Operation	Instruction Code	Privileged	T Bit	New
FMOV	DRm,XDn	DRm → XDn	1111nnn1mmm01100	—	—	—
FMOV	XDm,DRn	XDm → DRn	1111nnn0mmm11100	—	—	—
FMOV	XDm,XDn	XDm → XDn	1111nnn1mmm11100	—	—	—
FMOV	@Rm,XDn	(Rm) → XDn	1111nnn1mmmm1000	—	—	—
FMOV	@Rm+,XDn	(Rm) → XDn, Rm + 8 → Rm	1111nnn1mmmm1001	—	—	—
FMOV	@(R0,Rm),XDn	(R0 + Rm) → XDn	1111nnn1mmmm0110	—	—	—
FMOV	XDm,@Rn	XDm → (Rn)	1111nnnnmmm11010	—	—	—
FMOV	XDm,@-Rn	Rn – 8 → Rn, XDm → (Rn)	1111nnnnmmm11011	—	—	—
FMOV	XDm,@(R0,Rn)	XDm → (R0 + Rn)	1111nnnnmmm10111	—	—	—
FIPR	FVm,FVn	inner_product (FVm, FVn) → FR[n+3]	1111nnmm11101101	—	—	—
FTRV	XMTRX,FVn	transform_vector (XMTRX, FVn) → FVn	1111nn0111111101	—	—	—
FRCHG		~FPSCR.FR → FPSCR.FR	1111101111111101	—	—	—
FSCHG		~FPSCR.SZ → FPSCR.SZ	1111001111111101	—	—	—
FPCHG		~FPSCR.PR → FPSCR.PR	1111011111111101	—	—	New
FSRRA	FRn	1/sqrt(FRn) → FRn	1111nnnn01111101	—	—	New
FSCA	FPUL,DRn	sin(FPUL) → FRn cos(FPUL) → FR[n + 1]	1111nnn011111101	—	—	New

Note: * sqrt(FRn) is the square root of FRn.

Section 4 Pipelining

This LSI is a 2-ILP (instruction-level-parallelism) superscalar pipelining microprocessor. Instruction execution is pipelined, and two instructions can be executed in parallel.

4.1 Pipelines

Figure 4.1 shows the basic pipelines. Normally, a pipeline consists of eight stages: instruction fetch (I1/I2/I3), decode and register read (ID), execution (E1/E2/E3), and write-back (WB). An instruction is executed as a combination of basic pipelines.

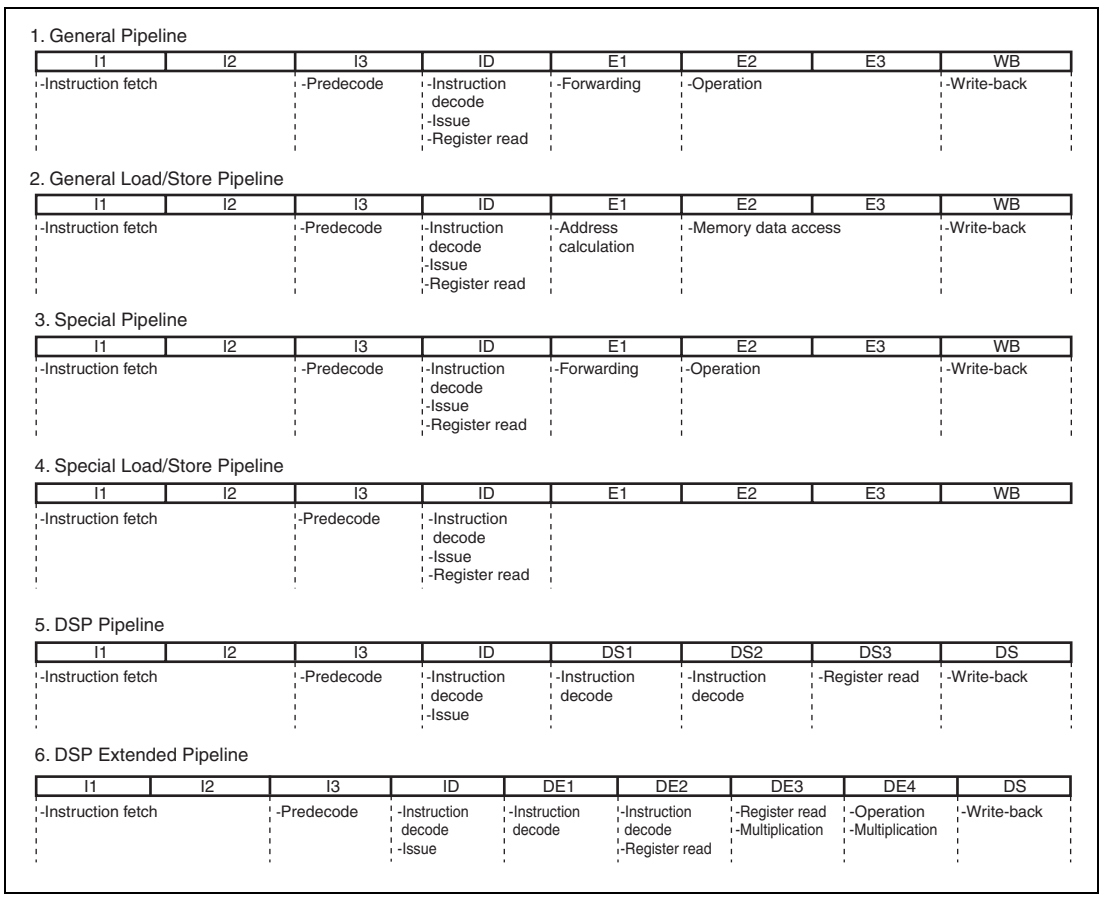


Figure 4.1 Basic Pipelines

Figure 4.2 shows the instruction execution patterns. Representations in figure 4.2 and their descriptions are listed in table 4.1.

Table 4.1 Representations of Instruction Execution Patterns

Representation	Description
E1 E2 E3 WB	CPU EX pipe is occupied
S1 S2 S3 WB	CPU LS pipe is occupied (with memory access)
s1 s2 s3 WB	CPU LS pipe is occupied (without memory access)
E1/S1	Either CPU EX pipe or CPU LS pipe is occupied
E1S1 , E1s1	Both CPU EX pipe and CPU LS pipe are occupied
M2 M3 MS	CPU MULT operation unit is occupied
DE1 DE2 DE3 DE4 DS	FPU-EX pipe is occupied
DS1 DS2 DS3 DS	FPU-LS pipe is occupied
ID	ID stage is locked
└──	Both CPU and FPU pipes are occupied

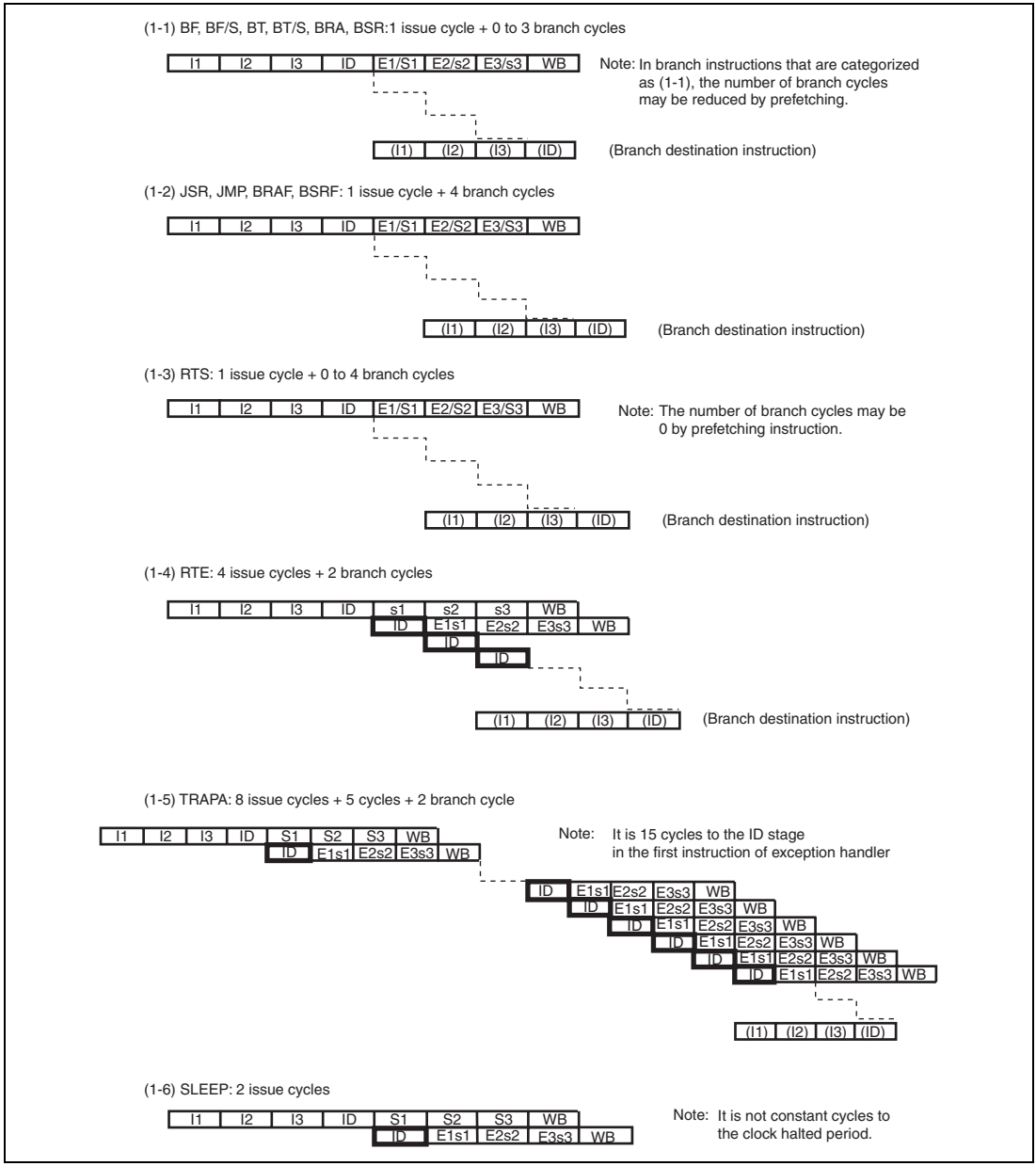


Figure 4.2 Instruction Execution Patterns (1)

(2-1) 1-step operation (EX type): 1 issue cycle

EXT[SU].[BW], MOVT, SWAP, XTRCT, ADD*, CMP*, DIV*, DT, NEG*, SUB*, AND, AND#, NOT, OR, OR#, TST, TST#, XOR, XOR#, ROT*, SHA*, SHL*, CLRS, CLRT, SETS, SETT, SETDMX, SETDMY, CLRDMXY

Notes: Except for AND#, OR#, TST#, and XOR# instructions using GBR relative addressing mode

I1	I2	I3	ID	E1	E2	E3	WB

(2-2) 1-step operation (LS type): 1 issue cycle

MOVA

I1	I2	I3	ID	s1	s2	s3	WB

(2-3) 1-step operation (MT type): 1 issue cycle

MOV#, NOP

I1	I2	I3	ID	E1/S1	E2/s2	E3/s3	WB

(2-4) MOV (MT type): 1 issue cycle

MOV

I1	I2	I3	ID	E1/s1	E2/s2	E3/S3	WB

(2-5) LDRS, LDRE (LS type): 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(2-6) LDRC, SETRC (CO type): 2 issue cycles

I1	I2	I3	ID	E1/s1	E2/s2	E3/s3	WB
				ID			

Figure 4.2 Instruction Execution Patterns (2)

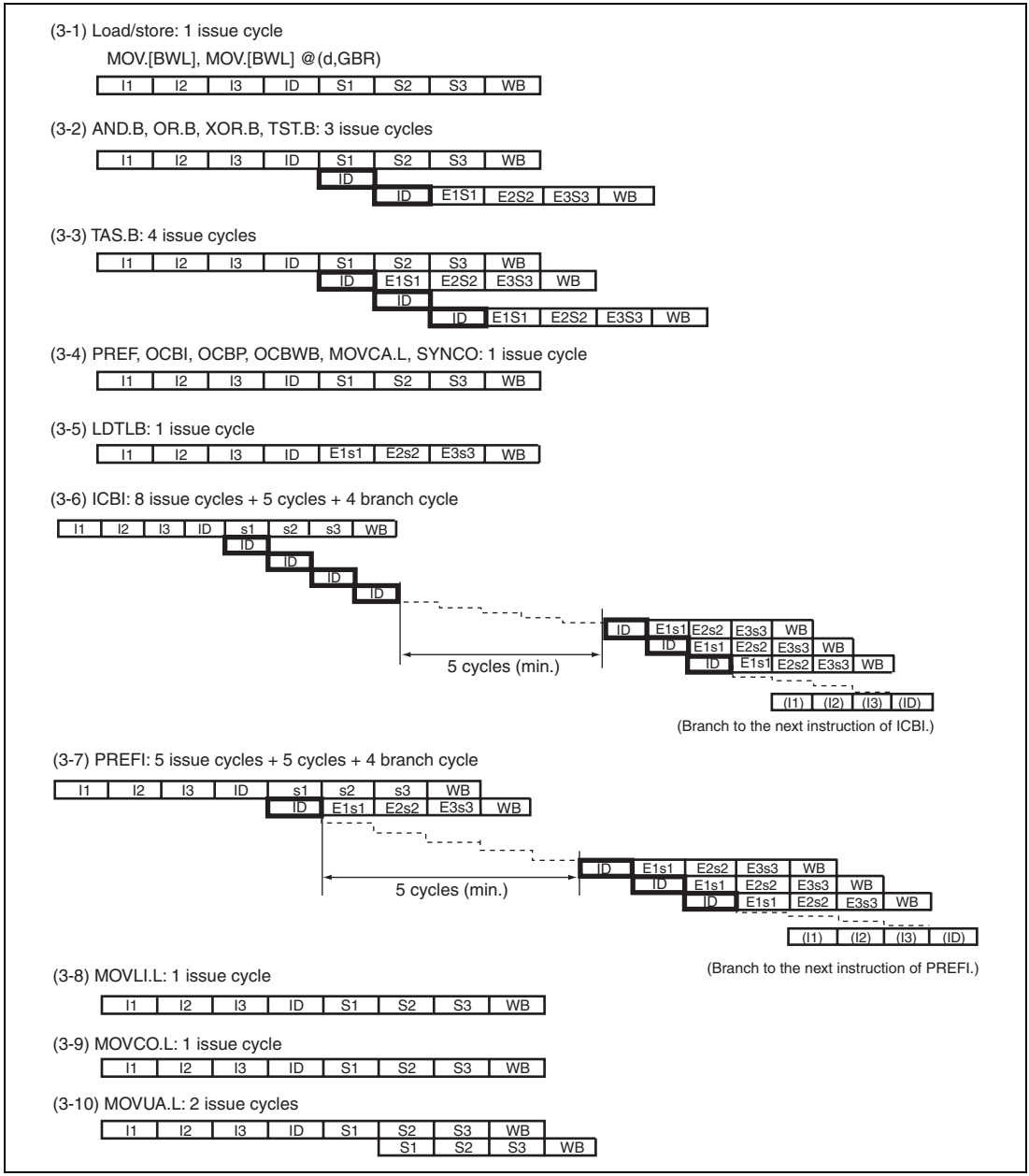


Figure 4.2 Instruction Execution Patterns (3)

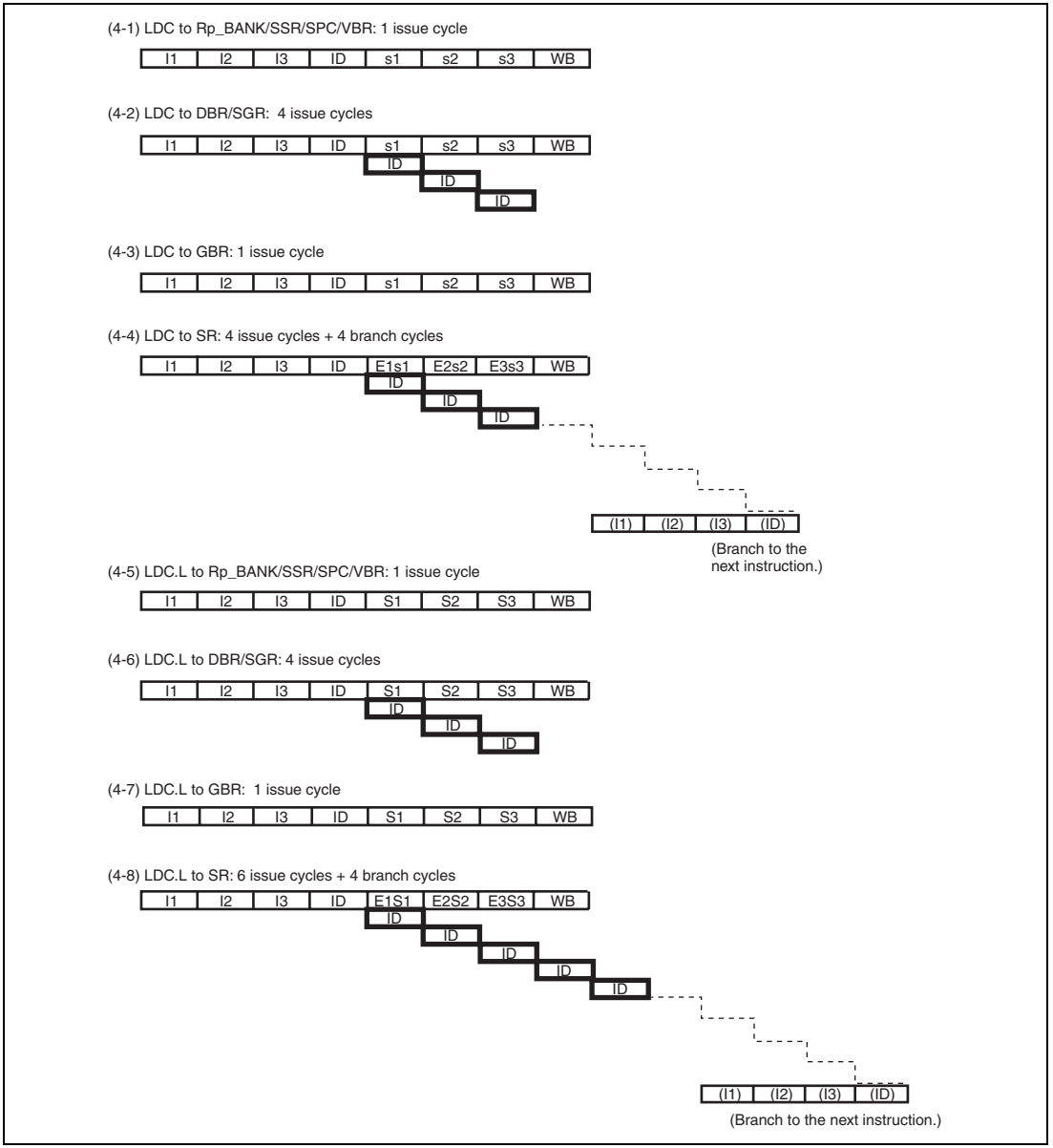


Figure 4.2 Instruction Execution Patterns (4)

(4-9) STC from DBR/GBR/Rp_BANK/SSR/SPC/VBR/SGR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-10) STC from RS/RE/MOD: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-11) STC from SR: 1 issue cycle

I1	I2	I3	ID	E1s1	E2s2	E3s3	WB

(4-12) STC.L from DBR/GBR/Rp_BANK/SSR/SPC/VBR/SGR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-13) STC.L from RS/RE/MOD: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-14) STC.L from SR: 1 issue cycle

I1	I2	I3	ID	E1S1	E2S2	E3S3	WB

(4-15) LDS to PR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-16) LDS.L to PR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-17) STS from PR: 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	WB

(4-18) STS.L from PR: 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB

(4-19) BSRF, BSR, JSR delay slot instructions (PR set): 0 issue cycle

(I1)	(I2)	(I3)	(ID)	(??1)	(??2)	(??3)	(WB)

Note: The value of PR is changed in the E3 stage of delay slot instruction.
When the STS and STS.L instructions from PR are used as delay slot instructions, changed PR value is used.

Figure 4.2 Instruction Execution Patterns (5)

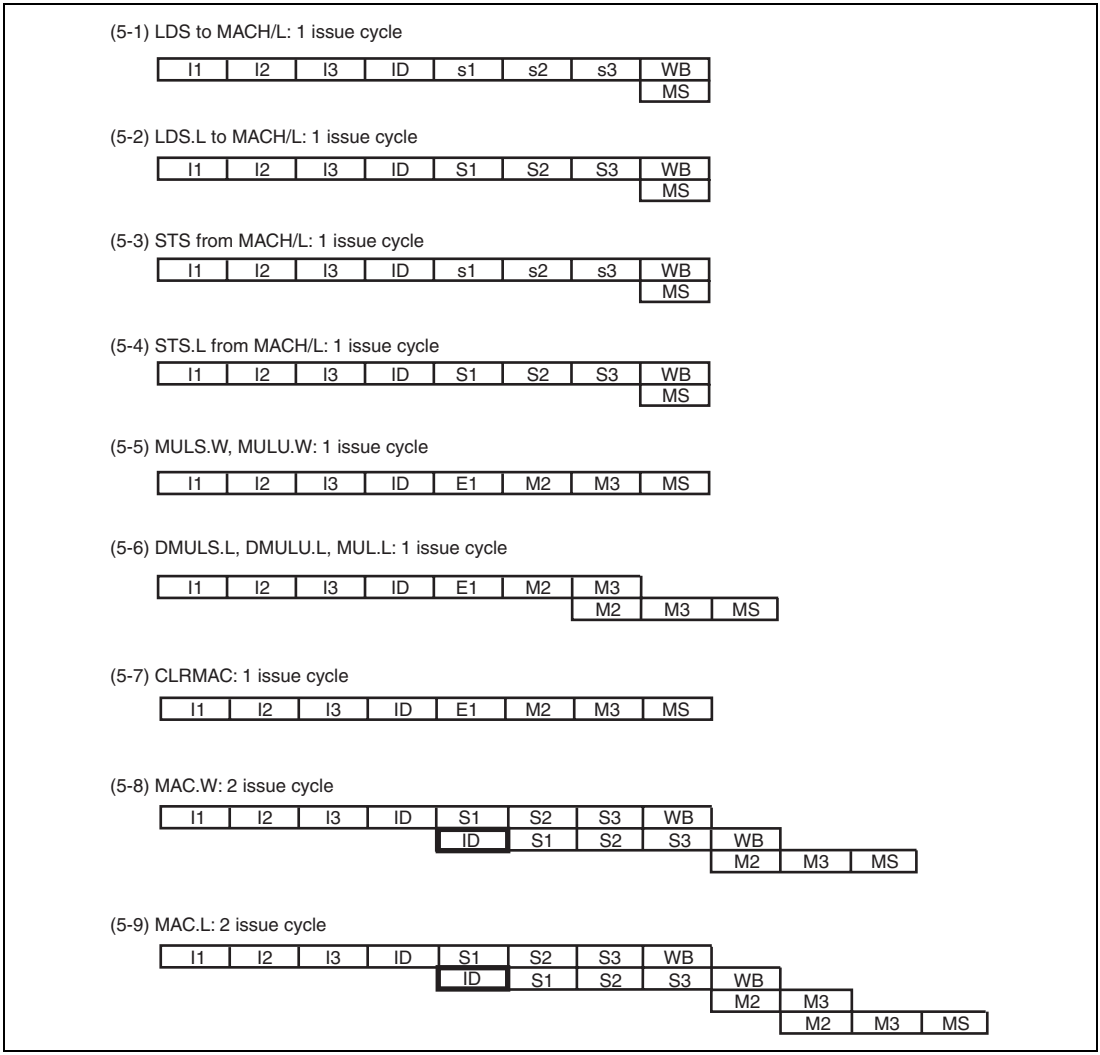


Figure 4.2 Instruction Execution Patterns (6)

(6-1) LDS to DSP (DSR, A0, X0, X1, Y0, Y1) : 1 issue cycle

I1	I2	I3	ID	s1	s2	s3	
				DS1	DS2	DS3	DS

(6-2) STS from DSP (DSR, A0, X0, X1, Y0, Y1) : 1 issue cycle

I1	I2	I3	ID	DS1	DS2	DS3	DS4
				s1	s2	s3	WB

(6-3) LDS.L to DSP (DSR, A0, X0, X1, Y0, Y1) : 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB
				DS1	DS2	DS3	DS

(6-4) STS.L from DSP (DSR, A0, X0, X1, Y0, Y1) : 1 issue cycle

I1	I2	I3	ID	DS1	DS2	DS3	DS4
				S1	S2	S3	WB

(6-5) MOVS : 1 issue cycle

I1	I2	I3	ID	S1	S2	S3	WB
				DS1	DS2	DS3	DS

(6-6) MOVX/NOPX, MOVY/NOPY (Without DSP operation, 16-bit instruction) : 1 issue cycle

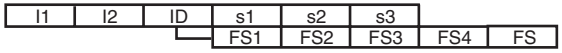
I1	I2	I3	ID	S1	S2	S3	WB
				DS1	DS2	DS3	DS

(6-7) MOVX/NOPX, MOVY/NOPY (With DSP operation, 32-bit instruction) : 1 issue cycle

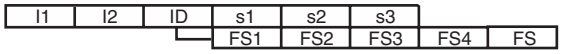
I1	I2	I3	ID	S1	S2	S3	WB	
				DS1	DS2	DS3	DS	
				DE1	DE2	DE3	DE4	DS

Figure 4.2 Instruction Execution Patterns (7)

(6-12) Single-precision FABS, FNEG/double-precision FABS, FNEG: 1 issue cycle

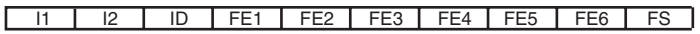


(6-13) FLDI0, FLDI1: 1 issue cycle

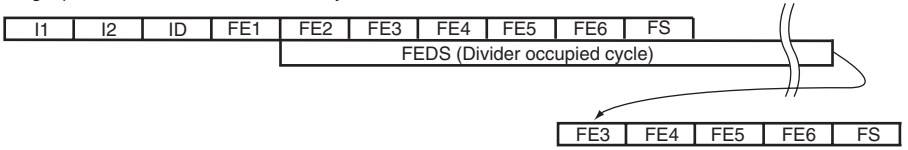


(6-14) Single-precision floating-point computation: 1 issue cycle

FCMP/EQ, FCMP/GT, FADD, FLOAT, FMAC, FMUL, FSUB, FTRC, FRCHG, FSCHG, FPCHG

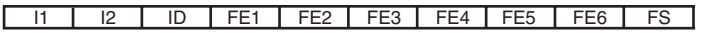


(6-15) Single-precision FDIV/FSQRT: 1 issue cycle



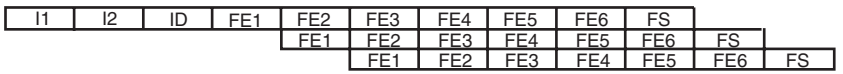
(6-16) Double-precision floating-point computation: 1 issue cycle

FCMP/EQ, FCMP/GT, FADD, FLOAT, FSUB, FTRC, FCNVSD, FCNVDS



(6-17) Double-precision floating-point computation: 1 issue cycle

FMUL



(6-18) Double-precision FDIV/FSQRT: 1 issue cycle

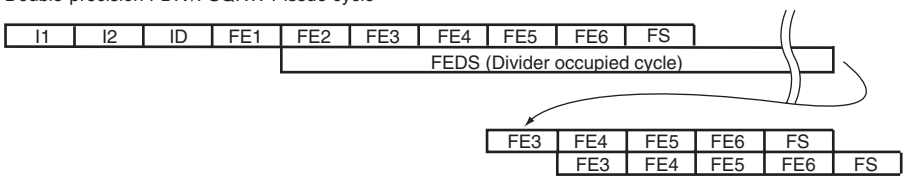


Figure 4.2 Instruction Execution Patterns (8)

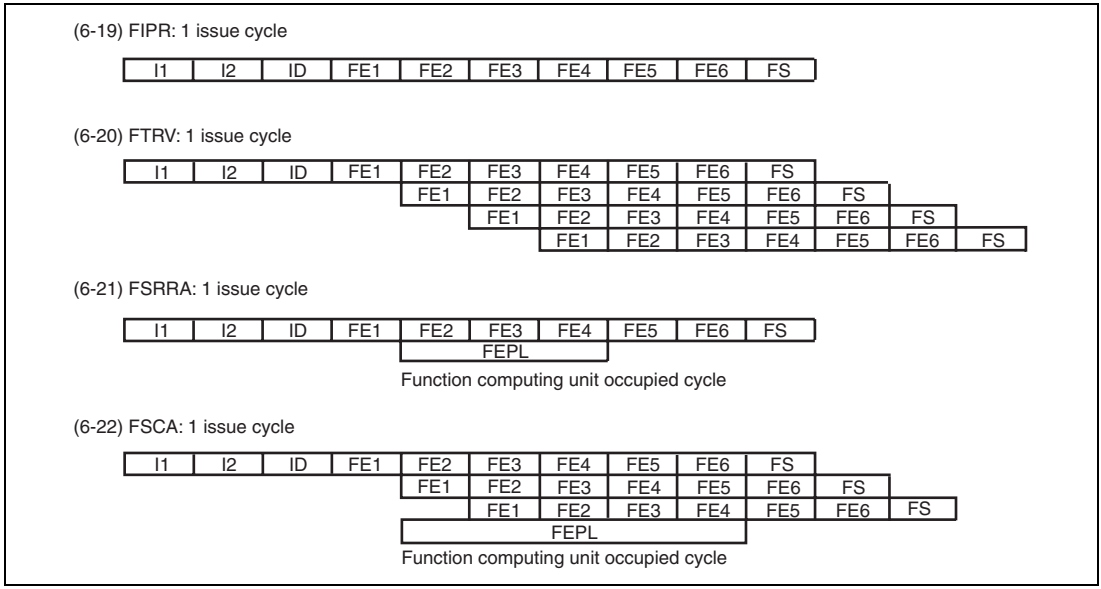


Figure 4.2 Instruction Execution Patterns (9)

4.2 Parallel-Executability

Instructions are categorized into six groups according to the internal function blocks used, as shown in table 4.2. Table 4.3 shows the parallel-executability of pairs of instructions in terms of groups. For example, ADD in the EX group and BRA in the BR group can be executed in parallel.

Table 4.2 Instruction Groups

Instruction Group	Instruction			
EX	ADD	DT	ROTL	SHLR8
	ADDC	EXTS	ROTR	SHLR16
	ADDV	EXTU	SETS	SUB
	AND #imm,R0	MOVT	SETT	SUBC
	AND Rm,Rn	MUL.L	SHAD	SUBV
	CLRMAC	MULS.W	SHAL	SWAP
	CLRS	MULU.W	SHAR	TST #imm,R0
	CLRT	NEG	SHLD	TST Rm,Rn
	CMP	NEGC	SHLL	XOR #imm,R0
	DIV0S	NOT	SHLL2	XOR Rm,Rn
	DIV0U	OR #imm,R0	SHLL8	XTRCT
	DIV1	OR Rm,Rn	SHLL16	
	DMUS.L	ROTCL	SHLR	
	DMULU.L	ROTCR	SHLR2	
MT	MOV #imm,Rn	MOV Rm,Rn	NOP	
BR	BF	BRAF	BT	JSR
	BF/S	BSR	BT/S	RTS
	BRA	BSRF	JMP	
LS	FABS	FMOV.S FR,@adr	MOV.[BWL] @adr,R	STC CR2,Rn
	FNEG	FSTS	MOV.[BWL] R,@adr	STC.L CR2,@-Rn
	FLDI0	LDC Rm,CR1	MOVA	STS SR2,Rn
	FLDI1	LDC.L @Rm+,CR1	MOVCA.L	STS.L SR2,@-Rn
	FLDS	LDS Rm,SR1	MOVUA	STS SR1,Rn
	FMOV @adr,FR	LDS Rm,SR2	OCBI	STS.L SR1,@-Rn
	FMOV FR,@adr	LDS.L @adr,SR2	OCBP	
	FMOV FR,FR	LDS.L @Rm+,SR1	OCBWB	
	FMOV.S @adr,FR	LDS.L @Rm+,SR2	PREF	

Instruction Group	Instruction			
FE	FADD	FDIV	FRCHG	FSCA
	FSUB	FIPR	FSCHG	FSRRA
	FCMP (S/D)	FLOAT	FSQRT	FPCHG
	FCNVDS	FMAC	FTRC	
	FCNVSD	FMUL	FTRV	
CO	AND.B #imm,@(R0,GBR)	LDC.L @Rm+,SR	PREFI	TRAPA
	ICBI	LDTLB	RTE	TST.B #imm,@(R0,GBR)
	LDC Rm,DBR	MAC.L	SLEEP	XOR.B #imm,@(R0,GBR)
	LDC Rm, SGR	MAC.W	STC SR,Rn	
	LDC Rm,SR	MOVCO	STC.L SR,@-Rn	
	LDC.L @Rm+,DBR	MOVLI	SYNCO	
	LDC.L @Rm+,SGR	OR.B #imm,@(R0,GBR)	TAS.B	

[Legend]
R: Rm/Rn
@adr: Address
SR1: MACH/MACL/PR
SR2: FPUL/FPSCR
CR1: GBR/Rp_BANK/SPC/SSR/VBR
CR2: CR1/DBR/SGR
FR: FRm/FRn/DRm/DRn/XDm/XDn

The parallel execution of two instructions can be carried out under following conditions.

1. Both addr (preceding instruction) and addr+2 (following instruction) are specified within the minimum page size (1 Kbyte).
2. The execution of these two instructions is supported in table 4.3, Combination of Preceding and Following Instructions.
3. Data used by an instruction of addr does not conflict with data used by a previous instruction
4. Data used by an instruction of addr+2 does not conflict with data used by a previous instruction
5. Both instructions are valid

Table 4.3 Combination of Preceding and Following Instructions

		Preceding Instruction (addr)					
		EX	**MT**	**BR**	**LS**	**FE**	**CO**
Following Instruction (addr+2)	EX	No	Yes	Yes	Yes	Yes	
	MT	Yes	Yes	Yes	Yes	Yes	
	BR	Yes	Yes	No	Yes	Yes	
	LS	Yes	Yes	Yes	No	Yes	
	FE	Yes	Yes	Yes	Yes	No	
	CO						No

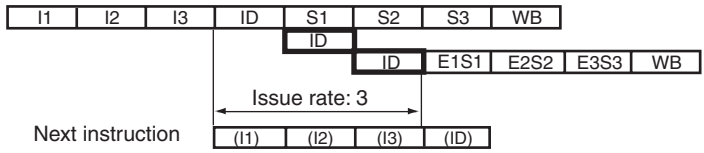
4.3 Issue Rates and Execution Cycles

Instruction execution cycles are summarized in table 4.4. Instruction Group in the table 4.4 corresponds to the category in the table 4.2. Penalty cycles due to a pipeline stall are not considered in the issue rates and execution cycles in this section.

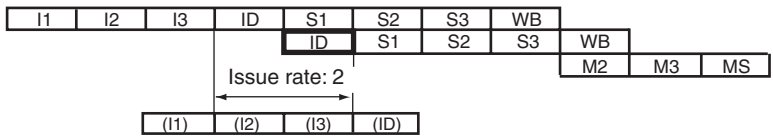
1. Issue Rate

Issue rates indicates the issue period between one instruction and next instruction.

E.g. AND.B instruction



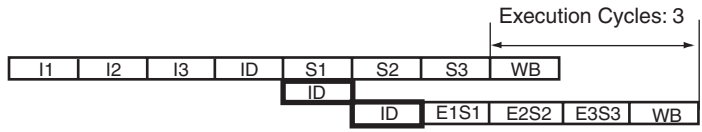
E.g. MAC.W instruction



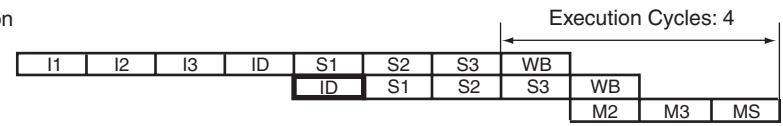
2. Execution Cycles

Execution cycles indicates the cycle counts an instruction occupied the pipeline based on the next rules.

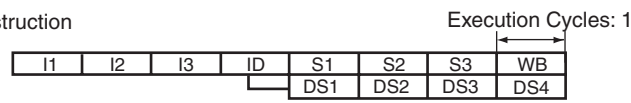
CPU instruction
E.g. AND.B instruction



E.g. MAC.W instruction



DSP instruction
E.g. MOVX, MOVY instruction



E.g. NOPX, NOPY, PADD instruction

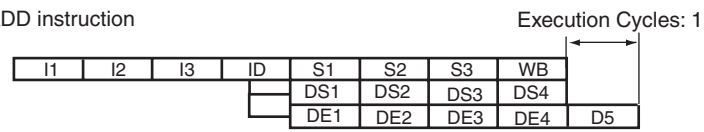


Table 4.4 Issue Rates and Execution Cycles

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Data transfer instructions	1	EXTS.B	Rm,Rn	EX	1	1	2-1
	2	EXTS.W	Rm,Rn	EX	1	1	2-1
	3	EXTU.B	Rm,Rn	EX	1	1	2-1
	4	EXTU.W	Rm,Rn	EX	1	1	2-1
	5	MOV	Rm,Rn	MT	1	1	2-4
	6	MOV	#imm,Rn	MT	1	1	2-3
	7	MOVA	@(disp,PC),R0	LS	1	1	2-2
	8	MOV.W	@(disp,PC),Rn	LS	1	1	3-1
	9	MOV.L	@(disp,PC),Rn	LS	1	1	3-1
	10	MOV.B	@Rm,Rn	LS	1	1	3-1
	11	MOV.W	@Rm,Rn	LS	1	1	3-1
	12	MOV.L	@Rm,Rn	LS	1	1	3-1
	13	MOV.B	@Rm+,Rn	LS	1	1	3-1
	14	MOV.W	@Rm+,Rn	LS	1	1	3-1
	15	MOV.L	@Rm+,Rn	LS	1	1	3-1
	16	MOV.B	@(disp,Rm),R0	LS	1	1	3-1
	17	MOV.W	@(disp,Rm),R0	LS	1	1	3-1
	18	MOV.L	@(disp,Rm),Rn	LS	1	1	3-1
	19	MOV.B	@(R0,Rm),Rn	LS	1	1	3-1
	20	MOV.W	@(R0,Rm),Rn	LS	1	1	3-1
	21	MOV.L	@(R0,Rm),Rn	LS	1	1	3-1
	22	MOV.B	@(disp,GBR),R0	LS	1	1	3-1
	23	MOV.W	@(disp, GBR),R0	LS	1	1	3-1
	24	MOV.L	@(disp, GBR),R0	LS	1	1	3-1
	25	MOV.B	Rm,@Rn	LS	1	1	3-1
	26	MOV.W	Rm,@Rn	LS	1	1	3-1
	27	MOV.L	Rm,@Rn	LS	1	1	3-1
	28	MOV.B	Rm,@-Rn	LS	1	1	3-1
	29	MOV.W	Rm,@-Rn	LS	1	1	3-1
	30	MOV.L	Rm,@-Rn	LS	1	1	3-1

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Data transfer instructions	31	MOV.B	R0,@(disp,Rn)	LS	1	1	3-1
	32	MOV.W	R0,@(disp,Rn)	LS	1	1	3-1
	33	MOV.L	Rm,@(disp,Rn)	LS	1	1	3-1
	34	MOV.B	Rm,@(R0,Rn)	LS	1	1	3-1
	35	MOV.W	Rm,@(R0,Rn)	LS	1	1	3-1
	36	MOV.L	Rm,@(R0,Rn)	LS	1	1	3-1
	37	MOV.B	R0,@(disp,GBR)	LS	1	1	3-1
	38	MOV.W	R0,@(disp,GBR)	LS	1	1	3-1
	39	MOV.L	R0,@(disp,GBR)	LS	1	1	3-1
	40	MOVCA.L	R0,@Rn	LS	1	1	3-4
	41	MOVCO.L	R0,@Rn	CO	1	1	3-9
	42	MOVLI.L	@Rm,R0	CO	1	1	3-8
	43	MOVUA.L	@Rm,R0	LS	2	2	3-10
	44	MOVUA.L	@Rm+,R0	LS	2	2	3-10
	45	MOVT	Rn	EX	1	1	2-1
	46	OCBI	@Rn	LS	1	1	3-4
	47	OCBP	@Rn	LS	1	1	3-4
	48	OCBWB	@Rn	LS	1	1	3-4
	49	PREF	@Rn	LS	1	1	3-4
	50	SWAP.B	Rm,Rn	EX	1	1	2-1
	51	SWAP.W	Rm,Rn	EX	1	1	2-1
	52	XTRCT	Rm,Rn	EX	1	1	2-1
Fixed-point arithmetic instructions	53	ADD	Rm,Rn	EX	1	1	2-1
	54	ADD	#imm,Rn	EX	1	1	2-1
	55	ADDC	Rm,Rn	EX	1	1	2-1
	56	ADDV	Rm,Rn	EX	1	1	2-1
	57	CMP/EQ	#imm,R0	EX	1	1	2-1
	58	CMP/EQ	Rm,Rn	EX	1	1	2-1
	59	CMP/GE	Rm,Rn	EX	1	1	2-1
	60	CMP/GT	Rm,Rn	EX	1	1	2-1
	61	CMP/HI	Rm,Rn	EX	1	1	2-1
	62	CMP/HS	Rm,Rn	EX	1	1	2-1

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Fixed-point arithmetic instructions	63	CMP/PL	Rn	EX	1	1	2-1
	64	CMP/PZ	Rn	EX	1	1	2-1
	65	CMP/STR	Rm,Rn	EX	1	1	2-1
	66	DIV0S	Rm,Rn	EX	1	1	2-1
	67	DIV0U		EX	1	1	2-1
	68	DIV1	Rm,Rn	EX	1	1	2-1
	69	DMULS.L	Rm,Rn	EX	1	2	5-6
	70	DMULU.L	Rm,Rn	EX	1	2	5-6
	71	DT	Rn	EX	1	1	2-1
	72	MAC.L	@Rm+,@Rn+	CO	2	5	5-9
	73	MAC.W	@Rm+,@Rn+	CO	2	4	5-8
	74	MUL.L	Rm,Rn	EX	1	2	5-6
	75	MULS.W	Rm,Rn	EX	1	1	5-5
	76	MULU.W	Rm,Rn	EX	1	1	5-5
	77	NEG	Rm,Rn	EX	1	1	2-1
	78	NEGC	Rm,Rn	EX	1	1	2-1
	79	SUB	Rm,Rn	EX	1	1	2-1
	80	SUBC	Rm,Rn	EX	1	1	2-1
	81	SUBV	Rm,Rn	EX	1	1	2-1
Logical instructionss	82	AND	Rm,Rn	EX	1	1	2-1
	83	AND	#imm,R0	EX	1	1	2-1
	84	AND.B	#imm,@(R0,GBR)	CO	3	3	3-2
	85	NOT	Rm,Rn	EX	1	1	2-1
	86	OR	Rm,Rn	EX	1	1	2-1
	87	OR	#imm,R0	EX	1	1	2-1
	88	OR.B	#imm,@(R0,GBR)	CO	3	3	3-2
	89	TAS.B	@Rn	CO	4	4	3-3
	90	TST	Rm,Rn	EX	1	1	2-1
	91	TST	#imm,R0	EX	1	1	2-1
	92	TST.B	#imm,@(R0,GBR)	CO	3	3	3-2
	93	XOR	Rm,Rn	EX	1	1	2-1
	94	XOR	#imm,R0	EX	1	1	2-1
	95	XOR.B	#imm,@(R0,GBR)	CO	3	3	3-2

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Shift instructions	96	ROTL	Rn	EX	1	1	2-1
	97	ROTR	Rn	EX	1	1	2-1
	98	ROTCL	Rn	EX	1	1	2-1
	99	ROTCR	Rn	EX	1	1	2-1
	100	SHAD	Rm,Rn	EX	1	1	2-1
	101	SHAL	Rn	EX	1	1	2-1
	102	SHAR	Rn	EX	1	1	2-1
	103	SHLD	Rm,Rn	EX	1	1	2-1
	104	SHLL	Rn	EX	1	1	2-1
	105	SHLL2	Rn	EX	1	1	2-1
	106	SHLL8	Rn	EX	1	1	2-1
	107	SHLL16	Rn	EX	1	1	2-1
	108	SHLR	Rn	EX	1	1	2-1
	109	SHLR2	Rn	EX	1	1	2-1
	110	SHLR8	Rn	EX	1	1	2-1
	111	SHLR16	Rn	EX	1	1	2-1
Branch instructions	112	BF	disp	BR	1+0 to 2	1	1-1
	113	BF/S	disp	BR	1+0 to 2	1	1-1
	114	BT	disp	BR	1+0 to 2	1	1-1
	115	BT/S	disp	BR	1+0 to 2	1	1-1
	116	BRA	disp	BR	1+0 to 2	1	1-1
	117	BRAF	Rm	BR	1+3	1	1-2
	118	BSR	disp	BR	1+0 to 2	1	1-1
	119	BSRF	Rm	BR	1+3	1	1-2
	120	JMP	@Rn	BR	1+3	1	1-2
	121	JSR	@Rn	BR	1+3	1	1-2
	122	RTS		BR	1+0 to 3	1	1-3

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
System control instruction	123	NOP		MT	1	1	2-3
	124	CLRMAC		EX	1	1	5-7
	125	CLRS		EX	1	1	2-1
	126	CLRT		EX	1	1	2-1
	127	ICBI	@Rn	CO	8+5+3	13	3-6
	128	SETS		EX	1	1	2-1
	129	SETT		EX	1	1	2-1
	130	PREFI	@Rn	CO	5+5+3	10	3-7
	131	SYNCO		CO	Undefined	Undefined	3-4
	132	TRAPA	#imm	CO	8+5+1	13	1-5
	133	RTE		CO	4+1	4	1-4
	134	SLEEP		CO	Undefined	Undefined	1-6
	135	LDTLB		CO	1	1	3-5
	136	LDC	Rm,DBR	CO	4	4	4-2
	137	LDC	Rm,SGR	CO	4	4	4-2
	138	LDC	Rm,GBR	LS	1	1	4-3
	139	LDC	Rm,Rp_BANK	LS	1	1	4-1
	140	LDC	Rm,SR	CO	4+3	4	4-4
	141	LDC	Rm,SSR	LS	1	1	4-1
	142	LDC	Rm,SPC	LS	1	1	4-1
	143	LDC	Rm,VBR	LS	1	1	4-1
	144	LDC.L	@Rm+,DBR	CO	4	4	4-6
	145	LDC.L	@Rm+,SGR	CO	4	4	4-6
	146	LDC.L	@Rm+,GBR	LS	1	1	4-7
	147	LDC.L	@Rm+,Rp_BANK	LS	1	1	4-5
	148	LDC.L	@Rm+,SR	CO	6+3	4	4-8
	149	LDC.L	@Rm+,SSR	LS	1	1	4-5
	150	LDC.L	@Rm+,SPC	LS	1	1	4-5
	151	LDC.L	@Rm+,VBR	LS	1	1	4-5
	152	LDS	Rm,MACH	LS	1	1	5-1
	153	LDS	Rm,MACL	LS	1	1	5-1
	154	LDS	Rm,PR	LS	1	1	4-13

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
System control instructions	155	LDS.L	@Rm+,MACH	LS	1	1	5-2
	156	LDS.L	@Rm+,MACL	LS	1	1	5-2
	157	LDS.L	@Rm+,PR	LS	1	1	4-14
	158	STC	DBR,Rn	LS	1	1	4-9
	159	STC	SGR,Rn	LS	1	1	4-9
	160	STC	GBR,Rn	LS	1	1	4-9
	161	STC	Rp_BANK,Rn	LS	1	1	4-9
	162	STC	SR,Rn	CO	1	1	4-10
	163	STC	SSR,Rn	LS	1	1	4-9
	164	STC	SPC,Rn	LS	1	1	4-9
	165	STC	VBR,Rn	LS	1	1	4-9
	166	STC.L	DBR,@-Rn	LS	1	1	4-11
	167	STC.L	SGR,@-Rn	LS	1	1	4-11
	168	STC.L	GBR,@-Rn	LS	1	1	4-11
	169	STC.L	Rp_BANK,@-Rn	LS	1	1	4-11
	170	STC.L	SR,@-Rn	CO	1	1	4-12
	171	STC.L	SSR,@-Rn	LS	1	1	4-11
	172	STC.L	SPC,@-Rn	LS	1	1	4-11
	173	STC.L	VBR,@-Rn	LS	1	1	4-11
	174	STS	MACH,Rn	LS	1	1	5-3
	175	STS	MACL,Rn	LS	1	1	5-3
	176	STS	PR,Rn	LS	1	1	4-15
	177	STS.L	MACH,@-Rn	LS	1	1	5-4
	178	STS.L	MACL,@-Rn	LS	1	1	5-4
	179	STS.L	PR,@-Rn	LS	1	1	4-16
Single-precision floating-point instructions	180	FLDI0	FRn	LS	1	1	6-13
	181	FLDI1	FRn	LS	1	1	6-13
	182	FMOV	FRm,FRn	LS	1	1	6-9
	183	FMOV.S	@Rm,FRn	LS	1	1	6-9
	184	FMOV.S	@Rm+,FRn	LS	1	1	6-9
	185	FMOV.S	@(R0,Rm),FRn	LS	1	1	6-9
	186	FMOV.S	FRm,@Rn	LS	1	1	6-9

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Single-precision floating-point instructions	187	FMOV.S	FRm,@-Rn	LS	1	1	6-9
	188	FMOV.S	FRm,@(R0,Rn)	LS	1	1	6-9
	189	FLDS	FRm,FPUL	LS	1	1	6-10
	190	FSTS	FPUL,FRn	LS	1	1	6-11
	191	FABS	FRn	LS	1	1	6-12
	192	FADD	FRm,FRn	FE	1	1	6-14
	193	FCMP/EQ	FRm,FRn	FE	1	1	6-14
	194	FCMP/GT	FRm,FRn	FE	1	1	6-14
	195	FDIV	FRm,FRn	FE	1	14	6-15
	196	FLOAT	FPUL,FRn	FE	1	1	6-14
	197	FMAC	FR0,FRm,FRn	FE	1	1	6-14
	198	FMUL	FRm,FRn	FE	1	1	6-14
	199	FNEG	FRn	LS	1	1	6-12
	200	FSQRT	FRn	FE	1	14	6-15
	201	FSUB	FRm,FRn	FE	1	1	6-14
	202	FTRC	FRm,FPUL	FE	1	1	6-14
	203	FMOV	DRm,DRn	LS	1	1	6-9
	204	FMOV	@Rm,DRn	LS	1	1	6-9
	205	FMOV	@Rm+,DRn	LS	1	1	6-9
	206	FMOV	@(R0,Rm),DRn	LS	1	1	6-9
	207	FMOV	DRm,@Rn	LS	1	1	6-9
	208	FMOV	DRm,@-Rn	LS	1	1	6-9
	209	FMOV	DRm,@(R0,Rn)	LS	1	1	6-9
Double-precision floating-point instructions	210	FABS	DRn	LS	1	1	6-12
	211	FADD	DRm,DRn	FE	1	1	6-16
	212	FCMP/EQ	DRm,DRn	FE	1	1	6-16
	213	FCMP/GT	DRm,DRn	FE	1	1	6-16
	214	FCNVDS	DRm,FPUL	FE	1	1	6-16
	215	FCNVSD	FPUL,DRn	FE	1	1	6-16
	216	FDIV	DRm,DRn	FE	1	30	6-18
	217	FLOAT	FPUL,DRn	FE	1	1	6-16
	218	FMUL	DRm,DRn	FE	1	3	6-17

Functional Category	No.	Instruction		Instruction Group	Issue Rate	Execution Cycles	Execution Pattern
Double-precision floating-point instructions	219	FNEG	DRn	LS	1	1	6-12
	220	FSQRT	DRn	FE	1	30	6-18
	221	FSUB	DRm,DRn	FE	1	1	6-16
	222	FTRC	DRm,FPUL	FE	1	1	6-16
FPU system control instructions	223	LDS	Rm,FPUL	LS	1	1	6-1
	224	LDS	Rm,FPSCR	LS	1	1	6-5
	225	LDS.L	@Rm+,FPUL	LS	1	1	6-3
	226	LDS.L	@Rm+,FPSCR	LS	1	1	6-7
	227	STS	FPUL,Rn	LS	1	1	6-2
	228	STS	FPSCR,Rn	LS	1	1	6-6
	229	STS.L	FPUL,@-Rn	LS	1	1	6-4
	230	STS.L	FPSCR,@-Rn	LS	1	1	6-8
Graphics acceleration instructions	231	FMOV	DRm,XDn	LS	1	1	6-9
	232	FMOV	XDm,DRn	LS	1	1	6-9
	233	FMOV	XDm,XDn	LS	1	1	6-9
	234	FMOV	@Rm,XDn	LS	1	1	6-9
	235	FMOV	@Rm+,XDn	LS	1	1	6-9
	236	FMOV	@(R0,Rm),XDn	LS	1	1	6-9
	237	FMOV	XDm,@Rn	LS	1	1	6-9
	238	FMOV	XDm,@-Rn	LS	1	1	6-9
	239	FMOV	XDm,@(R0,Rn)	LS	1	1	6-9
	240	FIPR	FVm,FVn	FE	1	1	6-19
	241	FRCHG		FE	1	1	6-14
	242	FSCHG		FE	1	1	6-14
	243	FPCHG		FE	1	1	6-14
	244	FSRRA	FRn	FE	1	1	6-21
	245	FSCA	FPUL,DRn	FE	1	3	6-22
	246	FTRV	XMTRX,FVn	FE	1	4	6-20

Section 5 Exception Handling

5.1 Summary of Exception Handling

Exception handling processing is handled by a special routine which is executed by a reset, general exception handling, or interrupt. For example, if the executing instruction ends abnormally, appropriate action must be taken in order to return to the original program sequence, or report the abnormality before terminating the processing. The process of generating an exception handling request in response to abnormal termination, and passing control to a user-written exception handling routine, in order to support such functions, is given the generic name of exception handling.

The exception handling in this LSI is of three kinds: resets, general exceptions, and interrupts.

5.2 Register Descriptions

Table 5.1 lists the configuration of registers related exception handling.

Table 5.1 Register Configuration

Register Name	Abbr.	R/W	P4 Address*	Area 7 Address*	Access Size
TRAPA exception register	TRA	R/W	H'FF00 0020	H'1F00 0020	32
Exception event register	EXPEVT	R/W	H'FF00 0024	H'1F00 0024	32
Interrupt event register	INTEVT	R/W	H'FF00 0028	H'1F00 0028	32
Non-support detection exception register	EXPMASK	R/W	H'FF2F 0004	H'1F2F 0004	32

Note: * P4 is the address when virtual address space P4 area is used. Area 7 is the address when physical address space area 7 is accessed by using the TLB.

Table 5.2 States of Register in Each Operating Mode

Abbr.	Power-on Reset	Manual Reset	Software Standby	Module Standby	U-Standby	Sleep
TRAPA	Undefined	Undefined	Retained	—	Undefined	Retained
EXPEVT	H'0000 0000	H'0000 0020	Retained	—	H'0000 0000	Retained
INTEVT	Undefined	Undefined	Retained	—	Undefined	Retained
EXPMASK	H'0000 001F	H'0000 0001F	Retained	—	H'0000 001F	Retained

5.2.1 TRAPA Exception Register (TRA)

The TRAPA exception register (TRA) consists of 8-bit immediate data (imm) for the TRAPA instruction. TRA is set automatically by hardware when a TRAPA instruction is executed. TRA can also be modified by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRACODE								—	—
Initial value:	0	0	0	0	0	0	—	—	—	—	—	—	—	—	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
9 to 2	TRACODE	Undefined	R/W	TRAPA Code 8-bit immediate data of TRAPA instruction is set
1, 0	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.

5.2.2 Exception Event Register (EXPEVT)

The exception event register (EXPEVT) consists of a 12-bit exception code. The exception code set in EXPEVT is that for a reset or general exception event. The exception code is set automatically by hardware when an exception occurs. EXPEVT can also be modified by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	EXPCODE											
Initial value:	0	0	0	0	0	0	0	0	0	0	0/1	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
11 to 0	EXPCODE	H'000 or H'020	R/W	Exception Code The exception code for a reset or general exception is set. For details, see table 5.3.

5.2.3 Interrupt Event Register (INTEVT)

The interrupt event register (INTEVT) consists of a 14-bit exception code. The exception code is set automatically by hardware when an exception occurs. INTEVT can also be modified by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	INTCODE													
Initial value:	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved For details on reading/writing this bit, see General Precautions on Handling of Product.
13 to 0	INTCODE	Undefined	R/W	Exception Code The exception code for an interrupt is set. For details, see table 5.3.

5.2.4 Non-Support Detection Exception Register (EXPMASK)

The non-support detection exception register (EXPMASK) is used to enable or disable the generation of exceptions in response to the use of any of functions 1 to 3 listed below. The functions of 1 to 3 are planned not to be supported in the future SuperH-family products. The exception generation functions of EXPMASK can be used in advance of execution; the detection function then checks for the use of these functions in the software. This will ease the transfer of software to the future SuperH-family products that do not support the respective functions.

1. Handling of an instruction other than the NOP instruction in the delay slot of the RTE instruction.
2. Handling of the SLEEP instruction in the delay slot of the branch instruction.
3. Performance of IC/OC memory-mapped associative write operations.

According to the value of EXPMASK, functions 1 and 2 can generate a slot illegal instruction exception, and 3 can generate a data address error exception.

Generation of each exception can be disabled by writing 1 to the corresponding bit in EXPMASK. However, it is recommended that the above functions should not be used when making a program to maintain the compatibility with the future products.

Use the store instruction of the CPU to update EXPMASK. After updating the register and then reading the register once, execute either of the following instructions. Executing either instruction guarantees the operation with the updated register value.

- Execute the RTE instruction.
- Execute the ICBI instruction for any address (including non-cacheable area).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	–	–	–	MM CAW	–	–	BRDS SLP	RTE DS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.
4	MMCAW	1	R/W	Memory-Mapped Cache Associative Write 0: Memory-mapped cache associative write is disabled. (A data address error exception will occur.) 1: Memory-mapped cache associative write is enabled. For further details, refer to section 8.6.5, Memory-Mapped Cache Associative Write Operation.
3, 2	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.
1	BRDSSLP	1	R/W	Delay Slot SLEEP Instruction 0: The SLEEP instruction in the delay slot is disabled. (The SLEEP instruction is taken as a slot illegal instruction.) 1: The SLEEP instruction in the delay slot is enabled.
0	RTEDS	1	R/W	RTE Delay Slot 0: An instruction other than the NOP instruction in the delay slot of the RTE instruction is disabled. (An instruction other than the NOP instruction is taken as a slot illegal instruction). 1: An instruction other than the NOP instruction in the delay slot of the RTE instruction is enabled.

5.3 Exception Handling Functions

5.3.1 Exception Handling Flow

In exception handling, the contents of the program counter (PC), status register (SR), and R15 are saved in the saved program counter (SPC), saved status register (SSR), and saved general register15 (SGR), and the CPU starts execution of the appropriate exception handling routine according to the vector address. An exception handling routine is a program written by the user to handle a specific exception. The exception handling routine is terminated and control returned to the original program by executing a return-from-exception instruction (RTE). This instruction restores the PC and SR contents and returns control to the normal processing routine at the point at which the exception occurred. The SGR contents are not written back to R15 with an RTE instruction.

The basic processing flow is as follows. For the meaning of the SR bits, see section 2, Programming Model.

1. The PC, SR, and R15 contents are saved in SPC, SSR, and SGR, respectively.
2. The block bit (BL) in SR is set to 1.
3. The mode bit (MD) in SR is set to 1.
4. The register bank bit (RB) in SR is set to 1.
5. In a reset, the FPU disable bit (FD) in SR is cleared to 0.
6. The exception code is written to the bits 11 to 0 in exception event register (EXPEVT) or the bits 13 to 0 in interrupt event register (INTEVT).
7. When the interrupt mode switch bit (INTMU) in CPUOPM has been 1, the interrupt mask level bit (IMASK) in SR is changed to accepted interrupt level.
8. The CPU branches to the determined exception handling vector address, and the exception handling routine begins.

5.3.2 Exception Handling Vector Addresses

The reset vector address is fixed at H'A0000000. Exception and interrupt vector addresses are determined by adding the offset for the specific event to the vector base address, which is set by software in the vector base register (VBR). In the case of the TLB miss exception, for example, the offset is H'00000400, so if H'9C080000 is set in VBR, the exception handling vector address will be H'9C080400. If a further exception occurs at the exception handling vector address, a duplicate exception will result, and recovery will be difficult; therefore, addresses that are not to be converted (in P1 and P2 areas) should be specified for vector addresses.

5.4 Exception Types and Priorities

Table 5.3 shows the types of exceptions, with their relative priorities, vector addresses, and exception/interrupt codes.

Table 5.3 Exceptions

Exception Category	Execution Mode	Exception	Priority Level*[2]	Priority Order*[2]	Exception Transition Direction*[3] Vector Address	Exception Transition Direction*[3] Offset	Exception Code*[4]
Reset	Abort type	Power-on reset	1	1	H'A000 0000	—	H'000
		Manual reset	1	2	H'A000 0000	—	H'020
		H-UDI reset	1	1	H'A000 0000	—	H'000
		Instruction TLB multiple-hit exception	1	3	H'A000 0000	—	H'140
		Data TLB multiple-hit exception	1	4	H'A000 0000	—	H'140
General exception	Re-execution type	User break before instruction execution*	2	0	(VBR/DBR)	H'100/—	H'1E0
		Instruction address error	2	1	(VBR)	H'100	H'0E0
		Instruction TLB miss exception	2	2	(VBR)	H'400	H'040
		Instruction TLB protection violation exception	2	3	(VBR)	H'100	H'0A0
		General illegal instruction exception	2	4	(VBR)	H'100	H'180
		Slot illegal instruction exception	2	4	(VBR)	H'100	H'1A0
		General FPU disable exception	2	4	(VBR)	H'100	H'800
		Slot FPU disable exception	2	4	(VBR)	H'100	H'820
		Data address error (read)	2	5	(VBR)	H'100	H'0E0
		Data address error (write)	2	5	(VBR)	H'100	H'100
		Data TLB miss exception (read)	2	6	(VBR)	H'400	H'040
		Data TLB miss exception (write)	2	6	(VBR)	H'400	H'060
		Data TLB protection violation exception (read)	2	7	(VBR)	H'100	H'0A0
		Data TLB protection violation exception (write)	2	7	(VBR)	H'100	H'0C0
		FPU exception	2	8	(VBR)	H'100	H'120
		Initial page write exception	2	9	(VBR)	H'100	H'080

Exception Category	Execution Mode	Exception	Priority Level*[2]	Priority Order*[2]	Exception Transition Direction*[3] Vector Address	Exception Transition Direction*[3] Offset	Exception Code*[4]
General exception	Completion type	Unconditional trap (TRAPA)	2	4	(VBR)	H'100	H'160
		User break after instruction execution*	2	10	(VBR/DBR)	H'100/—	H'1E0
Interrupt	Completion type	Nonmaskable interrupt	3	—	(VBR)	H'600	H'1C0
		General interrupt request	4	—	(VBR)	H'600	—

Note: 1. When UBDE in CBCR = 1, PC = DBR. In other cases, PC = VBR + H'100.
2. Priority is first assigned by priority level, then by priority order within each level (the lowest number represents the highest priority).
3. Control passes to H'A000 0000 in a reset, and to [VBR + offset] in other cases.
4. Stored in EXPEVT for a reset or general exception, and in INTEVT for an interrupt.

5.5 Exception Flow

5.5.1 Exception Flow

Figure 5.1 shows an outline flowchart of the basic operations in instruction execution and exception handling. For the sake of clarity, the following description assumes that instructions are executed sequentially, one by one. Figure 5.1 shows the relative priority order of the different kinds of exceptions (reset, general exception, and interrupt). Register settings in the event of an exception are shown only for SSR, SPC, SGR, EXPEVT/INTEVT, SR, and PC. However, other registers may be set automatically by hardware, depending on the exception. For details, see section 5.6, Description of Exceptions. Also, see section 5.6.4, Priority Order with Multiple Exceptions, for exception handling during execution of a delayed branch instruction and a delay slot instruction, or in the case of instructions in which two data accesses are performed.

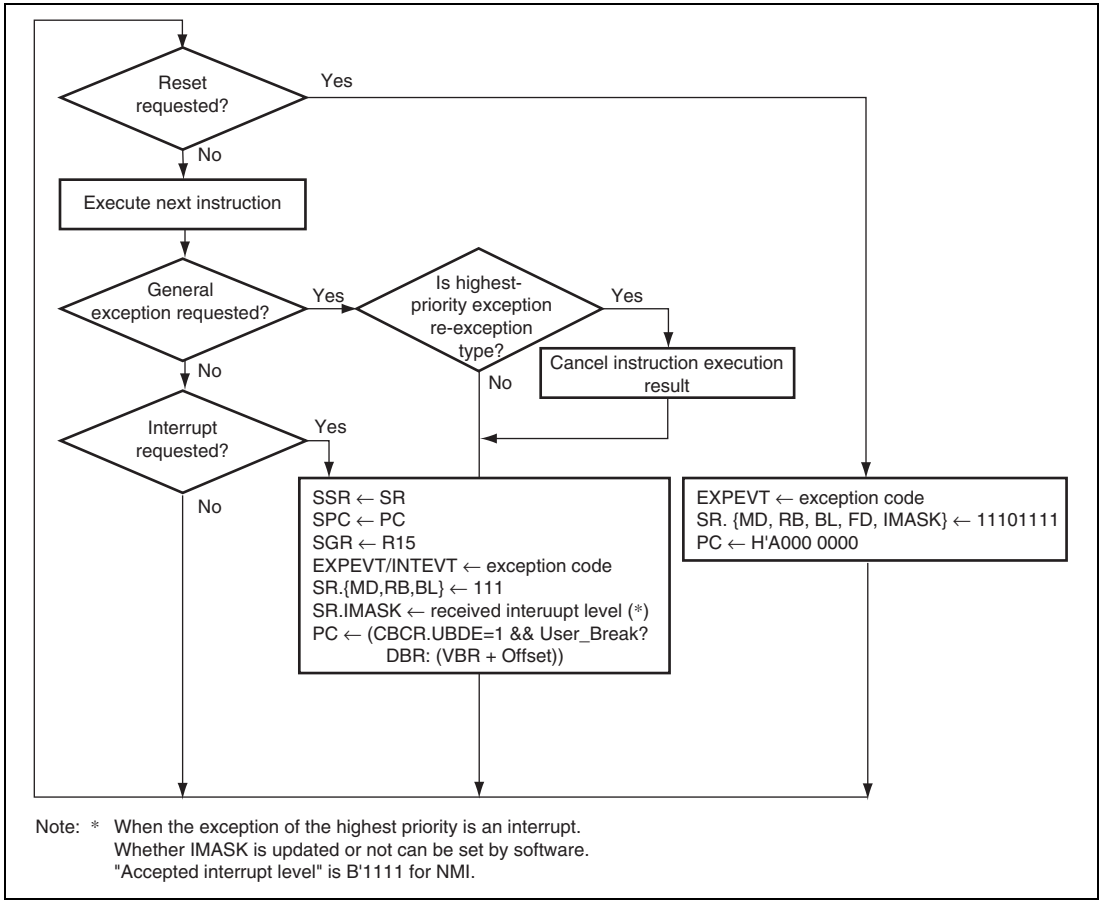


Figure 5.1 Instruction Execution and Exception Handling

5.5.2 Exception Source Acceptance

A priority ranking is provided for all exceptions for use in determining which of two or more simultaneously generated exceptions should be accepted. Five of the general exceptions—general illegal instruction exception, slot illegal instruction exception, general FPU disable exception, slot FPU disable exception, and unconditional trap exception—are detected in the process of instruction decoding, and do not occur simultaneously in the instruction pipeline. These exceptions therefore all have the same priority. General exceptions are detected in the order of instruction execution. However, exception handling is performed in the order of instruction flow (program order). Thus, an exception for an earlier instruction is accepted before that for a later instruction. An example of the order of acceptance for general exceptions is shown in figure 5.2.

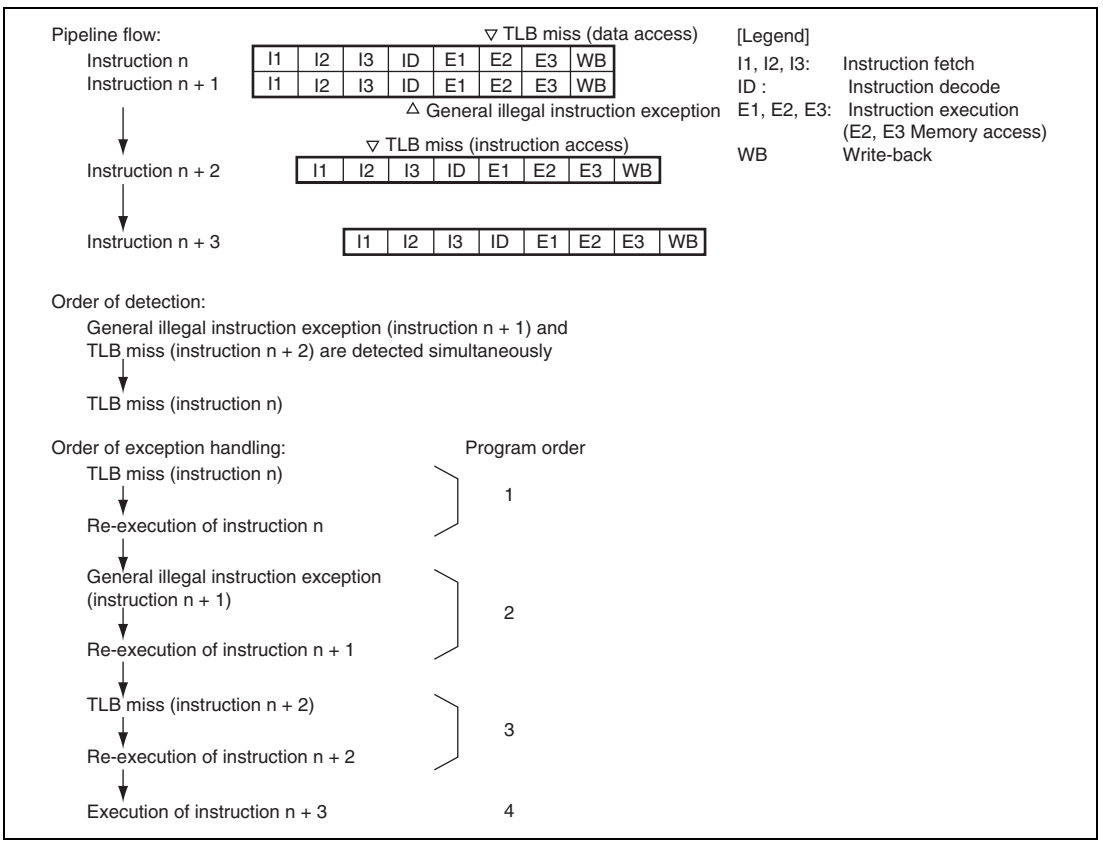


Figure 5.2 Example of General Exception Acceptance Order

5.5.3 Exception Requests and BL Bit

When the BL bit in SR is 0, general exceptions and interrupts are accepted.

When the BL bit in SR is 1 and an general exception other than a user break is generated, the CPU's internal registers and the registers of the other modules are set to their states following a manual reset, and the CPU branches to the same address as in a reset (H'A0000000). For the operation in the event of a user break, see section 49, User Break Controller (UBC). If an ordinary interrupt occurs, the interrupt request is held pending and is accepted after the BL bit has been cleared to 0 by software. If a nonmaskable interrupt (NMI) occurs, it can be held pending or accepted according to the setting made by software.

Thus, normally, SPC and SSR are saved and then the BL bit in SR is cleared to 0, to enable multiple exception state acceptance.

5.5.4 Return from Exception Handling

The RTE instruction is used to return from exception handling. When the RTE instruction is executed, the SPC contents are restored to PC and the SSR contents to SR, and the CPU returns from the exception handling routine by branching to the SPC address. If SPC and SSR were saved to external memory, set the BL bit in SR to 1 before restoring the SPC and SSR contents and issuing the RTE instruction.

5.6 Description of Exceptions

The various exception handling operations explained here are exception sources, transition address on the occurrence of exception, and processor operation when a transition is made.

5.6.1 Resets

(1) Power-On Reset

- Condition:
  Power-on reset request
- Operations:
  Exception code H'000 is set in EXPEVT, initialization of the CPU and on-chip peripheral module is carried out, and then a branch is made to the reset vector (H'A0000000). For details, see the register descriptions in the relevant sections. A power-on reset should be executed when power is supplied.

(2) Manual Reset

- Condition:
  Manual reset request
- Operations:
  Exception code H'020 is set in EXPEVT, initialization of the CPU and on-chip peripheral module is carried out, and then a branch is made to the branch vector (H'A0000000). The registers initialized by a power-on reset and manual reset are different. For details, see the register descriptions in the relevant sections.

(3) H-UDI Reset

- Source: SDIR.TI[7:4] = B'0110 (negation) or B'0111 (assertion)
- Transition address: H'A0000000
- Transition operations:
  Exception code H'000 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A0000000.
  CPU and on-chip peripheral module initialization is performed. For details, see the register descriptions in the relevant sections.

(4) Instruction TLB Multiple Hit Exception

- Source: Multiple ITLB address matches
- Transition address: H'A0000000
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  Exception code H'140 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A0000000.

  CPU and on-chip peripheral module initialization is performed in the same way as in a manual reset. For details, see the register descriptions in the relevant sections.

(5) Data TLB Multiple-Hit Exception

- Source: Multiple UTLB address matches
- Transition address: H'A0000000
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  Exception code H'140 is set in EXPEVT, initialization of VBR and SR is performed, and a branch is made to PC = H'A0000000.

  CPU and on-chip peripheral module initialization is performed in the same way as in a manual reset. For details, see the register descriptions in the relevant sections.

5.6.2 General Exceptions

(1) Data TLB Miss Exception

- Source: Address mismatch in UTLB address comparison
- Transition address: VBR + H'00000400
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'040 (for a read access) or H'060 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0400.

  To speed up TLB miss processing, the offset is separate from that of other exceptions.

```
Data_TLB_miss_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access ? H'0000 0040 : H'0000 0060;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0400;
}
```

(2) Instruction TLB Miss Exception

- Source: Address mismatch in ITLB address comparison
- Transition address: VBR + H'00000400
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'40 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0400.

  To speed up TLB miss processing, the offset is separate from that of other exceptions.

```
ITLB_miss_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0040;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0400;
}
```

(3) Initial Page Write Exception

- Source: TLB is hit in a store access, but dirty bit D = 0
- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'080 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
Initial_write_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0080;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(4) Data TLB Protection Violation Exception

- Source: The access does not accord with the UTLB protection information (PR bits or EPR bits) shown in table 5.4 and table 5.5.

Table 5.4 UTLB Protection Information (TLB Compatible Mode)

PR	Privileged Mode	User Mode
00	Only read access possible	Access not possible
01	Read/write access possible	Access not possible
10	Only read access possible	Only read access possible
11	Read/write access possible	Read/write access possible

Table 5.5 UTLB Protection Information (TLB Extended Mode)

EPR [5]	Read Permission in Privileged Mode
0	Read access possible
1	Read access not possible

EPR [4]	Write Permission in Privileged Mode
0	Write access possible
1	Write access not possible

EPR [2]	Read Permission in User Mode
0	Read access possible
1	Read access not possible

EPR [1]	Write Permission in User Mode
0	Write access possible
1	Write access not possible

- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0A0 (for a read access) or H'0C0 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
Data_TLB_protection_violation_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access ? H'0000 00A0 : H'0000 00C0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(5) Instruction TLB Protection Violation Exception

- Source: The access does not accord with the ITLB protection information (PR bits or EPR bits) shown in table 5.6 and table5.7.

Table 5.6 ITLB Protection Information (TLB Compatible Mode)

PR	Privileged Mode	User Mode
0	Access possible	Access not possible
1	Access possible	Access possible

Table 5.7 ITLB Protection Information (TLB Extended Mode)

EPR [5], EPR [3]	Execution Permission in Privileged Mode
11, 01	Execution of instructions possible
10	Instruction fetch not possible Execution of Rn access by ICBI possible
00	Execution of instructions not possible

EPR [2], EPR [0]	Execution Permission in User Mode
11, 01	Execution of instructions possible
10	Instruction fetch not possible Execution of Rn access by ICBI possible
00	Execution of instructions not possible

- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0A0 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
ITLB_protection_violation_exception()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 00A0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(6) Data Address Error

- Sources:
  - — Word data access from other than a word boundary (2n +1)
  - — Longword data access from other than a longword data boundary (4n +1, 4n + 2, or 4n +3) (Except MOVUA)
  - — Quadword data access from other than a quadword data boundary (8n +1, 8n + 2, 8n +3, 8n + 4, 8n + 5, 8n + 6, or 8n + 7)
  - — Access to area H'80000000 to H'FFFFFFFF in user mode

    Area H'E5000000 to H'E5FFFFFF can be accessed in user mode. For details, see section 7, Memory Management Unit (MMU) and section 10, On-chip Memory.
  - — IC/OC memory mapped associative write is performed when the MMCAW bit in EXPMASK is 0. For details of memory mapped associative write, see section 8.6.5, Memory-Mapped Cache Associative Write Operation.
- Transition address: VBR + H'0000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0E0 (for a read access) or H'100 (for a write access) is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. For details, see section 7, Memory Management Unit (MMU).

```
Data_address_error()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = read_access? H'0000 00E0: H'0000 0100;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(7) Instruction Address Error

- Sources:
  - — Instruction fetch from other than a word boundary (2n +1)
  - — Instruction fetch from area H'80000000 to H'FFFFFFFF in user mode

    Area H'E5000000 to H'E5FFFFFF can be accessed in user mode. For details, see section 10, On-Chip Memory.
- Transition address: VBR + H'00000100
- Transition operations:

  The virtual address (32 bits) at which this exception occurred is set in TEA, and the corresponding virtual page number (22 bits) is set in PTEH [31:10]. ASID in PTEH indicates the ASID when this exception occurred.

  The PC and SR contents for the instruction at which this exception occurred are saved in the SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'0E0 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. For details, see section 7, Memory Management Unit (MMU).

```
Instruction_address_error()
{
    TEA = EXCEPTION_ADDRESS;
    PTEH.VPN = PAGE_NUMBER;
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 00E0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(8) Unconditional Trap

- Source: Execution of TRAPA instruction
- Transition address: VBR + H'00000100
- Transition operations:

  As this is a processing-completion-type exception, the PC contents for the instruction following the TRAPA instruction are saved in SPC. The value of SR and R15 when the TRAPA instruction is executed are saved in SSR and SGR. The 8-bit immediate value in the TRAPA instruction is multiplied by 4, and the result is set in TRA [9:0]. Exception code H'160 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
TRAPA_exception()
{
    SPC = PC + 2;
    SSR = SR;
    SGR = R15;
    TRA = imm << 2;
    EXPEVT = H'0000 0160;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(9) General Illegal Instruction Exception

- Sources:
  - — Decoding of an undefined instruction not in a delay slot

    Delayed branch instructions: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT/S, BF/S

    Undefined instruction: H'FFFD
  - — Decoding in user mode of a privileged instruction not in a delay slot

    Privileged instructions: LDC, STC, RTE, LDTLB, SLEEP, but excluding LDC/STC instructions that access GBR
- Transition address: VBR + H'00000100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'180 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. Operation is not guaranteed if an undefined code other than H'FFFD is decoded.

```
General_illegal_instruction_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0180;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(10) Slot Illegal Instruction Exception

- Sources:
  - — Decoding of an undefined instruction in a delay slot
    Delayed branch instructions: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT/S, BF/S
    Undefined instruction: H'FFFD
  - — Decoding of an instruction that modifies PC in a delay slot
    Instructions that modify PC: JMP, JSR, BRA, BRAF, BSR, BSRF, RTS, RTE, BT, BF, BT/S, BF/S, TRAPA, LDC Rm,SR, LDC.L @Rm+,SR, ICBI, PREFI
  - — Decoding in user mode of a privileged instruction in a delay slot
    Privileged instructions: LDC, STC, RTE, LDTLB, SLEEP, but excluding LDC/STC instructions that access GBR
  - — Decoding of a PC-relative MOV instruction or MOVA instruction in a delay slot
  - — The BRDSSLP bit in EXPMASK is 0, and the SLEEP instruction in the delay slot is executed.
  - — The RTEDS bit in EXPMASK is 0, and an instruction other than the NOP instruction in the delay slot is executed.
- Transition address: VBR + H'0000 0100
- Transition operations:
  The PC contents for the preceding delayed branch instruction are saved in SPC. The SR and R15 contents when this exception occurred are saved in SSR and SGR.
  Exception code H'1A0 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. Operation is not guaranteed if an undefined code other than H'FFFD is decoded.

```
Slot_illegal_instruction_exception()
{
    SPC = PC - 2;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 01A0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(11) General FPU Disable Exception

- Source: Decoding of an FPU instruction* not in a delay slot with SR.FD = 1
- Transition address: VBR + H'00000100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'800 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

Note: * FPU instructions are instructions in which the first 4 bits of the instruction code are F (but excluding undefined instruction H'FFFD), and the LDS, STS, LDS.L, and STS.L instructions corresponding to FPUL and FPSCR.

```
General_fpu_disable_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0800;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(12) Slot FPU Disable Exception

- Source: Decoding of an FPU instruction in a delay slot with SR.FD =1
- Transition address: VBR + H'00000100
- Transition operations:

  The PC contents for the preceding delayed branch instruction are saved in SPC. The SR and R15 contents when this exception occurred are saved in SSR and SGR.

  Exception code H'820 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
Slot_fpu_disable_exception()
{
    SPC = PC - 2;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0820;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

(13) Pre-Execution User Break/Post-Execution User Break

- Source: Fulfilling of a break condition set in the user break controller
- Transition address: VBR + H'00000100, or DBR
- Transition operations:

  In the case of a post-execution break, the PC contents for the instruction following the instruction at which the breakpoint is set are set in SPC. In the case of a pre-execution break, the PC contents for the instruction at which the breakpoint is set are set in SPC.

  The SR and R15 contents when the break occurred are saved in SSR and SGR. Exception code H'1E0 is set in EXPEVT.

  The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100. It is also possible to branch to PC = DBR.

  For details of PC, etc., when a data break is set, see section 49, User Break Controller (UBC).

```
User_break_exception()
{
    SPC = (pre_execution break? PC : PC + 2);
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 01E0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = (CBCR.UBDE==1 ? DBR : VBR + H’0000 0100);
}
```

(14) FPU Exception

- Source: Exception due to execution of a floating-point operation
- Transition address: VBR + H'00000100
- Transition operations:

  The PC and SR contents for the instruction at which this exception occurred are saved in SPC and SSR . The R15 contents at this time are saved in SGR. Exception code H'120 is set in EXPEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0100.

```
FPU_exception()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    EXPEVT = H'0000 0120;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    PC = VBR + H'0000 0100;
}
```

5.6.3 Interrupts

(1) NMI (Nonmaskable Interrupt)

- Source: NMI pin edge detection
- Transition address: VBR + H'00000600
- Transition operations:

  The PC and SR contents for the instruction immediately after this exception is accepted are saved in SPC and SSR. The R15 contents at this time are saved in SGR.

  Exception code H'1C0 is set in INTEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to PC = VBR + H'0600. When the BL bit in SR is 0, this interrupt is not masked by the interrupt mask bits in SR, and is accepted at the highest priority level. When the BL bit in SR is 1, a software setting can specify whether this interrupt is to be masked or accepted. When the INTMU bit in CPUOPM is 1 and the NMI interrupt is accessed, B'1111 is set to IMASK bit in SR. For details, see section 13, Interrupt Controller (INTC).

```
NMI()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    INTEVT = H'0000 01C0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    If (cond) SR.IMASK = B'1111;
    PC = VBR + H'0000 0600;
}
```

(2) General Interrupt Request

- Source: The interrupt mask level bits setting in SR is smaller than the interrupt level of interrupt request, and the BL bit in SR is 0 (accepted at instruction boundary).
- Transition address: VBR + H'00000600
- Transition operations:

  The PC contents immediately after the instruction at which the interrupt is accepted are set in SPC. The SR and R15 contents at the time of acceptance are set in SSR and SGR.

The code corresponding to the each interrupt source is set in INTEVT. The BL, MD, and RB bits are set to 1 in SR, and a branch is made to VBR + H'0600. When the INTMU bit in CPUOPM is 1, IMASK bit in SR is changed to accepted interrupt level. For details, see section 13, Interrupt Controller (INTC).

```
Module_interruption()
{
    SPC = PC;
    SSR = SR;
    SGR = R15;
    INTEVT = H'0000 0400 ~ H'0000 3FE0;
    SR.MD = 1;
    SR.RB = 1;
    SR.BL = 1;
    if (cond) SR.IMASK = level_of accepted_interrupt ();
    PC = VBR + H'0000 0600;
}
```

5.6.4 Priority Order with Multiple Exceptions

With some instructions, such as instructions that make two accesses to memory, and the indivisible pair comprising a delayed branch instruction and delay slot instruction, multiple exceptions occur. Care is required in these cases, as the exception priority order differs from the normal order.

(1) Instructions that Make Two Accesses to Memory

With MAC instructions, memory-to-memory arithmetic/logic instructions, TAS instructions, and MOVUA instructions, two data transfers are performed by a single instruction, and an exception will be detected for each of these data transfers. In these cases, therefore, the following order is used to determine priority.

1. Data address error in first data transfer
2. TLB miss in first data transfer
3. TLB protection violation in first data transfer
4. Initial page write exception in first data transfer
5. Data address error in second data transfer
6. TLB miss in second data transfer
7. TLB protection violation in second data transfer

8. Initial page write exception in second data transfer

(2) Indivisible Delayed Branch Instruction and Delay Slot Instruction

As a delayed branch instruction and its associated delay slot instruction are indivisible, they are treated as a single instruction. Consequently, the priority order for exceptions that occur in these instructions differs from the usual priority order. The priority order shown below is for the case where the delay slot instruction has only one data transfer.

1. A check is performed for the interrupt type and re-execution type exceptions of priority levels 1 and 2 in the delayed branch instruction.
2. A check is performed for the interrupt type and re-execution type exceptions of priority levels 1 and 2 in the delay slot instruction.
3. A check is performed for the completion type exception of priority level 2 in the delayed branch instruction.
4. A check is performed for the completion type exception of priority level 2 in the delay slot instruction.
5. A check is performed for priority level 3 in the delayed branch instruction and priority level 3 in the delay slot instruction. (There is no priority ranking between these two.)
6. A check is performed for priority level 4 in the delayed branch instruction and priority level 4 in the delay slot instruction. (There is no priority ranking between these two.)

If the delay slot instruction has a second data transfer, two checks are performed in step 2, as in the above case (Instructions that make two accesses to memory).

If the accepted exception (the highest-priority exception) is a delay slot instruction re-execution type exception, the branch instruction PR register write operation (PC → PR operation performed in a BSR, BSRF, or JSR instruction) is not disabled. Note that in this case, the contents of PR register are not guaranteed.

5.7 Usage Notes

(1) Return from Exception Handling

A. Check the BL bit in SR with software. If SPC and SSR have been saved to memory, set the BL bit in SR to 1 before restoring them.

B. Issue an RTE instruction. When RTE is executed, the SPC contents are saved in PC, the SSR contents are saved in SR, and branch is made to the SPC address to return from the exception handling routine.

(2) If a General Exception or Interrupt Occurs When BL Bit in SR = 1

A. General exception

When a general exception other than a user break occurs, the PC value for the instruction at which the exception occurred in SPC, and a manual reset is executed. The value in EXPEVT at this time is H'00000020; the SSR contents are undefined.

B. Interrupt

If an ordinary interrupt occurs, the interrupt request is held pending and is accepted after the BL bit in SR has been cleared to 0 by software. If a nonmaskable interrupt (NMI) occurs, it can be held pending or accepted according to the setting made by software.

In sleep or standby mode, however, an interrupt is accepted even if the BL bit in SR is set to 1.

(3) SPC when an Exception Occurs

A. Re-execution type general exception

The PC value for the instruction at which the exception occurred is set in SPC, and the instruction is re-executed after returning from the exception handling routine. If an exception occurs in a delay slot instruction, however, the PC value for the delayed branch instruction is saved in SPC regardless of whether or not the preceding delay slot instruction condition is satisfied.

B. Completion type general exception or interrupt

The PC value for the instruction following that at which the exception occurred is set in SPC. If an exception occurs in a branch instruction with delay slot, however, the PC value for the branch destination is saved in SPC.

(4) RTE Instruction Delay Slot

A. The instruction in the delay slot of the RTE instruction is executed only after the value saved in SSR has been restored to SR. The acceptance of the exception related to the instruction access is determined depending on SR before restoring, while the acceptance of other exceptions is determined depending on the processing mode by SR after restoring or the BL bit. The completion type exception is accepted before branching to the destination of RTE instruction. However, if the re-execution type exception is occurred, the operation cannot be guaranteed.

B. The user break is not accepted by the instruction in the delay slot of the RTE instruction.

(5) Changing the SR Register Value and Accepting Exception

A. When the MD or BL bit in the SR register is changed by the LDC instruction, the acceptance of the exception is determined by the changed SR value, starting from the next instruction.* In the completion type exception, an exception is accepted after the next instruction has been executed. However, an interrupt of completion type exception is accepted before the next instruction is executed.

Note: * When the LDC instruction for SR is executed, following instructions are fetched again and the instruction fetch exception is evaluated again by the changed SR.

Section 6 Floating-Point Unit (FPU)

6.1 Features

The FPU has the following features.

- Conforms to IEEE754 standard
- 32 single-precision floating-point registers (can also be referenced as 16 double-precision registers)
- Two rounding modes: Round to Nearest and Round to Zero
- Two denormalization modes: Flush to Zero and Treat Denormalized Number
- Six exception sources: FPU Error, Invalid Operation, Divide By Zero, Overflow, Underflow, and Inexact
- Comprehensive instructions: Single-precision, double-precision, graphics support, and system control

When the FD bit in SR is set to 1, the FPU cannot be used, and an attempt to execute an FPU instruction will cause an FPU disable exception (general FPU disable exception or slot FPU disable exception).

6.2 Data Formats

6.2.1 Floating-Point Format

A floating-point number consists of the following three fields:

- Sign bit (s)
- Exponent field (e)
- Fraction field (f)

This LSI can handle single-precision and double-precision floating-point numbers, using the formats shown in figures 6.1 and 6.2.

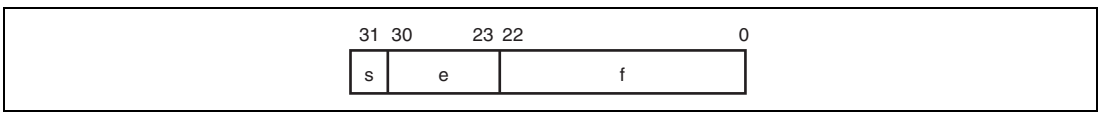


Figure 6.1 Format of Single-Precision Floating-Point Number

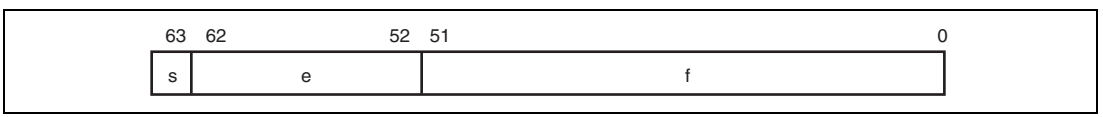


Figure 6.2 Format of Double-Precision Floating-Point Number

The exponent is expressed in biased form, as follows:

$$e = E + bias$$

The range of unbiased exponent E is $E_{min} - 1$ to $E_{max} + 1$. The two values $E_{min} - 1$ and $E_{max} + 1$ are distinguished as follows. $E_{min} - 1$ indicates zero (both positive and negative sign) and a denormalized number, and $E_{max} + 1$ indicates positive or negative infinity or a non-number (NaN). Table 6.1 shows floating-point formats and parameters.

Table 6.1 Floating-Point Number Formats and Parameters

Parameter	Single-Precision	Double-Precision
Total bit width	32 bits	64 bits
Sign bit	1 bit	1 bit
Exponent field	8 bits	11 bits
Fraction field	23 bits	52 bits
Precision	24 bits	53 bits
Bias	+127	+1023
E_{max}	+127	+1023
E_{min}	–126	–1022

Floating-point number value v is determined as follows:

If $E = E_{max} + 1$ and $f \neq 0$, v is a non-number (NaN) irrespective of sign s
If $E = E_{max} + 1$ and $f = 0$, $v = (-1)^s$ (infinity) [positive or negative infinity]
If $E_{min} \leq E \leq E_{max}$, $v = (-1)^s 2^E$ (1.f) [normalized number]
If $E = E_{min} - 1$ and $f \neq 0$, $v = (-1)^s 2^{Emin}$ (0.f) [denormalized number]
If $E = E_{min} - 1$ and $f = 0$, $v = (-1)^s 0$ [positive or negative zero]

Table 6.2 shows the ranges of the various numbers in hexadecimal notation. For the signaling non-number and quiet non-number, see section 6.2.2, Non-Numbers (NaN). For the denormalized number, see section 6.2.3, Denormalized Numbers.

Table 6.2 Floating-Point Ranges

Type	Single-Precision	Double-Precision
Signaling non-number	H'7FFF FFFF to H'7FC0 0000	H'7FFF FFFF FFFF FFFF to H'7FF8 0000 0000 0000
Quiet non-number	H'7FBF FFFF to H'7F80 0001	H'7FF7 FFFF FFFF FFFF to H'7FF0 0000 0000 0001
Positive infinity	H'7F80 0000	H'7FF0 0000 0000 0000
Positive normalized number	H'7F7F FFFF to H'0080 0000	H'7FEF FFFF FFFF FFFF to H'0010 0000 0000 0000
Positive denormalized number	H'007F FFFF to H'0000 0001	H'000F FFFF FFFF FFFF to H'0000 0000 0000 0001
Positive zero	H'0000 0000	H'0000 0000 0000 0000
Negative zero	H'8000 0000	H'8000 0000 0000 0000
Negative denormalized number	H'8000 0001 to H'807F FFFF	H'8000 0000 0000 0001 to H'800F FFFF FFFF FFFF
Negative normalized number	H'8080 0000 to H'FF7F FFFF	H'8010 0000 0000 0000 to H'FFEF FFFF FFFF FFFF
Negative infinity	H'FF80 0000	H'FFF0 0000 0000 0000
Quiet non-number	H'FF80 0001 to H'FFBF FFFF	H'FFF0 0000 0000 0001 to H'FFF7 FFFF FFFF FFFF
Signaling non-number	H'FFC0 0000 to H'FFFF FFFF	H'FFF8 0000 0000 0000 to H'FFFF FFFF FFFF FFFF

6.2.2 Non-Numbers (NaN)

Figure 6.3 shows the bit pattern of a non-number (NaN). A value is NaN in the following case:

- Sign bit: Don't care
- Exponent field: All bits are 1
- Fraction field: At least one bit is 1

The NaN is a signaling NaN (sNaN) if the MSB of the fraction field is 1, and a quiet NaN (qNaN) if the MSB is 0.

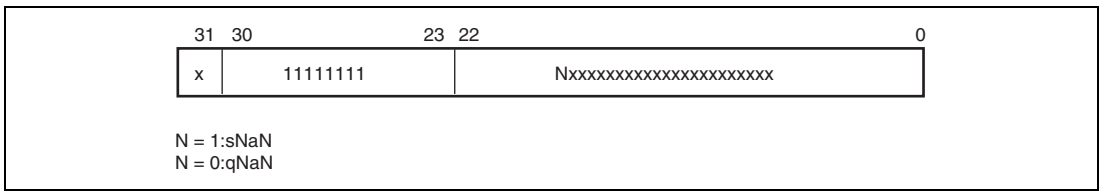


Figure 6.3 Single-Precision NaN Bit Pattern

An sNaN is assumed to be the input data in an operation, except the transfer instructions between registers, FABS, and FNEG, that generates a floating-point value.

- When the EN.V bit in FPSCR is 0, the operation result (output) is a qNaN.
- When the EN.V bit in FPSCR is 1, an invalid operation exception will be generated. In this case, the contents of the operation destination register are unchanged.

Following three instructions are used as transfer instructions between registers.

- FMOV FRm,FRn
- FLDS FRm,FPUL
- FSTS FPUL,FRn

If a qNaN is input in an operation that generates a floating-point value, and an sNaN has not been input in that operation, the output will always be a qNaN irrespective of the setting of the EN.V bit in FPSCR. An exception will not be generated in this case.

The qNAN values as operation results are as follows:

- Single-precision qNaN: H'7FBF FFFF
- Double-precision qNaN: H'7FF7 FFFF FFFF FFFF

See section 10, Instruction Descriptions of the SH-4A Extended Functions Software Manual for details of floating-point operations when a non-number (NaN) is input.

6.2.3 Denormalized Numbers

For a denormalized number floating-point value, the exponent field is expressed as 0, and the fraction field as a non-zero value.

When the DN bit in FPSCR of the FPU is 1, a denormalized number (source operand or operation result) is always positive or negative zero in a floating-point operation that generates a value (an operation other than transfer instructions between registers, FNEG, or FABS).

When the DN bit in FPSCR is 0, a denormalized number (source operand or operation result) is processed as it is. See section 10, Instruction Descriptions of the SH-4A Extended Functions Software Manual for details of floating-point operations when a denormalized number is input.

6.3 Register Descriptions

6.3.1 Floating-Point Registers

Figure 6.4 shows the floating-point register configuration. There are thirty-two 32-bit floating-point registers comprised with two banks: FPR0_BANK0 to FPR15_BANK0, and FPR0_BANK1 to FPR15_BANK1. These thirty-two registers are referenced as FR0 to FR15, DR0/2/4/6/8/10/12/14, FV0/4/8/12, XF0 to XF15, XD0/2/4/6/8/10/12/14, and XMTRX. Corresponding registers to FPR0_BANK0 to FPR15_BANK0, and FPR0_BANK1 to FPR15_BANK1 are determined according to the FR bit of FPSCR.

1. Floating-point registers, FPRi_BANKj (32 registers)
   FPR0_BANK0 to FPR15_BANK0
   FPR0_BANK1 to FPR15_BANK1

2. Single-precision floating-point registers, FRi (16 registers)
   When FPSCR.FR = 0, FR0 to FR15 are allocated to FPR0_BANK0 to FPR15_BANK0;
   when FPSCR.FR = 1, FR0 to FR15 are allocated to FPR0_BANK1 to FPR15_BANK1.

3. Double-precision floating-point registers, DRi (8 registers): A DR register comprises two FR registers.
   DR0 = {FR0, FR1}, DR2 = {FR2, FR3}, DR4 = {FR4, FR5}, DR6 = {FR6, FR7},
   DR8 = {FR8, FR9}, DR10 = {FR10, FR11}, DR12 = {FR12, FR13}, DR14 = {FR14, FR15}

4. Single-precision floating-point vector registers, FVi (4 registers): An FV register comprises four FR registers.
   FV0 = {FR0, FR1, FR2, FR3}, FV4 = {FR4, FR5, FR6, FR7},
   FV8 = {FR8, FR9, FR10, FR11}, FV12 = {FR12, FR13, FR14, FR15}

5. Single-precision floating-point extended registers, XFi (16 registers)
   When FPSCR.FR = 0, XF0 to XF15 are allocated to FPR0_BANK1 to FPR15_BANK1;
   when FPSCR.FR = 1, XF0 to XF15 are allocated to FPR0_BANK0 to FPR15_BANK0.

6. Double-precision floating-point extended registers, XDi (8 registers): An XD register comprises two XF registers.
   XD0 = {XF0, XF1}, XD2 = {XF2, XF3}, XD4 = {XF4, XF5}, XD6 = {XF6, XF7},
   XD8 = {XF8, XF9}, XD10 = {XF10, XF11}, XD12 = {XF12, XF13}, XD14 = {XF14, XF15}

7. Single-precision floating-point extended register matrix, XMTRX: XMTRX comprises all 16 XF registers.

$$XMTRX = \begin{bmatrix} XF0 & XF4 & XF8 & XF12 \\ XF1 & XF5 & XF9 & XF13 \\ XF2 & XF6 & XF10 & XF14 \\ XF3 & XF7 & XF11 & XF15 \end{bmatrix}$$

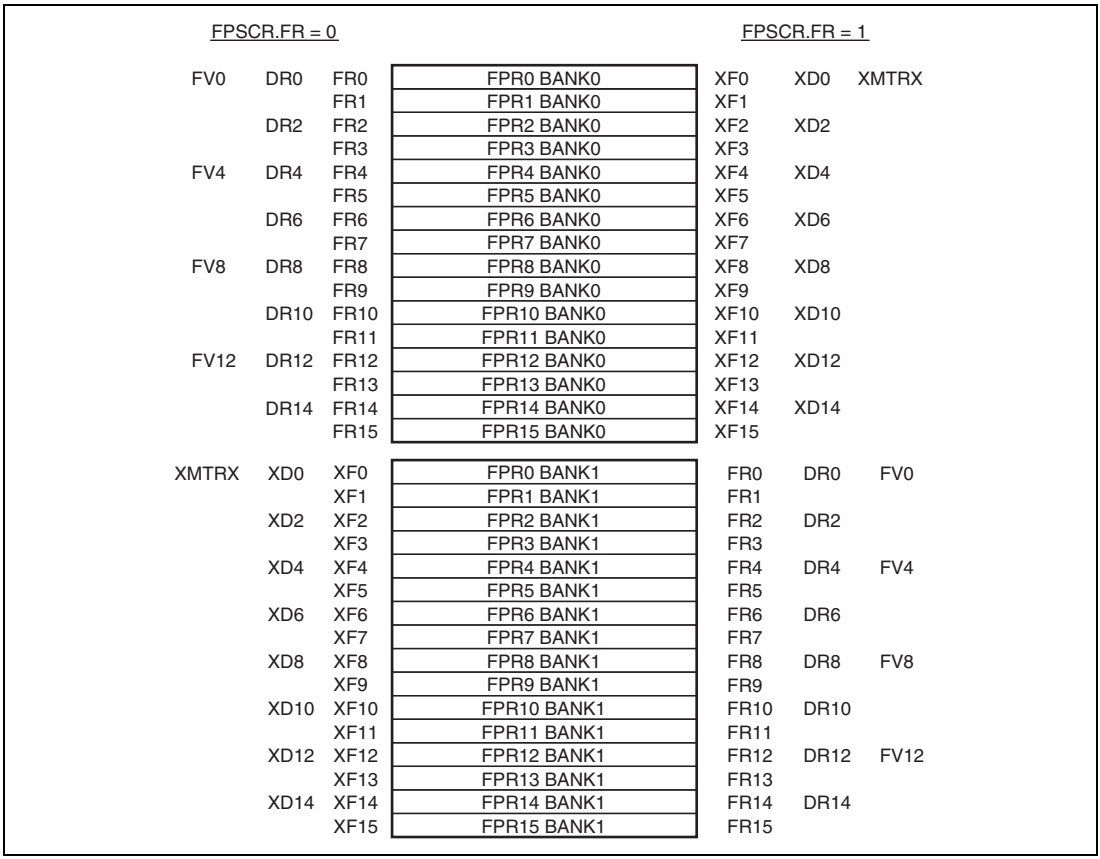


Figure 6.4 Floating-Point Registers

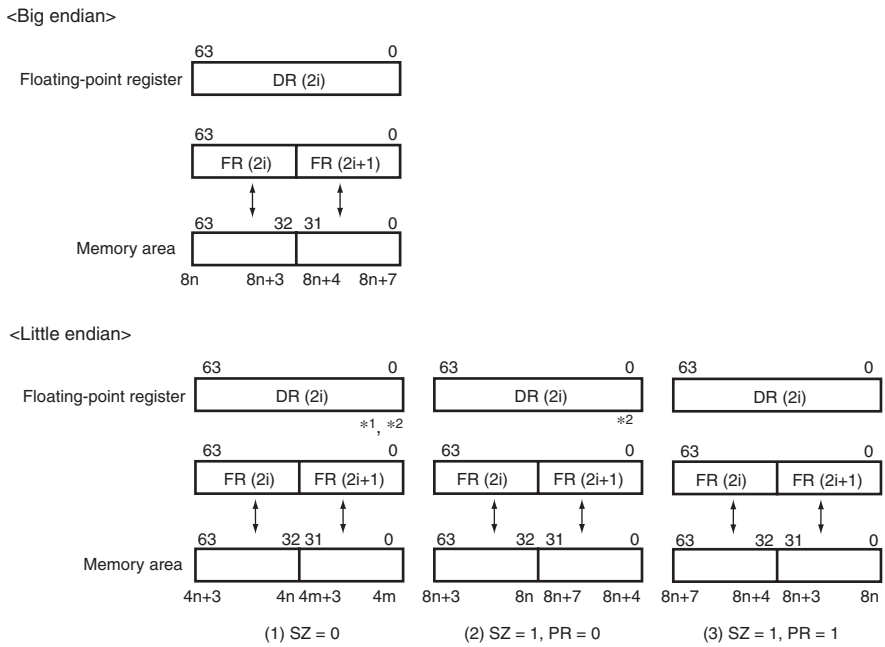
6.3.2 Floating-Point Status/Control Register (FPSCR)

bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	FR	SZ	PR	DN	Cause	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cause				Enable (EN)					Flag					RM	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	FR	0	R/W	Floating-Point Register Bank 0: FPR0_BANK0 to FPR15_BANK0 are assigned to FR0 to FR15 and FPR0_BANK1 to FPR15_BANK1 are assigned to XF0 to XF15 1: FPR0_BANK0 to FPR15_BANK0 are assigned to XF0 to XF15 and FPR0_BANK1 to FPR15_BANK1 are assigned to FR0 to FR15
20	SZ	0	R/W	Transfer Size Mode 0: Data size of FMOV instruction is 32-bits 1: Data size of FMOV instruction is a 32-bit register pair (64 bits) For relations between endian and the SZ and PR bits, see figure 6.5.
19	PR	0	R/W	Precision Mode 0: Floating-point instructions are executed as single-precision operations 1: Floating-point instructions are executed as double-precision operations (graphics support instructions are undefined) For relations between endian and the SZ and PR bits, see figure 6.5.
18	DN	1	R/W	Denormalization Mode 0: Denormalized number is treated as such 1: Denormalized number is treated as zero

Bit	Bit Name	Initial Value	R/W	Description
17 to 12	Cause	All 0	R/W	FPU Exception Cause Field
11 to 7	Enable	All 0	R/W	FPU Exception Enable Field
6 to 2	Flag	All 0	R/W	FPU Exception Flag Field Each time an FPU operation instruction is executed, the FPU exception cause field is cleared to 0. When an FPU exception occurs, the bits corresponding to FPU exception cause field and flag field are set to 1. The FPU exception flag field remains set to 1 until it is cleared to 0 by software. For bit allocations of each field, see table 6.3.
1	RM1	0	R/W	Rounding Mode
0	RM0	1	R/W	These bits select the rounding mode. 00: Round to Nearest 01: Round to Zero 10: Reserved 11: Reserved



Notes: 1. In the case of SZ = 0 and PR = 0, DR register can not be used.

2. The bit-location of DR register is used for double precision format when PR = 1. (In the case of (2), it is used when PR is changed from 0 to 1.)

Figure 6.5 Relation between SZ Bit and Endian

Table 6.3 Bit Allocation for FPU Exception Handling

Field Name		FPU Error (E)	Invalid Operation (V)	Division by Zero (Z)	Overflow (O)	Underflow (U)	Inexact (I)
Cause	FPU exception cause field	Bit 17	Bit 16	Bit 15	Bit 14	Bit 13	Bit 12
Enable	FPU exception enable field	None	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7
Flag	FPU exception flag field	None	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2

6.3.3 Floating-Point Communication Register (FPUL)

Information is transferred between the FPU and CPU via FPUL. FPUL is a 32-bit system register that is accessed from the CPU side by means of LDS and STS instructions. For example, to convert the integer stored in general register R1 to a single-precision floating-point number, the processing flow is as follows:

R1 → (LDS instruction) → FPUL → (single-precision FLOAT instruction) → FR1

6.4 Rounding

In a floating-point instruction, rounding is performed when generating the final operation result from the intermediate result. Therefore, the result of combination instructions such as FMAC, FTRV, and FIPR will differ from the result when using a basic instruction such as FADD, FSUB, or FMUL. Rounding is performed once in FMAC, but twice in FADD, FSUB, and FMUL.

Which of the two rounding methods is to be used is determined by the RM bits in FPSCR.

FPSCR.RM[1:0] = 00: Round to Nearest
FPSCR.RM[1:0] = 01: Round to Zero

(1) Round to Nearest

The operation result is rounded to the nearest expressible value. If there are two nearest expressible values, the one with an LSB of 0 is selected.

If the unrounded value is $2^{Emax} (2 - 2^{-P})$ or more, the result will be infinity with the same sign as the unrounded value. The values of Emax and P, respectively, are 127 and 24 for single-precision, and 1023 and 53 for double-precision.

(2) Round to Zero

The digits below the round bit of the unrounded value are discarded.

If the unrounded value is larger than the maximum expressible absolute value, the value will become the maximum expressible absolute value with the same sign as unrounded value.

6.5 Floating-Point Exceptions

6.5.1 General FPU Disable Exceptions and Slot FPU Disable Exceptions

FPU-related exceptions are occurred when an FPU instruction is executed with SR.FD set to 1. When the FPU instruction is in other than delayed slot, the general FPU disable exception is occurred. When the FPU instruction is in the delay slot, the slot FPU disable exception is occurred.

6.5.2 FPU Exception Sources

The exception sources are as follows:

- FPU error (E): When FPSCR.DN = 0 and a denormalized number is input
- Invalid operation (V): In case of an invalid operation, such as NaN input
- Division by zero (Z): Division with a zero divisor
- Overflow (O): When the operation result overflows
- Underflow (U): When the operation result underflows
- Inexact exception (I): When overflow, underflow, or rounding occurs

The FPU exception cause field in FPSCR contains bits corresponding to all of above sources E, V, Z, O, U, and I, and the FPU exception flag and enable fields in FPSCR contain bits corresponding to sources V, Z, O, U, and I, but not E. Thus, FPU errors cannot be disabled.

When an FPU exception occurs, the corresponding bit in the FPU exception cause field is set to 1, and 1 is added to the corresponding bit in the FPU exception flag field. When an FPU exception does not occur, the corresponding bit in the FPU exception cause field is cleared to 0, but the corresponding bit in the FPU exception flag field remains unchanged.

6.5.3 FPU Exception Handling

FPU exception handling is initiated in the following cases:

- FPU error (E): FPSCR.DN = 0 and a denormalized number is input
- Invalid operation (V): FPSCR.Enable.V = 1 and (instruction = FTRV or invalid operation)
- Division by zero (Z): FPSCR.Enable.Z = 1 and division with a zero divisor or the input of FSRRA is zero
- Overflow (O): FPSCR.Enable.O = 1 and possibility of operation result overflow
- Underflow (U): FPSCR.Enable.U = 1 and possibility of operation result underflow
- Inexact exception (I): FPSCR.Enable.I = 1 and instruction with possibility of inexact operation result

Please refer section 11, Instruction Descriptions of the SH-4A Extended Functions Software Manual about the FPU exception case in detail.

All exception events that originate in the FPU are assigned as the same exception event. The meaning of an exception is determined by software by reading from FPSCR and interpreting the information it contains. Also, the destination register is not changed by any FPU exception handling operation.

If the FPU exception sources except for above are generated, the bit corresponding to source V, Z, O, U, or I is set to 1, and a default value is generated as the operation result.

- Invalid operation (V): qNaN is generated as the result.
- Division by zero (Z): Infinity with the same sign as the unrounded value is generated.
- Overflow (O):
  When rounding mode = RZ, the maximum normalized number, with the same sign as the unrounded value, is generated.
  When rounding mode = RN, infinity with the same sign as the unrounded value is generated.
- Underflow (U):
  When FPSCR.DN = 0, a denormalized number with the same sign as the unrounded value, or zero with the same sign as the unrounded value, is generated.
  When FPSCR.DN = 1, zero with the same sign as the unrounded value, is generated.
- Inexact exception (I): An inexact result is generated.

6.6 Graphics Support Functions

This LSI supports two kinds of graphics functions: new instructions for geometric operations, and pair single-precision transfer instructions that enable high-speed data transfer.

6.6.1 Geometric Operation Instructions

Geometric operation instructions perform approximate-value computations. To enable high-speed computation with a minimum of hardware, this LSI ignores comparatively small values in the partial computation results of four multiplications. Consequently, the error shown below is produced in the result of the computation:

$$\text{Maximum error} = \text{MAX (individual multiplication result} \times 2^{-\text{MIN (number of multiplier significant digits}-1,\ \text{number of multiplicand significant digits}-1)}) + \text{MAX (result value} \times 2^{-23},\ 2^{-149})$$

The number of significant digits is 24 for a normalized number and 23 for a denormalized number (number of leading zeros in the fractional part).

In a future version of the SH Series, the above error is guaranteed, but the same result between different processor cores is not guaranteed.

(1) FIPR FVm, FVn (m, n: 0, 4, 8, 12)

This instruction is basically used for the following purposes:

- Inner product ($m \neq n$):
  This operation is generally used for surface/rear surface determination for polygon surfaces.
- Sum of square of elements ($m = n$):
  This operation is generally used to find the length of a vector.

Since an inexact exception is not detected by an FIPR instruction, the inexact exception (I) bit in both the FPU exception cause field and flag field are always set to 1 when an FIPR instruction is executed. Therefore, if the I bit is set in the FPU exception enable field, FPU exception handling will be executed.

(2) FTRV XMTRX, FVn (n: 0, 4, 8, 12)

This instruction is basically used for the following purposes:

- Matrix (4 × 4) · vector (4):
  This operation is generally used for viewpoint changes, angle changes, or movements called vector transformations (4-dimensional). Since affine transformation processing for angle + parallel movement basically requires a 4 × 4 matrix, this LSI supports 4-dimensional operations.
- Matrix (4 × 4) × matrix (4 × 4):
  This operation requires the execution of four FTRV instructions.

Since an inexact exception is not detected by an FIRV instruction, the inexact exception (I) bit in both the FPU exception cause field and flag field are always set to 1 when an FTRV instruction is executed. Therefore, if the I bit is set in the FPU exception enable field, FPU exception handling will be executed. It is not possible to check all data types in the registers beforehand when executing an FTRV instruction. If the V bit is set in the FPU exception enable field, FPU exception handling will be executed.

(3) FRCHG

This instruction modifies banked registers. For example, when the FTRV instruction is executed, matrix elements must be set in an array in the background bank. However, to create the actual elements of a translation matrix, it is easier to use registers in the foreground bank. When the LDS instruction is used on FPSCR, this instruction takes four to five cycles in order to maintain the FPU state. With the FRCHG instruction, the FR bit in FPSCR can be changed in one cycle.

6.6.2 Pair Single-Precision Data Transfer

In addition to the powerful new geometric operation instructions, this LSI also supports high-speed data transfer instructions.

When the SZ bit is 1, this LSI can perform data transfer by means of pair single-precision data transfer instructions.

- FMOV DRm/XDm, DRn/XDRn (m, n: 0, 2, 4, 6, 8, 10, 12, 14)
- FMOV DRm/XDm, @Rn (m: 0, 2, 4, 6, 8, 10, 12, 14; n: 0 to 15)

These instructions enable two single-precision (2 × 32-bit) data items to be transferred; that is, the transfer performance of these instructions is doubled.

- FSCHG

This instruction changes the value of the SZ bit in FPSCR, enabling fast switching between use and non-use of pair single-precision data transfer.

Section 7 Memory Management Unit (MMU)

This LSI supports an 8-bit address space identifier, a 32-bit virtual address space, and a 29-bit or 32-bit physical address space. Address translation from virtual addresses to physical addresses is enabled by the memory management unit (MMU) in this LSI. The MMU performs high-speed address translation by caching user-created address translation table information in an address translation buffer (translation lookaside buffer: TLB).

This LSI has four instruction TLB (ITLB) entries and 64 unified TLB (UTLB) entries. UTLB copies are stored in the ITLB by hardware. A paging system is used for address translation. It is possible to set the virtual address space access right and implement memory protection independently for privileged mode and user mode.

The MMU of this LSI runs in several operating modes. In view of physical address mapping ranges, 29-bit address mode and 32-bit address extended mode are provided. In view of flag functions of the MMU, TLB compatible mode (four paging sizes with four protection bits) and TLB extended mode (eight paging sizes with six protection bits) are provided.

Selection between 29-bit address mode and 32-bit address extended mode is made by setting the relevant control register (bit SE in the PASCR register) by software.

Selection between TLB compatible mode and TLB extended mode is made by setting the relevant control register (bit ME in the MMUCR register) by software. The range of physical address mapping is explained through sections 7.1, Overview of MMU, to 7.7, Memory-Mapped TLB Configuration, for the case of 29-bit address mode, which is followed by section 7.8, 32-Bit Address Extended Mode, where differences from 29-bit address mode are explained.

The flag functions of the MMU are explained in parallel for both TLB compatible mode and TLB extended mode.

Note: The 32-bit address extended mode is an option.
For support/unsupport of this mode, see the hardware manual of the product.

7.1 Overview of MMU

The MMU was conceived as a means of making efficient use of physical memory. As shown in (0) in figure 7.1, when a process is smaller in size than the physical memory, the entire process can be mapped onto physical memory, but if the process increases in size to the point where it does not fit into physical memory, it becomes necessary to divide the process into smaller parts, and map the parts requiring execution onto physical memory as occasion arises ((1) in figure 7.1). Having this mapping onto physical memory executed consciously by the process itself imposes a heavy burden on the process. The virtual memory system was devised as a means of handling all physical memory mapping to reduce this burden ((2) in figure 7.1). With a virtual memory system, the size of the available virtual memory is much larger than the actual physical memory, and processes are mapped onto this virtual memory. Thus processes only have to consider their operation in virtual memory, and mapping from virtual memory to physical memory is handled by the MMU. The MMU is normally managed by the OS, and physical memory switching is carried out so as to enable the virtual memory required by a process to be mapped smoothly onto physical memory. Physical memory switching is performed via secondary storage, etc.

The virtual memory system that came into being in this way works to best effect in a time sharing system (TSS) that allows a number of processes to run simultaneously ((3) in figure 7.1). Running a number of processes in a TSS did not increase efficiency since each process had to take account of physical memory mapping. Efficiency is improved and the load on each process reduced by the use of a virtual memory system ((4) in figure 7.1). In this virtual memory system, virtual memory is allocated to each process. The task of the MMU is to map a number of virtual memory areas onto physical memory in an efficient manner. It is also provided with memory protection functions to prevent a process from inadvertently accessing another process's physical memory.

When address translation from virtual memory to physical memory is performed using the MMU, it may happen that the translation information has not been recorded in the MMU, or the virtual memory of a different process is accessed by mistake. In such cases, the MMU will generate an exception, change the physical memory mapping, and record the new address translation information.

Although the functions of the MMU could be implemented by software alone, having address translation performed by software each time a process accessed physical memory would be very inefficient. For this reason, a buffer for address translation (the translation lookaside buffer: TLB) is provided by hardware, and frequently used address translation information is placed here. The TLB can be described as a cache for address translation information. However, unlike a cache, if address translation fails—that is, if an exception occurs—switching of the address translation information is normally performed by software. Thus memory management can be performed in a flexible manner by software.

There are two methods by which the MMU can perform mapping from virtual memory to physical memory: the paging method, using fixed-length address translation, and the segment method, using variable-length address translation. With the paging method, the unit of translation is a fixed-size address space called a page.

In the following descriptions, the address space in virtual memory in this LSI is referred to as virtual address space, and the address space in physical memory as physical address space.

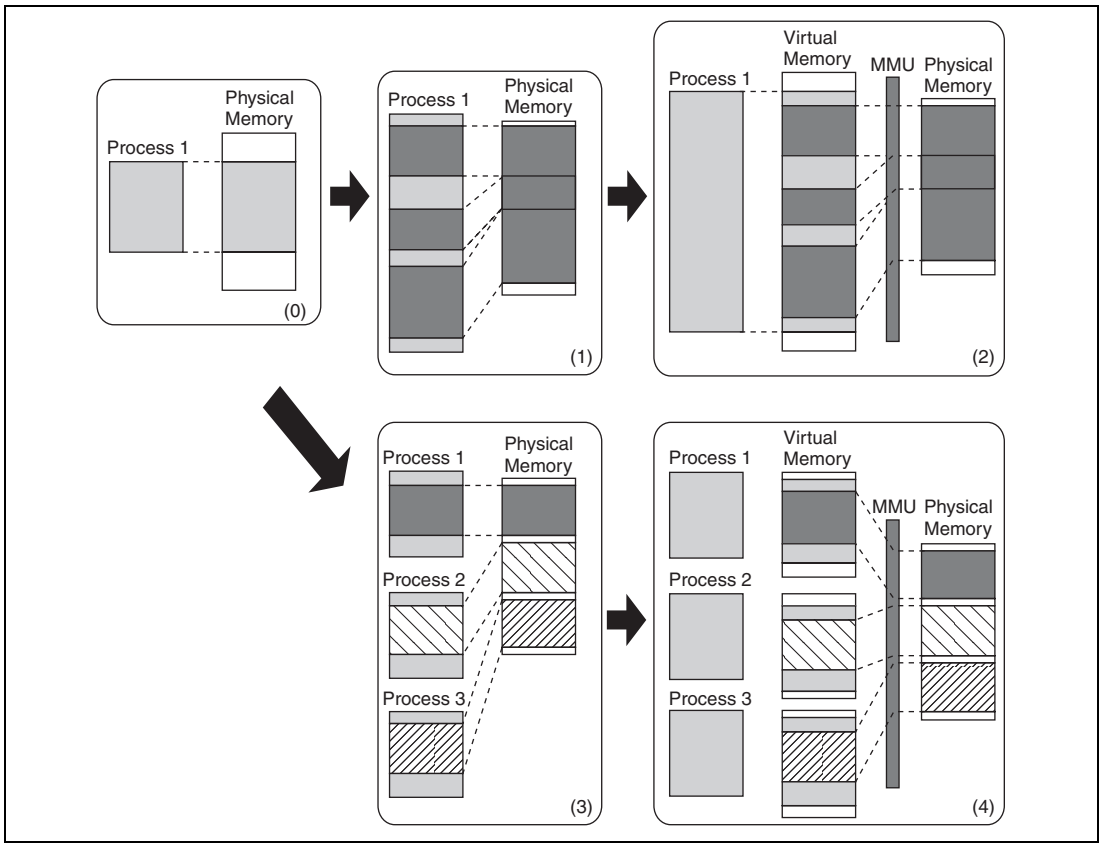


Figure 7.1 Role of MMU

7.1.1 Address Spaces

(1) Virtual Address Space

This LSI supports a 32-bit virtual address space, and can access a 4-Gbyte address space. The virtual address space is divided into a number of areas, as shown in figures 7.2 and 7.3. In privileged mode, the 4-Gbyte space from the P0 area to the P4 area can be accessed. In user mode, a 2-Gbyte space in the U0 area can be accessed. When the SQMD bit in the MMU control register (MMUCR) is 0, a 64-Mbyte space in the store queue area can be accessed. When the RMD bit in the on-chip memory control register (RAMCR) is 1, a 16-Mbyte space in on-chip memory area can be accessed. Accessing areas other than the U0 area, store queue area, and on-chip memory area in user mode will cause an address error.

When the AT bit in MMUCR is set to 1 and the MMU is enabled, the P0, P3, and U0 areas can be mapped onto any physical address space in 1-, 4-, 64-Kbyte, or 1-Mbyte page units in TLB compatible mode and in 1-, 4-, 8-, 64-, 256-Kbyte, 1-, 4-, or 64-Mbyte page units in TLB extended mode. By using an 8-bit address space identifier, the P0, P3, and U0 areas can be increased to a maximum of 256. Mapping from the virtual address space to the 29-bit physical address space is carried out using the TLB.

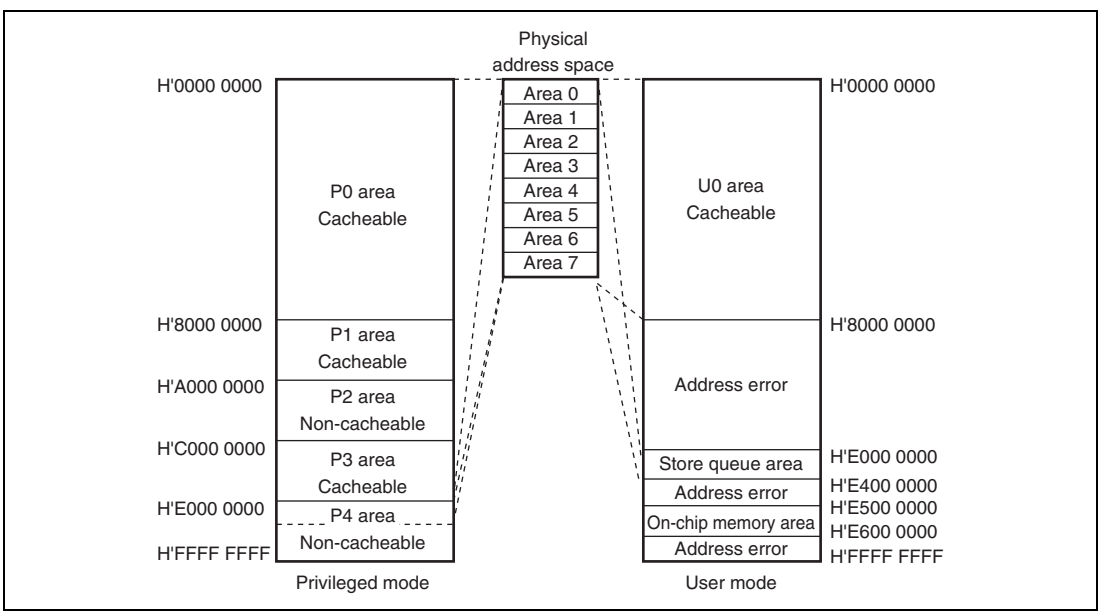


Figure 7.2 Virtual Address Space (AT in MMUCR= 0)

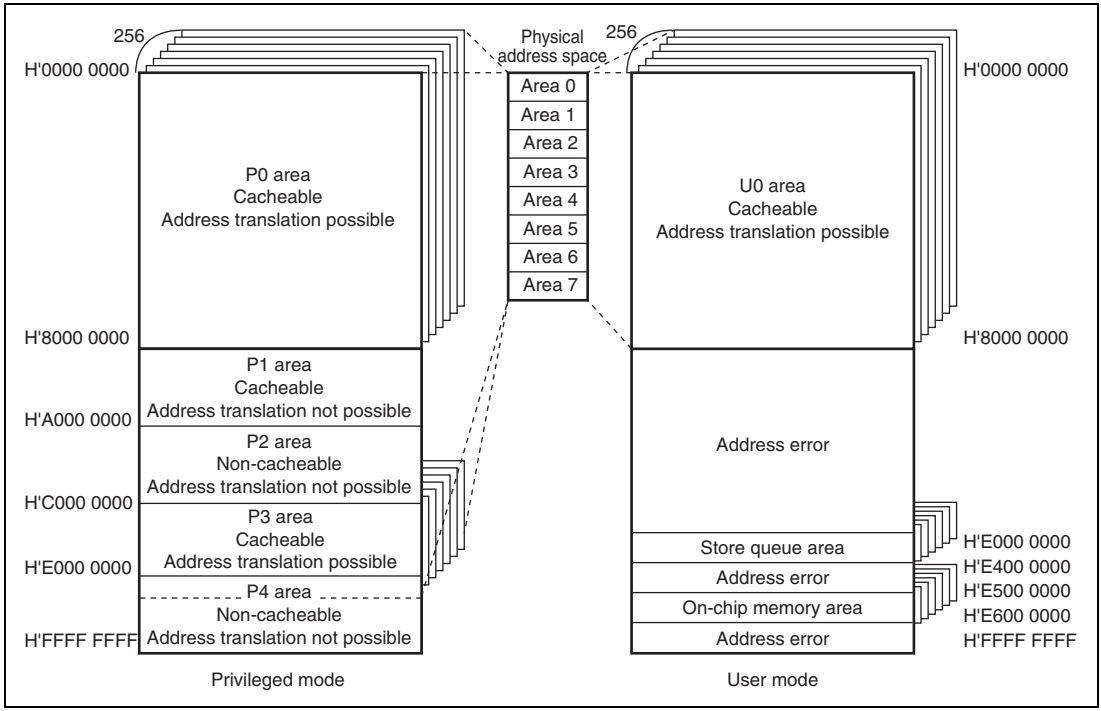


Figure 7.3 Virtual Address Space (AT in MMUCR= 1)

(a) P0, P3, and U0 Areas

The P0, P3, and U0 areas allow address translation using the TLB and access using the cache.

When the MMU is disabled, replacing the upper 3 bits of an address with 0s gives the corresponding physical address. Whether or not the cache is used is determined by the CCR setting. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the WT bit in CCR.

When the MMU is enabled, these areas can be mapped onto any physical address space in 1-, 4-, 64-Kbyte, or 1-Mbyte page units in TLB compatible mode and in 1-, 4-, 8-, 64, 256-Kbyte, 1-, 4-, or 64-Mbyte page units in TLB extended mode using the TLB. When CCR is in the cache enabled state and the C bit for the corresponding page of the TLB entry is 1, accesses can be performed using the cache. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the WT bit of the TLB entry.

When the P0, P3, and U0 areas are mapped onto the control register area which is allocated in the area 7 in physical address space by means of the TLB, the C bit for the corresponding page must be cleared to 0.

(b) P1 Area

The P1 area does not allow address translation using the TLB but can be accessed using the cache.

Regardless of whether the MMU is enabled or disabled, clearing the upper 3 bits of an address to 0 gives the corresponding physical address. Whether or not the cache is used is determined by the CCR setting. When the cache is used, switching between the copy-back method and the write-through method for write accesses is specified by the CB bit in CCR.

(c) P2 Area

The P2 area does not allow address translation using the TLB and access using the cache.

Regardless of whether the MMU is enabled or disabled, clearing the upper 3 bits of an address to 0 gives the corresponding physical address.

(d) P4 Area

The P4 area is mapped onto the internal resource of this LSI. This area except the store queue and on-chip memory areas does not allow address translation using the TLB. This area cannot be accessed using the cache. The P4 area is shown in detail in figure 7.4.

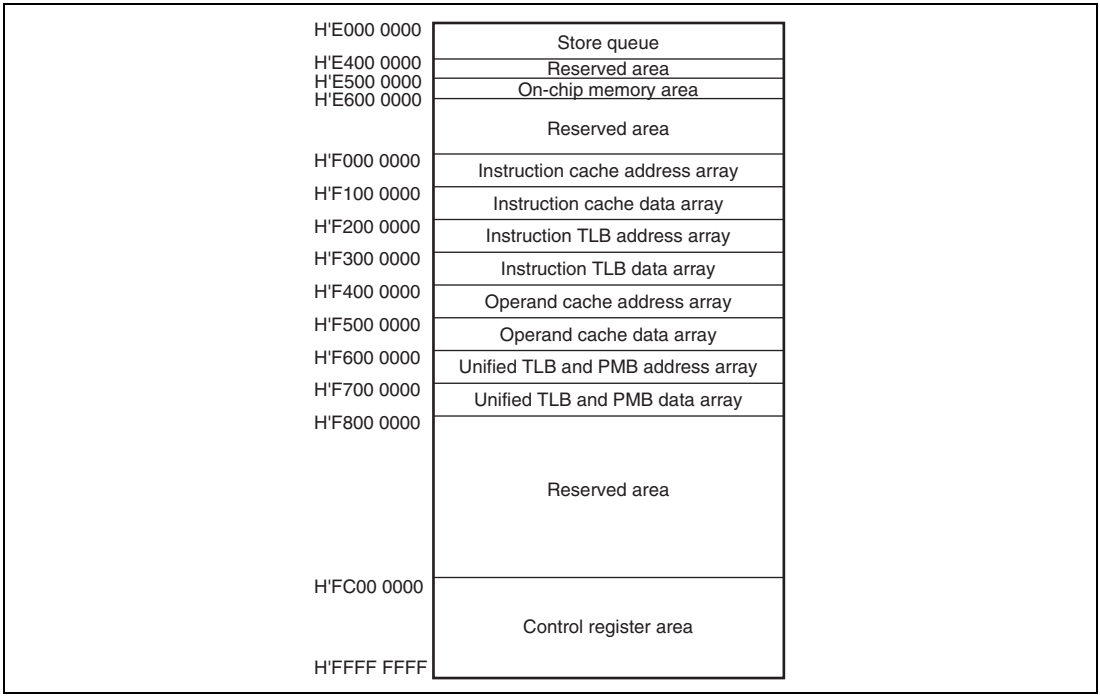


Figure 7.4 P4 Area

The area from H'E000 0000 to H'E3FF FFFF comprises addresses for accessing the store queues (SQs). In user mode, the access right is specified by the SQMD bit in MMUCR. For details, see section 8.7, Store Queues.

The area from H'E500 0000 to H'E5FF FFFF comprises addresses for accessing the on-chip memory. In user mode, the access right is specified by the RMD bit in RAMCR. For details, see section 10, On-Chip Memory.

The area from H'F000 0000 to H'F0FF FFFF is used for direct access to the instruction cache address array. For details, see section 8.6.1, IC Address Array.

The area from H'F100 0000 to H'F1FF FFFF is used for direct access to the instruction cache data array. For details, see section 8.6.2, IC Data Array.

The area from H'F200 0000 to H'F2FF FFFF is used for direct access to the instruction TLB address array. For details, see section 7.7.1, ITLB Address Array.

The area from H'F300 0000 to H'F37F FFFF is used for direct access to instruction TLB data array. For details, see section 7.7.2, ITLB Data Array (TLB Compatible Mode) and section 7.7.3, ITLB Data Array (TLB Extended Mode).

The area from H'F400 0000 to H'F4FF FFFF is used for direct access to the operand cache address array. For details, see section 8.6.3, OC Address Array.

The area from H'F500 0000 to H'F5FF FFFF is used for direct access to the operand cache data array. For details, see section 8.6.4, OC Data Array.

The area from H'F600 0000 to H'F60F FFFF is used for direct access to the unified TLB address array. For details, see section 7.7.4, UTLB Address Array.

The area from H'F610 0000 to H'F61F FFFF is used for direct access to the PMB address array. For details, see section 7.8.5, Memory-Mapped PMB Configuration.

The area from H'F700 0000 to H'F70F FFFF is used for direct access to unified TLB data array. For details, see section 7.7.5, UTLB Data Array (TLB Compatible Mode) and 7.7.6, UTLB Data Array (TLB Extended Mode).

The area from H'F710 0000 to H'F71F FFFF is used for direct access to the PMB data array. For details, see section 7.8.5, Memory-Mapped PMB Configuration.

The area from H'FC00 0000 to H'FFFF FFFF is the on-chip peripheral module control register area. For details, see register descriptions in each section of the hardware manual of the product.

(2) Physical Address Space

This LSI supports a 29-bit physical address space. The physical address space is divided into eight areas as shown in figure 7.5. Area 7 is a reserved area. For details, see the Bus State Controller (BSC) section of the hardware manual of the product.

Only when area 7 in the physical address space is accessed using the TLB, addresses H'1C00 0000 to H'1FFF FFFF of area 7 are not designated as a reserved area, but are equivalent to the control register area in the P4 area, in the virtual address space.

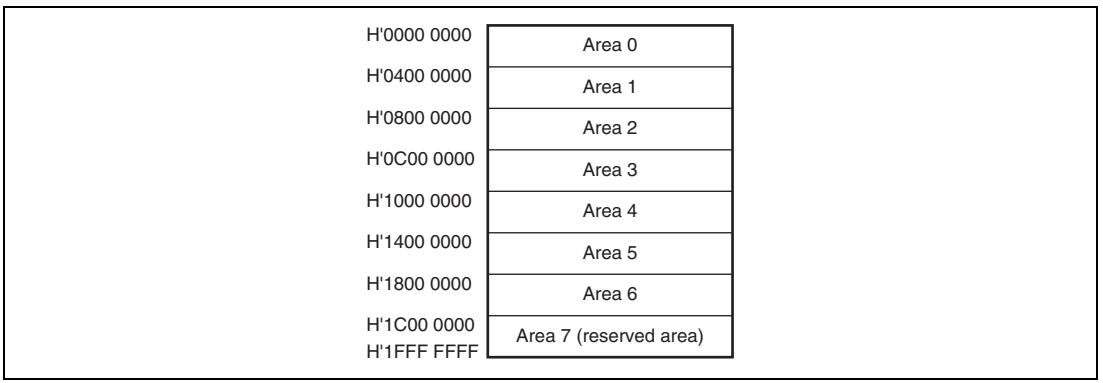


Figure 7.5 Physical Address Space

(3) Address Translation

When the MMU is used, the virtual address space is divided into units called pages, and translation to physical addresses is carried out in these page units. The address translation table in external memory contains the physical addresses corresponding to virtual addresses and additional information such as memory protection codes. Fast address translation is achieved by caching the contents of the address translation table located in external memory into the TLB. In this LSI, basically, the ITLB is used for instruction accesses and the UTLB for data accesses. In the event of an access to an area other than the P4 area, the accessed virtual address is translated to a physical address. If the virtual address belongs to the P1 or P2 area, the physical address is uniquely determined without accessing the TLB. If the virtual address belongs to the P0, U0, or P3 area, the TLB is searched using the virtual address, and if the virtual address is recorded in the TLB, a TLB hit is made and the corresponding physical address is read from the TLB. If the accessed virtual address is not recorded in the TLB, a TLB miss exception is generated and processing switches to the TLB miss exception handling routine. In the TLB miss exception handling routine, the address translation table in external memory is searched, and the corresponding physical address and page management information are recorded in the TLB. After the return from the exception handling routine, the instruction which caused the TLB miss exception is re-executed.

(4) Single Virtual Memory Mode and Multiple Virtual Memory Mode

There are two virtual memory systems, single virtual memory and multiple virtual memory, either of which can be selected with the SV bit in MMUCR. In the single virtual memory system, a number of processes run simultaneously, using virtual address space on an exclusive basis, and the physical address corresponding to a particular virtual address is uniquely determined. In the multiple virtual memory system, a number of processes run while sharing the virtual address space, and particular virtual addresses may be translated into different physical addresses depending on the process. The only difference between the single virtual memory and multiple virtual memory systems in terms of operation is in the TLB address comparison method (see section 7.3.3, Address Translation Method).

(5) Address Space Identifier (ASID)

In multiple virtual memory mode, an 8-bit address space identifier (ASID) is used to distinguish between multiple processes running simultaneously while sharing the virtual address space. Software can set the 8-bit ASID of the currently executing process in PTEH in the MMU. The TLB does not have to be purged when processes are switched by means of ASID.

In single virtual memory mode, ASID is used to provide memory protection for multiple processes running simultaneously while using the virtual address space on an exclusive basis.

Note: Two or more entries with the same virtual page number (VPN) but different ASID must not be set in the TLB simultaneously in single virtual memory mode.

7.2 Register Descriptions

The following registers are related to MMU processing.

Table 7.1 Register Configuration

Register Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Size
Page table entry high register	PTEH	R/W	H'FF00 0000	H'1F00 0000	32
Page table entry low register	PTEL	R/W	H'FF00 0004	H'1F00 0004	32
Translation table base register	TTB	R/W	H'FF00 0008	H'1F00 0008	32
TLB exception address register	TEA	R/W	H'FF00 000C	H'1F00 000C	32
MMU control register	MMUCR	R/W	H'FF00 0010	H'1F00 0010	32
Page table entry assistance register	PTEA	R/W	H'FF00 0034	H'1F00 0034	32
Physical address space control register	PASCR	R/W	H'FF00 0070	H'1F00 0070	32
Instruction re-fetch inhibit control register	IRMCR	R/W	H'FF00 0078	H'1F00 0078	32

Note: * These P4 addresses are for the P4 area in the virtual address space. These area 7 addresses are accessed from area 7 in the physical address space by means of the TLB.

Table 7.2 Register States in Each Processing State

Abbreviation	Power-on Reset	Manual Reset	Software-Standby	Module-Standby	R-Standby	U-Standby	Sleep
PTEH	Undefined	Undefined	Retained	Retained	Retained	Retained	Retained
PTEL	Undefined	Undefined	Retained	Retained	Retained	Retained	Retained
TTB	Undefined	Undefined	Retained	Retained	Retained	Retained	Retained
TEA	Undefined	Retained	Retained	Retained	Retained	Retained	Retained
MMUCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained
PTEA	H'0000 xxx0	H'0000 xxx0	Retained	Retained	H'0000 xxx0	H'0000 xxx0	Retained
PASCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained
IRMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained

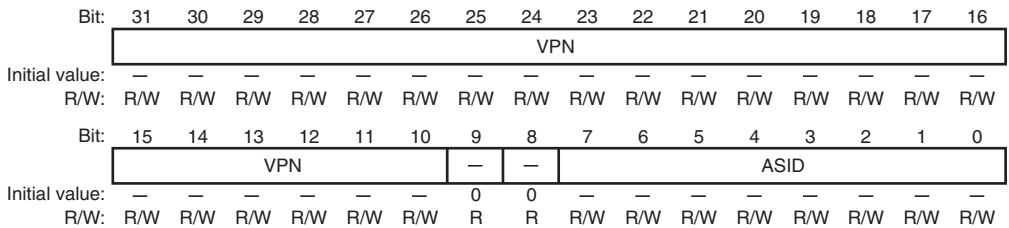
7.2.1 Page Table Entry High Register (PTEH)

PTEH consists of the virtual page number (VPN) and address space identifier (ASID). When an MMU exception or address error exception occurs, the VPN of the virtual address at which the exception occurred is set in the VPN bit by hardware. VPN varies according to the page size, but the VPN set by hardware when an exception occurs consists of the upper 22 bits of the virtual address which caused the exception. VPN setting can also be carried out by software. The number of the currently executing process is set in the ASID bit by software. ASID is not updated by hardware. VPN and ASID are recorded in the UTLB by means of the LDTLB instruction.

After the ASID field in PTEH has been updated, execute one of the following three methods before an access (including an instruction fetch) to the P0, P3, or U0 area that uses the updated ASID value is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the P0, P3, or U0 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating the ASID field, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after the ASID field has been updated.

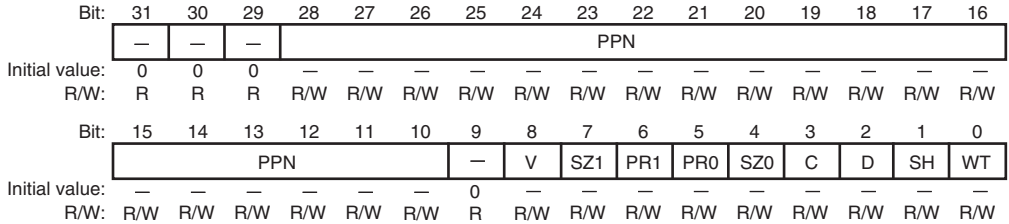
Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.



Bit	Bit Name	Initial Value	R/W	Description
31 to 10	VPN	Undefined	R/W	Virtual Page Number
9, 8	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
7 to 0	ASID	Undefined	R/W	Address Space Identifier

7.2.2 Page Table Entry Low Register (PTEL)

PTEL is used to hold the physical page number and page management information to be recorded in the UTLB by means of the LDTLB instruction. The contents of this register are not changed unless a software directive is issued.



Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
28 to 10	PPN	Undefined	R/W	Physical Page Number
9	—	0	R	Reserved For details on reading from or writing to this bit, see description in General Precautions on Handling of Product.
8	V	Undefined	R/W	Page Management Information The meaning of each bit is same as that of corresponding bit in Common TLB (UTLB). For details, see section 7.3, TLB Functions (TLB Compatible Mode; MMUCR.ME = 0) and section 7.4, TLB Functions (TLB Extended Mode; MMUCR.ME = 1). Note: SZ1, PR1, SZ0, and PR0 bits are valid only in TLB compatible mode.
7	SZ1	Undefined	R/W	
6	PR1	Undefined	R/W	
5	PR0	Undefined	R/W	
4	SZ0	Undefined	R/W	
3	C	Undefined	R/W	
2	D	Undefined	R/W	
1	SH	Undefined	R/W	
0	WT	Undefined	R/W	

7.2.3 Translation Table Base Register (TTB)

TTB is used to store the base address of the currently used page table, and so on. The contents of TTB are not changed unless a software directive is issued. This register can be used freely by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TTB															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TTB															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

7.2.4 TLB Exception Address Register (TEA)

After an MMU exception or address error exception occurs, the virtual address at which the exception occurred is stored. The contents of this register can be changed by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEA								Virtual address at which MMU exception or address error occurred							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEA								Virtual address at which MMU exception or address error occurred							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

7.2.5 MMU Control Register (MMUCR)

The individual bits perform MMU settings as shown below. Therefore, MMUCR rewriting should be performed by a program in the P1 or P2 area.

After MMUCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the P0, P3, U0, or store queue area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the P0, P3, or U0 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating MMUCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

MMUCR contents can be changed by software. However, the LRUI and URC bits may also be updated by hardware.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LRUI						—	—	URB						—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	URC						SQMD	SV	ME	—	—	—	—	TI	—	AT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	LRUI	000000	R/W	Least Recently Used ITLB These bits indicate the ITLB entry to be replaced. The LRU (least recently used) method is used to decide the ITLB entry to be replaced in the event of an ITLB miss. The entry to be purged from the ITLB can be confirmed using the LRUI bits. LRUI is updated by means of the algorithm shown below. x means that updating is not performed. 000xxx: ITLB entry 0 is used 1xx00x: ITLB entry 1 is used x1x1x0: ITLB entry 2 is used xx1x11: ITLB entry 3 is used xxxxxx: Other than above When the LRUI bit settings are as shown below, the corresponding ITLB entry is updated by an ITLB miss. Ensure that values for which "Setting prohibited" is indicated below are not set at the discretion of software. After a power-on or manual reset, the LRUI bits are initialized to 0, and therefore a prohibited setting is never made by a hardware update. x means "don't care". 111xxx: ITLB entry 0 is updated 0xx11x: ITLB entry 1 is updated x0x0x1: ITLB entry 2 is updated xx0x00: ITLB entry 3 is updated Other than above: Setting prohibited
25, 24	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
23 to 18	URB	000000	R/W	UTLB Replace Boundary These bits indicate the UTLB entry boundary at which replacement is to be performed. Valid only when URB ≠ 0.
17, 16	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	URC	000000	R/W	UTLB Replace Counter These bits serve as a random counter for indicating the UTLB entry for which replacement is to be performed with an LDTLB instruction. This bit is incremented each time the UTLB is accessed. If URB > 0, URC is cleared to 0 when the condition URC = URB is satisfied. Also note that if a value is written to URC by software which results in the condition of URC > URB, incrementing is first performed in excess of URB until URC = H'3F. URC is not incremented by an LDTLB instruction.
9	SQMD	0	R/W	Store Queue Mode Specifies the right of access to the store queues. 0: User/privileged access possible 1: Privileged access possible (address error exception in case of user access)
8	SV	0	R/W	Single Virtual Memory Mode/Multiple Virtual Memory Mode Switching When this bit is changed, ensure that 1 is also written to the TI bit. 0: Multiple virtual memory mode 1: Single virtual memory mode
7	ME	0	R/W	TLB Extended Mode Switching 0: TLB compatible mode 1: TLB extended mode For modifying the ME bit value, always set the TI bit to 1 to invalidate the contents of ITLB and UTLB. The selection of TLB operating mode made by the ME bit does not affect the functionality or operation of the PMB.
6 to 3	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
2	TI	0	R/W	TLB Invalidate Bit Writing 1 to this bit invalidates (clears to 0) all valid UTLB/ITLB bits. This bit is always read as 0.
1	—	0	R	Reserved For details on reading from or writing to this bit, see description in General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
0	AT	0	R/W	Address Translation Enable Bit These bits enable or disable the MMU. 0: MMU disabled 1: MMU enabled MMU exceptions are not generated when the AT bit is 0. In the case of software that does not use the MMU, the AT bit should be cleared to 0.

7.2.6 Page Table Entry Assistance Register (PTEA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	EPR						ESZ				–	–	–	–
Initial value:	0	0	–	–	–	–	–	–	–	–	–	–	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.
13 to 8	EPR	Undefined	R/W	Page Control Information
7 to 4	ESZ	Undefined	R/W	Each bit has the same function as the corresponding bit of the unified TLB (UTLB). For details, see section 7.4, TLB Functions (TLB Extended Mode; MMUCR.ME = 1)
3 to 0	—	All 0	R	Reserved For details on reading/writing these bits, see General Precautions on Handling of Product.

7.2.7 Physical Address Space Control Register (PASCR)

PASCR controls the operation in the physical address space.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UB							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
7 to 0	UB	H'00	R/W	Buffered Write Control for Each Area (64 Mbytes) When writing is performed without using the cache or in the cache write-through mode, these bits specify whether the next bus access from the CPU waits for the end of writing for each area. 0 : Buffered write (The CPU does not wait for the end of writing bus access and starts the next bus access) 1 : Unbuffered write (The CPU waits for the end of writing bus access and starts the next bus access) UB[7]: Corresponding to the control register area UB[6]: Corresponding to area 6 UB[5]: Corresponding to area 5 UB[4]: Corresponding to area 4 UB[3]: Corresponding to area 3 UB[2]: Corresponding to area 2 UB[1]: Corresponding to area 1 UB[0]: Corresponding to area 0

7.2.8 Instruction Re-Fetch Inhibit Control Register (IRMCR)

When the specific resource is changed, IRMCR controls whether the instruction fetch is performed again for the next instruction. The specific resource means the part of control registers, TLB, and cache.

In the initial state, the instruction fetch is performed again for the next instruction after changing the resource. However, the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction every time the resource is changed. Therefore, it is recommended that each bit in IRMCR is set to 1 and the specific instruction should be executed after all necessary resources have been changed prior to execution of the program which uses changed resources.

For details on the specific sequence, see descriptions in each resource.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	R2	R1	LT	MT	MC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
4	R2	0	R/W	Re-Fetch Inhibit 2 after Register Change When MMUCR, PASCR, CCR, PTEH, or RAMCR is changed, this bit controls whether re-fetch is performed for the next instruction. 0: Re-fetch is performed 1: Re-fetch is not performed

Bit	Bit Name	Initial Value	R/W	Description
3	R1	0	R/W	Re-Fetch Inhibit 1 after Register Change When a register allocated in addresses H'FF200000 to H'FF2FFFFF is changed, this bit controls whether re-fetch is performed for the next instruction. 0: Re-fetch is performed 1: Re-fetch is not performed
2	LT	0	R/W	Re-Fetch Inhibit after LDTLB Execution This bit controls whether re-fetch is performed for the next instruction after the LDTLB instruction has been executed. 0: Re-fetch is performed 1: Re-fetch is not performed
1	MT	0	R/W	Re-Fetch Inhibit after Writing Memory-Mapped TLB This bit controls whether re-fetch is performed for the next instruction after writing memory-mapped ITLB/UTLB while the AT bit in MMUCR is set to 1. 0: Re-fetch is performed 1: Re-fetch is not performed
0	MC	0	R/W	Re-Fetch Inhibit after Writing Memory-Mapped IC This bit controls whether re-fetch is performed for the next instruction after writing memory-mapped IC while the ICE bit in CCR is set to 1. 0: Re-fetch is performed 1: Re-fetch is not performed

7.3 TLB Functions (TLB Compatible Mode; MMUCR.ME = 0)

7.3.1 Unified TLB (UTLB) Configuration

The UTLB is used for the following two purposes:

1. To translate a virtual address to a physical address in a data access
2. As a table of address translation information to be recorded in the ITLB in the event of an ITLB miss

The UTLB is so called because of its use for the above two purposes. Information in the address translation table located in external memory is cached into the UTLB. The address translation table contains virtual page numbers and address space identifiers, and corresponding physical page numbers and page management information. Figure 7.6 shows the UTLB configuration. The UTLB consists of 64 fully-associative type entries. Figure 7.7 shows the relationship between the page size and address format.

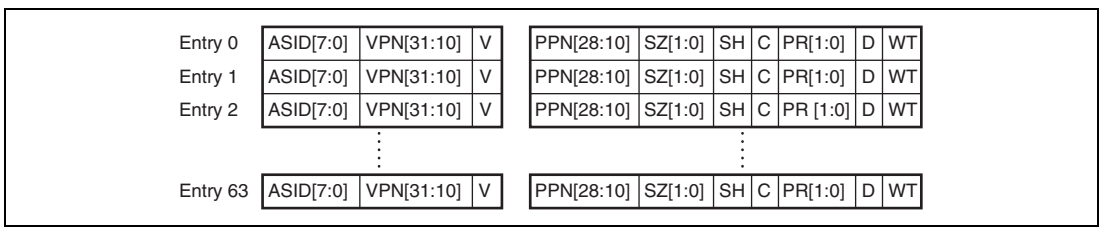


Figure 7.6 UTLB Configuration (TLB Compatible Mode)

[Legend]

- VPN: Virtual page number
  For 1-Kbyte page: Upper 22 bits of virtual address
  For 4-Kbyte page: Upper 20 bits of virtual address
  For 64-Kbyte page: Upper 16 bits of virtual address
  For 1-Mbyte page: Upper 12 bits of virtual address

- ASID: Address space identifier
  Indicates the process that can access a virtual page.
  In single virtual memory mode and user mode, or in multiple virtual memory mode, if the SH bit is 0, this identifier is compared with the ASID in PTEH when address comparison is performed.

- SH: Share status bit
  When 0, pages are not shared by processes.
  When 1, pages are shared by processes.

- SZ[1:0]: Page size bits
  Specify the page size.
  00: 1-Kbyte page
  01: 4-Kbyte page
  10: 64-Kbyte page
  11: 1-Mbyte page

- V: Validity bit
  Indicates whether the entry is valid.
  0: Invalid
  1: Valid
  Cleared to 0 by a power-on reset.
  Not affected by a manual reset.

- PPN: Physical page number
  Upper 22 bits of the physical address of the physical page number.
  With a 1-Kbyte page, PPN[28:10] are valid.
  With a 4-Kbyte page, PPN[28:12] are valid.
  With a 64-Kbyte page, PPN[28:16] are valid.
  With a 1-Mbyte page, PPN[28:20] are valid.
  The synonym problem must be taken into account when setting the PPN (see section 7.5.5, Avoiding Synonym Problems).

- PR[1:0]: Protection key data
  2-bit data expressing the page access right as a code.
  00: Can be read from only in privileged mode
  01: Can be read from and written to in privileged mode
  10: Can be read from only in privileged or user mode
  11: Can be read from and written to in privileged mode or user mode

- C: Cacheability bit
  Indicates whether a page is cacheable.
  0: Not cacheable
  1: Cacheable
  When the control register area is mapped, this bit must be cleared to 0.

- D: Dirty bit
  Indicates whether a write has been performed to a page.
  0: Write has not been performed
  1: Write has been performed

- WT: Write-through bit
  Specifies the cache write mode.
  0: Copy-back mode
  1: Write-through mode

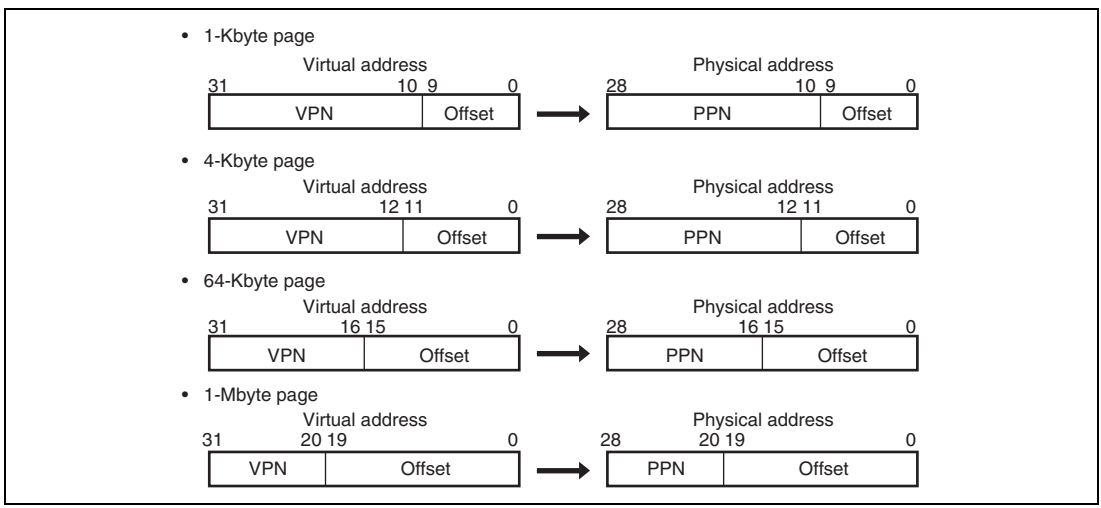


Figure 7.7 Relationship between Page Size and Address Format (TLB Compatible Mode)

7.3.2 Instruction TLB (ITLB) Configuration

The ITLB is used to translate a virtual address to a physical address in an instruction access. Information in the address translation table located in the UTLB is cached into the ITLB. Figure 7.8 shows the ITLB configuration. The ITLB consists of four fully-associative type entries.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR
Entry 3	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	SZ[1:0]	SH	C	PR

Notes: 1. The D and WT bits are not supported.
2. There is only one PR bit, corresponding to the upper bit of the PR bits in the UTLB.

Figure 7.8 ITLB Configuration (TLB Compatible Mode)

7.3.3 Address Translation Method

Figure 7.9 shows a flowchart of a memory access using the UTLB.

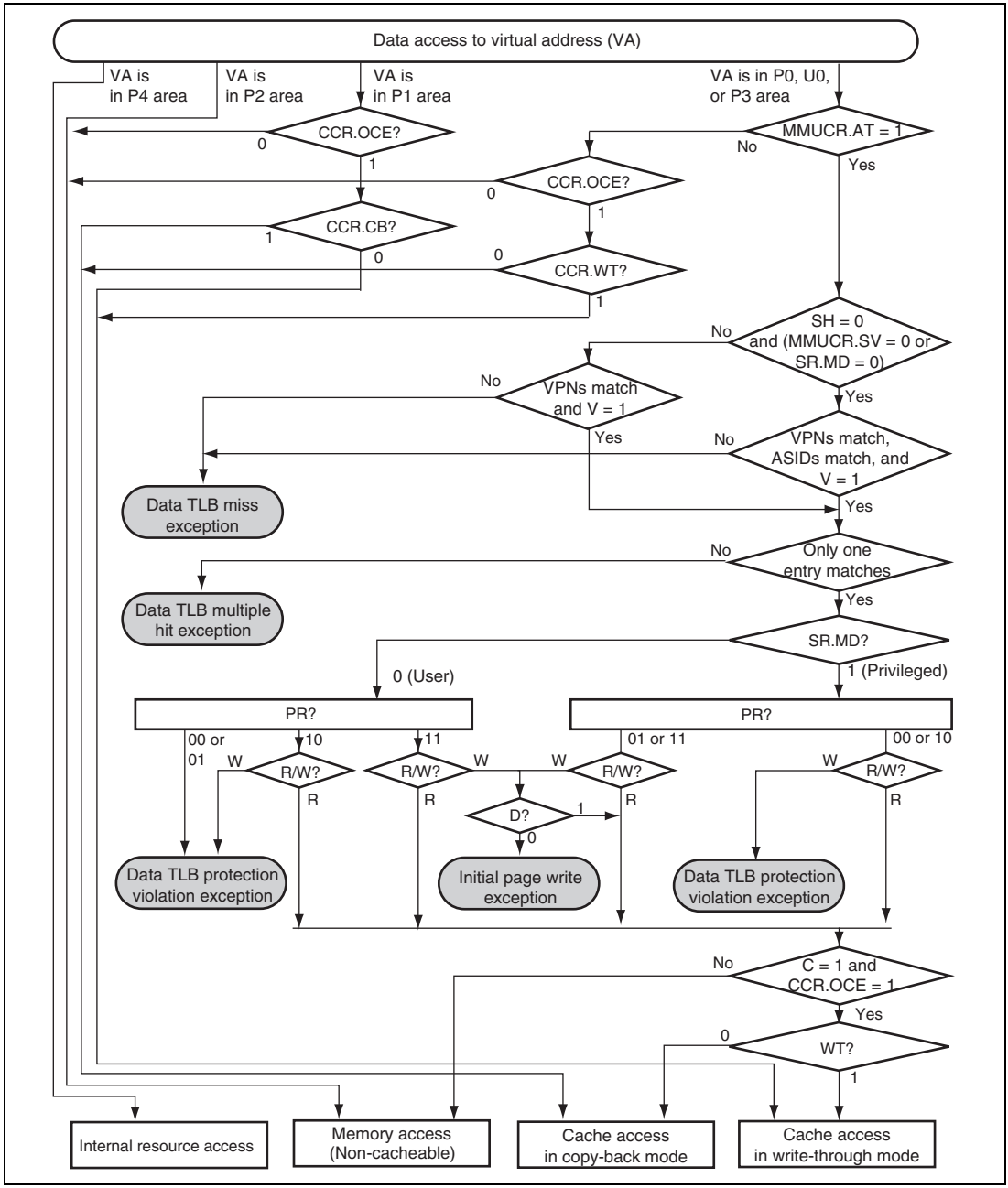


Figure 7.9 Flowchart of Memory Access Using UTLB (TLB Compatible Mode)

Figure 7.10 shows a flowchart of a memory access using the ITLB.

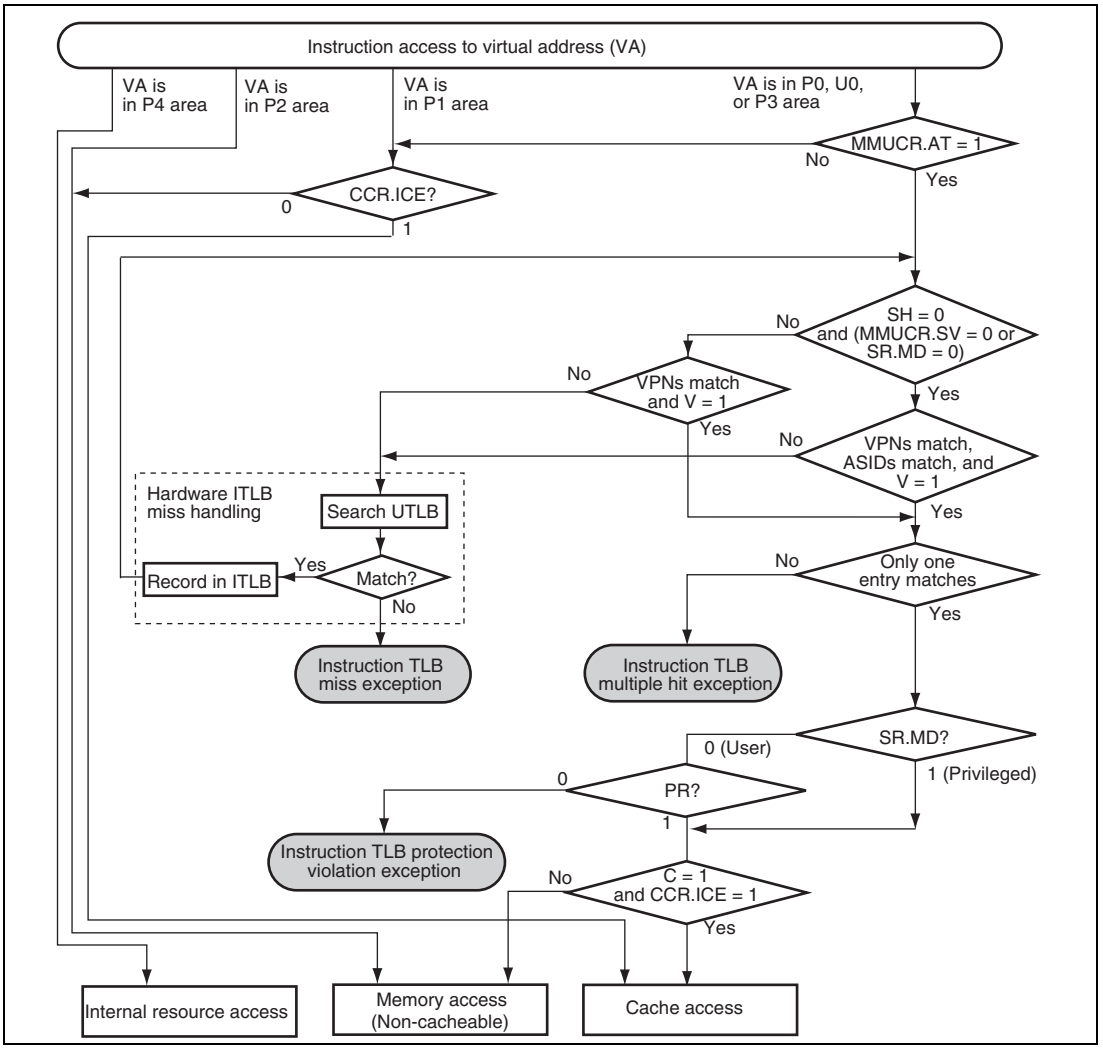


Figure 7.10 Flowchart of Memory Access Using ITLB (TLB Compatible Mode)

7.4 TLB Functions (TLB Extended Mode; MMUCR.ME = 1)

7.4.1 Unified TLB (UTLB) Configuration

Figure 7.11 shows the configuration of the UTLB in TLB extended mode. Figure 7.12 shows the relationship between the page size and address format.

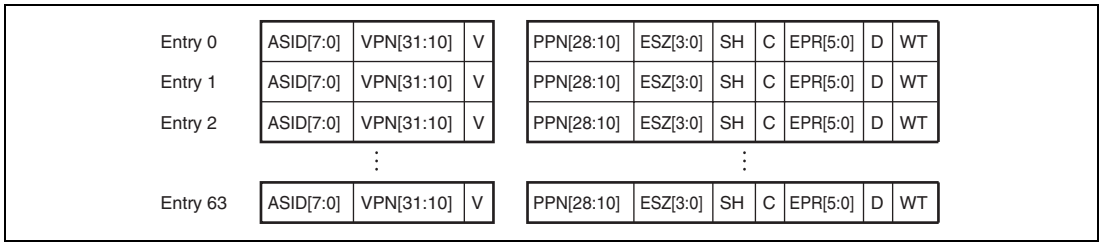


Figure 7.11 UTLB Configuration (TLB Extended Mode)

[Legend]

- VPN: Virtual page number
  For 1-Kbyte page: Upper 22 bits of virtual address
  For 4-Kbyte page: Upper 20 bits of virtual address
  For 8-Kbyte page: Upper 19 bits of virtual address
  For 64-Kbyte page: Upper 16 bits of virtual address
  For 256-Kbyte page: Upper 14 bits of virtual address
  For 1-Mbyte page: Upper 12 bits of virtual address
  For 4-Mbyte page: Upper 10 bits of virtual address
  For 64-Mbyte page: Upper 6 bits of virtual address
- ASID: Address space identifier
  Indicates the process that can access a virtual page.
  In single virtual memory mode and user mode, or in multiple virtual memory mode, if the SH bit is 0, this identifier is compared with the ASID in PTEH when address comparison is performed.
- SH: Share status bit
  When 0, pages are not shared by processes.
  When 1, pages are shared by processes.

- ESZ: Page size bits
  Specify the page size.
  0000: 1-Kbyte page
  0001: 4-Kbyte page
  0010: 8-Kbyte page
  0100: 64-Kbyte page
  0101: 256-Kbyte page
  0111: 1-Mbyte page
  1000: 4-Mbyte page
  1100: 64-Mbyte page

Note: When a value other than those listed above is recorded, operation is not guaranteed.

- V: Validity bit
  Indicates whether the entry is valid.
  0: Invalid
  1: Valid
  Cleared to 0 by a power-on reset.
  Not affected by a manual reset.
- PPN: Physical page number
  Upper 19 bits of the physical address.
  With a 1-Kbyte page, PPN[28:10] are valid.
  With a 4-Kbyte page, PPN[28:12] are valid.
  With a 8-Kbyte page, PPN[28:13] are valid.
  With a 64-Kbyte page, PPN[28:16] are valid.
  With a 256-Kbyte page, PPN[28:18] are valid.
  With a 1-Mbyte page, PPN[28:20] are valid.
  With a 4-Mbyte page, PPN[28:22] are valid.
  With a 64-Mbyte page, PPN[28:26] are valid.
  The synonym problem must be taken into account when setting the PPN (see section 7.5.5, Avoiding Synonym Problems).

- EPR: Protection key data

  6-bit data expressing the page access right as a code.

  Reading, writing, and execution (instruction fetch) in privileged mode and reading, writing, and execution (instruction fetch) in user mode can be set independently. Each bit is disabled by 0 and enabled by 1.

  EPR[5]: Reading in privileged mode

  EPR[4]: Writing in privileged mode

  EPR[3]: Execution in privileged mode (instruction fetch)

  EPR[2]: Reading in user mode

  EPR[1]: Writing in user mode

  EPR[0]: Execution in user mode (instruction fetch)

- C: Cacheability bit

  Indicates whether a page is cacheable.

  0: Not cacheable

  1: Cacheable

  When the control register area is mapped, this bit must be cleared to 0.

- D: Dirty bit

  Indicates whether a write has been performed to a page.

  0: Write has not been performed.

  1: Write has been performed.

- WT: Write-through bit

  Specifies the cache write mode.

  0: Copy-back mode

  1: Write-through mode

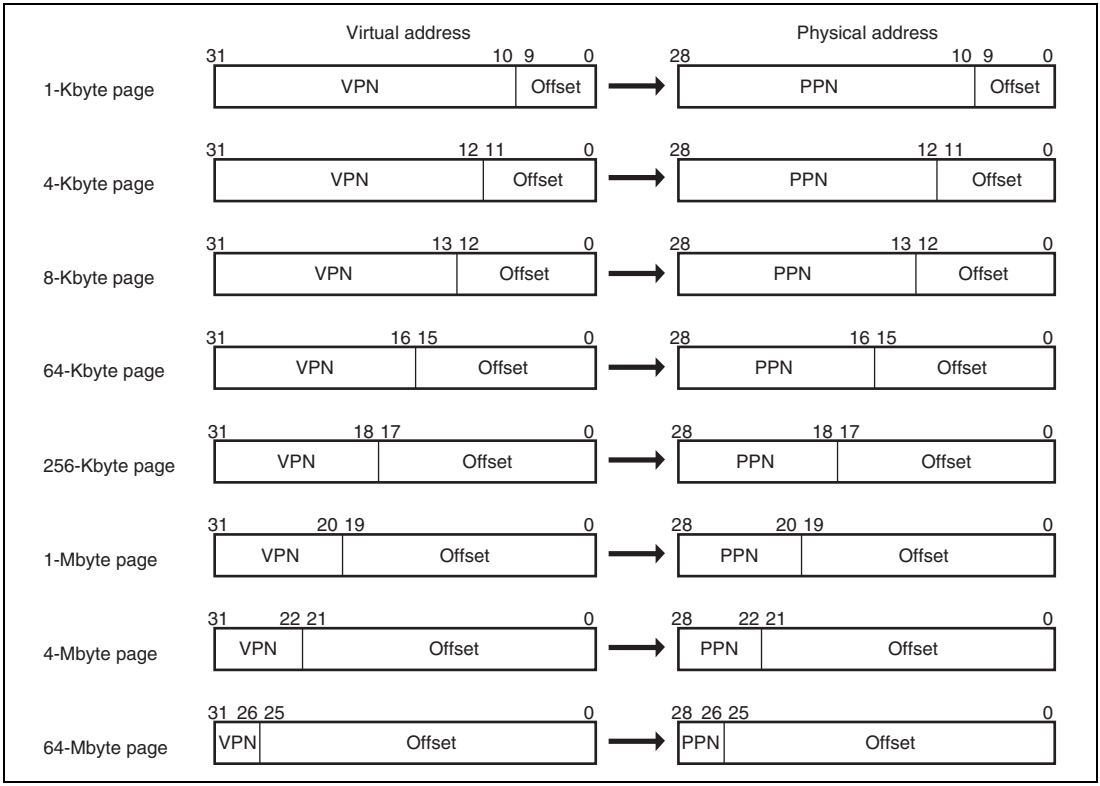


Figure 7.12 Relationship between Page Size and Address Format (TLB Extended Mode)

7.4.2 Instruction TLB (ITLB) Configuration

Figure 7.13 shows the configuration of the ITLB in TLB extended mode.

Entry 0	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 1	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 2	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]
Entry 3	ASID[7:0]	VPN[31:10]	V	PPN[28:10]	ESZ[3:0]	SH	C	EPR[5]	EPR[3]	EPR[2]	EPR[0]

Note: Bits EPR[4], EPR[1], D, and WT are not supported.

Figure 7.13 ITLB Configuration (TLB Extended Mode)

7.4.3 Address Translation Method

Figure 7.14 is a flowchart of memory access using the UTLB in TLB extended mode.

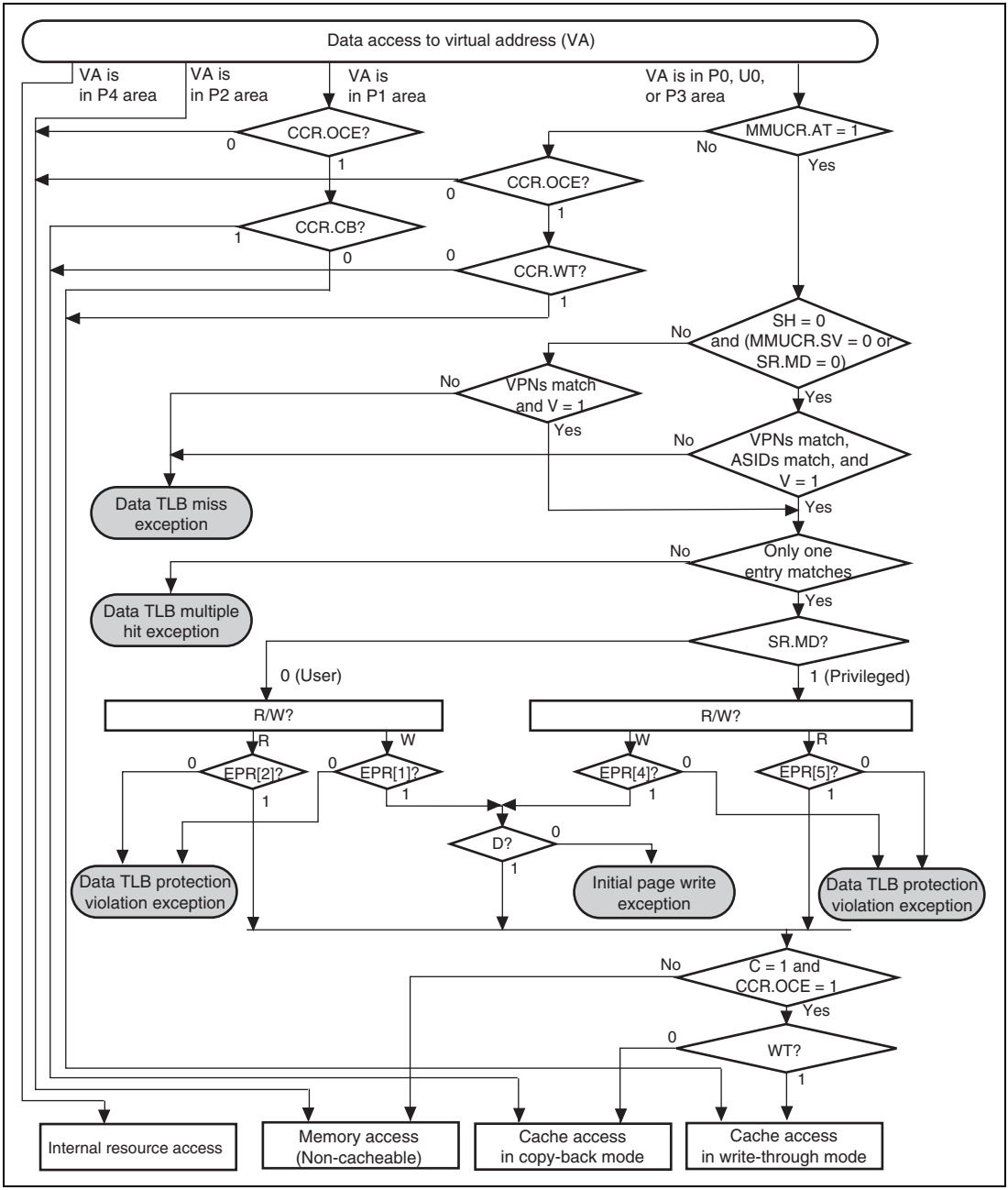


Figure 7.14 Flowchart of Memory Access Using UTLB (TLB Extended Mode)

Figure 7.15 is a flowchart of memory access using the ITLB in TLB extended mode.

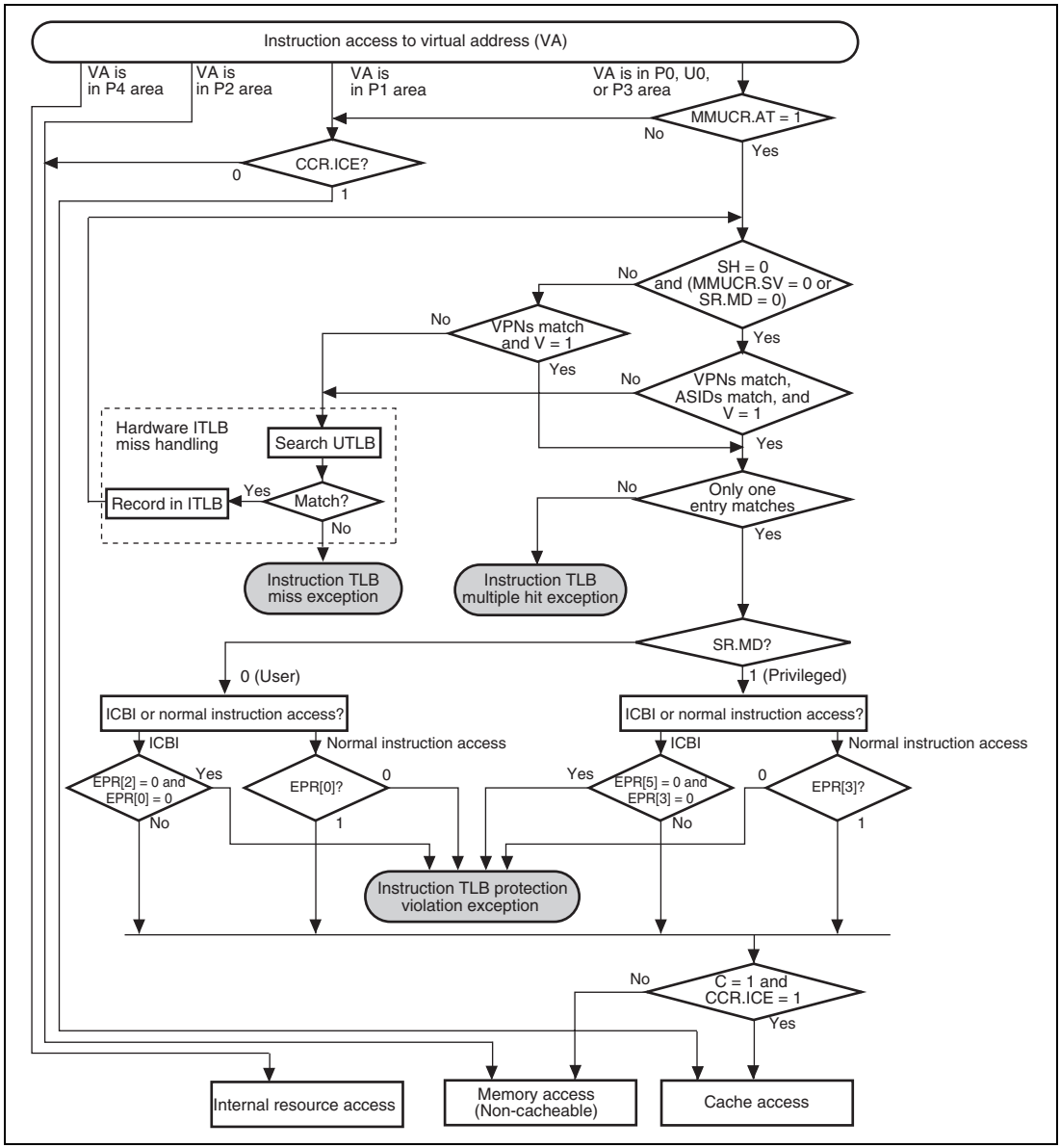


Figure 7.15 Flowchart of Memory Access Using ITLB (TLB Extended Mode)

7.5 MMU Functions

7.5.1 MMU Hardware Management

This LSI supports the following MMU functions.

1. The MMU decodes the virtual address to be accessed by software, and performs address translation by controlling the UTLB/ITLB in accordance with the MMUCR settings.
2. The MMU determines the cache access status on the basis of the page management information read during address translation (C and WT bits).
3. If address translation cannot be performed normally in a data access or instruction access, the MMU notifies software by means of an MMU exception.
4. If address translation information is not recorded in the ITLB in an instruction access, the MMU searches the UTLB. If the necessary address translation information is recorded in the UTLB, the MMU copies this information into the ITLB in accordance with the LRUI bit setting in MMUCR.

7.5.2 MMU Software Management

Software processing for the MMU consists of the following:

1. Setting of MMU-related registers. Some registers are also partially updated by hardware automatically.
2. Recording, deletion, and reading of TLB entries. There are two methods of recording UTLB entries: by using the LDTLB instruction, or by writing directly to the memory-mapped UTLB. ITLB entries can only be recorded by writing directly to the memory-mapped ITLB. Deleting or reading UTLB/ITLB entries is enabled by accessing the memory-mapped UTLB/ITLB.
3. MMU exception handling. When an MMU exception occurs, processing is performed based on information set by hardware.

7.5.3 MMU Instruction (LDTLB)

A TLB load instruction (LDTLB) is provided for recording UTLB entries. When an LDTLB instruction is issued, this LSI copies the contents of PTEH and PTEL (also the contents of PTEA in TLB extended mode) to the UTLB entry indicated by the URC bit in MMUCR. ITLB entries are not updated by the LDTLB instruction, and therefore address translation information purged from the UTLB entry may still remain in the ITLB entry. As the LDTLB instruction changes address translation information, ensure that it is issued by a program in the P1 or P2 area.

After the LDTLB instruction has been executed, execute one of the following three methods before an access (include an instruction fetch) the area where TLB is used to translate the address is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the area where TLB is used to translate the address.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the LT bit in IRMCR is 0 (initial value) before executing the LDTLB instruction, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note: that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The operation of the LDTLB instruction is shown in figure 7.16 and 7.17.

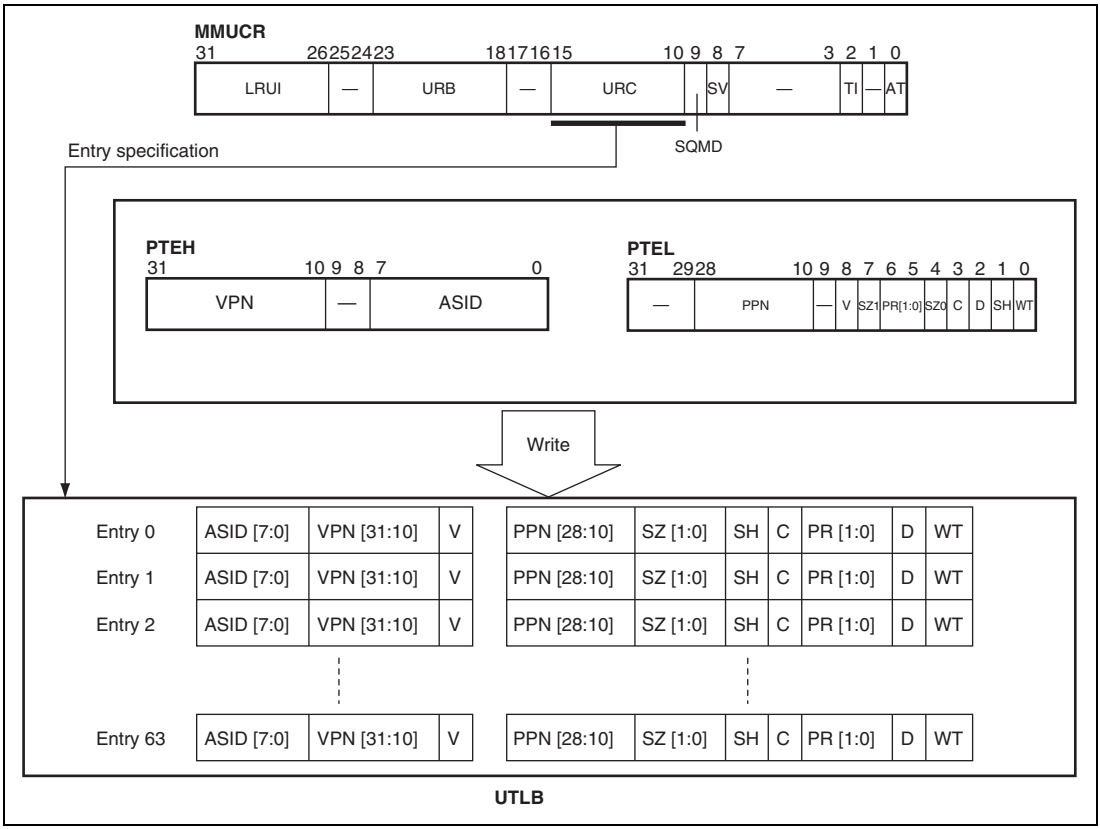


Figure 7.16 Operation of LDTLB Instruction (TLB Compatible Mode)

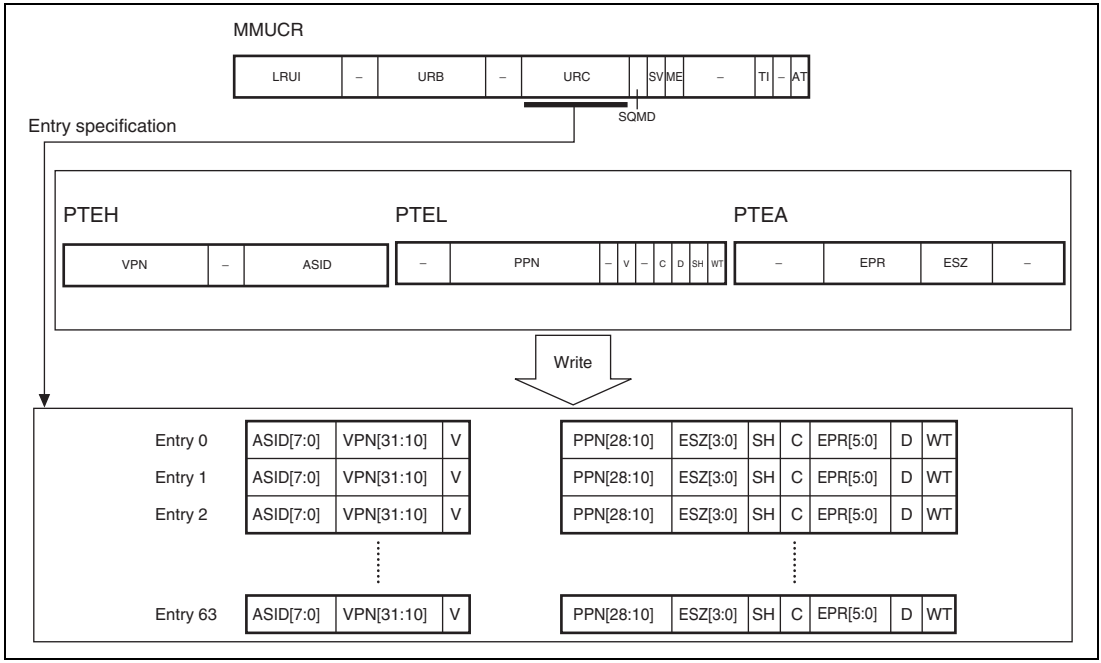


Figure 7.17 Operation of LDTLB Instruction (TLB Extended Mode)

7.5.4 Hardware ITLB Miss Handling

In an instruction access, this LSI searches the ITLB. If it cannot find the necessary address translation information (ITLB miss occurred), the UTLB is searched by hardware, and if the necessary address translation information is present, it is recorded in the ITLB. This procedure is known as hardware ITLB miss handling. If the necessary address translation information is not found in the UTLB search, an instruction TLB miss exception is generated and processing passes to software.

7.5.5 Avoiding Synonym Problems

When information on 1- or 4-Kbyte pages is written as TLB entries, a synonym problem may arise. The problem is that, when a number of virtual addresses are mapped onto a single physical address, the same physical address data is written to a number of cache entries, and it becomes impossible to guarantee data integrity. This problem does not occur with the instruction TLB and instruction cache because only data is read in these cases. In this LSI, entry specification is performed using bits 12 to 5 of the virtual address in order to achieve fast operand cache operation. However, bits 12 to 10 of the virtual address in the case of a 1-Kbyte page, and bit 12 of the virtual address in the case of a 4-Kbyte page, are subject to address translation. As a result, bits 12 to 10 of the physical address after translation may differ from bits 12 to 10 of the virtual address.

Consequently, the following restrictions apply to the writing of address translation information as UTLB entries.

- When address translation information whereby a number of 1-Kbyte page UTLB entries are translated into the same physical address is written to the UTLB, ensure that the VPN[12:10] values are the same.
- When address translation information whereby a number of 4-Kbyte page UTLB entries are translated into the same physical address is written to the UTLB, ensure that the VPN[12] value is the same.
- Do not use 1-Kbyte page UTLB entry physical addresses with UTLB entries of a different page size.
- Do not use 4-Kbyte page UTLB entry physical addresses with UTLB entries of a different page size.

The above restrictions apply only when performing accesses using the cache.

For cache sizes other than 32 Kbytes, the page sizes that can lead to synonym problems and the bits in VPN the value of which should be matched at the time of writing entries to the UTBL are different from those shown in the above explanation. The page sizes that can lead to synonym problems are shown in table 7.3 for cache sizes of 8 Kbytes to 64 Kbytes.

7.6 MMU Exceptions

There are seven MMU exceptions: instruction TLB multiple hit exception, instruction TLB miss exception, instruction TLB protection violation exception, data TLB multiple hit exception, data TLB miss exception, data TLB protection violation exception, and initial page write exception. Refer to figures 7.9, 7.10, 7.14, 7.15, and section 5, Exception Handling for the conditions under which each of these exceptions occurs.

7.6.1 Instruction TLB Multiple Hit Exception

An instruction TLB multiple hit exception occurs when more than one ITLB entry matches the virtual address to which an instruction access has been made. If multiple hits occur when the UTLB is searched by hardware in hardware ITLB miss handling, an instruction TLB multiple hit exception will result.

When an instruction TLB multiple hit exception occurs, a reset is executed and cache coherency is not guaranteed.

(1) Hardware Processing

In the event of an instruction TLB multiple hit exception, hardware carries out the following processing:

1. Sets the virtual address at which the exception occurred in TEA.
2. Sets exception code H'140 in EXPEVT.
3. Branches to the reset handling routine (H'A000 0000).

(2) Software Processing (Reset Routine)

The ITLB entries which caused the multiple hit exception are checked in the reset handling routine. This exception is intended for use in program debugging, and should not normally be generated.

7.6.2 Instruction TLB Miss Exception

An instruction TLB miss exception occurs when address translation information for the virtual address to which an instruction access is made is not found in the UTLB entries by the hardware ITLB miss handling routine. The instruction TLB miss exception processing carried out by hardware and software is shown below. This is the same as the processing for a data TLB miss exception.

(1) Hardware Processing

In the event of an instruction TLB miss exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'040 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0400 to the contents of VBR, and starts the instruction TLB miss exception handling routine.

(2) Software Processing (Instruction TLB Miss Exception Handling Routine)

Software is responsible for searching the external memory page table and assigning the necessary page table entry. Software should carry out the following processing in order to find and assign the necessary page table entry.

1. In TLB compatible mode, write to PTEL the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to PTEL and PTEA the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
2. When the entry to be replaced in entry replacement is specified by software, write the value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.

3. In TLB compatible mode, execute the LDTLB instruction and write the contents of PTEH and PTEL to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of PTEH, PTEL, PTEA to the UTLB.
4. Finally, execute the exception handling return instruction (RTE) to terminate the exception handling routine and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

For the execution of the LDTLB instruction, see section 7.9.1, Note on Using LDTLB Instruction.

7.6.3 Instruction TLB Protection Violation Exception

An instruction TLB protection violation exception occurs when, even though an ITLB entry contains address translation information matching the virtual address to which an instruction access is made, the actual access type is not permitted by the access right specified by the PR or EPR bit. The instruction TLB protection violation exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of an instruction TLB protection violation exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'0A0 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the instruction TLB protection violation exception handling routine.

(2) Software Processing (Instruction TLB Protection Violation Exception Handling Routine)

Resolve the instruction TLB protection violation, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.6.4 Data TLB Multiple Hit Exception

A data TLB multiple hit exception occurs when more than one UTLB entry matches the virtual address to which a data access has been made.

When a data TLB multiple hit exception occurs, a reset is executed, and cache coherency is not guaranteed. The contents of PPN in the UTLB prior to the exception may also be corrupted.

(1) Hardware Processing

In the event of a data TLB multiple hit exception, hardware carries out the following processing:

1. Sets the virtual address at which the exception occurred in TEA.
2. Sets exception code H'140 in EXPEVT.
3. Branches to the reset handling routine (H'A000 0000).

(2) Software Processing (Reset Routine)

The UTLB entries which caused the multiple hit exception are checked in the reset handling routine. This exception is intended for use in program debugging, and should not normally be generated.

7.6.5 Data TLB Miss Exception

A data TLB miss exception occurs when address translation information for the virtual address to which a data access is made is not found in the UTLB entries. The data TLB miss exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of a data TLB miss exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.

3. Sets exception code H'040 in the case of a read, or H'060 in the case of a write in EXPEVT (OCBP, OCBWB: read; OCBI, MOVCA.L: write).
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0400 to the contents of VBR, and starts the data TLB miss exception handling routine.

(2) Software Processing (Data TLB Miss Exception Handling Routine)

Software is responsible for searching the external memory page table and assigning the necessary page table entry. Software should carry out the following processing in order to find and assign the necessary page table entry.

1. In TLB compatible mode, write to PTEL the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to PTEL and PTEA the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
2. When the entry to be replaced in entry replacement is specified by software, write the value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
3. In TLB compatible mode, execute the LDTLB instruction and write the contents of PTEH and PTEL to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of PTEH, PTEL, PTEA to the UTLB.
4. Finally, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

For the execution of the LDTLB instruction, see section 7.9.1, Note on Using LDTLB Instruction.

7.6.6 Data TLB Protection Violation Exception

A data TLB protection violation exception occurs when, even though a UTLB entry contains address translation information matching the virtual address to which a data access is made, the actual access type is not permitted by the access right specified by the PR or EPR bit. The data TLB protection violation exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of a data TLB protection violation exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'0A0 in the case of a read, or H'0C0 in the case of a write in EXPEVT (OCBP, OCBWB: read; OCBI, MOVCA.L: write).
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the data TLB protection violation exception handling routine.

(2) Software Processing (Data TLB Protection Violation Exception Handling Routine)

Resolve the data TLB protection violation, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.6.7 Initial Page Write Exception

An initial page write exception occurs when the D bit is 0 even though a UTLB entry contains address translation information matching the virtual address to which a data access (write) is made, and the access is permitted. The initial page write exception processing carried out by hardware and software is shown below.

(1) Hardware Processing

In the event of an initial page write exception, hardware carries out the following processing:

1. Sets the VPN of the virtual address at which the exception occurred in PTEH.
2. Sets the virtual address at which the exception occurred in TEA.
3. Sets exception code H'080 in EXPEVT.
4. Sets the PC value indicating the address of the instruction at which the exception occurred in SPC. If the exception occurred at a delay slot, sets the PC value indicating the address of the delayed branch instruction in SPC.
5. Sets the SR contents at the time of the exception in SSR. The R15 contents at this time are saved in SGR.
6. Sets the MD bit in SR to 1, and switches to privileged mode.
7. Sets the BL bit in SR to 1, and masks subsequent exception requests.
8. Sets the RB bit in SR to 1.
9. Branches to the address obtained by adding offset H'0000 0100 to the contents of VBR, and starts the initial page write exception handling routine.

(2) Software Processing (Initial Page Write Exception Handling Routine)

Software is responsible for the following processing:

1. Retrieve the necessary page table entry from external memory.
2. Write 1 to the D bit in the external memory page table entry.
3. In TLB compatible mode, write to PTEL the values of the PPN, PR, SZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
   In TLB extended mode, write to PTEL and PTEA the values of the PPN, EPR, ESZ, C, D, SH, V, and WT bits in the page table entry stored in the address translation table for external memory.
4. When the entry to be replaced in entry replacement is specified by software, write that value to the URC bits in MMUCR. If URC is greater than URB at this time, the value should be changed to an appropriate value after issuing an LDTLB instruction.
5. In TLB compatible mode, execute the LDTLB instruction and write the contents of PTEH and PTEL to the TLB.
   In TLB extended mode, execute the LDTLB instruction and write the contents of PTEH, PTEL, PTEA to the UTLB.
6. Finally, execute the exception handling return instruction (RTE), terminate the exception handling routine, and return control to the normal flow. The RTE instruction should be issued at least one instruction after the LDTLB instruction.

7.7 Memory-Mapped TLB Configuration

To enable the ITLB and UTLB to be managed by software, their contents are allowed to be read from and written to by a program in the P1/P2 area with a MOV instruction in privileged mode. Operation is not guaranteed if access is made from a program in another area.

After the memory-mapped TLB has been accessed, execute one of the following three methods before an access (including an instruction fetch) to an area other than the P1/P2 area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be an area other than the P1/P2 area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the MT bit in IRMCR is 0 (initial value) before accessing the memory-mapped TLB, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after MMUCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

The ITLB and UTLB are allocated to the P4 area in the virtual address space.

In TLB compatible mode, VPN, V, and ASID in the ITLB can be accessed as an address array, PPN, V, SZ, PR, C, and SH as a data array. VPN, D, V, and ASID in the UTLB can be accessed as an address array, PPN, V, SZ, PR, C, D, WT, and SH as a data array. V and D can be accessed from both the address array side and the data array side.

In TLB extended mode, VPN, V, and ASID in the ITLB can be accessed as an address array, PPN, V, ESZ, EPR, C, and SH as a data array. VPN, D, V, and ASID in the UTLB can be accessed as an address array, PPN, V, ESZ, EPR, C, D, WT, and SH as a data array. V and D can be accessed from both the address array side and the data array side.

In both TLB compatible mode and TLB extended mode, only longword access is possible. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of 0 should be specified; their read value is undefined.

7.7.1 ITLB Address Array

The ITLB address array is allocated to addresses H'F200 0000 to H'F2FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and VPN, V, and ASID to be written to the address array are specified in the data field.

In the address field, bits [31:24] have the value H'F2 indicating the ITLB address array and the entry is specified by bits [9:8]. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, bits [31:10] indicate VPN, bit [8] indicates V, and bits [7:0] indicate ASID.

The following two kinds of operation can be used on the ITLB address array:

1. ITLB address array read

   VPN, V, and ASID are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB address array write

   VPN, V, and ASID specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

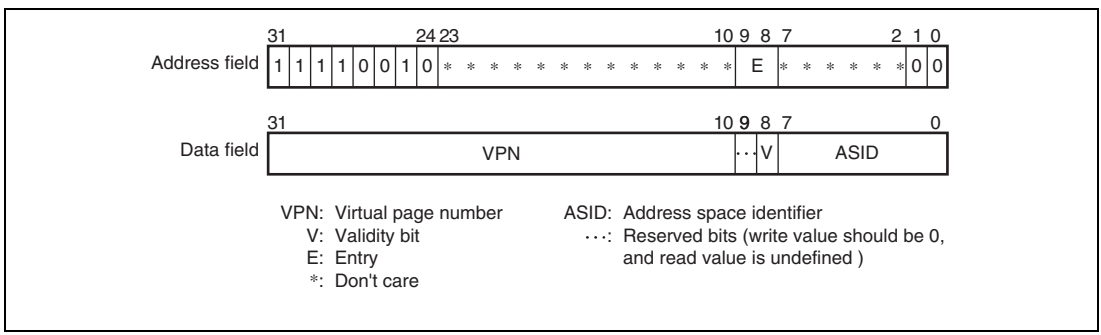


Figure 7.18 Memory-Mapped ITLB Address Array

7.7.2 ITLB Data Array (TLB Compatible Mode)

The ITLB data array is allocated to addresses H'F300 0000 to H'F37F FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and PPN, V, SZ, PR, C, and SH to be written to the data array are specified in the data field.

In the address field, bits [31:23] have the value H'F30 indicating ITLB data array and the entry is specified by bits [9:8].

In the data field, bits [28:10] indicate PPN, bit [8] indicates V, bits [7] and [4] indicate SZ, bit [6] indicates PR, bit [3] indicates C, and bit [1] indicates SH.

The following two kinds of operation can be used on ITLB data array:

1. ITLB data array read

   PPN, V, SZ, PR, C, and SH are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB data array write

   PPN, V, SZ, PR, C, and SH specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

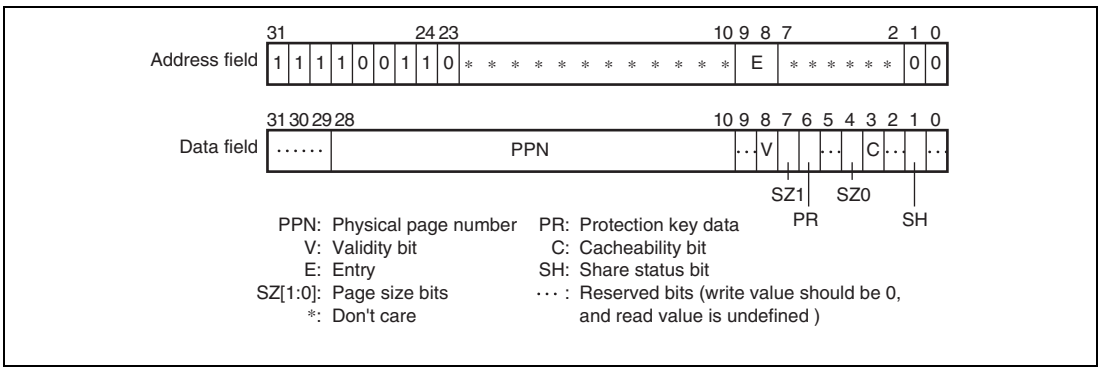


Figure 7.19 Memory-Mapped ITLB Data Array (TLB Compatible Mode)

7.7.3 ITLB Data Array (TLB Extended Mode)

In TLB extended mode the names of the data arrays have been changed from ITLB data array to ITLB data array 1, ITLB data array 2 is added, and the EPR and ESZ bits are accessible. In TLB extended mode, the PR and SZ bits of ITLB data array 1 are reserved and 0 should be specified as the write value for these bits. In addition, when a write to ITLB data array 1 is performed, a write to ITLB data array 2 of the same entry should always be performed.

In TLB compatible mode (MMUCR.ME = 0), ITLB data array 2 cannot be accessed. Operation if they are accessed is not guaranteed.

(1) ITLB Data Array 1

In TLB extended mode, bits 7, 6, and 4 in the data field, which correspond to the PR and SZ bits in compatible mode, are reserved. Specify 0 as the write value for these bits.

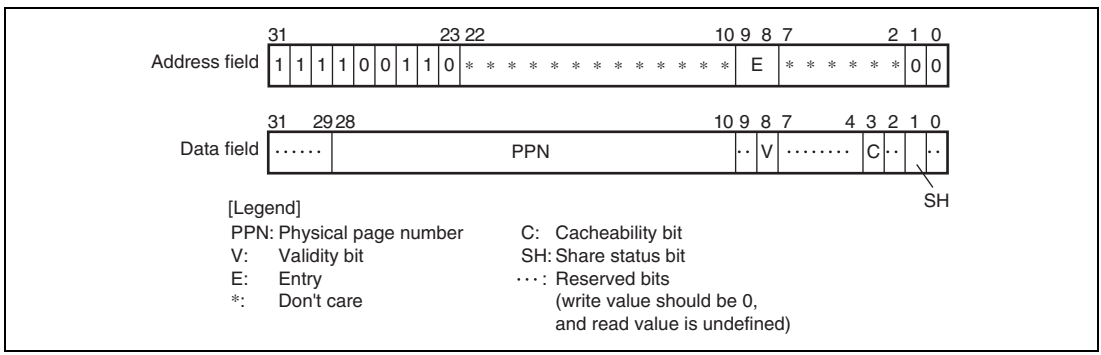


Figure 7.20 Memory-Mapped ITLB Data Array 1 (TLB Extended Mode)

(2) ITLB Data Array 2

The ITLB data array is allocated to addresses H'F380 0000 to H'F3FF FFFF in the P4 area. Access to data array 2 requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and EPR and ESZ to be written to data array 2 are specified in the data field.

In the address field, bits [31:23] have the value H'F38 indicating ITLB data array 2 and the entry is specified by bits [9:8].

In the data field, bits [13], [11], [10], and [8] indicate EPR[5], [3], [2], and [0], and bits [7:4] indicate ESZ, respectively.

The following two kinds of operation can be applied to ITLB data array 2:

1. ITLB data array 2 read

   EPR and ESZ are read into the data field from the ITLB entry corresponding to the entry set in the address field.
2. ITLB data array 2 write

   EPR and ESZ specified in the data field are written to the ITLB entry corresponding to the entry set in the address field.

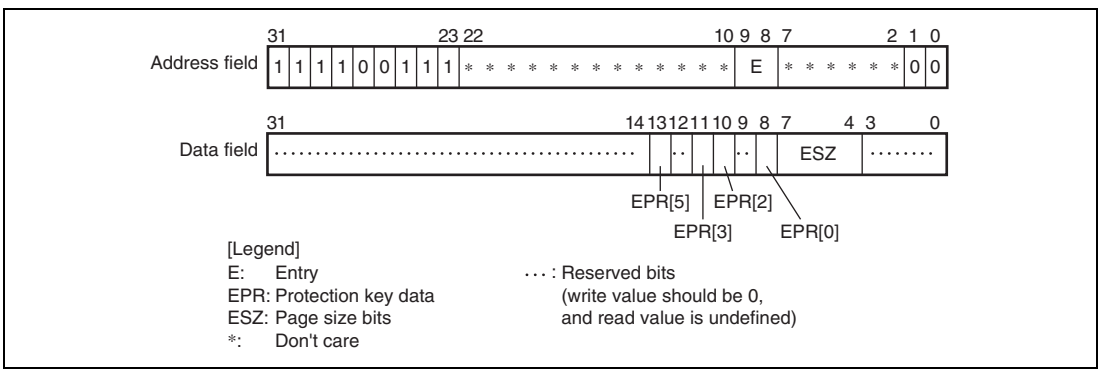


Figure 7.21 Memory-Mapped ITLB Data Array 2 (TLB Extended Mode)

7.7.4 UTLB Address Array

The UTLB address array is allocated to addresses H'F600 0000 to H'F60F FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and VPN, D, V, and ASID to be written to the address array are specified in the data field.

In the address field, bits [31:20] have the value H'F60 indicating the UTLB address array and the entry is specified by bits [13:8]. Bit [7] that is the association bit (A bit) in the address field specifies whether address comparison is performed in a write to the UTLB address array.

In the data field, bits [31:10] indicate VPN, bit [9] indicates D, bit [8] indicates V, and bits [7:0] indicate ASID.

The following three kinds of operation can be used on the UTLB address array:

1. UTLB address array read

   VPN, D, V, and ASID are read into the data field from the UTLB entry corresponding to the entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is 1 or 0.
2. UTLB address array write (non-associative)

   VPN, D, V, and ASID specified in the data field are written to the UTLB entry corresponding to the entry set in the address field. The A bit in the address field should be cleared to 0.
3. UTLB address array write (associative)

   When a write is performed with the A bit in the address field set to 1, comparison of all the UTLB entries is carried out using the VPN specified in the data field and ASID in PTEH. The usual address comparison rules are followed, but if a UTLB miss occurs, the result is no operation, and an exception is not generated. If the comparison identifies a UTLB entry corresponding to the VPN specified in the data field, D and V specified in the data field are written to that entry. This associative operation is simultaneously carried out on the ITLB, and if a matching entry is found in the ITLB, V is written to that entry. Even if the UTLB comparison results in no operation, a write to the ITLB is performed as long as a matching entry is found in the ITLB. If there is a match in both the UTLB and ITLB, the UTLB information is also written to the ITLB.

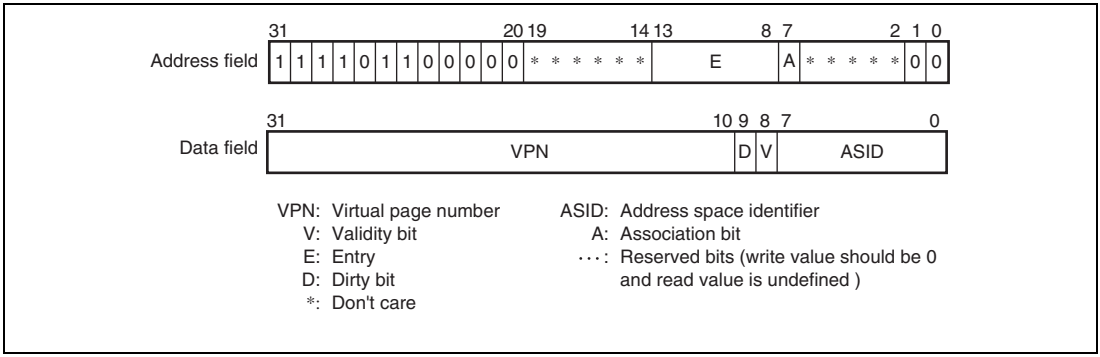


Figure 7.22 Memory-Mapped UTLB Address Array

7.7.5 UTLB Data Array (TLB Compatible Mode)

The UTLB data array is allocated to addresses H'F700 0000 to H'F70F FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and PPN, V, SZ, PR, C, D, SH, and WT to be written to data array are specified in the data field.

In the address field, bits [31:20] have the value H'F70 indicating UTLB data array and the entry is specified by bits [13:8].

In the data field, bits [28:10] indicate PPN, bit [8] indicates V, bits [7] and [4] indicate SZ, bits [6:5] indicate PR, bit [3] indicates C, bit [2] indicates D, bit [1] indicates SH, and bit [0] indicates WT.

The following two kinds of operation can be used on UTLB data array:

1. UTLB data array read

   PPN, V, SZ, PR, C, D, SH, and WT are read into the data field from the UTLB entry corresponding to the entry set in the address field.
2. UTLB data array write

   PPN, V, SZ, PR, C, D, SH, and WT specified in the data field are written to the UTLB entry corresponding to the entry set in the address field.

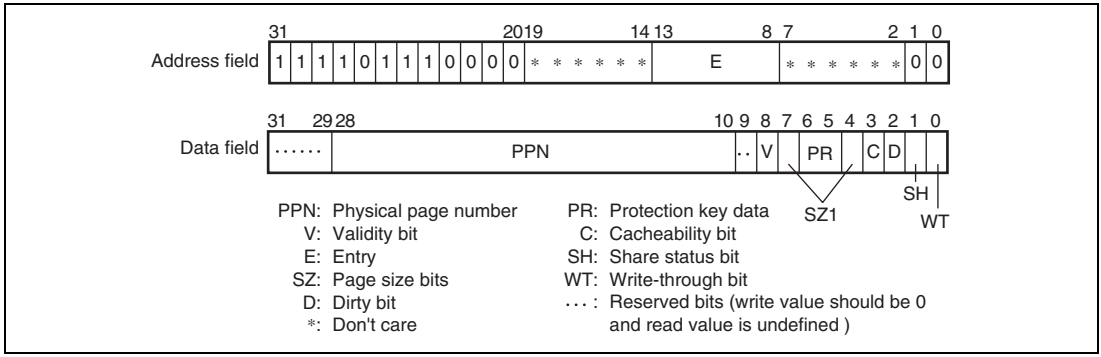


Figure 7.23 Memory-Mapped UTLB Data Array (TLB Compatible Mode)

7.7.6 UTLB Data Array (TLB Extended Mode)

In TLB extended mode, the names of the data arrays have been changed from UTLB data array to UTLB data array 1, UTLB data array 2 is added, and the EPR and ESZ bits are accessible. In TLB extended mode, the PR and SZ bits of UTLB data array 1 are reserved and 0 should be specified as the write value for these bits. In addition, when a write to UTLB data array 1 is performed, a write to UTLB data array 2 of the same entry should always be performed after that.

In TLB compatible mode (MMUCR.ME = 0), UTLB data array 2 cannot be accessed. Operation if they are accessed is not guaranteed.

(1) UTLB Data Array 1

In TLB extended mode, bits 7 to 4 in the data field, which correspond to the PR and SZ bits in compatible mode, are reserved. Specify 0 as the write value for these bits.

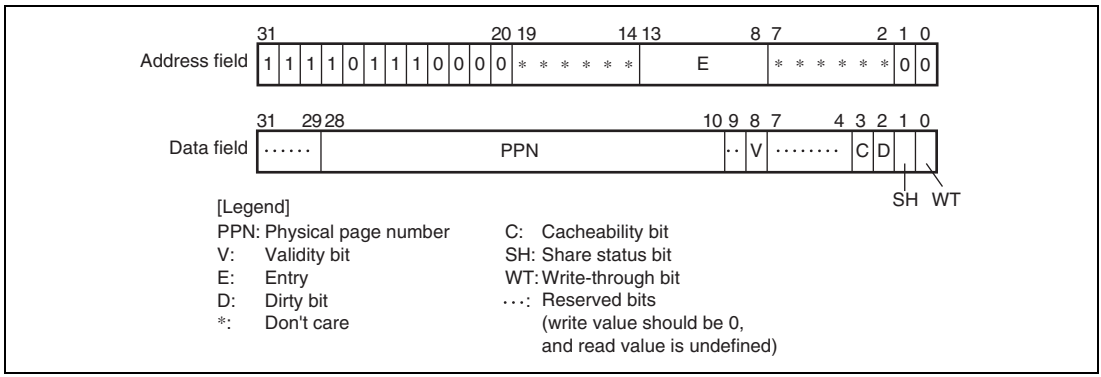


Figure 7.24 Memory-Mapped UTLB Data Array 1 (TLB Extended Mode)

(2) UTLB Data Array 2

The UTLB data array is allocated to addresses H'F780 0000 to H'F78F FFFF in the P4 area. Access to data array 2 requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification (when writing). Information for selecting the entry to be accessed is specified in the address field, and EPR and ESZ to be written to data array 2 are specified in the data field.

In the address field, bits [31:20] have the value H'F78 indicating UTLB data array 2 and the entry is specified by bits [13:8].

In the data field, bits [13:8] indicate EPR, and bits [7:4] indicate ESZ, respectively.

The following two kinds of operation can be applied to UTLB data array 2:

1. UTLB data array 2 read

   EPR and ESZ are read into the data field from the UTLB entry corresponding to the entry set in the address field.
2. UTLB data array 2 write

   EPR and ESZ specified in the data field are written to the UTLB entry corresponding to the entry set in the address field.

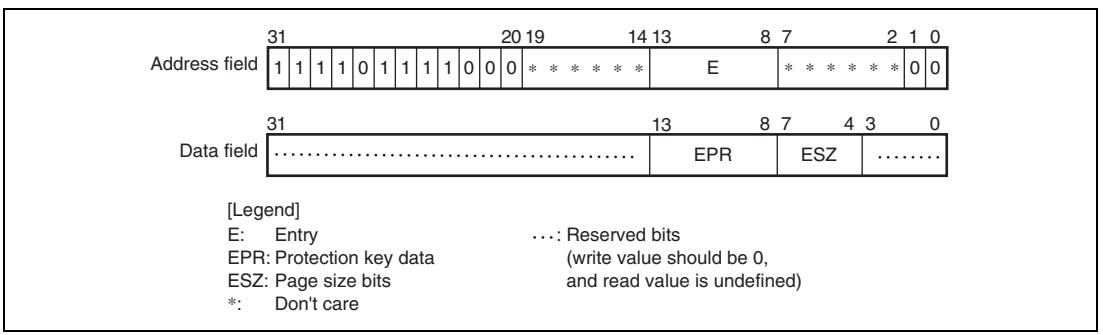


Figure 7.25 Memory-Mapped UTLB Data Array 2 (TLB Extended Mode)

7.8 32-Bit Address Extended Mode

Setting the SE bit in PASCR to 1 changes mode from 29-bit address mode which handles the 29-bit physical address space to 32-bit address extended mode which handles the 32-bit physical address space.

Note: For support/Unsupport of the 32-bit address extended mode, see the hardware manual of the product.

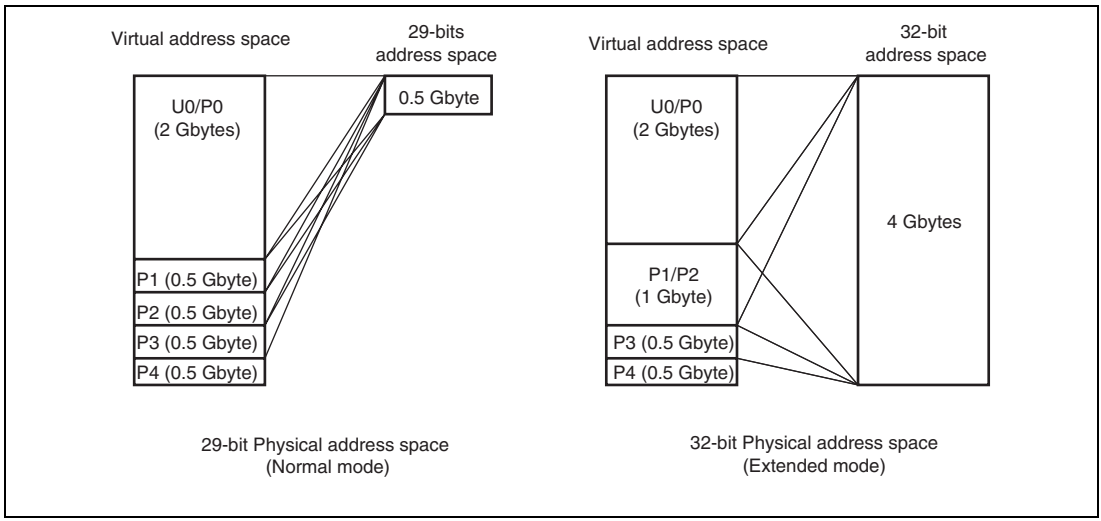


Figure 7.26 Physical Address Space (32-Bit Address Extended Mode)

7.8.1 Overview of 32-Bit Address Extended Mode

In 32-bit address extended mode, the privileged space mapping buffer (PMB) is introduced. The PMB maps virtual addresses in the P1 or P2 area which are not translated in 29-bit address mode to the 32-bit physical address space. In areas which are target for address translation of the TLB (UTLB/ITLB), upper three bits in the PPN field of the UTLB or ITLB are extended and then addresses after the TLB translation can handle the 32-bit physical addresses.

As for the cache operation, P1 area is cacheable and P2 area is non-cacheable in the case of 29-bit address mode, but the cache operation of both P1 and P2 area are determined by the C bit and WT bit in the PMB in the case of 32-bit address mode.

7.8.2 Transition to 32-Bit Address Extended Mode

This LSI enters 29-bit address mode after a power-on reset. Transition is made to 32-bit address extended mode by setting the SE bit in PASCR to 1. In 32-bit address extended mode, the MMU operates as follows.

1. When the AT bit in MMUCR is 0, virtual addresses in the U0, P0, or P3 area become 32-bit physical addresses. Addresses in the P1 or P2 area are translated according to the PMB mapping information. B'10 should be set to the upper 2 bits of virtual page number (VPN[31:30]) in the PMB in order to indicate P1 or P2 area. The operation is not guaranteed when the value except B'10 is set to these bits.
2. When the AT bit in MMUCR is 1, virtual addresses in the U0, P0, or P3 area are translated to 32-bit physical addresses according to the TLB conversion information. Addresses in the P1 or P2 area are translated according to the PMB mapping information. B'10 should be set to the upper 2 bits of virtual page number (VPN[31:30]) in the PMB in order to indicate P1 or P2 area. The operation is not guaranteed when the value except B'10 is set to these bits.
3. Regardless of the setting of the AT bit in MMUCR, bits 31 to 29 in physical addresses become B'111 in the control register area (addresses H'FC00 0000 to H'FFFF FFFF). When the control register area is recorded in the UTLB and accessed, B'111 should be set to PPN[31:29].

7.8.3 Privileged Space Mapping Buffer (PMB) Configuration

In 32-bit address extended mode, virtual addresses in the P1 or P2 area are translated according to the PMB mapping information. The PMB has 16 entries and configuration of each entry is as follows.

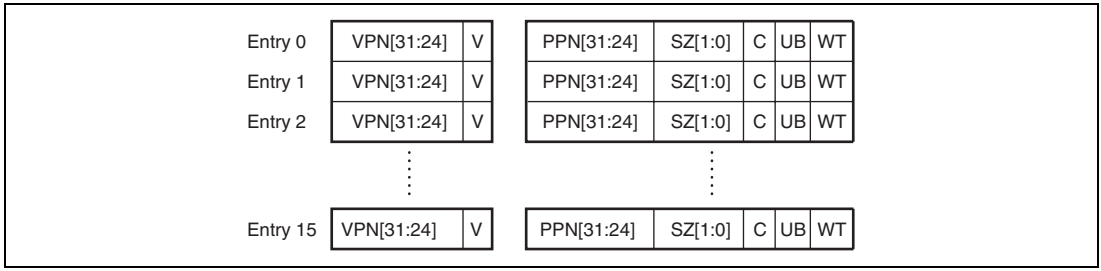


Figure 7.27 PMB Configuration

[Legend]

- VPN: Virtual page number
  For 16-Mbyte page: Upper 8 bits of virtual address
  For 64-Mbyte page: Upper 6 bits of virtual address
  For 128-Mbyte page: Upper 5 bits of virtual address
  For 512-Mbyte page: Upper 3 bits of virtual address

Note: B'10 should be set to the upper 2 bits of VPN in order to indicate P1 or P2 area.

- SZ: Page size bits
  Specify the page size.
  00: 16-Mbyte page
  01: 64-Mbyte page
  10: 128-Mbyte page
  11: 512-Mbyte page
- V: Validity bit
  Indicates whether the entry is valid.
  0: Invalid
  1: Valid
  Cleared to 0 by a power-on reset.
  Not affected by a manual reset.
- PPN: Physical page number
  Upper 8 bits of the physical address of the physical page number.
  With a 16-Mbyte page, PPN[31:24] are valid.
  With a 64-Mbyte page, PPN[31:26] are valid.
  With a 128-Mbyte page, PPN[31:27] are valid.
  With a 512-Mbyte page, PPN[31:29] are valid.

- C: Cacheability bit
  Indicates whether a page is cacheable.
  0: Not cacheable
  1: Cacheable
- WT: Write-through bit
  Specifies the cache write mode.
  0: Copy-back mode
  1: Write-through mode
- UB: Buffered write bit
  Specifies whether a buffered write is performed.
  0: Buffered write (Data access of subsequent processing proceeds without waiting for the write to complete.)
  1: Unbuffered write (Data access of subsequent processing is stalled until the write has completed.)

7.8.4 PMB Function

This LSI supports the following PMB functions.

1. Only memory-mapped write can be used for writing to the PMB. The LDTLB instruction cannot be used to write to the PMB.
2. Software must ensure that every accessed P1 or P2 address has a corresponding PMB entry before the access occurs. When an access to an address in the P1 or P2 area which is not recorded in the PMB is made, this LSI is reset by the TLB. In this case, the accessed address in the P1 or P2 area which causes the TLB reset is stored in the TEA and code H′140 in the EXPEVT.
3. This LSI does not guarantee the operation when multiple hit occurs in the PMB. Special care should be taken when the PMB mapping information is recorded by software.
4. The PMB does not have an associative write function.
5. Since there is no PR field in the PMB, read/write protection cannot be preformed. The address translation target of the PMB is the P1 or P2 address. In user mode access, an address error exception occurs.
6. Both entries from the UTLB and PMB are mixed and recorded in the ITLB by means of the hardware ITLB miss handling. However, these entries can be identified by checking whether VPN[31:30] is 10 or not. When an entry from the PMB is recorded in the ITLB, H′00, 01, and 1 are recorded in the ASID, PR, and SH fields which do not exist in the PMB, respectively.

7.8.5 Memory-Mapped PMB Configuration

To enable the PMB to be managed by software with a MOV instruction in privileged mode, the PMB address array is allocated to addresses H'F610 0000 to H'F61F FFFF in the P4 area and the PMB data array to addresses H'F710 0000 to H'F71F FFFF in the P4 area. VPN and V in the PMB can be accessed as an address array, PPN, V, SZ, C, WT, and UB as a data array. V can be accessed from both the address array side and the data array side. A program which executes a PMB memory-mapped access should be placed in the page area at which the C bit in PMB is cleared to 0.

1. PMB address array read

   When memory reading is performed while bits 31 to 20 in the address field are specified as H'F61 which indicates the PMB address array and bits 11 to 8 in the address field as an entry, bits 31 to 24 in the data field are read as VPN and bit 8 in the data field as V.

2. PMB address array write

   When memory writing is performed while bits 31 to 20 in the address field are specified as H'F61 which indicates the PMB address array and bits 11 to 8 in the address field as an entry, and bits 31 to 24 in the data field are specified as VPN and bit 8 in the data field as V, data is written to the specified entry.

3. PMB data array read

   When memory reading is performed while bits 31 to 20 in the address field are specified as H'F71 which indicates the PMB data array and bits 11 to 8 in the address field as an entry, bits 31 to 24 in the data field are read as PPN, bit 9 in the data field as UB, bit 8 in the data field as V, bits 7 and 4 in the data field as SZ, bit 3 in the data field as C, and bit 0 in the data field as WT.

4. PMB data array write

   When memory writing is performed while bits 31 to 20 in the address field are specified as H'F71 which indicates the PMB data array and bits 11 to 8 in the address field as an entry, and bits 31 to 24 in the data field are specified as PPN, bit 9 in the data field as UB, bit 8 in the data field as V, bits 7 and 4 in the data field as SZ, bit 3 in the data field as C, and bit 0 in the data field as WT, data is written to the specified entry.

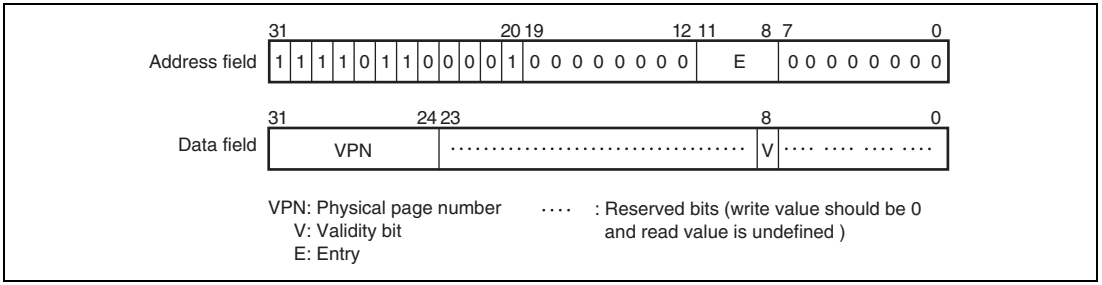


Figure 7.28 Memory-Mapped PMB Address Array

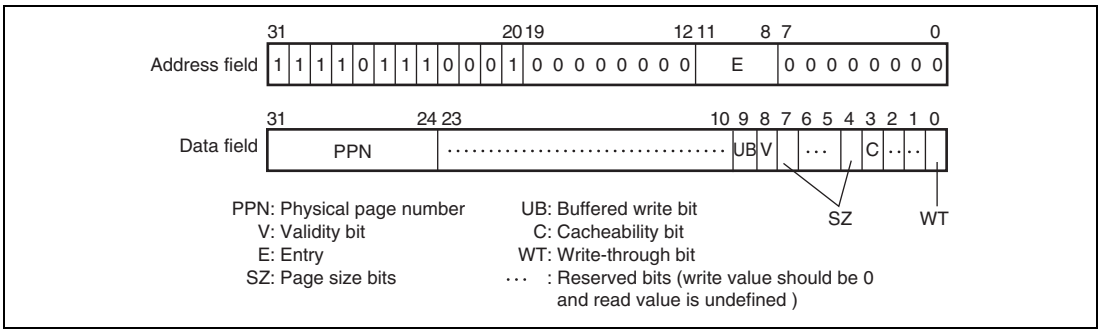


Figure 7.29 Memory-Mapped PMB Data Array

7.8.6 Notes on Using 32-Bit Address Extended Mode

When using 32-bit address extended mode, note that the items described in this section are extended or changed as follows.

(1) PASCR

The SE bit is added in bit 31 in the control register (PASCR). The bits 6 to 0 of the UB in the PASCR are invalid (Note that the bit 7 of the UB is still valid). When writing to the P1 or P2 area, the UB bit in the PMB controls whether a buffered write is performed or not. When the MMU is enabled, the UB bit in the TLB controls writing to the P0, P3, or U0 area. When the MMU is disabled, writing to the P0, P3, or U0 area is always performed as a buffered write.

Bit	Bit Name	Initial Value	R/W	Description
31	SE	0	R/W	0: 29-bit address mode 1: 32-bit address extended mode

Bit	Bit Name	Initial Value	R/W	Description
30 to 8	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
7 to 0	UB	All 0	R/W	Buffered Write Control for Each Area (64 Mbytes) When writing is performed without using the cache or in the cache write-through mode, these bits specify whether the CPU waits for the end of writing for each area. 0: The CPU does not wait for the end of writing 1: The CPU stalls and waits for the end of writing UB[7]: Corresponding to the control register area UB[6:0]: These bits are invalid in 32-bit address extended mode.

(2) ITLB

The PPN field in the ITLB is extended to bits 31 to 10.

(3) UTLB

The PPN field in the UTLB is extended to bits 31 to 10. The same UB bit as that in the PMB is added in each entry of the UTLB.

- UB: Buffered write bit
  Specifies whether a buffered write is performed.
  0: Buffered write (Subsequent processing proceeds without waiting for the write to complete.)
  1: Unbuffered write (Subsequent processing is stalled until the write has completed.)

In a memory-mapped TLB access, the UB bit can be read from or written to by bit 9 in the data array.

(4) PTEL

The same UB bit as that in the PMB is added in bit 9 in PTEL. This UB bit is written to the UB bit in the UTLB by the LDTLB instruction. The PPN field is extended to bits 31 to 10.

(5) CCR.CB

The CB bit in CCR is invalid. Whether a cacheable write for the P1 area is performed in copy-back mode or write-though mode is determined by the WT bit in the PMB.

(6) IRMCR.MT

The MT bit in IRMCR is valid for a memory-mapped PMB write.

(7) QACR0, QACR1

AREA0[4:2]/AREA1[4:2] fields of QACR0/QACR1 are extended to AREA0[7:2]/AREA1[7:2] corresponding to physical address [31:26].

(8) LSA0, LSA1, LDA0, LDA1

L0SADR, L1SADR, L0DADR, and L1DADR fields are extended to bits 31 to 10.

When using 32-bit address mode, the following notes should be applied to software.

1. For the SE bit switching, switching from 0 to 1 is only supported in a boot routine which is allocated in an area where caching and TLB-based address translation are not allowed and runs after a power-on reset or manual reset.
2. After switching the SE bit, an area in which the program is allocated becomes the target of the PMB address translation. Therefore, the area should be recorded in the PMB before switching the SE bit. An address which may be accessed in the P1 or P2 area such as the exception handler should also be recorded in the PMB.
3. When an external memory access occurs by an operand memory access located before the MOV.L instruction which switches the SE bit, external memory space addresses accessed in both address modes should be the same.
4. Note that the V bit is mapped to both address array and data array in PMB registration. That is, first write 0 to the V bit in one of arrays and then write 1 to the V bit in another array.

7.9 Usage Notes

7.9.1 Note on Using LDTLB Instruction

When using an LDTLB instruction instead of software to a value to the MMUCR. URC, execute 1 or 2 below.

1. In 29-bit address mode, follow A. and B. below. In 32-bit address mode, follow A. through D. below.
   A. Place the TLB miss exception handling routine*[1] only in the P1, P2 area ,or the on-chip memory so that all the instruction accesses*[3] in the TLB miss exception handling routine should occur solely in the P1, P2 area, or the on-chip memory. Clear the RP bit in the RAMCR register to 0 (initial value), when the TLB miss exception handling routine is placed in the on-chip memory.
   B. Use only one page of the PMB for instruction accesses*[3] in the TLB miss exception handling routine*[1]. In 32-bit address mode, do not place them in the last 64 bytes of a page of the PMB.
   C. In 32-bit address mode, obey 1 and 2 below when recording information in the UTLB in the MMU-related exception*[2] handling routine.
      a. When thea TLB miss exception occurs, and recording the information of a page with the access right in the UTLB, do not record the page, in which the exception has occurred, in the UTLB using the following two operations.
         - Specifies the protection key data that causes a protection violation exception upon re-execution of the instruction that has caused the TLB miss exception and records the page, in which the TLB miss exception has occurred, in the UTLB.
         - Specifies the protection key data that does not cause a protection violation exception in the protection violation exception handling routine to record the page in the UTLB and re-executes the instruction that has caused the protection violation exception.
      b. When an initial page write exception occurs and the TLB entry in the UTLB of which the dirty bit is 1 is replaced, before the write instruction for the page corresponding to this replaced TLB entry is completed, register the TLB entry of which the dirty bit is 1.
   D. Do not make an attempt to execute the FDIV or FSQRT instruction in the TLB miss exception handling routine.

2. If a TLB miss exception occurs, add 1 to MMUCR.URC before executing an LDTLB instruction.

Notes: 1. An exception handling routine is an entire set of instructions that are executed from the address (VBR + offset) upon occurrence of an exception to the RTE for returning to the original program or to the RTE delay slot.

2. MMU-related exceptions are: instruction TLB miss exception, instruction TLB miss protection violation exception, data TLB miss exception, data TLB protection violation exception, and initial page write exception.

3. Instruction accesses include the PREFI and ICBI instructions.

Section 8 Caches

This LSI has an on-chip 32-Kbyte instruction cache (IC) for instructions and an on-chip 32-Kbyte operand cache (OC) for data.

Note: For the size of instruction cache and operand cache, see the hardware manual of the product. This manual describes the 32-Kbyte case for each cache memory.
For different cache sizes, bit positions different from those shown in figures 8.1, 8.2, and 8.5 to 8.8 apply. The bit positions in ways and entries for various cache sizes are given in the table below. The bit positions in ways apply to figures 8.5 to 8.8, and those in entries apply to figures 8.1, 8.2, 8.5, 8.7, and 8.8.

Cache size	Way	Entry
8 Kbytes	bit[12:11]	bit[10:5]
16 Kbytes	bit[13:12]	bit[11:5]
32 Kbytes	bit[14:13]	bit[12:5]
64 Kbytes	bit[15:14]	bit[13:5]

8.1 Features

The features of the cache are given in table 8.1.

This LSI supports two 32-byte store queues (SQs) to perform high-speed writes to external memory. The features of the store queues are given in table 8.2.

Table 8.1 Cache Features

Item	Instruction Cache	Operand Cache
Capacity	32-Kbyte cache	32-Kbyte cache
Type	4-way set-associative, virtual address index/physical address tag	4-way set-associative, virtual address index/physical address tag
Line size	32 bytes	32 bytes
Entries	256 entries/way	256 entries/way
Write method	—	Copy-back/write-through selectable
Replacement method	LRU (least-recently-used) algorithm	LRU (least-recently-used) algorithm

Table 8.2 Store Queue Features

Item	Store Queues
Capacity	32 bytes × 2
Addresses	H'E000 0000 to H'E3FF FFFF
Write	Store instruction (1-cycle write)
Write-back	Prefetch instruction (PREF instruction)
Access right	When MMU is disabled: Determined by SQMD bit in MMUCR When MMU is enabled: Determined by PR for each page

The operand cache of this LSI is 4-way set associative, each may comprising 256 cache lines. Figure 8.1 shows the configuration of the operand cache.

The instruction cache is 4-way set-associative, each way comprising 256 cache lines. Figure 8.2 shows the configuration of the instruction cache.

This LSI has an IC way prediction scheme to reduce power consumption. In addition, memory-mapped associative writing, which is detectable as an exception, can be enabled by using the non-support detection exception register (EXPMASK). For details, see section 5, Exception Handling.

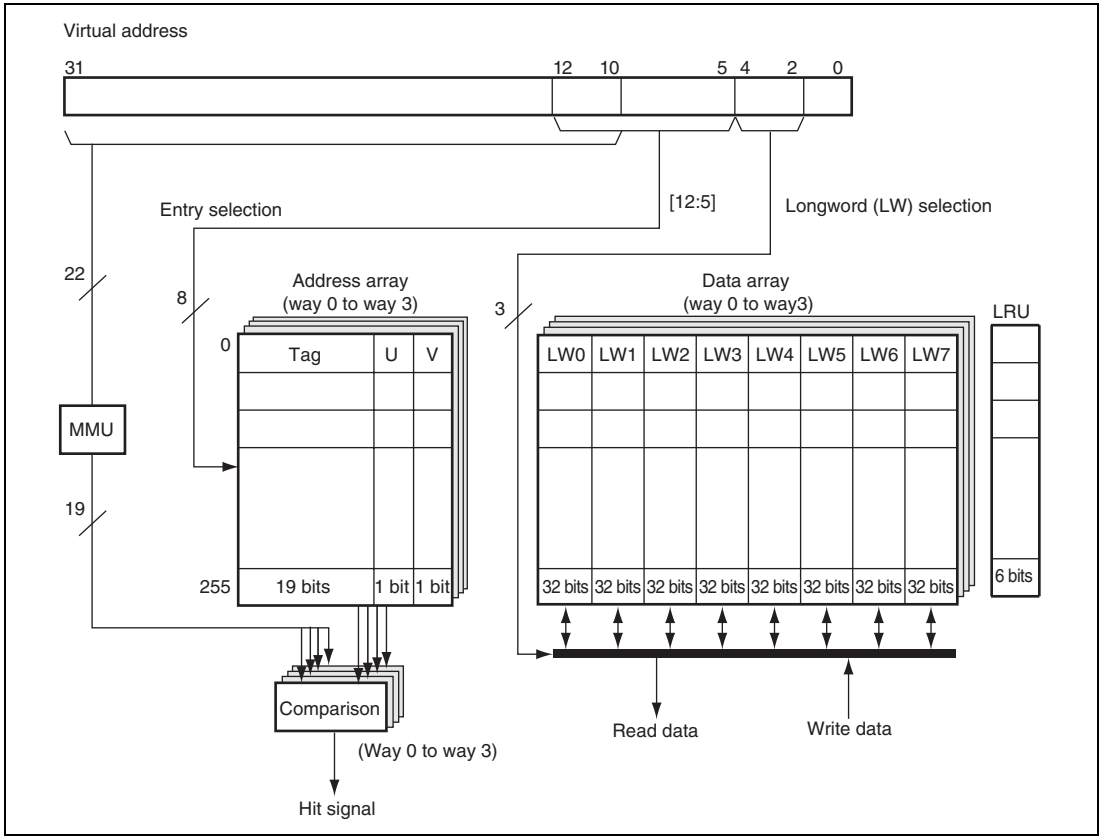


Figure 8.1 Configuration of Operand Cache (Cache size = 32 Kbytes)

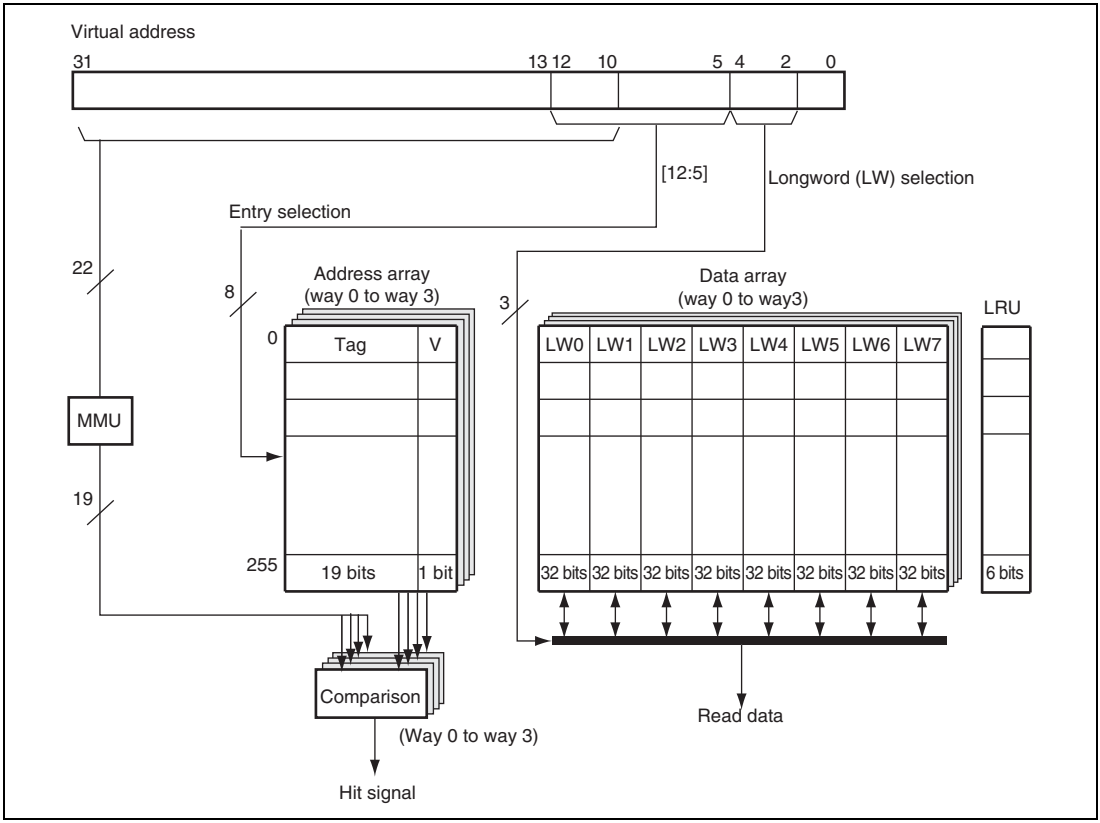


Figure 8.2 Configuration of Instruction Cache (Cache size = 32 Kbytes)

- Tag

  Stores the upper 19 bits of the 29-bit physical address of the data line to be cached. The tag is not initialized by a power-on or manual reset.

- V bit (validity bit)

  Indicates that valid data is stored in the cache line. When this bit is 1, the cache line data is valid. The V bit is initialized to 0 by a power-on reset, but retains its value in a manual reset.

- U bit (dirty bit)

  The U bit is set to 1 if data is written to the cache line while the cache is being used in copy-back mode. That is, the U bit indicates a mismatch between the data in the cache line and the data in external memory. The U bit is never set to 1 while the cache is being used in write-through mode, unless it is modified by accessing the memory-mapped cache (see section 8.6, Memory-Mapped Cache Configuration). The U bit is initialized to 0 by a power-on reset, but retains its value in a manual reset.

- Data array

  The data field holds 32 bytes (256 bits) of data per cache line. The data array is not initialized by a power-on or manual reset.

- LRU

  In a 4-way set-associative method, up to 4 items of data can be registered in the cache at each entry address. When an entry is registered, the LRU bit indicates which of the 4 ways it is to be registered in. The LRU mechanism uses 6 bits of each entry, and its usage is controlled by hardware. The LRU (least-recently-used) algorithm is used for way selection, and selects the less recently accessed way. The LRU bits are initialized to 0 by a power-on reset but not by a manual reset. The LRU bits cannot be read from or written to by software.

8.2 Register Descriptions

The following registers are related to cache.

Table 8.3 Register Configuration

Register Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Size
Cache control register	CCR	R/W	H'FF00 001C	H'1F00 001C	32
Queue address control register 0	QACR0	R/W	H'FF00 0038	H'1F00 0038	32
Queue address control register 1	QACR1	R/W	H'FF00 003C	H'1F00 003C	32
On-chip memory control register	RAMCR	R/W	H'FF00 0074	H'1F00 0074	32

Note: * These P4 addresses are for the P4 area in the virtual address space. These area 7 addresses are accessed from area 7 in the physical address space by means of the TLB.

Table 8.4 Register States in Each Processing State

Register Name	Abbreviation	Power-on Reset	Manual Reset	Sleep	Standby
Cache control register	CCR	H'0000 0000	H'0000 0000	Retained	Retained
Queue address control register 0	QACR0	Undefined	Undefined	Retained	Retained
Queue address control register 1	QACR1	Undefined	Undefined	Retained	Retained
On-chip memory control register	RAMCR	H'0000 0000	H'0000 0000	Retained	Retained

8.2.1 Cache Control Register (CCR)

CCR controls the cache operating mode, the cache write mode, and invalidation of all cache entries.

CCR modifications must only be made by a program in the non-cacheable P2 area or IL memory. After CCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the cacheable area is performed.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the cacheable area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating CCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after CCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ICI	—	—	ICE	—	—	—	—	OCI	CB	WT	OCE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
11	ICI	0	R/W	IC Invalidation Bit When 1 is written to this bit, the V bits of all IC entries are cleared to 0. This bit is always read as 0.

Bit	Bit Name	Initial Value	R/W	Description
10, 9	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
8	ICE	0	R/W	IC Enable Bit Selects whether the IC is used. Note however when address translation is performed, the IC cannot be used unless the C bit in the page management information is also 1. 0: IC not used 1: IC used
7 to 4	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
3	OCI	0	R/W	OC Invalidation Bit When 1 is written to this bit, the V and U bits of all OC entries are cleared to 0. This bit is always read as 0.
2	CB	0	R/W	Copy-Back Bit Indicates the P1 area cache write mode. 0: Write-through mode 1: Copy-back mode
1	WT	0	R/W	Write-Through Mode Indicates the P0, U0, and P3 area cache write mode. When address translation is performed, the value of the WT bit in the page management information has priority. 0: Copy-back mode 1: Write-through mode
0	OCE	0	R/W	OC Enable Bit Selects whether the OC is used. Note however when address translation is performed, the OC cannot be used unless the C bit in the page management information is also 1. 0: OC not used 1: OC used

8.2.2 Queue Address Control Register 0 (QACR0)

QACR0 specifies the area onto which store queue 0 (SQ0) is mapped when the MMU is disabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AREA0			—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
4 to 2	AREA0	Undefined	R/W	When the MMU is disabled, these bits generate physical address bits [28:26] for SQ0.
1, 0	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.

8.2.3 Queue Address Control Register 1 (QACR1)

QACR1 specifies the area onto which store queue 1 (SQ1) is mapped when the MMU is disabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AREA1			—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
4 to 2	AREA1	Undefined	R/W	When the MMU is disabled, these bits generate physical address bits [28:26] for SQ1.
1, 0	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.

8.2.4 On-Chip Memory Control Register (RAMCR)

RAMCR controls the number of ways in the IC and OC and prediction of the IC way.

RAMCR modifications must only be made by a program in the non-cacheable P2 area. After RAMCR has been updated, execute one of the following three methods before an access (including an instruction fetch) to the cacheable area or the IL memory area.

1. Execute a branch using the RTE instruction. In this case, the branch destination may be the cacheable area or the IL memory area.
2. Execute the ICBI instruction for any address (including non-cacheable area).
3. If the R2 bit in IRMCR is 0 (initial value) before updating RAMCR, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after RAMCR has been updated.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPW	—	—	—	L2FC	L2E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
9	RMD	0	R/W	On-Chip Memory Access Mode Bit For details, see section 10.4, IL Memory Protective Functions.
8	RP	0	R/W	On-Chip Memory Protection Enable Bit For details, see section 10.4, IL Memory Protective Functions.

Bit	Bit Name	Initial Value	R/W	Description
7	IC2W	0	R/W	IC Two-Way Mode bit 0: IC is a four-way operation 1: IC is a two-way operation For details, see section 8.4.3, IC Two-Way Mode.
6	OC2W	0	R/W	OC Two-Way Mode bit 0: OC is a four-way operation 1: OC is a two-way operation For details, see section 8.3.6, OC Two-Way Mode.
5	ICWPD	0	R/W	IC Way Prediction Stop Selects whether the IC way prediction is used. 0: Instruction cache performs way prediction. 1: Instruction cache does not perform way prediction.
4 to 2	—	All 0	R	Reserved For details on reading from or writing to these bits, see description in General Precautions on Handling of Product.
1	L2FC	0	R/W	Secondary Cache Forcible Coherency Mode For details, see section 9.3.2, Operation of Secondary Cache.
0	L2E	0	R/W	Secondary Cache Enable For details, see section 9.3.2, Operation of Secondary Cache.

8.3 Operand Cache Operation

8.3.1 Read Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is read from a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tags read from the each way is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and its V bit is 1, see No. 3.
- If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 0, see No. 4.
- If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 1, see No. 5.

3. Cache hit

   The data indexed by virtual address bits [4:0] is read from the data field of the cache line on the hitted way in accordance with the access size. Then the LRU bits are updated to indicate the hitted way is the latest one.

4. Cache miss (no write-back)

   Data is read into the cache line on the way, which is selected to replace, from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data(8 bytes) including the cache-missed data. When the corresponding data arrives in the cache, the read data is returned to the CPU. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and 0 is written to the U bit on the way. Then the LRU bit is updated to indicate the way is latest one.

5. Cache miss (with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then data is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data, and when the corresponding data arrives in the cache, the read data is returned to the CPU. While the remaining one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit, and 0 to the U bit. And the LRU bits are updated to indicate the way is latest one. The data in the write-back buffer is then written back to external memory.

8.3.2 Prefetch Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is prefetched from a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and its V bit is 1, see No. 3.
- If there is no way whose tag matches and the V bit is 1, and the U bit of the way which is selected to replace using the LRU bits is 0, see No. 4.
- If there is no way whose tag matches and the V bit is 1, and the U bit of the way which is selected to replace using the LRU bits is 1, see No. 5.

3. Cache hit

   Then the LRU bits are updated to indicate the hitted way is the latest one.

4. Cache miss (no write-back)

   Data is read into the cache line on the way, which is selected to replace, from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation the CPU doesn't wait the data arrives. While the one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and 0 is written to the U bit on the way. And the LRU bit is updated to indicate the way is latest one.

5. Cache miss (with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then data is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation the CPU doesn't wait the data arrives. While the one cache line of data is being read, the CPU can execute the next processing. And the LRU bits are updated to indicate the way is latest one. The data in the write-back buffer is then written back to external memory.

8.3.3 Write Operation

When the Operand Cache (OC) is enabled (OCE = 1 in CCR) and data is written to a cacheable area, the cache operates as follows:

1. The tag, V bit, U bit, and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and its V bit is 1, see No. 3 for copy-back and No. 4 for write-through.
- I If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 0, see No. 5 for copy-back and No. 7 for write-through.
- If there is no way whose tag matches and its V bit is 1 and the U bit of the way which is selected to replace using the LRU bits is 1, see No. 6 for copy-back and No. 7 for write-through.

3. Cache hit (copy-back)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then 1 is written to the U bit. The LRU bits are updated to indicate the way is the latest one.

4. Cache hit (write-through)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. A write is also performed to external memory corresponding to the virtual address. Then the LRU bits are updated to indicate the way is the latest one. In this case, the U bit isn't updated.

5. Cache miss (copy-back, no write-back)

   A data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then, the data, excluding the cache-missed data which is written already, is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address.

   Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and the U bit on the way. Then the LRU bit is updated to indicate the way is latest one.

6. Cache miss (copy-back, with write-back)

   The tag and data field of the cache line on the way which is selected to replace are saved in the write-back buffer. Then a data write in accordance with the access size is performed for the data field on the hit way which is indexed by virtual address bits [4:0]. Then, the data, excluding the cache-missed data which is written already, is read into the cache line on the way which is selected to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. While the remaining data on the cache line is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, 1 is written to the V bit and the U bit on the way. Then the LRU bit is updated to indicate the way is latest one. Then the data in the write-back buffer is then written back to external memory.

7. Cache miss (write-through)

   A write of the specified access size is performed to the external memory corresponding to the virtual address. In this case, a write to cache is not performed.

8.3.4 Write-Back Buffer

In order to give priority to data reads to the cache and improve performance, this LSI has a write-back buffer which holds the relevant cache entry when it becomes necessary to purge a dirty cache entry into external memory as the result of a cache miss. The write-back buffer contains one cache line of data and the physical address of the purge destination.

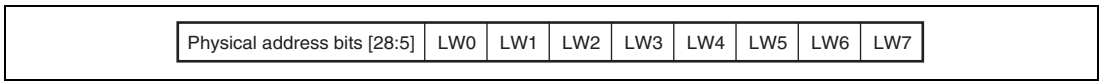


Figure 8.3 Configuration of Write-Back Buffer

8.3.5 Write-Through Buffer

This LSI has a 64-bit buffer for holding write data when writing data in write-through mode or writing to a non-cacheable area. This allows the CPU to proceed to the next operation as soon as the write to the write-through buffer is completed, without waiting for completion of the write to external memory.

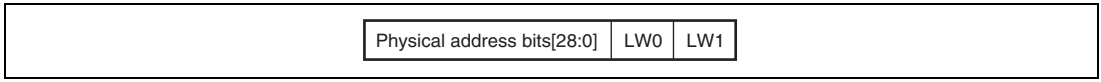


Figure 8.4 Configuration of Write-Through Buffer

8.3.6 OC Two-Way Mode

When the OC2W bit in RAMCR is set to 1, OC two-way mode which only uses way 0 and way 1 in the OC is entered. Thus, power consumption can be reduced. In this mode, only way 0 and way 1 are used even if a memory-mapped OC access is made.

The OC2W bit should be modified by a program in the P2 area. At that time, if the valid line has already been recorded in the OC, data should be written back by software, if necessary, 1 should be written to the OCI bit in CCR, and all entries in the OC should be invalid before modifying the OC2W bit.

8.4 Instruction Cache Operation

8.4.1 Read Operation

When the IC is enabled (ICE = 1 in CCR) and instruction fetches are performed from a cacheable area, the instruction cache operates as follows:

1. The tag, V bit, U bit and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and the V bit is 1, see No. 3.
- If there is no way whose tag matches and the V bit is 1, see No. 4.

3. Cache hit

   The data indexed by virtual address bits [4:2] is read as an instruction from the data field on the hit way. The LRU bits are updated to indicate the way is the latest one.

4. Cache miss

   Data is read into the cache line on the way which selected using LRU bits to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data, and when the corresponding data arrives in the cache, the read data is returned to the CPU as an instruction. While the remaining one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, and 1 is written to the V bit, the LRU bits are updated to indicate the way is the latest one.

8.4.2 Prefetch Operation

When the IC is enabled (ICE = 1 in CCR) and instruction prefetches are performed from a cacheable area, the instruction cache operates as follows:

1. The tag, V bit, Ubit and LRU bits on each way are read from the cache line indexed by virtual address bits [12:5].
2. The tag, read from each way, is compared with bits [28:10] of the physical address resulting from virtual address translation by the MMU:

- If there is a way whose tag matches and the V bit is 1, see No. 3.
- If there is no way whose tag matches and the V bit is 1, see No. 4.

3. Cache hit

   The LRU bits is updated to indicate the way is the latest one.

4. Cache miss

   Data is read into the cache line on a way which selected using the LRU bits to replace from the physical address space corresponding to the virtual address. Data reading is performed, using the wraparound method, in order from the quad-word data (8 bytes) including the cache-missed data. In the prefetch operation, the CPU doesn't wait the data arrived. While the one cache line of data is being read, the CPU can execute the next processing. When reading of one line of data is completed, the tag corresponding to the physical address is recorded in the cache, and 1 is written to the V bit, the LRU bits is updated to indicate the way is the latest one.

8.4.3 IC Two-Way Mode

When the IC2W bit in RAMCR is set to 1, IC two-way mode which only uses way 0 and way 1 in the IC is entered. Thus, power consumption can be reduced. In this mode, only way 0 and way 1 are used even if a memory-mapped IC access is made.

The IC2W bit should be modified by a program in the P2 area. At that time, if the valid line has already been recorded in the IC, 1 should be written to the ICI bit in CCR and all entries in the IC should be invalid before modifying the IC2W bit.

8.4.4 Instruction Cache Way Prediction Operation

This LSI incorporates an instruction cache (IC) way prediction scheme to reduce power consumption. This is achieved by activating only the data array that corresponds to a predicted way. When way prediction misses occur, data must be re-read from the right way, which may lead to lower performance in instruction fetching. Setting the ICWPD bit to 1 disables the IC way prediction scheme. Since way prediction misses do not occur in this mode, there is no loss of performance in instruction fetching but the IC consumes more power. The ICWPD bit should be modified by a program in the non-cacheable P2 area. If a valid line has already been recorded in the IC at this time, invalidate all entries in the IC by writing 1 to the ICI bit in CCR before modifying the ICWPD bit.

8.5 Cache Operation Instruction

8.5.1 Coherency between Cache and External Memory

(1) Cache Operation Instruction

Coherency between cache and external memory should be assured by software. In this LSI, the following six instructions are supported for cache operations. Details of these instructions are given in section 11, Instruction Descriptions of this LSI Extended Functions Software Manual.

- Operand cache invalidate instruction: OCBI @Rn
  Operand cache invalidation (no write-back)
- Operand cache purge instruction: OCBP @Rn
  Operand cache invalidation (with write-back)
- Operand cache write-back instruction: OCBWB @Rn
  Operand cache write-back
- Operand cache allocate instruction: MOVCA.L R0,@Rn
  Operand cache allocation
- Instruction cache invalidate instruction: ICBI @Rn
  Instruction cache invalidation
- Operand access synchronization instruction: SYNCO
  Wait for data transfer completion

(2) Coherency Control

The operand cache can receive "PURGE" and "FLUSH" transaction from SuperHyway bus to control the cache coherency. Since the address used by the PURGE and FLUSH transaction is a physical address, do not use the 1 Kbyte page size to avoid cache synonym problem in MMU enable mode.

- PURGE transaction

  When the operand cache is enabled, the PURGE transaction checks the operand cache and invalidates the hit entry. If the invalidated entry is dirty, the data is written back to the external memory. If the transaction is not hit to the cache, it is no-operation.
- FLUSH transaction

  When the operand cache is enabled, the FLUSH transaction checks the operand cache and if the hit line is dirty, then the data is written back to the external memory. If the transaction is not hit to the cache or the hit entry is not dirty, it is no-operation.

8.5.2 Prefetch Operation

This LSI supports a prefetch instruction to reduce the cache fill penalty incurred as the result of a cache miss. If it is known that a cache miss will result from a read or write operation, it is possible to fill the cache with data beforehand by means of the prefetch instruction to prevent a cache miss due to the read or write operation, and so improve software performance. If a prefetch instruction is executed for data already held in the cache, or if the prefetch address results in a UTLB miss or a protection violation, the result is no operation, and an exception is not generated. Details of the prefetch instruction are given in section 11, Instruction Descriptions of this LSI Extended Functions Software Manual.

- Prefetch instruction (OC) : PREF @Rn
- Prefetch instruction (IC) : PREFI @Rn

8.6 Memory-Mapped Cache Configuration

The IC and OC can be managed by software. The contents of IC data array can be read from or written to by a program in the P2 area by means of a MOV instruction in privileged mode. The contents of IC address array can also be read from or written to in privileged mode by a program in the P2 area or the IL memory area by means of a MOV instruction. Operation is not guaranteed if access is made from a program in another area. In this case, execute one of the following three methods for executing a branch to the P0, U0, P1, or P3 area.

1. Execute a branch using the RTE instruction.
2. Execute a branch to the P0, U0, P1, or P3 area after executing the ICBI instruction for any address (including non-cacheable area).
3. If the MC bit in IRMCR is 0 (initial value) before making an access to the memory-mapped IC, the specific instruction does not need to be executed. However, note that the CPU processing performance will be lowered because the instruction fetch is performed again for the next instruction after making an access to the memory-mapped IC.

Note that the method 3 may not be guaranteed in the future SuperH Series. Therefore, it is recommended that the method 1 or 2 should be used for being compatible with the future SuperH Series.

In privileged mode, the OC contents can be read from or written to by a program in the P1 or P2 area by means of a MOV instruction. Operation is not guaranteed if access is made from a program in another area. The IC and OC are allocated to the P4 area in the virtual address space. Only data accesses can be used on both the IC address array and data array and the OC address array and data array, and accesses are always longword-size. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of 0 should be specified and the read value is undefined.

8.6.1 IC Address Array

The IC address array is allocated to addresses H'F000 0000 to H'F0FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the write tag and V bit are specified in the data field.

In the address field, bits [31:24] have the value H'F0 indicating the IC address array, and the way is specified by bits [14:13] and the entry by bits [12:5]. The association bit (A bit) [3] in the address field specifies whether or not association is performed when writing to the IC address array. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [31:10], and the V bit by bit [0]. As the IC address array tag is 19 bits in length, data field bits [31:29] are not used in the case of a write in which association is not performed. Data field bits [31:29] are used for the virtual address specification only in the case of a write in which association is performed.

The following three kinds of operation can be used on the IC address array:

1. IC address array read

   The tag and V bit are read into the data field from the IC entry corresponding to the way and entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is 1 or 0.
2. IC address array write (non-associative)

   The tag and V bit specified in the data field are written to the IC entry corresponding to the way and entry set in the address field. The A bit in the address field should be cleared to 0.
3. IC address array write (associative)

   When a write is performed with the A bit in the address field set to 1, the tag in each way stored in the entry specified in the address field is compared with the tag specified in the data field. The way numbers of bits [14:13] in the address field are not used. If the MMU is enabled at this time, comparison is performed after the virtual address specified by data field bits [31:10] has been translated to a physical address using the ITLB. If the addresses match and the V bit in the way is 1, the V bit specified in the data field is written into the IC entry. In other cases, no operation is performed. This operation is used to invalidate a specific IC entry. If an ITLB miss occurs during address translation, or the comparison shows a mismatch, an exception is not generated, no operation is performed, and the write is not executed.

Note: IC address array associative writing function may not be supported in the future SuperH Series. Therefore, it is recommended that the ICBI instruction should be used to operate the IC definitely by handling ITLB miss and reporting ITLB miss exception.

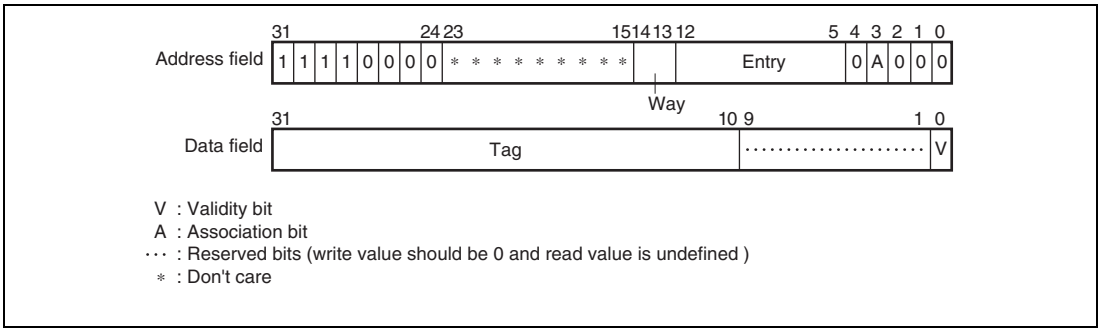


Figure 8.5 Memory-Mapped IC Address Array (Cache size = 32 Kbytes)

8.6.2 IC Data Array

The IC data array is allocated to addresses H'F100 0000 to H'F1FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F1 indicating the IC data array, and the way is specified by bits [14:13] and the entry by bits [12:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, 0 should be specified for address field bits [1:0].

The data field is used for the longword data specification.

The following two kinds of operation can be used on the IC data array:

1. IC data array read

   Longword data is read into the data field from the data specified by the longword specification bits in the address field in the IC entry corresponding to the way and entry set in the address field.
2. IC data array write

   The longword data specified in the data field is written for the data specified by the longword specification bits in the address field in the IC entry corresponding to the way and entry set in the address field.

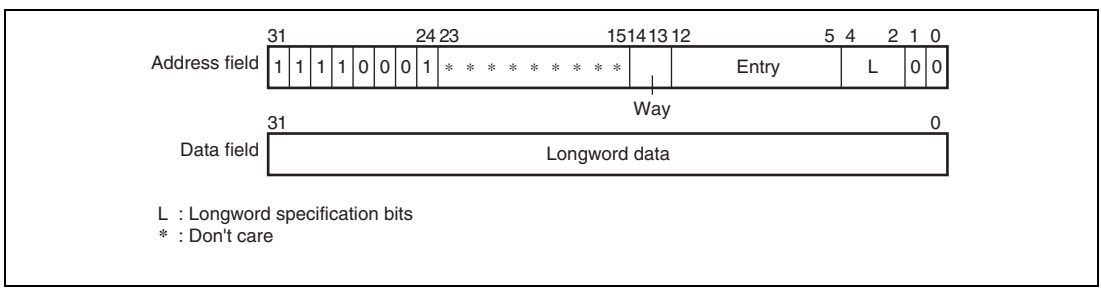


Figure 8.6 Memory-Mapped IC Data Array (Cache size = 32 Kbytes)

8.6.3 OC Address Array

The OC address array is allocated to addresses H'F400 0000 to H'F4FF FFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the write tag, U bit, and V bit are specified in the data field.

In the address field, bits [31:24] have the value H'F4 indicating the OC address array, and the way is specified by bits [14:13] and the entry by bits [12:5]. The association bit (A bit) [3] in the address field specifies whether or not association is performed when writing to the OC address array. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [31:10], the U bit by bit [1], and the V bit by bit [0]. As the OC address array tag is 19 bits in length, data field bits [31:29] are not used in the case of a write in which association is not performed. Data field bits [31:29] are used for the virtual address specification only in the case of a write in which association is performed.

The following three kinds of operation can be used on the OC address array:

1. OC address array read

   The tag, U bit, and V bit are read into the data field from the OC entry corresponding to the way and entry set in the address field. In a read, associative operation is not performed regardless of whether the association bit specified in the address field is 1 or 0.
2. OC address array write (non-associative)

   The tag, U bit, and V bit specified in the data field are written to the OC entry corresponding to the way and entry set in the address field. The A bit in the address field should be cleared to 0.

   When a write is performed to a cache line for which the U bit and V bit are both 1, after write-back of that cache line, the tag, U bit, and V bit specified in the data field are written.

3. OC address array write (associative)

   When a write is performed with the A bit in the address field set to 1, the tag in each way stored in the entry specified in the address field is compared with the tag specified in the data field. The way numbers of bits [14:13] in the address field are not used. If the MMU is enabled at this time, comparison is performed after the virtual address specified by data field bits [31:10] has been translated to a physical address using the UTLB. If the addresses match and the V bit in the way is 1, the U bit and V bit specified in the data field are written into the OC entry. In other cases, no operation is performed. This operation is used to invalidate a specific OC entry. If the OC entry U bit is 1, and 0 is written to the V bit or to the U bit, write-back is performed. If a UTLB miss occurs during address translation, or the comparison shows a mismatch, an exception is not generated, no operation is performed, and the write is not executed.

Note: OC address array associative writing function may not be supported in the future SuperH Series. Therefore, it is recommended that the OCBI, OCBP, or OCBWB instruction should be used to operate the OC definitely by reporting data TLB miss exception.

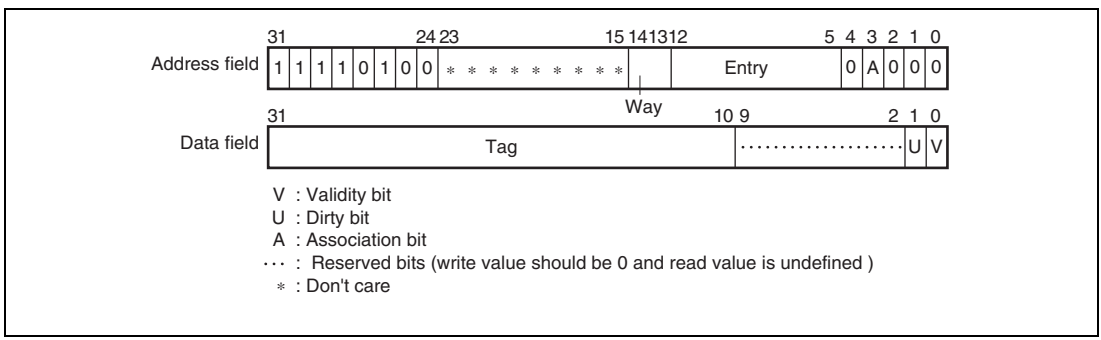


Figure 8.7 Memory-Mapped OC Address Array (Cache size = 32 Kbytes)

8.6.4 OC Data Array

The OC data array is allocated to addresses H'F500 0000 to H'F5FF FFFF in the P4 area. A data array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The way and entry to be accessed are specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F5 indicating the OC data array, and the way is specified by bits [14:13] and the entry by bits [12:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, 0 should be specified for address field bits [1:0].

The data field is used for the longword data specification.

The following two kinds of operation can be used on the OC data array:

1. OC data array read

   Longword data is read into the data field from the data specified by the longword specification bits in the address field in the OC entry corresponding to the way and entry set in the address field.
2. OC data array write

   The longword data specified in the data field is written for the data specified by the longword specification bits in the address field in the OC entry corresponding to the way and entry set in the address field. This write does not set the U bit to 1 on the address array side.

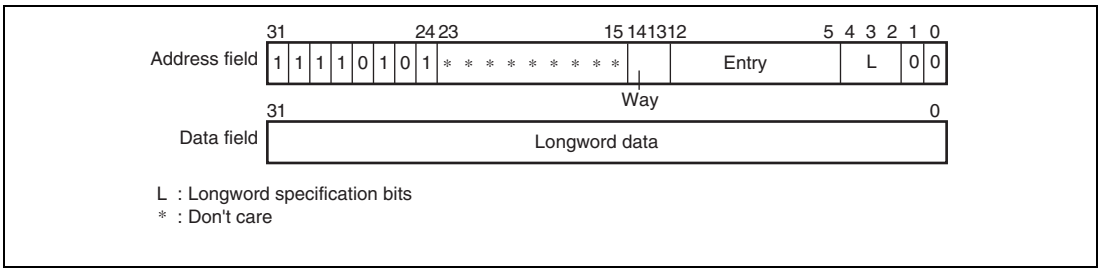


Figure 8.8 Memory-Mapped OC Data Array (Cache size = 32 Kbytes)

8.6.5 Memory-Mapped Cache Associative Write Operation

Associative writing to the IC and OC address arrays may not be supported in future SuperH-family products. The use of instructions ICBI, OCBI, OCBP, and OCBWB is recommended. These instructions handle ITLB misses, and notify instruction TLB miss exceptions and data TLB miss exceptions, thus providing a sure way of controlling the IC and OC. As a transitional measure, this LSI generates address errors when this function is used. If compatibility with previous products is a crucial consideration, on the other hand, the MMCAW bit in EXPMASK (H'FF2F 0004) can be set to 1 to enable this function. However, instructions ICBI, OCBI, OCBP, and OCBWB should be used to guarantee compatibility with future SuperH-family products.

8.7 Store Queues

This LSI supports two 32-byte store queues (SQs) to perform high-speed writes to external memory.

8.7.1 SQ Configuration

There are two 32-byte store queues, SQ0 and SQ1, as shown in figure 8.9. These two store queues can be set independently.

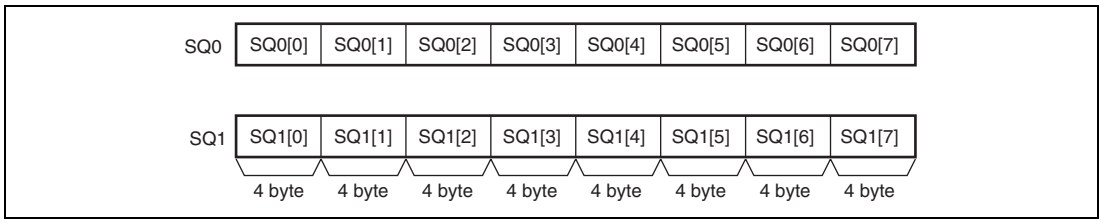


Figure 8.9 Store Queue Configuration

8.7.2 Writing to SQ

A write to the SQs can be performed using a store instruction for addresses H'E000 0000 to H'E3FF FFFC in the P4 area. A longword or quadword access size can be used. The meanings of the address bits are as follows:

[31:26]	: 111000	Store queue specification
[25:6]	: Don't care	Used for external memory transfer/access right
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification
[4:2]	: LW specification	Specifies longword position in SQ0/SQ1
[1:0]	: 00	Fixed at 0

8.7.3 Transfer to External Memory

Transfer from the SQs to external memory can be performed with a prefetch instruction (PREF). Issuing a PREF instruction for addresses H'E000 0000 to H'E3FF FFFC in the P4 area starts a transfer from the SQs to external memory. The transfer length is fixed at 32 bytes, and the start address is always at a 32-byte boundary. While the contents of one SQ are being transferred to external memory, the other SQ can be written to without a penalty cycle. However, writing to the SQ involved in the transfer to external memory is kept waiting until the transfer is completed.

The physical address bits [28:0] of the SQ transfer destination are specified as shown below, according to whether the MMU is enabled or disabled.

- When MMU is enabled (AT = 1 in MMUCR)

  The SQ area (H'E000 0000 to H'E3FF FFFF) is set in VPN of the UTLB, and the transfer destination physical address in PPN. The ASID, V, SZ, SH, PR, and D bits have the same meaning as for normal address translation, but the C and WT bits have no meaning with regard to this page. When a prefetch instruction is issued for the SQ area, address translation is performed and physical address bits [28:10] are generated in accordance with the SZ bit specification. For physical address bits [9:5], the address prior to address translation is generated in the same way as when the MMU is disabled. Physical address bits [4:0] are fixed at 0. Transfer from the SQs to external memory is performed to this address.
- When MMU is disabled (AT = 0 in MMUCR)

  The SQ area (H'E000 0000 to H'E3FF FFFF) is specified as the address at which a PREF instruction is issued. The meanings of address bits [31:0] are as follows:

[31:26]	: 111000	Store queue specification
[25:6]	: Address	Transfer destination physical address bits [25:6]
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification and transfer destination physical address bit [5]
[4:2]	: Don't care	No meaning in a prefetch
[1:0]	: 00	Fixed at 0

  Physical address bits [28:26], which cannot be generated from the above address, are generated from QACR0 and QACR1.

  QACR0[4:2] : Physical address bits [28:26] corresponding to SQ0
  QACR1[4:2] : Physical address bits [28:26] corresponding to SQ1

  Physical address bits [4:0] are always fixed at 0 since burst transfer starts at a 32-byte boundary.

8.7.4 Determination of SQ Access Exception

Determination of an exception in a write to an SQ or transfer to external memory (PREF instruction) is performed as follows according to whether the MMU is enabled or disabled. If an exception occurs during a write to an SQ, the SQ contents before the write are retained. If an exception occurs in a data transfer from an SQ to external memory, the transfer to external memory will be aborted.

- When MMU is enabled (AT = 1 in MMUCR)

  Operation is in accordance with the address translation information recorded in the UTLB, and the SQMD bit in MMUCR. Write type exception judgment is performed for writes to the SQs, and read type exception judgment for transfer from the SQs to external memory (using a PREF instruction). As a result, a TLB miss exception or protection violation exception is generated as required. However, if SQ access is enabled in privileged mode only by the SQMD bit in MMUCR, an address error will occur even if address translation is successful in user mode.

- When MMU is disabled (AT = 0 in MMUCR)

  Operation is in accordance with the SQMD bit in MMUCR.

  0: Privileged/user mode access possible

  1: Privileged mode access possible

  If the SQ area is accessed in user mode when the SQMD bit in MMUCR is set to 1, an address error will occur.

8.7.5 Reading from SQ

In privileged mode in this LSI, reading the contents of the SQs may be performed by means of a load instruction for addresses H'FF00 1000 to H'FF00 103C in the P4 area. Only longword access is possible.

[31:6]	: H'FF00 1000	Store queue specification
[5]	: 0/1	0: SQ0 specification 1: SQ1 specification
[4:2]	: LW specification	Specifies longword position in SQ0/SQ1
[1:0]	: 00	Fixed at 0

8.8 Notes on Using 32-Bit Address Extended Mode

In 32-bit address extended mode, the items described in this section are extended as follows.

1. The tag bits [28:10] (19 bits) in the IC and OC are extended to bits [31:10] (22 bits).
2. An instruction which operates the IC (a memory-mapped IC access and writing to the ICI bit in CCR) should be located in the P1 or P2 area. The cacheable bit (C bit) in the corresponding entry in the PMB should be 0.
3. Bits [4:2] (3 bits) for the AREA0 bit in QACR0 and the AREA1 bit in QACR1 are extended to bits [7:2] (6 bits).

Section 9 Secondary Cache

This LSI includes an on-chip 256-Kbyte secondary cache with an instruction/data unified structure.

9.1 Features

Table 9.1 shows the features of the secondary cache.

Table 9.1 Secondary Cache Features

Features	Secondary Cache
Capacity	256 Kbytes
Structure	4-way set associative, physical address index/physical address tag
Line size	32 bytes
Number of entries	2048 entries/way
Write system	Write through
Replacement method	Least-recently-used (LRU) algorithm
Operating clock	(CPU operating clock)/n (n = 1, 2, 3, 4, 6, or 8)

9.2 Register Descriptions

Table 9.2 shows the configuration of the registers related to the secondary cache control.

Table 9.2 Register Configuration

Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Size
On-chip memory control register	RAMCR	R/W	H'FF000074	H'1F000074	32

Note: * The P4 address is the address used when using P4 area in the virtual address space. The area 7 address is the address used when accessing from area 7 in the physical address space using the TLB.

Table 9.3 Register Status in Each Processing State

Name	Abbreviation	Power-On Reset	Manual Reset	Sleep	Standby
On-chip memory control register	RAMCR	H'00000000	H'00000000	Retained	Retained

9.2.1 On-Chip Memory Control Register (RAMCR)

RAMCR controls the secondary cache.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPD	—	—	—	L2FC	L2E
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
9	RMD	0	R/W	On-Chip Memory Access Mode For further details, refer to section 10.4, IL Memory Protective Functions.
8	RP	0	R/W	On-Chip Memory Protection Enable For further details, refer to section 10.4, IL Memory Protective Functions.
7	IC2W	0	R/W	IC Two-Way Mode For further details, refer to section 8.4.3, IC Two-Way Mode.
6	OC2W	0	R/W	OC Two-Way Mode For further details, refer to section 8.3.6, OC Two-Way Mode.
5	ICWPD	0	RW	IC Way Prediction Stop For further details, refer to section 8.4.4, Instruction Cache Way Prediction Operation.

Bit	Bit Name	Initial Value	R/W	Description
4 to 2	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
1	L2FC	0	R/W	Secondary Cache Forcible Coherency Mode Controls whether to perform a data write in the memory-mapped secondary cache address array when a (non-associative) data write in the memory-mapped OC address array is performed. 0: Data is not written in the secondary cache address array. 1: Data is written in the secondary cache address array.
0	L2E	0	R/W	Secondary Cache Enable 0: The secondary cache is not used. 1: The secondary cache is used.

9.3 Secondary Cache Configuration and Operation

9.3.1 Configuration

Figure 9.1 shows the configuration of the secondary cache.

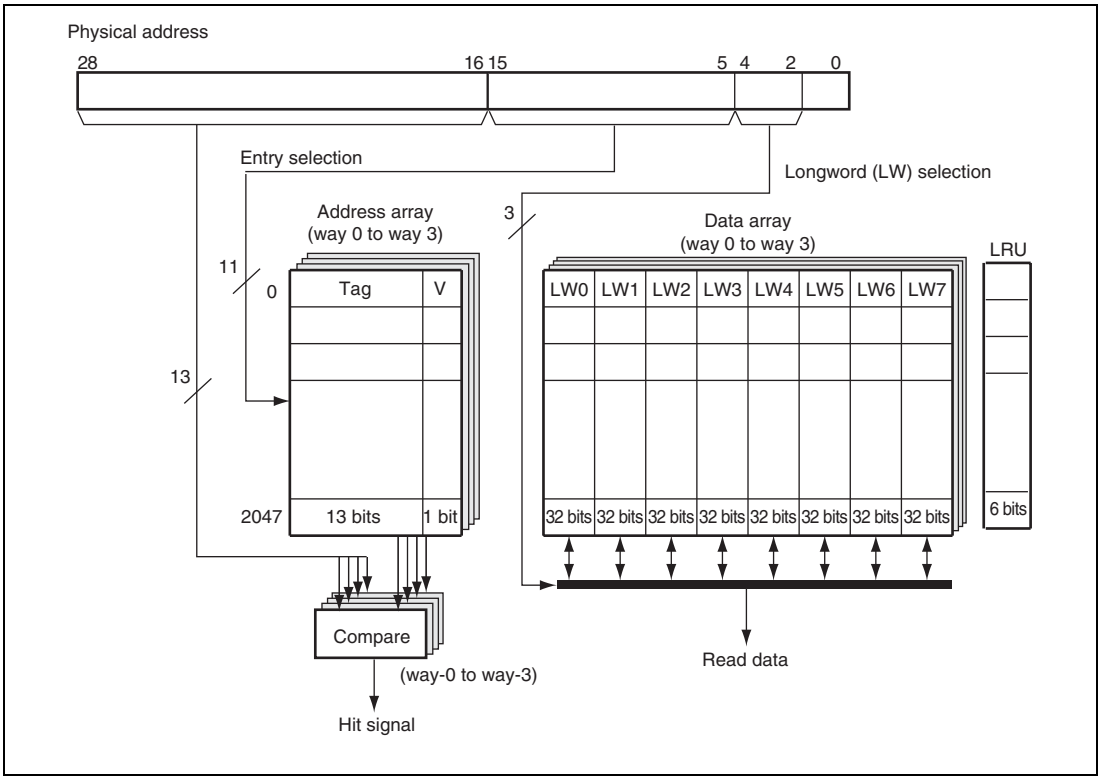


Figure 9.1 Secondary Cache Configuration

The secondary cache consists of 2048 cache lines. Each cache line is composed of a 13-bit tag, V bit, 6-bit LRU bit, and 32-byte data.

Tag: Stores the upper 13 bits of the 29-bit physical address of the data line to be cached. The tag is not initialized by a power-on or manual reset.

V Bit (Validity Bit): Indicates that valid data is stored in the cache line. When this bit is 1, the cache line data is valid. The V bit is initialized to 0 by a power-on reset, but retains its value in a manual reset.

LRU Bit: In a 4-way set-associative system, up to four kinds of data that have the same entry address can be registered in cache. The LRU bit indicates in which of the four ways the entry is to be registered. There are six LRU bits in each entry, and they are controlled by hardware. The least recently used (LRU) algorithm, which selects the least recently accessed way, is used for way selection. The LRU bit is initialized to 0 by a power-on reset, but retains its value in a manual reset.

Data Field: The data field holds 32 bytes (256 bits) of data per cache line. The data field is not initialized by a power-on reset or manual reset.

9.3.2 Operation of Secondary Cache

In this LSI, the instruction cache (IC) and operand cache (OC), which comprise the primary cache, must be enabled in order to use the secondary cache. The V and LRU bits of the secondary cache are hardware initialized either by a power-on reset, by an IC invalidate through writing in CCR.ICI = 1, or by an OC invalidate through writing of CCR.OCI = 1. The number of secondary cache operating clock cycles required by this operation equals the number of entries. If memory access by the CPU misses the primary cache during this operation, then the secondary cache is not used, and the external memory is accessed instead.

Methods for accessing the secondary cache are:

- Operation via the primary cache through instruction fetch and operand access by the CPU:
  See 1. below.
- Operation through cache manipulation instruction by the CPU:
  See 2. below.
- Coherency access by the SuperHyway
  See 3. below.
- Access through memory-mapped primary cache (IC or OC) address array writing
  See 4. below.
- Access through memory-mapped secondary cache access (refer to section 9.4, Configuration of Memory-Mapped Secondary Cache.)

1. Operation via the primary cache through the instruction fetch and the operand access by the CPU

   Operations via the primary cache through the instruction fetch and the operand access by the CPU are:

- Read from the Primary Cache
- Write-Through-Write from the Primary Cache
- Copy-Back-Write from the Primary Cache

A. Read from the Primary Cache

This is a block read operation when a cache miss occurs in the primary cache, and single-line (32-byte) data transfer is always performed. The tag and the V bit are read out from the secondary cache line of each way indexed by physical address bits [15:5], which is the primary cache miss address, transferred from the primary cache.

The tag is compared with the physical address bits [28:16]:

— If the tag matches and V bit is 1 See a. below.

— Other than the above See b. below.

a. Secondary cache hit

The 32-byte data indexed by physical address bits [15:5] is read out from the data field of the secondary cache line of the hit way, and the data is filled into the primary cache.

b. Secondary cache miss

Data is read into the secondary cache line from the external memory space corresponding to the physical address. Data reading is performed, using the wraparound method, in order from the data corresponding to the physical address where the cache miss occurs, and when the first data arrives in the secondary cache, the read data is returned to the primary cache. When writing of one line data is completed, the secondary cache registers the tag corresponding to the physical address, and writes 1 to the V bit.

B. Write-Through-Write from the Primary Cache

A single write operation (quadword/longword/word/byte) when write-through-write occurs in the primary cache.

The tag and the V bit are read from the cache line of each way indexed by the physical address bits [15:5] transferred from the primary cache.

The tag is compared with the physical address bits [28:16]:

— If the tag matches and the V bit is 1 See a. below.

— Other than the above See b. below.

a. Secondary cache hit

A data write in accordance with the access size (quadword/longword/word/byte) is performed for the data indexed by the physical address bits [15:5], and indexed by the physical address bits [4:0] in the data field in the cache line of the hit way. The data write is also performed to the external memory in accordance with the specified access size.

b. Secondary cache miss

A data write in accordance with the specified access size is performed to the external memory corresponding to the physical address. In this case, the data write to secondary cache is not performed.

C. Copy-Back-Write from the Primary Cache

This is a block write operation when copy-back-write occurs in the primary cache, and a single-line (32-byte) data transfer is always performed. The tag and the V bit are read from the cache line of each way indexed by the physical address bits [15:5] which are transferred from the primary cache.

The tag is compared with bits [28:16] of the physical address:

— If the tag matches and the V bit is 1 See a. below.

— Other than the above See b. below.

a. Secondary cache hit

32-byte data is written to the data of way where the cache line is hit, indexed by physical address bits [15:5]. A 32-byte data is also written to the external memory.

b. Secondary cache miss

A 32-byte data is written to the external memory corresponding to the physical address. In this case, data write to the secondary cache is not performed.

2. Operation through the cache manipulation instruction by the CPU

Operations through the cache manipulation instruction by the CPU include:

- Through ICBI @Rn, OCBI @Rn, or OCBP @Rn instruction
- Through OCBWB @Rn instruction

A. ICBI @Rn, OCBI @Rn, and OCBP @Rn Instructions

Invalidate the secondary cache line including the physical address corresponding to the virtual address indicated by Rn. The secondary cache is write-through cache, therefore it is not dirty, and a copy-back-write does not occur.

B. OCBWB @Rn Instruction

The secondary cache is write-through cache, therefore it is not dirty, and no operation is performed.

3. Coherency access by SuperHyway bus

   Coherency accesses by the SuperHyway bus include:

- When the PURGE transaction is received
- When the FLUSH transaction is received

   A. When the PURGE Transaction is Received

      The secondary cache line including indicated physical address is invalidated.

   B. When the FLUSH Transaction is Received

      The secondary cache is write-through mode, therefore it is not dirty, and no operation is performed.

4. Memory-mapped primary cache (IC/OC) access

   Memory-mapped primary cache (IC/OC) access includes:

- Memory-mapped OC address array write (non associative)
- Memory-mapped IC address array write (associative)
- Memory-mapped OC address array write (associative)

   A. Memory-Mapped OC Address Array (Non Associative)

      When the secondary cache is enabled (RAMCR.L2E = 1) and the L2FC bit in the RAMCR register is 1, the tag and the V bit specified by the data field are written into the secondary cache entry specified by the address field bits [15:5] in addition to the operation into the OC address array.

      When the L2FC bit of the RAMCR register is 0, the secondary cache is not accessed.

   B. Memory-Mapped IC Address Array Write (Associative)

      When 0 is written to the V bit while the secondary cache is enabled (RAMCR.L2E = 1), the secondary cache line including the physical address corresponding to the virtual address specified by the data field is invalidated in addition to the operation into the IC address array. When 1 is written to the V bit, no operation is performed.

   C. Memory-Mapped OC Address Array Write (Associative)

      When 0 is written to the V bit during the secondary cache is enabled (RAMCR.L2E = 1), the secondary cache line including physical address corresponding to the virtual address specified by the data field is invalidated in addition to the operation into the OC address array. When 1 is written to the V bit, no operation is performed.

9.4 Configuration of Memory-Mapped Secondary Cache

To enable the secondary cache to be maneged by software, the contents of the secondary cache can be read/written through the MOV instruction in privileged mode. The secondary cache is allocated in P4 area in the virtual address space. Only data accesses can be used on both the address array and data array of the secondary cache, and the access size is always longword. Instruction fetches cannot be performed in these areas. For reserved bits, a write value of 0 should be specified, and read values are undefined.

9.4.1 Secondary Cache Address Array

The secondary cache address array is allocated to addresses H'F8000000 to H'F8FFFFFF in the P4 area. An address array access requires a 32-bit address field specification (when reading or writing) and a 32-bit data field specification. The entry to be accessed is specified in the address field, and the tag, V bit, and LRU bits for writing are specified in the data filed.

In the address fields, bits [31:24] have the value H'F8 indicating the secondary cache address array, and the way is specified by bits [17:16], and the entry by bits [15:5]. As only longword access is used, 0 should be specified for address field bits [1:0].

In the data field, the tag is indicated by bits [29:16], the LRU bits by bits [9:4], and the V bit by bit [0].

The following two kinds of operation can be used on the secondary cache address array:

- Secondary cache address array read

  The tag, LRU bits, and V bit are read into the data field from the secondary cache entry corresponding to the entry which is set in the address field.
- Secondary cache address array write

  The tag, LRU bits, and V bit specified in the data filed are written in the secondary cache entry corresponding to the entry which is set in the address field.

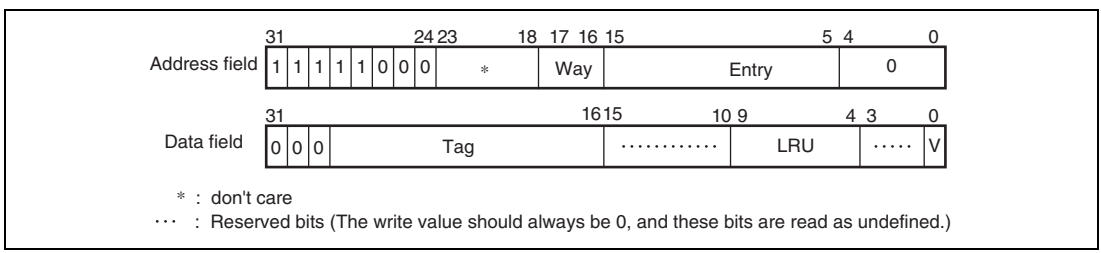


Figure 9.2 Memory-Mapped Secondary Cache Address Array

9.4.2 Secondary Cache Data Array

The secondary cache data array is allocated to the addresses H'F9000000 to H'F9FFFFFF in area P4. Data array access requires a 32-bit address field specification (when reading/writing) and a 32-bit data field specification. The entry to be accessed is specified in the address field, and the longword data to be written is specified in the data field.

In the address field, bits [31:24] have the value H'F9 indicating the secondary cache data array, and the way is specified by bits [17:16], and the entry by bits [15:5]. Address field bits [4:2] are used for the longword data specification in the entry. As only longword access is used, 0 should be specified for address field bits [1:0].

The data field is used for longword data specification.

The following two kinds of operation can be used on the secondary cache data array:

- Secondary cache data array read

  Longword data is read into the data field from the data in the secondary cache corresponding to the entry and longword specification bits which is set in the address field.
- Secondary cache data array write

  The longword data specified in the data field is written to the data in the secondary cache corresponding to the entry and longword specification bits which is set in the address field.

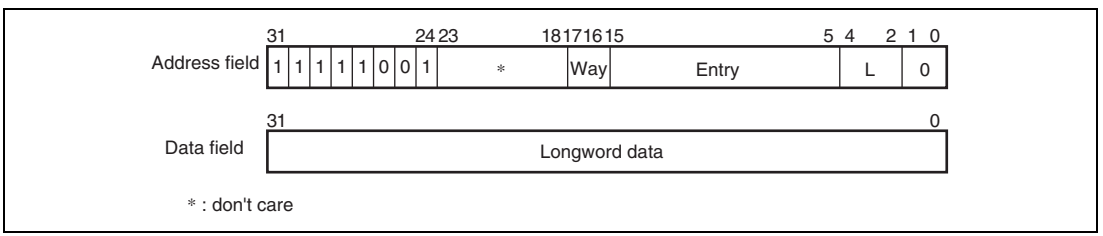


Figure 9.3 Memory-Mapped Secondary Cache Data Array

9.5 Usage Notes

9.5.1 Coherency Control

To enable the secondary cache in the operating system, which does not control the secondary cache directly, the L2E bit and L2FC bit in the RAMCR register should be set to 1. In addition, the number of entries to be given to a routine purging all OC entries should be set as the number of entries of the secondary cache. Accordingly, all entries of OC and the secondary cache are purged (invalidated) through the operation described in (3) of section 9.3.2, Operation of Secondary Cache.

9.5.2 32-Bit Address Extended Mode

In the 32-bit address extended mode, the tags of the secondary cache are also extended to bits [31:16].

Section 10 On-chip Memory

This LSI incorporates a 16-Kbyte IL memory, which is suitable for instruction storage.

10.1 Features

(1) IL Memory

- Capacity16 Kbytes
- Page

  The IL memory is divided into four pages (pages 0, 1, 2, and 3).
- Memory map

  The IL memory is allocated to the addresses shown in Table 10.1 in both the virtual address space and the physical address space.

Table 10.1 IL Memory Addresses

	Memory Size
Page	**16 Kbyte**
Page 0	H'E520 0000 to H'E520 0FFF
Page 1	H'E520 1000 to H'E520 1FFF
Page 2	H'E520 2000 to H'E520 2FFF
Page 3	H'E520 3000 to H'E520 3FFF

- Ports

  The page has three independent read/write ports and is connected to the SuperHyway bus, the cache/RAM internal bus, and the instruction bus. The instruction bus is used when the IL memory is accessed through instruction fetch. The cache/RAM internal bus is used when the IL memory is accessed through operand access. The SuperHyway bus is used for IL memory access from the SuperHyway bus master module.
- Priority

  In the event of simultaneous accesses to the same page from different buses, the access requests are processed according to priority. The priority order is: SuperHyway bus > cache/RAM internal bus > instruction bus.

10.2 Register Descriptions

The following registers are related to the IL memory.

Table 10.2 Register Configuration

Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Access Size
On-chip memory control register	RAMCR	R/W	H'FF00 0074	H'1F00 0074	32

Note: * The P4 address is the address used when using P4 area in the virtual address space. The area 7 address is the address used when accessing from area 7 in the physical address space using the TLB.

Table 10.3 Register States in Each Processing Mode

Name	Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
On-chip memory control register	RAMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained

10.2.1 On-Chip Memory Control Register (RAMCR)

RAMCR controls the protective functions in IL memory.

When updating RAMCR, please follow limitation described at section 8.2.4 and 9.2.1, On-Chip Memory Control Register (RAMCR).

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RMD	RP	IC2W	OC2W	ICWPD	—	—	—	L2FC	L2E
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
9	RMD	0	R/W	IL Memory Access Mode Specifies the right of access to the on-chip memory from the virtual address space. 0: An access in privileged mode is allowed. (An address error exception occurs in user mode.) 1: An access in user/ privileged mode is allowed.
8	RP	0	R/W	IL Memory Protection Enable Selects whether or not to use the protective functions using ITLB and UTLB for accessing the IL memory from the virtual address space. 0: Protective functions are not used. 1: Protective functions are used. For further details, refer to section 10.4, IL Memory Protective Functions.
7	IC2W	0	R/W	IC Two-Way Mode For further details, refer to section 8.4.3, IC Two-Way Mode.
6	OC2W	0	R/W	OC Two-Way Mode For further details, refer to section 8.3.6, OC Two-Way Mode.

Bit	Bit Name	Initial Value	R/W	Description
5	ICWPD	0	R/W	IC Way Prediction Disable For further details, refer to section 8.4.4, Instruction Cache Way Prediction Operation.
4 to 2	—	All 0	R	Reserved For read/write in these bits, refer to General Precautions on Handling of Product.
1	L2FC	0	R/W	Secondary Cache Forcible Coherency Mode For further details, refer to section 9, Secondary cache.
0	L2E	0	R/W	Secondary Cache Enable For further details, refer to section 9, Secondary cache.

10.3 Operation

10.3.1 Instruction Fetch Access from the CPU

Instruction fetch access from the CPU is performed directly via the instruction bus for a given virtual address. In the case of successive accesses to the same page of IL memory and as long as no page conflict occurs, the access takes one cycle.

10.3.2 Operand Access from the CPU and Access from the FPU

Operand access from the CPU and access from the FPU are performed via the cache/RAM internal bus. Access via the cache/RAM internal bus takes more than one cycle.

10.3.3 Access from the SuperHyway Bus Master Module

IL memory is always accessed by the SuperHyway bus master module, such as DMAC, via the SuperHyway bus which is a physical address bus. The same addresses as for the virtual addresses must be used.

10.4 IL Memory Protective Functions

This LSI implements the following protective functions to the IL memory by using the IL memory access mode bit (RMD) and the IL memory protection enable bit (RP) in the on-chip memory control register (RAMCR).

- Protective functions for access from the CPU and FPU

  When RAMCR.RMD = 0, and the IL memory is accessed in user mode, it is determined to be an address error exception.

  When MMUCR.AT = 1 and RAMCR.RP = 1, MMU exception and address error exception are checked in the IL memory area which is a part of area P4 as with the area P0/P3/U0.

The above descriptions are summarized in Table 10.4.

Table 10.4 Protective Function Exceptions to Access IL Memory

MMUCR.AT	RAMCR.RP	SR.MD	RAMCR. RMD	Always Occurring Exceptions	Possibly Occurring Exceptions
0	x	0	0	Address error exception	—
			1	—	—
		1	x	—	—
1	0	0	0	Address error exception	—
			1	—	—
		1	x	—	—
	1	0	0	Address error exception	—
			1	—	MMU exception
		1	x	—	MMU exception

[Legend] x: Don't care

10.5 Usage Notes

10.5.1 Page Conflict

In the event of simultaneous access to the same page from different buses, page conflict occurs. Although each access is completed correctly, this kind of conflict tends to lower memory accessibility. Therefore it is advisable to provide all possible preventative software measures. For example, conflicts will not occur if each bus accesses different pages.

10.5.2 Access Across Different Pages

Access from the instruction bus is performed in one cycle when the access is made successively to the same page but takes multiple cycles (a maximum of two wait cycles may be required) when the access is made across pages or the previous access was made to memory other than IL memory. For this reason, from the viewpoint of performance optimization, it is recommended to design the software such that the target page does not change so often in access from the instruction bus. For example, allocating a separate program for each page will deliver better efficiency.

10.5.3 IL Memory Coherency

In order to allocate instructions in the IL memory, write an instruction to the IL memory, execute the following sequence, then branch to the rewritten instruction.

- SYNCO
- ICBI @Rn

In this case, the target for the ICBI instruction can be any address (IL memory address may be possible) within the range where no address error exception occurs, and cache hit/miss is possible.

10.5.4 Sleep Mode

The SuperHyway bus master module, such as DMAC, cannot access IL memory in sleep mode.

Section 11 RS Memory

This LSI includes an on-chip RS memory module that stores instructions or data.

11.1 Features

- Capacity
  This LSI includes 2 Kbytes of the RS memory.
- Memory map
  The RS memory is allocated to the range of the P4 area from H'FD80 0000 to H'FD80 07FF.
- Port
  The RS memory is connected to the SuperHyway bus.
- Backup area
  The RS memory is allocated to the backup area and can be used to store the program for use on returning from R-standby.

11.2 Operation

11.2.1 Access from the CPU

Virtual addresses should always be used in access to this memory by the CPU.

11.2.2 Access from the DMAC

Virtual addresses should always be used in access to this memory by the DMAC.

Section 12 SuperHyway Packet Router

12.1 SuperHyway Bus Connection

Figure 12.1 shows the initiators and targets that are connected via the SuperHyway packet router (GPR) on the SuperHyway bus.

The following numbers of modules are connected.

- Initiators: 10
- Targets: 12 (including the target incorporated in the GPR)

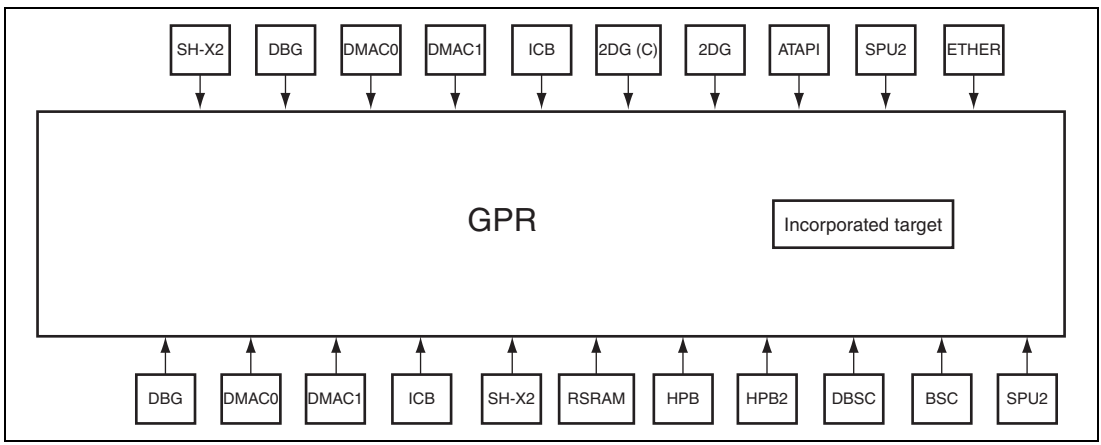


Figure 12.1 Connection of SuperHyway Packet Router (GPR)

12.2 SuperHyway (GPR) Arbitration

Table 12.1 lists the arbitration schemes employed by the SuperHyway packet router (GPR).

Table 12.1 Arbitration Schemes

Resource	Resource #	Target	Arbitration Scheme
Request	0	DBSC target	Priority determination by full LRU plus priority value
	1	Other targets	Priority determination by full LRU plus priority value
Response	0	SH-X2 initiator	Fixed priority
	1	Other initiators	Fixed priority

Note: LRU (Least recently used)

12.2.1 Arbitration of Request Resources

(1) Arbitration Schemes

The order of priority in which requests from the initiators are accepted can be set according to the priority value of the request. When there are simultaneous requests from multiple initiators within the same resource, the request that has the highest priority value is accepted first. The full LRU scheme is used for the requests having the same priority value. In the full LRU scheme, highest priority is given to the initiator whose request has been least recently accepted.

(2) Priority Transfers based on Priority Value

Any of the following priority values can be selected by the PRI control registers incorporated in the GPR.

(a) Fixed value (H'0)
(b) Priority values output by each initiator (default)
(c) Priority values set in the PRI control registers (H'0 to H'F)

Priority values can be output only by the initiators SH-X2, DBG, ICB, and 2DG. Table 12.2 shows the priority values output by the initiators and the output conditions of the initiators.

Table 12.2 Priority Values Output by Initiators

Initiator	Priority Values Output by Initiators*	Output Conditions
ICB	H'F	—
SH-X2	H'8 or H'C	SR.BL = 0: H'8 SR.BL = 1: H'C
2DG (commands)	H'8 to H'9	Transferring the start packet: H'8
2DG (other than commands)		Transferring the packet other than the start packet: H'9
DBG	H'8	—
DMAC0	—	—
DMAC1	—	—
SPU	—	—
ETHER	—	—
ATAPI	—	—

Note: * For the initiators not outputting the priority value, the value is fixed to H'0.

(3) Lock Transfers

Some of the initiators can make a lock transfer request. Requests from other initiators can be blocked when a lock transfer is being executed.

In this LSI, lock transfer requests can be made only by the initiators SH-X2 (memory access in TAS command), DMAC0 and DMAC1. Whether or not the GPR can receive the lock request output by the initiator is set by the LCK control register.

- Behavior in Lock Transfers

Transfer permission is granted to the initiator identified as highest in priority determination, and when a lock transfer request has been made for the transferred request packet, requests from all the initiators except the initiator with the transfer permission are masked. Resetting forcibly clears the lock condition.

- Behavior in Non-lock Transfers

When no lock transfer request is made for a transferred request packet, the requests that were masked in lock transfer are unmasked.

12.2.2 Arbitration of Response Resources

In arbitration of response resources, the fixed priority scheme below is used irrespective of resources.

DBG > DMAC0 > DMAC1 > ICB > SH-X > RSRAM > HPB > HPB2 > DSBSC > BSC > SPU > Incorporated target

12.3 Register Descriptions

Table 12.3 shows the register configuration of the SuperHyway packet rouger (GPR). Table 12.4 shows the register states in each processing mode.

Table 12.3 List of PRI Control Registers

Name	Target Initiator	Abbreviation	Address	R/W	Access Size	Initial Value
PRI control register 0	SH-X2	PRPRICR0	H'FF80 0028	R/W	32 bits	H'0000 0018
PRI control register 1	DBG	PRPRICR1	H'FF80 0030	R/W	32 bits	H'0000 0018
PRI control register 2	DMAC0	PRPRICR2	H'FF80 0038	R/W	32 bits	H'0000 0018
PRI control register 3	DMAC1	PRPRICR3	H'FF80 0040	R/W	32 bits	H'0000 0018
PRI control register 4	ICB	PRPRICR4	H'FF80 0048	R/W	32 bits	H'0000 0018
PRI control register 5	Reserved	PRPRICR5	H'FF80 0050	R/W	32 bits	H'0000 0018
PRI control register 6	2DG (command)	PRPRICR6	H'FF80 0058	R/W	32 bits	H'0000 0018
PRI control register 7	2DG	PRPRICR7	H'FF80 0060	R/W	32 bits	H'0000 0018
PRI control register 8	ATAPI	PRPRICR8	H'FF80 0068	R/W	32 bits	H'0000 0018
PRI control register 9	SPU2	PRPRICR9	H'FF80 0070	R/W	32 bits	H'0000 0018
PRI control register 10	ETHER	PRPRICR10	H'FF80 0078	R/W	32 bits	H'0000 0018
LCK control register	—	PRLCKCR	H'FF80 0018	R/W	32 bits	H'0000 07FF

Table 12.4 lists the PRI control registers and table 12.5 shows the details of the PRI control registers.

Table 12.4 Register State in each Processing Mode

Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
PRPRICRn	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PRLCKCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Note: n = 0 to 10

12.3.1 PRI Control Registers (PRPRICR0 to PRPRICR10)

PRPRICR0 to PRPRICR10 are 32-bit registers that select the priority value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	PRIEN		PRI			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved
5, 4	PRIEN[1:0]	01	R/W	Select the priority value to be used for emergency transfers. 00: Normal transfer (B’0000) 01: Operating with the priority values output by each initiator (default). (Refer to table 12.2.) 10: Operating with the priority values set by PRI[3:0] 11: Setting prohibited
3 to 0	PRI[3:0]	1000	R/W	Priority value Set by Register Select the priority value to be used when PRIEN[1:0] = B'10.

12.3.2 LCK Control Register (PRLCKCR)

PRLCKCR is a 32-bit register that enables or disables the lock transfer request from each initiator.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	LCKC										
Initial value:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	LCKC[10:0]	7FF	R/W	LCKC[10:4]: Setting disabled LCKC[3]: The lock transfer request from the initiator DMAC1 is enabled (= 1)/disabled. (= 0) LCKC[2]: The lock transfer request from the initiator DMAC0 is enabled (= 1)/disabled. (= 0) LCKC[1]: Setting prohibited LCKC[0]: The lock transfer request from the initiator SH-X2 is enabled (= 1)/disabled. (= 0)

Section 13 Interrupt Controller (INTC)

The interrupt controller (INTC) determines the priority of interrupt sources and controls interrupt requests to the CPU. Some INTC registers set the priority of each interrupt and interrupt requests are processed according to the user-set priority.

13.1 Features

The INTC has the following features.

- Fifteen levels of interrupt priority can be set

  By setting the interrupt priority registers, the priorities of on-chip peripheral module interrupts can be selected from 15 levels for individual request sources.
- NMI noise canceler function

  An NMI input-level bit indicates the NMI pin state. By reading this bit in the interrupt exception handling routine, the pin state can be checked, enabling it to be used as a noise canceler.
- NMI request masking when the block bit (BL) in the status register (SR) is set to 1

  Whether to mask NMI requests when the BL bit in SR is set to 1 can be selected.
- User-mode interrupt disabling function

  Specifying an interrupt mask level in the user interrupt mask level register (USERIMASK) disables interrupts which are not higher in priority than the specified mask level in user mode.

Figure 13.1 shows a block diagram of the INTC.

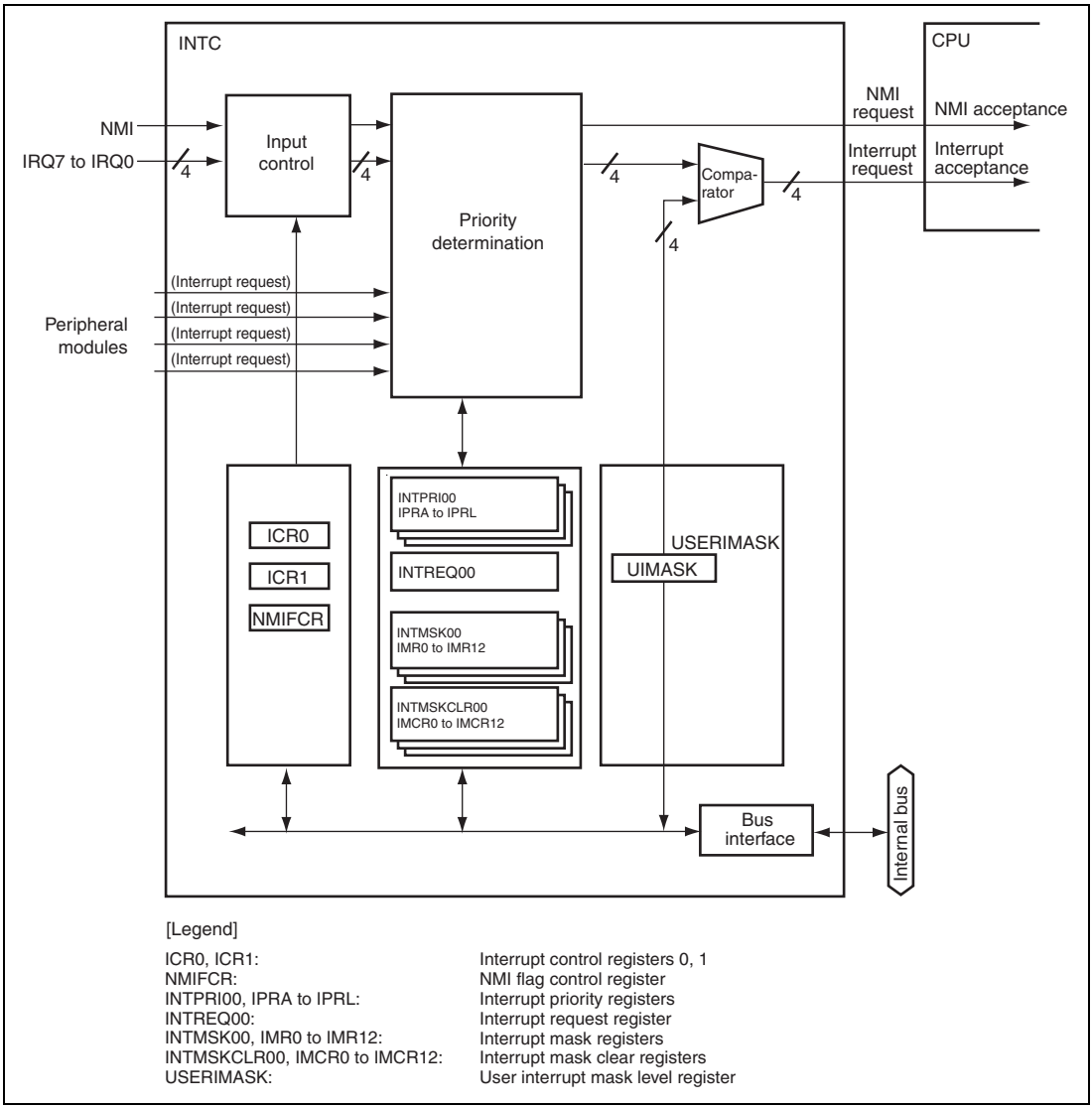


Figure 13.1 Block Diagram of INTC

13.2 Input/Output Pins

Table 13.1 shows the INTC pin configuration.

Table 13.1 Pin Configuration

Pin Name	Function	I/O	Description
NMI	Nonmaskable interrupt input pin	Input	Input of interrupt request signal that is not maskable
IRQ7 to IRQ0	IRQ7 to IRQ0 interrupt input pins	Input	Input of IRQ7 to IRQ0 interrupt request signals (maskable by the IMASK bit setting in SR)

13.3 Register Descriptions

Table 13.2 shows the INTC register configuration. Table 13.3 shows the register states in each operating mode.

Table 13.2 Register Configuration

Register	Abbreviation	R/W	Address	Access Size
Interrupt control register 0	ICR0	R/W	H'A414 0000	16
Interrupt control register 1	ICR1	R/W	H'A414 001C	16
Interrupt priority register 00	INTPRI00	R/W	H'A414 0010	32
Interrupt request register 00	INTREQ00	R/W	H'A414 0024	8
Interrupt mask register 00	INTMSK00	R/W	H'A414 0044	8
Interrupt mask clear register 00	INTMSKCLR00	W	H'A414 0064	8
NMI flag control register	NMIFCR	R/W	H'A414 00C0	16
User interrupt mask level register	USERIMSK	R/W	H'A470 0000	32
Interrupt priority register A	IPRA	R/W	H'A408 0000	16
Interrupt priority register B	IPRB	R/W	H'A408 0004	16
Interrupt priority register C	IPRC	R/W	H'A408 0008	16
Interrupt priority register D	IPRD	R/W	H'A408 000C	16
Interrupt priority register E	IPRE	R/W	H'A408 0010	16
Interrupt priority register F	IPRF	R/W	H'A408 0014	16
Interrupt priority register G	IPRG	R/W	H'A408 0018	16
Interrupt priority register H	IPRH	R/W	H'A408 001C	16

Register	Abbreviation	R/W	Address	Access Size
Interrupt priority register I	IPRI	R/W	H'A408 0020	16
Interrupt priority register J	IPRJ	R/W	H'A408 0024	16
Interrupt priority register K	IPRK	R/W	H'A408 0028	16
Interrupt priority register L	IPRL	R/W	H'A408 002C	16
Interrupt mask register 0	IMR0	R/W	H'A408 0080	8
Interrupt mask register 1	IMR1	R/W	H'A408 0084	8
Interrupt mask register 2	IMR2	R/W	H'A408 0088	8
Interrupt mask register 3	IMR3	R/W	H'A408 008C	8
Interrupt mask register 4	IMR4	R/W	H'A408 0090	8
Interrupt mask register 5	IMR5	R/W	H'A408 0094	8
Interrupt mask register 6	IMR6	R/W	H'A408 0098	8
Interrupt mask register 7	IMR7	R/W	H'A408 009C	8
Interrupt mask register 8	IMR8	R/W	H'A408 00A0	8
Interrupt mask register 9	IMR9	R/W	H'A408 00A4	8
Interrupt mask register 10	IMR10	R/W	H'A408 00A8	8
Interrupt mask register 11	IMR11	R/W	H'A408 00AC	8
Interrupt mask register 12	IMR12	R/W	H'A408 00B0	8
Interrupt mask clear register 0	IMCR0	W	H'A408 00C0	8
Interrupt mask clear register 1	IMCR1	W	H'A408 00C4	8
Interrupt mask clear register 2	IMCR2	W	H'A408 00C8	8
Interrupt mask clear register 3	IMCR3	W	H'A408 00CC	8
Interrupt mask clear register 4	IMCR4	W	H'A408 00D0	8
Interrupt mask clear register 5	IMCR5	W	H'A408 00D4	8
Interrupt mask clear register 6	IMCR6	W	H'A408 00D8	8
Interrupt mask clear register 7	IMCR7	W	H'A408 00DC	8
Interrupt mask clear register 8	IMCR8	W	H'A408 00E0	8
Interrupt mask clear register 9	IMCR9	W	H'A408 00E4	8
Interrupt mask clear register 10	IMCR10	W	H'A408 00E8	8
Interrupt mask clear register 11	IMCR11	W	H'A408 00EC	8
Interrupt mask clear register 12	IMCR12	W	H'A408 00F0	8

Table 13.3 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
ICR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
INTPRI00	Initialized	Initialized	Retained	Retained	Initialized	Retained
INTREQ00	Initialized	Initialized	Retained	Retained	Initialized	Retained
INTMSK00	Initialized	Initialized	Retained	Retained	Initialized	Retained
INTMSKCLR00	Initialized	Initialized	Retained	Retained	Initialized	Retained
NMIFCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
USERIMASK	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRA	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRB	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRC	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRD	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRE	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRF	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRG	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRH	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRI	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRJ	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRK	Initialized	Initialized	Retained	Retained	Initialized	Retained
IPRL	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR6	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR7	Initialized	Initialized	Retained	Retained	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
IMR8	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR9	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR10	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR11	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMR12	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR6	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR7	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR8	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR9	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR10	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR11	Initialized	Initialized	Retained	Retained	Initialized	Retained
IMCR12	Initialized	Initialized	Retained	Retained	Initialized	Retained

13.3.1 Interrupt Control Register 0 (ICR0)

ICR0 is a 16-bit register that sets the input signal detection mode for the external interrupt pins (IQR, NMI), and indicates the input signal level at the NMI pin.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	MAI	—	—	—	—	NMIB	NMIE	—	—	LVL MODE	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R/W	R/W	R	R	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	NMIL	0	R	NMI Input Level Indicates the level of the signal input at the NMI pin. This bit can be read to determine the NMI pin level. This bit cannot be modified. 0: NMI input level is low 1: NMI input level is high
14	MAI	0	R/W	NMI Interrupt Mask Selects whether to mask all interrupts while the NMI input level is low regardless of the BL bit setting in SR. 0: Enables interrupts while the NMI input level is low 1: Disables interrupts while the NMI input level is low
13 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	NMIB	0	R/W	NMI Block Mode Selects whether to detect the NMI interrupt immediately or keep it pending until the BL bit in SR is cleared to 0 if the NMI interrupt is input while the BL bit is set to 1. 0: Keeps the NMI interrupt pending while the BL bit in SR is set to 1 1: Detects the NMI interrupt even while the BL bit in SR is set to 1
8	NMIE	0	R/W	NMI Edge Select Selects whether the falling or rising edge of the interrupt request signal at the NMI pin is detected. 0: Interrupt request is detected on falling edge of NMI input 1: Interrupt request is detected on rising edge of NMI input

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1. The write value should always be 1.
5	LVLMODE	0	R/W	Interrupt Source Retention Mode for Level-Sensed IRQ Signals Specifies whether or not to retain interrupt request signal (IRQ) levels in the INTC. 1: Retained 0: Not retained When level-sensing is selected for IRQ signals, we recommend setting this bit to 1. If this bit is set to 0, the corresponding bit in interrupt request register 00 (INTREQ00) must be cleared to 0 after withdrawal of an interrupt request on an IRQ pin.
4 to 0	—	All 0	—	Reserved These bits are always read as 0. The write value should always be 0.

13.3.2 Interrupt Control Register 1 (ICR1)

ICR1 specifies the detection mode for the external interrupt input pins IRQ7 to IRQ0 individually: rising edge, falling edge, low level, or high level.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ0S		IRQ1S		IRQ2S		IRQ3S		IRQ4S		IRQ5S		IRQ6S		IRQ7S	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	IRQ0S	00	R/W	IRQn Sense Select
13, 12	IRQ1S	00	R/W	These bits select whether interrupt request signals corresponding to pins IRQ7 to IRQ0 are detected by a rising edge, falling edge, low level, or high level.
11, 10	IRQ2S	00	R/W	
9, 8	IRQ3S	00	R/W	
7, 6	IRQ4S	00	R/W	
5, 4	IRQ5S	00	R/W	
3, 2	IRQ6S	00	R/W	
1, 0	IRQ7S	00	R/W	

IRQnS	Detection Mode
00	Interrupt request is detected on falling edge of IRQn input
01	Interrupt request is detected on rising edge of IRQn input
10	Interrupt request is detected on low level of IRQn input
11	Interrupt request is detected on high level of IRQn input

[Legend] n = 0 to 7

13.3.3 Interrupt Priority Register 00 (INTPRI00)

INTPRI00 is a 32-bit register that specifies priority levels from 15 to 0 for the external interrupt input pins IRQ7 to IRQ0.

Each 4-bit group is set with a value from H'F (1111) to H'0 (0000) to specify the interrupt priority level for the corresponding interrupt. Setting H'F means priority level 15 (the highest level); H'0 means priority level 0 (masking is requested).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRQ0				IRQ1				IRQ2				IRQ3			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ4				IRQ5				IRQ6				IRQ7			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	IRQ0	H'0	R/W	These bits set the priority level for each interrupt source in 4-bit units.
27 to 24	IRQ1	H'0	R/W	
23 to 20	IRQ2	H'0	R/W	
19 to 16	IRQ3	H'0	R/W	
15 to 12	IRQ4	H'0	R/W	
11 to 8	IRQ5	H'0	R/W	
7 to 4	IRQ6	H'0	R/W	
3 to 0	IRQ7	H'0	R/W	

13.3.4 Interrupt Priority Registers A to L (IPRA to IPRL)

IPRA to IPRL are 16-bit registers that specify priority levels from 15 to 0 for on-chip peripheral module interrupts.

On-chip peripheral module interrupts are assigned to four 4-bit groups in each register. These 4-bit groups are set with values from H'F (1111) to H'0 (0000) to specify the interrupt priority level for the corresponding interrupt. Setting H'F means priority level 15 (the highest level); H'0 means priority level 0 (masking is requested).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IPR0n				IPR1n				IPR2n				IPR3n			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	IPR0n	H'0	R/W	These bits set the priority level for each interrupt source in 4-bit units. For details, see Table 13.4.
11 to 8	IPR1n	H'0	R/W	
7 to 4	IPR2n	H'0	R/W	
3 to 0	IPR3n	H'0	R/W	

Table 13.4 Interrupt Sources and IPRA to IPRL

Register	IPR0n	IPR1n	IPR2n	IPR3n
IPRA	TMU0_0	TMU0_1	TMU0_2	IrDA
IPRB	JPU	LCDC	DMAC1A	BEU2_1
IPRC	TMU1_0	TMU1_1	TMU1_2	SPU
IPRD	—	MMCIF	—	ATAPI
IPRE	DMAC0A	CEU2_0/BEU2_0/ VEU3F1/VOU2	SCIFA3	VPU5F
IPRF	KEYSC	DMAC0B	USB0/USB1	CMT
IPRG	SCIF0	SCIF1	SCIF2	VEU3F0
IPRH	MSIOF0	MSIOF1	I^2C1	I^2C0
IPRI	SCIFA4	ICB	TSIF	2DG/ICB
IPRJ	CEU2_1	EtherMAC	FSI	SDHI1
IPRK	RTC	DMAC1B	ICB	SDHI0
IPRL	SCIFA5	—	TPU	2DDMAC

Note: —: Reserved. An undefined value will be read. The write value should always be 0.

13.3.5 Interrupt Request Register 00 (INTREQ00)

INTREQ00 is an 8-bit register that indicates interrupt requests from external input pins IRQ7 to IRQ0. This register value is not affected by interrupt mask with the INTPRI00 or INTMSK00 settings.

When edge-detection mode is set for an IRQ pin (ICR1.IRQnS = B'00 or B'01), an interrupt request is cleared by writing 0 to the corresponding IRQn bit after reading IRQn = 1.

Writing 1 to this register is ignored. Write 1 to all bits that are not to be cleared to 0. To clear an interrupt-request bit when level-sensing mode is set for an IRQ pin (ICR1.IRQnS = B'10 or B'11), withdraw the interrupt request by changing the state of the IRQ pin. After that, write 1 to the corresponding bit in the INTMSK00 register. Finally, read the INTREQ00 register to check that the corresponding bit has actually been cleared to 0.

Bit:	7	6	5	4	3	2	1	0
	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4	IRQ5	IRQ6	IRQ7
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ0	0	R/W	IRQn Interrupt Request Indicates whether there is an interrupt request input to the IRQn pin. [For pins in edge-detection mode (ICR1.IRQnS = B'00 or B'01)] • When reading 0: No interrupt request has been detected 1: Interrupt request has been detected • When writing 0: Each bit is cleared by writing 0 after reading 1 1: Writing 1 is ignored. Write 1 to all bits that are not to be cleared to 0. [For pins in level-sensing mode (ICR1.IRQnS = B'10 or B'11)] • When reading (if ICR0.LVLMODE = 0) 0: The corresponding interrupt has not been detected. 1: The corresponding interrupt has been detected. Note: '1' is retained until the CPU accepts the interrupt, or as long as the corresponding bit in the INTMSK00 register. • When reading (if ICR0.LVLMODE = 1) 0: The corresponding interrupt request signal is being asserted. 1: The corresponding interrupt request signal is not being asserted. • When writing Writing 0 or 1 is ignored
6	IRQ1	0	R/W	
5	IRQ2	0	R/W	
4	IRQ3	0	R/W	
3	IRQ4	0	R/W	
2	IRQ5	0	R/W	
1	IRQ6	0	R/W	
0	IRQ7	0	R/W	

[Legend]
n = 0 to 7

When edge-detection mode, or level-sensing mode with the LVLMODE bit in ICR0 set to 0 is selected, bits in this register can be cleared by the following procedure.

(1) When edge-detection mode is selected

An interrupt source can be cleared by writing 0 to the corresponding IRQn bit after reading 1 from the IRQn bit. Write 1 to bits that are not to be cleared.

(2) When level-sensing mode is selected

The bit for an interrupt source can be cleared by writing 1 to the corresponding bit in the INTMSK00 register after having changed the pin state of the IRQ interrupt to withdraw the request. After that, read the INTREQ00 register to check that the corresponding bit has actually been cleared to 0.

When the level-sensing mode is selected and the LVLMODE bit in ICR0 is set to 1, the bits of this register indicate the input of valid IRQ0 interrupt requests. To clear this register, withdraw any interrupt requests by changing the states of IQR pins. Clearing by software is not necessary.

13.3.6 Interrupt Mask Register 00 (INTMSK00)

INTMSK00 is an 8-bit register that masks interrupt requests from external interrupt input pins IRQ7 to IRQ0.

To clear an interrupt mask, write 1 to the corresponding bit in INTMSKCLR00. Writing 0 to the corresponding bit in INTMSK00 does not affect the bit value.

Bit:	7	6	5	4	3	2	1	0
	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4	IRQ5	IRQ6	IRQ7
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ0	0	R/W	IRQn Interrupt Mask 0: The corresponding interrupt is not masked 1: The corresponding interrupt is masked
6	IRQ1	0	R/W	
5	IRQ2	0	R/W	
4	IRQ3	0	R/W	
3	IRQ4	0	R/W	
2	IRQ5	0	R/W	
1	IRQ6	0	R/W	
0	IRQ7	0	R/W	

[Legend]
n = 7 to 0

13.3.7 Interrupt Mask Clear Register 00 (INTMSKCLR00)

INTMSKCLR00 is an 8-bit write-only register that clears the mask settings for interrupts from external interrupt input pins IRQ7 to IRQ0.

Bit:	7	6	5	4	3	2	1	0
	IRQ0	IRQ1	IRQ2	IRQ3	IRQ4	IRQ5	IRQ6	IRQ7
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ0	0	W	IRQn Interrupt Mask Clear 0: Writing 0 is ignored 1: Clears the corresponding interrupt mask
6	IRQ1	0	W	
5	IRQ2	0	W	
4	IRQ3	0	W	
3	IRQ4	0	W	
2	IRQ5	0	W	
1	IRQ6	0	W	
0	IRQ7	0	W	

[Legend]
n = 7 to 0

13.3.8 Interrupt Mask Registers 0 to 12 (IMR0 to IMR12)

IMR0 to IMR12 are 8-bit registers that mask on-chip peripheral module interrupts. To mask an interrupt, write 1 to the corresponding bit in IMR0 to IMR12.

To clear an interrupt mask, write 1 to the corresponding bit in IMCR0 to IMCR12. Writing 0 to the corresponding bit in IMR0 to IMR12 does not affect the bit value.

Table 13.5 shows the relationship between IMR0 to IMR12 and each interrupt source.

Bit:	7	6	5	4	3	2	1	0
	IMRn0	IMRn1	IMRn2	IMRn3	IMRn4	IMRn5	IMRn6	IMRn7
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	IMRn0	0	R/W	Interrupt Mask
6	IMRn1	0	R/W	Masks the interrupt request corresponding to each bit. See Table 13.5 for the relationship between IMR and each interrupt source.
5	IMRn2	0	R/W	
4	IMRn3	0	R/W	When writing:
3	IMRn4	0	R/W	0: Writing 0 is ignored
2	IMRn5	0	R/W	1: Masks the corresponding interrupt request
1	IMRn6	0	R/W	When reading:
0	IMRn7	0	R/W	0: The corresponding interrupt request is not masked 1: The corresponding interrupt request is masked

[Legend]
n = 0 to 12

13.3.9 Interrupt Mask Clear Registers 0 to 12 (IMCR0 to IMCR12)

IMCR0 to IMCR12 are 8-bit write-only registers that clear the mask settings for the on-chip peripheral module interrupts. Table 13.5 shows the relationship between IMCR0 to IMCR12 and each interrupt source.

Bit:	7	6	5	4	3	2	1	0
	IMCRn0	IMCRn1	IMCRn2	IMCRn3	IMCRn4	IMCRn5	IMCRn6	IMCRn7
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
7	IMCRn0	0	W	Interrupt Mask Clear
6	IMCRn1	0	W	Clears the mask setting for the interrupt corresponding to each bit. See Table 13.5 for the relationship between IMCR and each interrupt source.
5	IMCRn2	0	W	
4	IMCRn3	0	W	When writing:
3	IMCRn4	0	W	0: Writing 0 is ignored
2	IMCRn5	0	W	1: Clears the corresponding interrupt mask
1	IMCRn6	0	W	When reading:
0	IMCRn7	0	W	An undefined value will be read

[Legend]
n = 0 to 12

Table 13.5 Correspondence between On-Chip Peripheral Module Interrupt Sources and IMR0 to IMR12, and IMCR0 to IMCR12

	Bit Name (Function Name)							
	7	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Register Name	IMR/CR n0	IMR/CR n1	IMR/CR n2	IMR/CR n3	IMR/CR n4	IMR/CR n5	IMR/CR n6	IMR/CR n7
IMR0/IMCR0	—	TUNI2	TUNI1	TUNI0	SDHII3	SDHII2	SDHII1	SDHII0
	(TMU1)				(SDHI1)			
IMR1/IMCR1	VOUI	VEU1I	BEU0I	CEU0I	DEI3	DEI2	DEI1	DEI0
	(VIO)				(DMAC0A)			
IMR2/IMCR2	—	—	—	VPUI	ATAPI	EtherMAC	—	SCIFA0
	(VPU)				(ATAPI)	(Ether)	(SCIFA)	
IMR3/IMCR3	DEI3	DEI2	DEI1	DEI0	—	—	—	IRDAI
	(DMAC1A)				—			(IrDA)
IMR4/IMCR4	—	TUNI2	TUNI1	TUNI0	JPUI	—	—	LCDCI
	(TMU0)				(JPU)	—	—	(LCDC)
IMR5/IMCR5	KEYI	DADERR	DEI5	DEI4	VEU0I	SCIF2	SCIF1	SCIF0
	(KEYSC)		(DMAC0B)		(VEU3F0)		(SCIF)	
IMR6/IMCR6	—	—	ICBI	SCIFA4	CEU1I	—	MSIOFI1	MSIOFI0
	—		(ICB)	(SCIFA4)	(CEU1)	—	(MSIOF)	
IMR7/IMCR7	DTE0I	WAIT0I	TACK0I	AL0I	DTE1I	WAIT1I	TACK1I	AL1I
	(I^2C0)				(I^2C1)			
IMR8/IMCR8	SDHII3	SDHII2	SDHII1	SDHII0	—	—	SCIFA5	FSI
	(SDHI0)				—	—	(SCIFA5)	(FSI)
IMR9/IMCR9	—	—	—	CMTI	—	USI1	USI0	—
	—			(CMT)	—	(USB1)	(USB0)	—
IMR10/IMCR10	—	DADERR	DEI5	DEI4	—	ATI	PRI	CUI
	(DMAC1B)				(RTC)			
IMR11/IMCR11	BRK	CEI	INI	TRI	—	TPUI	LMBI	TSIFI
	(2DG)				—	(TPU)	(ICB)	(TSIF)
IMR12/IMCR12	—	—	—	—	—	—	—	2DDMAC
	—				—	—	—	(2DDMAC)

Note: —: Reserved. An undefined value will be read. The write value should always be 0.

13.3.10 User Interrupt Mask Level Register (USERIMASK)

USERIMASK is a 32-bit register that specifies the level of interrupts to be accepted. As this register is allocated to a different 64-Kbyte page than where the other INTC registers are allocated, it can be accessed in user mode by translating its address to the corresponding address in area 7 through the MMU.

When the level of an interrupt is not higher than the interrupt level specified in the UIMASK bit, the interrupt is masked. Specifying H'F masks all interrupts except for NMI. The interrupt with a higher level than that specified in the UIMASK bit is accepted only when the corresponding interrupt mask bit in the interrupt mask register is 0 (interrupt enabled) and the IMASK bit setting in SR is lower than the level of the interrupt. The UIMASK value does not change even after an interrupt is accepted.

This register is initialized to H'0000 0000 (all interrupts enabled) by a power-on reset or manual reset. To prevent unintentional modification, this register can only be written to with bits 31 to 24 set to H'A5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UIMASK				—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R/W	Reserved These bits are always read as 0. When writing to the UIMASK bit, be sure to write H'A5 to these bits.
23 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	UIMASK	0000	R/W	User Interrupt Mask Level When the level of an interrupt is not higher than this value, the interrupt is masked.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

13.3.11 NMI Flag Control Register (NMIFCR)

NMIFCR is a 16-bit register that has an NMI flag (NMIFL bit) that can be read and cleared by software. The NMIFL bit is automatically set to 1 by hardware when the INTC detects an NMI, and can be cleared by writing 0 through software.

The NMIFL bit does not affect CPU processing with regard to NMI acceptance. The NMI request detected by the INTC is cleared when the CPU accepts it, but the NMIFL bit is not cleared automatically. Even if 0 is written to the NMIFL bit before the CPU accepts the NMI request, the NMI request is not canceled.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	NMIFL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	NMIL	0	R	NMI Input Level Indicates the level of the signal input at the NMI pin. This bit can be read to determine the NMI pin level. This bit cannot be modified. 0: NMI input level is low 1: NMI input level is high
14 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	NMIFL	0	R/W	NMI Interrupt Request Detection Indicates whether an NMI interrupt request signal has been detected. This bit is automatically set to 1 when the INTC detects an NMI interrupt request. Write 0 to clear the bit. Writing 1 is ignored. 0: NMI interrupt request has not been detected 1: NMI interrupt request has been detected

13.4 Interrupt Sources

There are three types of interrupt sources: NMI, IRQ, and on-chip peripheral modules. Each interrupt has a priority level (16 to 0), with 1 the lowest and 16 the highest. Priority level 0 masks an interrupt, so the interrupt request is ignored.

13.4.1 NMI Interrupt

The NMI interrupt has the highest priority level of 16. When the BL bit in SR of the CPU is 0, NMI interrupts are always accepted. In sleep or standby mode, NMI interrupts are accepted regardless of the BL setting. In addition, NMI interrupts are accepted by setting the NMIB bit in ICR0 regardless of the BL setting.

The NMI signal is edge-detected. The NMIE bit in ICR0 is used to select either rising or falling edge detection. After the NMIE bit in ICR0 is modified, NMI interrupts are not detected for a maximum of six bus clock cycles.

If the INTMU bit of the CPU operation mode register (CPUOPM.INTMU) is set to 1, interrupt mask level in SR (SR.IMASK) is automatically set to the level 15. If the INTMU bit in CPUOPM is set to 0, interrupt handling does not affect the interrupt mask level in SR (SR.IMASK).

13.4.2 IRQ Interrupts

IRQ interrupts are input on pins IRQ7 to IRQ0. The IRQnS bits (n = 0 to 7) in ICR1 are used to set these pins for detection of an edge or level as the interrupt. When level-sensing is selected for IRQ interrupts, operation differs according to the settings of the LVLMODE bit in the ICR0 register.

(1) When ICR0.LVLMODE = 0

The INTC does not keep IRQn bits at the high level if the interrupt request is withdrawn. The level on the pin must be maintained until the CPU accepts the interrupt and starts interrupt handling.

(2) When ICR0.LVLMODE = 1

When level-sensing is selected for IRQ interrupts, the pin level should still be maintained until the CPU accepts the interrupt and starts interrupt handling. However, if an interrupt request is withdrawn before the CPU has accepted it, the INTC keeps the IRQn bit at the high level until the CPU accepts the interrupt. Retention of the value continues until the CPU accepts the interrupt (not necessarily an IRQ interrupt) or the corresponding bit in the INTMSK00 register is set to 1. To clear an IRQn bit in the INTREQ00 register, withdraw the request by changing the pin state of

the IRQ interrupt from within the interrupt processing routine. After that, set the corresponding bit in the INTMSK00 register to 1.

When the INTMU bit in CPUOPM is set to 1, the interrupt mask level (IMASK) in SR is automatically modified to the level of the accepted interrupt. When the INTMU bit is cleared to 0, the IMASK value in SR is not affected by the accepted interrupt.

13.4.3 On-Chip Peripheral Module Interrupts

On-chip peripheral module interrupts are generated by the peripheral modules.

Not every interrupt source is assigned a different interrupt vector. Sources are reflected in the interrupt event register (INTEVT). It is easy to identify sources by using the value of INTEVT as a branch offset in the exception handling routine.

A priority level (from 15 to 0) can be set for each module by writing to IPRA to IPRL.

When the INTMU bit in the CPUOPM is set to 1, the interrupt mask level (IMASK) in SR is automatically modified to the level of the accepted interrupt. When the INTMU bit in CPUOPM is cleared to 0, the IMASK value in SR is not affected by the accepted interrupt.

The interrupt source flags and interrupt enable flags in each peripheral module must be updated only while the BL bit in SR is set to 1 or corresponding interrupt request is masked by the IMASK bit in SR, IMRs, or USERIMASK. To prevent accepting unintentional interrupts that should have been updated, read the on-chip peripheral register with the corresponding flag, wait for the priority determination time for peripheral modules shown in Table 13.8 (e.g. a period required to read a register in INTC once which are driven by the peripheral module clock), and then clear the BL bit to 0 or clear the corresponding interrupt mask by changing the mask setting. Thus, the necessary interval for internal processing is ensured. To update multiple flags, after updating the last flag, read only the register that includes the last flag.

If a flag is updated while the BL bit is 0, execution may branch to the interrupt handling routine with INTEVT = 0; interrupt handling may start depending on the timing relationship between flag updating and interrupt request detection in the LSI. In this case, operation can be continued without causing any problems by executing the RTE instruction.

13.4.4 Interrupt Exception Handling and Priority

Tables 10.6 and 10.7 show the interrupt sources, the codes for the interrupt event register (INTEVT), and the interrupt priority.

Each interrupt source is assigned to a unique INTEVT code. The start address of the exception handling routine is common for all interrupt sources. This is why, for instance, the value of INTEVT is used as an offset at the start of the exception handling routine to branch execution in order to identify the interrupt source.

On-chip peripheral module interrupt priorities can be set freely between 15 and 0 for each module by using IPRA to IPRL. A reset assigns priority level 0 to the on-chip peripheral module interrupts.

If the same priority level is assigned to two or more interrupt sources and interrupts from those sources occur simultaneously, their priority is determined according to the default priority indicated at the right in tables 10.6 and 10.7.

Interrupt priority registers and interrupt mask registers must be updated only while the BL bit in SR is set to 1. To prevent accepting unintentional interrupts, read any interrupt priority register and then clear the BL bit to 0, which ensures the necessary interval for internal processing.

Table 13.6 External Interrupt Sources and Priority

Interrupt Source		INTEVT Code	Interrupt Priority (Initial Value)	IPR (Bit Numbers)	Priority within IPR Setting Range	Default Priority
NMI		H'1C0	16	—	—	High
IRQ	IRQ0	H'600	15 to 0 (0)	INTPRI00 (31 to 28)	—	↑
	IRQ1	H'620	15 to 0 (0)	INTPRI00 (27 to 24)	—	
	IRQ2	H'640	15 to 0 (0)	INTPRI00 (23 to 20)	—	
	IRQ3	H'660	15 to 0 (0)	INTPRI00 (19 to 16)	—	
	IRQ4	H'680	15 to 0 (0)	INTPRI00 (15 to 12)	—	
	IRQ5	H'6A0	15 to 0 (0)	INTPRI00 (11 to 8)	—	
	IRQ6	H'6C0	15 to 0 (0)	INTPRI00 (7 to 4)	—	
	IRQ7	H'6E0	15 to 0 (0)	INTPRI00 (3 to 0)	—	Low

Table 13.7 On-Chip Peripheral Module Interrupt Sources and Priority

Interrupt Source		INTEVT Code	Interrupt Priority (Initial Value)	Corresponding IPR (Bit Numbers)	Priority within IPR Setting Range	Default Priority
HUDI		H'5E0	15	—	—	High
DMAC1A	DEI0	H'700	15 to 0 (0)	IPRB (7 to 4)	High	
	DEI1	H'720	15 to 0 (0)			
	DEI2	H'740	15 to 0 (0)			
	DEI3	H'760	15 to 0 (0)		Low	
2DG	TRI	H'780	15 to 0 (0)	IPRI (3 to 0)	High	
	INI	H'7A0	15 to 0 (0)			
	CEI	H'7C0	15 to 0 (0)			
	BRK	H'7E0	15 to 0 (0)		Low	
DMAC0A	DEI0	H'800	15 to 0 (0)	IPRE (15 to 12)	High	
	DEI1	H'820	15 to 0 (0)			
	DEI2	H'840	15 to 0 (0)			
	DEI3	H'860	15 to 0 (0)		Low	
VIO	CEU0I	H'880	15 to 0 (0)	IPRE (11 to 8)	High	
	BEU0I	H'8A0	15 to 0 (0)			
	VEU3F1I	H'8C0	15 to 0 (0)			
	VOUI	H'8E0	15 to 0 (0)		Low	
SCIFA3	SCIFA3	H'900	15 to 0 (0)	IPRE (7 to 4)	—	
VPU	VPUI	H'980	15 to 0 (0)	IPRE (3 to 0)	—	
TPU	TPUI	H'9A0	15 to 0 (0)	IPRL (7 to 4)	—	
CEU2_1	CEU1I	H'9E0	15 to 0 (0)	IPRJ (15 to 12)	—	
BEU2_1	BEU1I	H'A00	15 to 0 (0)	IPRB (3 to 0)	—	
USB0	USI0	H'A20	15 to 0 (0)	IPRF (7 to 4)	—	
USB1	USI1	H'A40	15 to 0 (0)	IPRF (7 to 4)	—	
ATAPI	ATAPI	H'A60	15 to 0 (0)	IPRD (3 to 0)	—	
RTC	ATI	H'A80	15 to 0 (0)	IPRK (15 to 12)	High	
	PRI	H'AA0	15 to 0 (0)			
	CUI	H'AC0	15 to 0 (0)		Low	
DMAC1B	DEI4	H'B00	15 to 0 (0)	IPRK (11 to 8)	High	
	DEI5	H'B20	15 to 0 (0)			
	DADERR	H'B40	15 to 0 (0)		Low	Low

Interrupt Source		INTEVT Code	Interrupt Priority (Initial Value)	Corresponding IPR (Bit Numbers)	Priority within IPR Setting Range	Default Priority
DMAC0B	DEI4	H'B80	15 to 0 (0)	IPRF (11 to 8)	High	High
	DEI5	H'BA0	15 to 0 (0)		↕	↕
	DADERR	H'BC0	15 to 0 (0)		Low	
KEYSC	KEYI	H'BE0	15 to 0 (0)	IPRF (15 to 12)	—	
SCIF	SCIF0	H'C00	15 to 0 (0)	IPRG (15 to 12)	—	
	SCIF1	H'C20	15 to 0 (0)	IPRG (11 to 8)	—	
	SCIF2	H'C40	15 to 0 (0)	IPRG (7 to 4)	—	
VEU3F0	VEU3F0I	H'C60	15 to 0 (0)	IPRG (3 to 0)	—	
MSIOF	MSIOFI0	H'C80	15 to 0 (0)	IPRH (15 to 12)	—	
	MSIOFI1	H'CA0	15 to 0 (0)	IPRH (11 to 8)	—	
SPU	SPUI0	H'CC0	15 to 0 (0)	IPRC (3 to 0)	—	
	SPUI1	H'CE0	15 to 0 (0)		—	
SCIFA4	SCIFA4	H'D00	15 to 0 (0)	IPRI (15 to 12)	—	
ICB	ICBI	H'D20	15 to 0 (0)	IPRI (11 to 8)	—	
EtherMAC	ETHI	H'D60	15 to 0 (0)	IPRJ (11 to 8)	—	
I^2C1	AL1I	H'D80	15 to 0 (0)	IPRH (7 to 4)	High	
	TACK1I	H'DA0	15 to 0 (0)		↕	
	WAIT1I	H'DC0	15 to 0 (0)			
	DTE1I	H'DE0	15 to 0 (0)		Low	
I^2C0	AL0I	H'E00	15 to 0 (0)	IPRH (3 to 0)	High	
	TACK0I	H'E20	15 to 0 (0)		↕	
	WAIT0I	H'E40	15 to 0 (0)			
	DTE0I	H'E60	15 to 0 (0)		Low	
SDHI0	SDHII0	H'E80	15 to 0 (0)	IPRK (3 to 0)	High	
	SDHII1	H'EA0	15 to 0 (0)		↕	
	SDHII2	H'EC0	15 to 0 (0)			
	SDHII3	H'EE0	15 to 0 (0)		Low	
CMT	CMTI	H'F00	15 to 0 (0)	IPRF (3 to 0)	—	
TSIF	TSIFI	H'F20	15 to 0 (0)	IPRI (7 to 4)	—	
ICB	LMBI	H'F60	15 to 0 (0)	IPRK (7 to 4)	—	Low

Interrupt Source		INTEVT Code	Interrupt Priority (Initial Value)	Corresponding IPR (Bit Numbers)	Priority within IPR Setting Range	Default Priority
FSI	FSII	H'F80	15 to 0 (0)	IPRJ (7 to 4)	—	High
SCIFA5	SCIFA5	H'FA0	15 to 0 (0)	IPRL (15 to 12)	—	
TMU0	TUNI0	H'400	15 to 0 (0)	IPRA (15 to 12)	—	
	TUNI1	H'420	15 to 0 (0)	IPRA (11 to 8)	—	
	TUNI2	H'440	15 to 0 (0)	IPRA (7 to 4)	—	
IrDA	IRDAI	H'480	15 to 0 (0)	IPRA (3 to 0)	—	
SDHI1	SDHII0	H'4E0	15 to 0 (0)	IPRJ (3 to 0)	High	
	SDHII1	H'500	15 to 0 (0)			
	SDHII2	H'520	15 to 0 (0)			
	SDHII3	H'540	15 to 0 (0)		Low	
JPU	JPUI	H'560	15 to 0 (0)	IPRB (15 to 12)	—	
2DDMAC	2DDMAC	H'4A0	15 to 0 (0)	IPRL (3 to 0)	—	
MMC	MMCI2	H'5A0	15 to 0 (0)	IPRD (11 to 8)	—	
	MMCI3	H'5C0	15 to 0 (0)	IPRD (11 to 8)	—	
LCDC	LCDCI	H'F40	15 to 0 (0)	IPRB (11 to 8)	—	
TMU1	TUNI0	H'920	15 to 0 (0)	IPRC (15 to 12)	—	
	TUNI1	H'940	15 to 0 (0)	IPRC (11 to 8)	—	
	TUNI2	H'960	15 to 0 (0)	IPRC (7 to 4)	—	Low

13.5 Operation

13.5.1 Interrupt Sequence

The sequence of interrupt operations is described below. Figures 10.2 and 10.3 are flowcharts of the operations.

1. The interrupt request sources send interrupt request signals to the INTC.
2. The INTC selects the highest-priority interrupt from the sent interrupt requests according to the interrupt priority registers. Lower-priority interrupts are held pending. If two of these interrupts have the same priority level or if multiple interrupts occur within a single module, the interrupt with the highest priority is selected according to tables 10.6 and 10.7.
3. The priority level of the interrupt selected by the INTC is compared with the interrupt mask level (IMASK) set in SR of the CPU. If the priority level is higher than the mask level, the INTC accepts the interrupt and sends an interrupt request signal to the CPU.
4. The CPU accepts an interrupt at a break in instructions.
5. The interrupt source code is set in the interrupt event register (INTEVT).
6. SR and program counter (PC) are saved to SSR and SPC, respectively. R15 is saved to SGR at this time.
7. The BL, MD, and RB bits in SR are set to 1.
8. Execution jumps to the start address of the interrupt exception handling routine (the sum of the value set in the vector base register (VBR) and H'0000 0600).

In the exception handling routine, execution may branch with the INTEVT value used as its offset in order to identify the interrupt source. This enables execution to branch to the handling routine for the individual interrupt source.

Notes: 1. When the INTMU bit in the CPU operating mode register (CPUOPM) is set to 1, the interrupt mask level (IMASK) in SR is automatically set to the level of the accepted interrupt. When the INTMU bit is cleared to 0, the IMASK value in SR is not affected by the accepted interrupt.

2. The interrupt source flag should be cleared in the interrupt handler. To ensure that an interrupt source that should have been cleared is not inadvertently accepted again, read the interrupt source flag, wait for the priority determination time for peripheral modules shown in Table 13.8 (e.g. a period required to read a register in INTC once which is driven by the peripheral module clock), and then clear the BL bit or execute an RTE instruction.

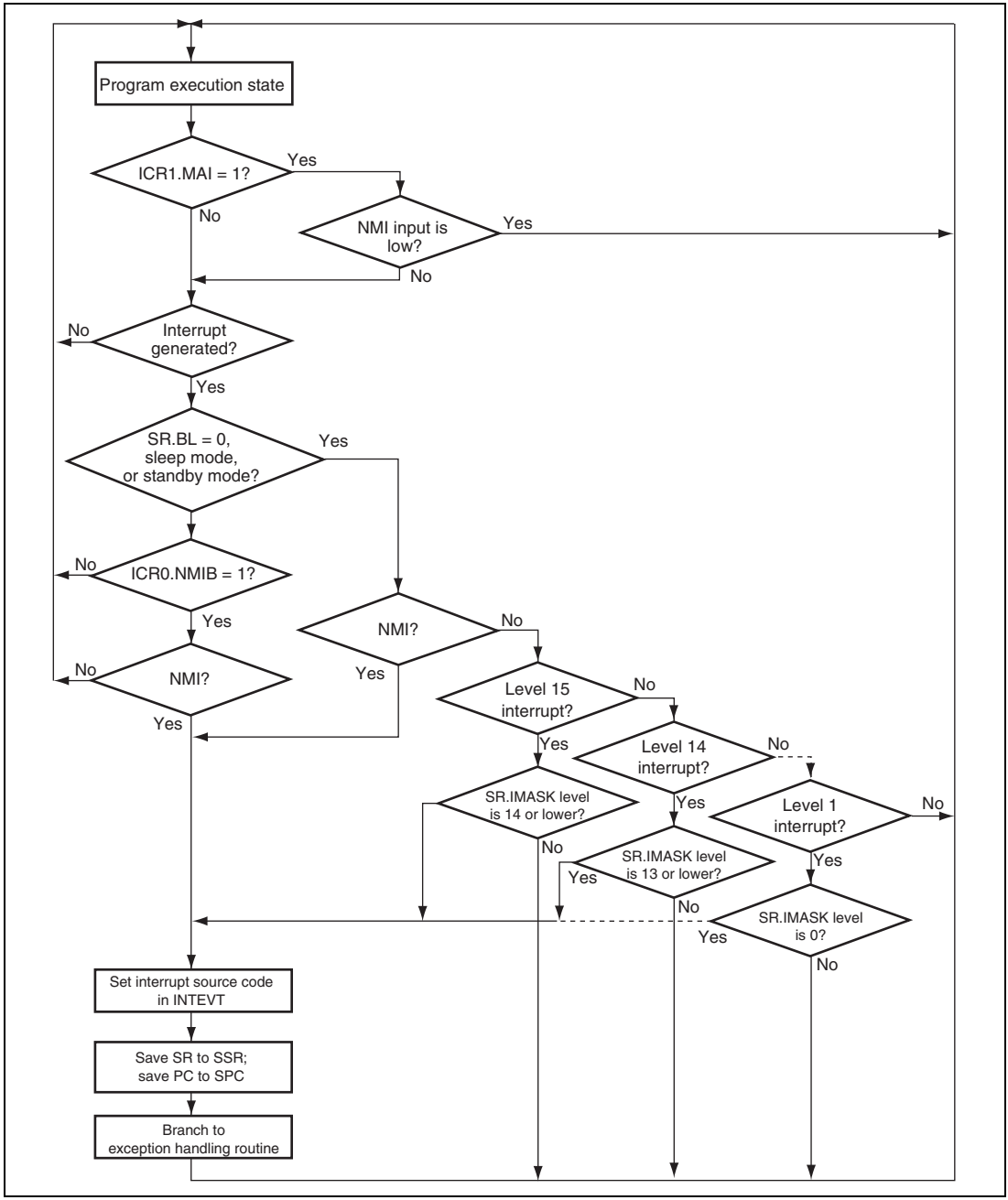


Figure 13.2 Interrupt Operation Flowchart (when CPUOPM.INTMU = 0)

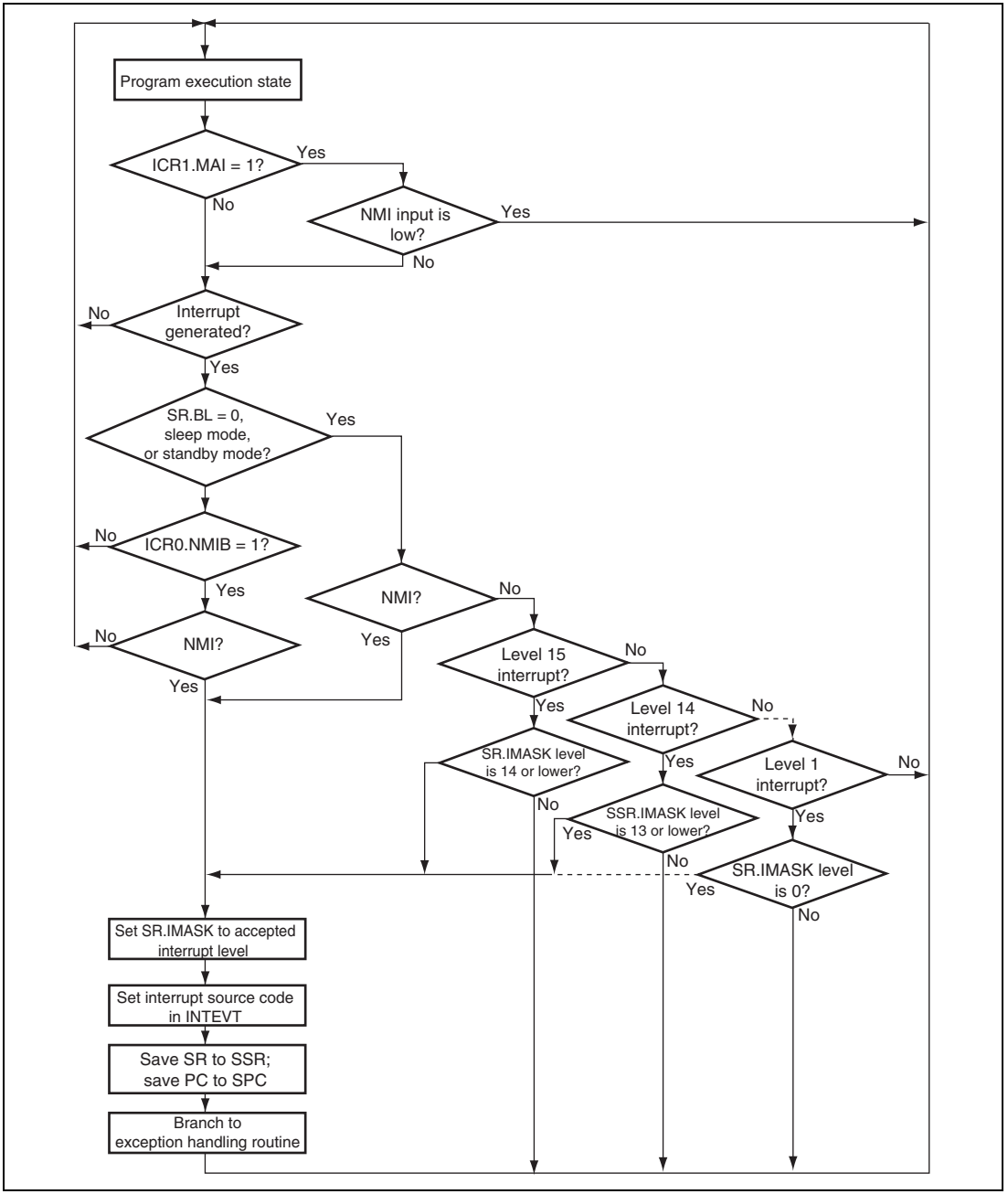


Figure 13.3 Interrupt Operation Flowchart (when CPUOPM.INTMU = 1)

13.5.2 Multiple Interrupts

When handling multiple interrupts, an interrupt handling routine should include the following procedures:

1. To identify the interrupt source, branch to a specific interrupt handling routine for the interrupt source by using the INTEVT code as an offset.
2. Clear the interrupt source in each specific interrupt handling routine.
3. Save SSR and SPC to the stack.
4. Clear the BL bit in SR. When the INTMU bit in CPUOPM is set to 1, the interrupt mask level (IMASK) in SR is automatically modified to the level of the accepted interrupt. When the INTMU bit in CPUOPM is cleared to 0, set the IMASK bit in SR by software to the accepted interrupt level.
5. Handle the interrupt as required.
6. Set the BL bit in SR to 1.
7. Restore SSR and SPC from memory.
8. Execute the RTE instruction.

When these procedures are followed in order, an interrupt of higher priority than the one being handled can be accepted if multiple interrupts occur after step 4. This reduces the interrupt response time for urgent processing.

13.5.3 Interrupt Masking by MAI Bit

Setting the MAI bit in ICR0 to 1 masks interrupts while the NMI signal is low regardless of the BL and IMASK bit settings in SR.

- Normal operation or sleep mode

  All interrupts are masked while the NMI signal is low. Note that only NMI interrupts due to NMI signal input occur.
- Standby mode

  All interrupts including NMI are masked while the NMI signal is low. And an NMI interrupt is not generated by the change of NMI pin state, too. While the MAI bit is set to 1, the NMI interrupt cannot be used to clear standby mode.

13.5.4 Interrupt Disabling Function in User Mode

Setting the interrupt mask level in USERIMASK disables interrupts having an equal or lower priority level than the specified mask level. This function can disable less-urgent interrupts in a task (such as device driver) operating in user mode to accelerate urgent processing.

USERIMASK is allocated to a different 64-Kbyte page than where the other INTC registers are allocated. When accessing this register in user mode, translate the address through the MMU. In the system that uses a multitasking OS, processes that can access USERIMASK must be controlled by using memory protection functions of the MMU. When terminating the task or switching to another task, be sure to clear USERIMASK to 0 before quitting the task. If the UIMASK bits are left set to a non-zero value, interrupts which are not higher in priority than the UIMASK level are held disabled, and correct operation may not be performed (for example, the OS cannot switch tasks).

A sample sequence of user-mode interrupt disabling operation is described below.

1. Classify interrupts into A and B shown below, and assign higher interrupt levels to A than B.
   A. Interrupts that should be accepted in the device driver
      (interrupts used by the OS, such as timer interrupts)
   B. Interrupts that should be disabled in the device driver
2. Make the MMU settings so that the address space including USERIMASK can only be accessed by the device driver in which interrupts should be disabled.
3. Branch to the device driver.
4. Specify the UIMASK bits so that interrupts B are masked in the device driver operating in user mode.
5. Perform urgent processing in the device driver.
6. Clear the UIMASK bits to 0 to return from the device driver processing.

13.6 Interrupt Response Time

Table 13.8 shows the interrupt response time, which is the interval from when an interrupt request occurs until the interrupt exception handling is started and the start instruction of the exception handling routine is fetched.

Table 13.8 Interrupt Response Time

Item		Number of States			Remarks
		NMI	IRQ	Peripheral Module	
Priority determination time		5 Bcyc + 2 Pcyc	4 Bcyc + 2 Pcyc	5 Pcyc	
Wait time until the CPU finishes the current sequence			S - 1 (≥ 0) × Icyc		
Interval from when interrupt exception handling begins (saving SR and PC) until a SuperHyway bus request is issued to fetch the start instruction of the exception handling routine			11 Icyc + 1 Scyc		
Response time	Total	(S + 10) Icyc + 1 Scyc + 5 Bcyc + 2 Pcyc	(S + 10) Icyc + 1 Scyc + 4 Bcyc + 2 Pcyc	(S + 10) Icyc + 1 Scyc + 5 Pcyc	
	Minimum	18 Icyc + S × Icyc	17 Icyc + S × Icyc	16 Icyc + S × Icyc	When Icyc:Scyc:Bcyc: Pcyc = 1:1:1:1

[Legend]

Icyc: Period for one CPU clock cycle

Scyc: Period for one SH clock cycle

Bcyc: Period for one bus clock cycle

Pcyc: Period for one peripheral clock cycle

S: Number of instruction execution states

Section 14 Bus State Controller (BSC)

The bus state controller (BSC) outputs control signals for various types of memory that are connected to the external address space and external devices. The BSC functions enable this LSI to connect directly with SRAM, burst ROM, other types of memory, and external devices. DDR2-SDRAM/MobileDDR-SDRAM is controlled by the bus state controller for SDRAM (DBSC).

14.1 Features

The BSC has the following features:

1. External address space
- Supports totally 256 Mbytes at a maximum. The space is divided into either six or four areas as shown below.
  - Address map 1: Six areas of $\overline{CS0}$, $\overline{CS4}$, $\overline{CS5A}$, $\overline{CS5B}$, $\overline{CS6A}$, and $\overline{CS6B}$
  - Address map 2: Four areas of $\overline{CS0}$, $\overline{CS4}$, $\overline{CS5}$, and $\overline{CS6}$
- Can specify the normal space interface, SRAM interface with byte selection, burst ROM (clock asynchronous), or various PCMCIA interfaces for each address space
- Can select the data bus width (8, 16, or 32 bits) for each address space
- Controls insertion of wait cycle for each address space
- Controls insertion of wait cycle for each read access and write access
- Can set independent idle cycles in the continuous access for five cases: read-write (in the same space/different spaces), read-read (in the same space/different spaces), or the first cycle is a write access

2. Normal space interface
- Supports the interface that can be connected directly to SRAM

3. Burst ROM interface (clock asynchronous)
- High-speed access to ROM that has the page mode function

4. SRAM interface with byte selection
- Supports the interface that can be connected directly to SRAM with byte selection

5. PCMCIA direct-connection interfaces
- Supports the "IC memory card and I/O card interface" provided with JEIDA Ver4.2 (PCMCIA2.1)
- Controls the insertion of wait states by the program
- Supports the bus-sizing function of the I/O bus width. (only in little endian mode.)

Note: For the PCMCIA direct-connection interfaces, the BSC supports only the signals and bus protocols listed in Table 14.1. Use an external circuit for the other control signals.

A block diagram of the BSC is shown in Figure 14.1.

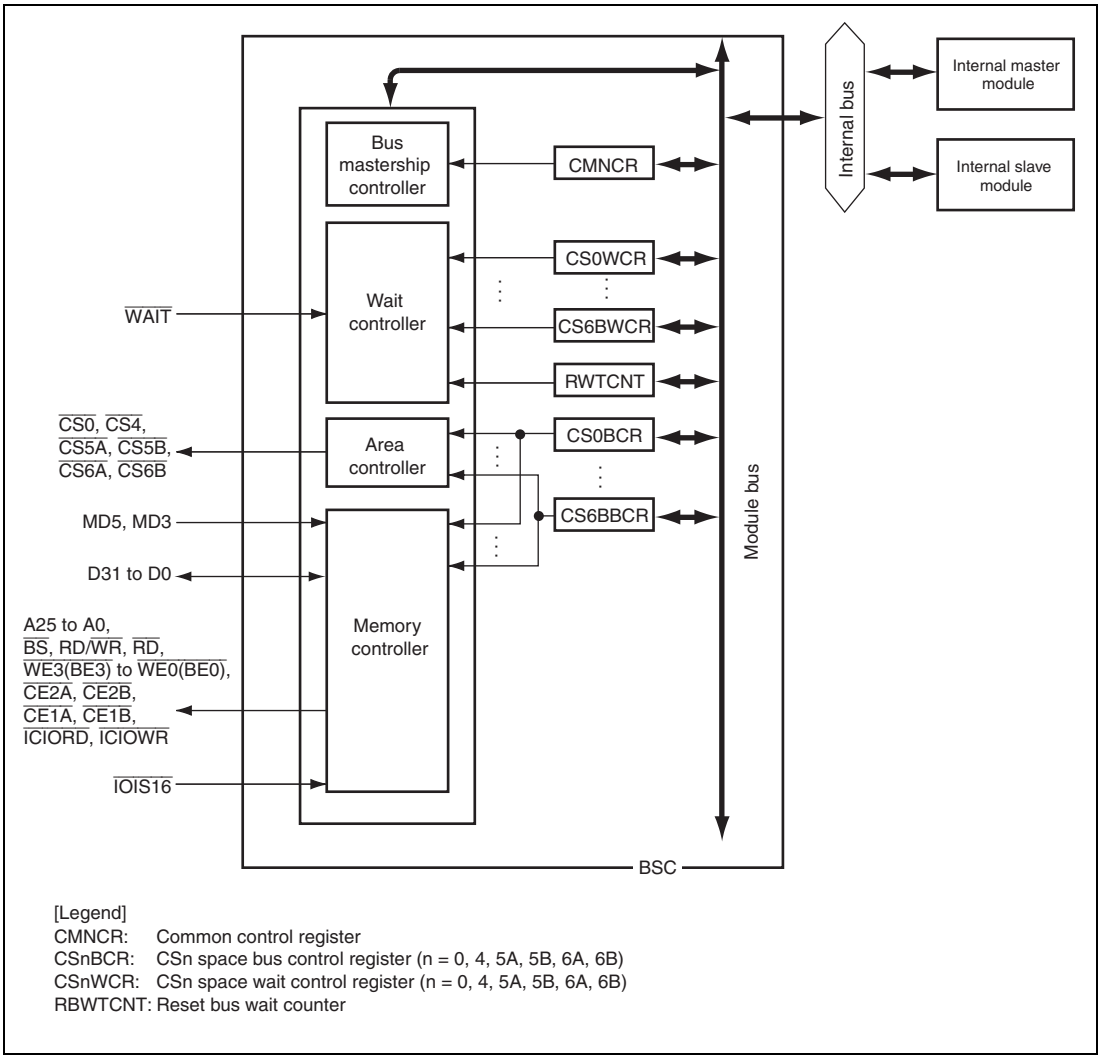


Figure 14.1 Block Diagram of BSC

14.2 Input/Output Pins

Table 14.1 lists the BSC pin configuration.

Table 14.1 Pin Configuration

Name	I/O	Description
A25 to A0	Output	Address output
D31 to D0	I/O	Data bus
$\overline{BS}$	Output	Signal to indicate the start of bus cycles At the start of each bus cycle, this signal is asserted for one cycle of the bus clock. In the case of memory that includes SyncBurst functionality, this is connected to the address-valid pin. On this chip, the pin function is multiplexed on the same pin as A25. This signal is asserted in access to the normal space, burst ROM (clock synchronous or asynchronous), and PCMCIA devices.
$\overline{CS0}$, $\overline{CS4}$	Output	Chip select
$\overline{CS5A}$/$\overline{CE2A}$	Output	Chip select Activated only when address map 1 is selected Correspond to PCMCIA card select signals D15 to D8 when PCMCIA is used
$\overline{CS5B}$/$\overline{CE1A}$	Output	Chip select Correspond to PCMCIA card select signals D7 to D0 when PCMCIA is used
$\overline{CS6A}$/$\overline{CE2B}$	Output	Chip select Activated only when address map 1 is selected Correspond to PCMCIA card select signals D15 to D8 when PCMCIA is used
$\overline{CS6B}$/$\overline{CE1B}$	Output	Chip select Correspond to PCMCIA card select signals D7 to D0 when PCMCIA is used
RDWR	Output	Read/write signal Connected to the $\overline{WE}$ pin when SRAM with byte selection is connected
$\overline{RD}$	Output	Read pulse signal (read data output enable signal) Strobe signal to indicate memory read cycles when PCMCIA is used

Name	I/O	Description
$\overline{\text{WE3(BE3)}}$/ $\overline{\text{ICIOWR}}$	Output	Byte write indication signal corresponding to D31 to D24 Connected to the byte select pin when SRAM with byte selection is connected Strobe signal to indicate the I/O write when PCMCIA is used
$\overline{\text{WE2(BE2)}}$/ $\overline{\text{ICIORD}}$	Output	Byte write indication signal corresponding to D23 to D16 Connected to the byte select pin when SRAM with byte selection is connected Strobe signal to indicate the I/O read when PCMCIA is used
$\overline{\text{WE1(BE1)}}$/ $\overline{\text{WE}}$	Output	Byte write indication signal corresponding to D15 to D8 Connected to the byte select pin when SRAM with byte selection is connected Strobe signal to indicate the memory write cycles when PCMCIA is used
$\overline{\text{WE0(BE0)}}$	Output	Byte write indication signal corresponding to D7 to D0 Connected to the byte select pin when SRAM with byte selection is connected
$\overline{\text{IOIS16}}$	Input	Signal to indicate the 16-bit I/O of PCMCIA Enabled only in little endian mode. In big endian mode, drive this pin low.
$\overline{\text{WAIT}}$	Input	External wait input
MD5, MD3	Input	MD5: Data alignment (big/little endian selectable) MD3: Bus width of area 0 (16/32 bits)

14.3 Area Overview

14.3.1 Area Division

In the architecture, this LSI has 32-bit virtual address spaces. This LSI is supported both 32-bit and 29-bit physical address spaces. The BSC performs control for this 29-bit space. The cache access method that is classified into P0 to P4 spaces by the upper three bits is shown. For details, see section 7, Memory Management Unit (MMU). The remaining 29 bits are used for division of the space into ten areas (address map 1) or eight areas (address map 2) according to the setting of the MAP bit in CMNCR.

As listed in tables 14.2 and 14.3, this LSI can connect eight or six physical areas to each type of memory, and it outputs chip select signals ($\overline{CS0}$, $\overline{MCS}$, $\overline{CS4}$, $\overline{CS5A}$, $\overline{CS5B}$, $\overline{CS6A}$, and $\overline{CS6B}$) for each of them. $\overline{MCS}$ is a control signal of DRAM area used for DDR2-SDRAM or Mobile-DDR SDRAM exclusively (areas 2,3,4,5 or areas 2,3) and controls 256-Mbyte or 128-Mbyte spaces by a chip select.

14.3.2 Address Map

The external address space has a capacity of 384 Mbytes and is used divided into eight partial spaces (address map 1) or six partial spaces (address map 2). The type of memory to be connected and the data bus width are specified in each partial space. The address map for the external address space is listed in tables 14.2 and 14.3.

Table 14.2 Address Map 1 (CMNCR.MAP = 0)

Address	Area	Chip select	Memory to be Connected	Capacity
H'00000000 to H'03FFFFFF	Area 0	$\overline{CS0}$	Normal memory Burst ROM (synchronous/ asynchronous) SRAM with byte selection	64 Mbytes
H'04000000 to H'07FFFFFF	Area 1	—	Internal I/O register area*[1]	64 Mbytes
H'08000000 to H'0FFFFFFF	Areas 2 and 3	$\overline{MCS}$	DRAM area *[2]	128 Mbytes

Address	Area	Chip select	Memory to be Connected	Capacity
H'10000000 to H'13FFFFFF	Area 4*[3]	$\overline{\text{MCS}}$	DRAM area *[2]	64 Mbytes
		$\overline{\text{CS4}}$	Normal memory SRAM with byte selection Burst ROM (synchronous/ asynchronous)	64 Mbytes
H'14000000 to H'15FFFFFF	Area 5A*[3]	$\overline{\text{MCS}}$	DRAM area *[2]	32 Mbytes
		$\overline{\text{CS5A}}$	Normal memory	32 Mbytes
H'16000000 to H'17FFFFFF	Area 5B*[3]	$\overline{\text{MCS}}$	DRAM area *[2]	32 Mbytes
		$\overline{\text{CS5B}}$	Normal memory SRAM with byte selection	32 Mbytes
H'18000000 to H'19FFFFFF	Area 6A	$\overline{\text{CS6A}}$	Normal memory	32 Mbytes
H'1A000000 to H'1BFFFFFF	Area 6B	$\overline{\text{CS6B}}$	Normal memory SRAM with byte selection	32 Mbytes
H'1C000000 to H'1FFFFFFF	Area 7	—	Reserved area*[4]	64 Mbytes

Notes: 1. Set the top three bits of the address of the internal I/O register to B'101 for allocation in area P2.
2. DRAM area is controlled by the DBSC.
3. Areas 4 and 5 are selected DRAM or Normal memory area by MMSELR register.
4. Do not access the reserved area. If the reserved area is accessed, correct operation cannot be guaranteed.

Table 14.3 Address Map 2 (CMNCR.MAP = 1)

Address	Area	Chip select	Memory to be Connected	Capacity
H'00000000 to H'03FFFFFF	Area 0	$\overline{CS0}$	Normal memory Burst ROM (synchronous/ asynchronous)	64 Mbytes
H'04000000 to H'07FFFFFF	Area 1	—	Internal I/O register area*[1]	64 Mbytes
H'08000000 to H'0FFFFFFF	Areas 2 and 3	$\overline{MCS}$	DRAM area *[2]	128 Mbytes
H'10000000 to H'13FFFFFF	Area 4*[4]	$\overline{MCS}$	DRAM area *[2]	64 Mbytes
		$\overline{CS4}$	Normal memory SRAM with byte selection Burst ROM (synchronous/ asynchronous)	64 Mbytes
H'14000000 to H'17FFFFFF	Area 5*[4]	$\overline{MCS}$	DRAM area *[2]	64 Mbytes
		$\overline{CS5B}$*[3]	Normal memory SRAM with byte selection PCMCIA	64 Mbytes
H'18000000 to H'1BFFFFFF	Area 6*[4]	$\overline{CS6B}$*[3]	Normal memory SRAM with byte selection PCMCIA	64 Mbytes
H'1C000000 to H'1FFFFFFF	Area 7	—	Reserved area*[4]	64 Mbytes

Notes: 1. Set the top three bits of the address of the internal I/O register to B'101 for allocation in area P2.
2. DRAM area is controlled by the DBSC.
3. For area 5, CS5BBCR and CS5BWCR are valid.
For area 6, CS6BBCR and CS6BWCR are valid.
4. Areas 4 and 5 are selected DRAM or Normal memory area by MMSELR register.
5. Do not access the reserved area. If the reserved area is accessed, correct operation cannot be guaranteed.

14.3.3 Memory Bus Width

The memory bus width of this LSI can be specified for each address space. In area 0, the bus width of 16 or 32 bits is selected by the external pin (MD3) at a power-on reset. In other areas except area 0 and DRAM area, the bus width is specified by the register. The memory type of area 0 at a power-on reset is normal space.

Table 14.4 Correspondence between External Pin (MD3) and Bus Width

MD3	Bus Width of Area 0
0	16 bits
1	32 bits

14.3.4 Data Alignment

This LSI supports the big endian and little endian methods of data alignment. The data alignment method is specified using the external pin (MD5) at a power-on reset.

Table 14.5 Correspondence between External Pin (MD5) and Endians

MD5	Endian
0	Big endian
1	Little endian

14.4 Register Descriptions

Table 14.6 shows the BSC register configuration. Table 14.7 shows the register states in each operating mode.

Do not access spaces other than CS0 until the completion of the memory interface setting.

Table 14.6 Register Configuration

Register	Abbreviation	R/W	Address	Access Size
Memory address-map selection register	MMSELR	R/W	H'FF800020	32
Common control register	CMNCR	R/W	H'FEC10000	32
CS0 space bus control register	CS0BCR	R/W	H'FEC10004	32
CS4 space bus control register	CS4BCR	R/W	H'FEC10010	32
CS5A space bus control register	CS5ABCR	R/W	H'FEC10014	32
CS5B space bus control register	CS5BBCR	R/W	H'FEC10018	32
CS6A space bus control register	CS6ABCR	R/W	H'FEC1001C	32
CS6B space bus control register	CS6BBCR	R/W	H'FEC10020	32
CS0 space wait control register	CS0WCR	R/W	H'FEC10024	32
CS4 space wait control register	CS4WCR	R/W	H'FEC10030	32
CS5A space wait control register	CS5AWCR	R/W	H'FEC10034	32
CS5B space wait control register	CS5BWCR	R/W	H'FEC10038	32
CS6A space wait control register	CS6AWCR	R/W	H'FEC1003C	32
CS6B space wait control register	CS6BWCR	R/W	H'FEC10040	32
Reset bus wait counter	RBWTCNT	—	H'FEC10054	32

Table 14.7 Register States in Each Operating Mode

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
MMSELR	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
CMNCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained
CS0BCR	Initialized	Retained	Retained	—	Retained*	Initialized	Retained
CS4BCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS5ABCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS5BBCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS6ABCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS6BBCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS0WCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained
CS4WCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS5AWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS5BWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS6AWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
CS6BWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
RBWTCNT	Initialized	Retained	Retained	—	Initialized	Initialized	Retained

Note *: After canceling R-standby, TYPE[3:0] bits in CS0BCR are not retained but initialized.

14.4.1 Memory Address-map Selection Register (MMSELR)

MMSELR is a 32-bit register that selects the memory address map for areas 4 and 5. This register should be accessed at the address H'FF80 0020 in long-word. To prevent incorrect writing, writing is accepted only when the upper 16-bit data is H'A5A5. The upper 29 bits are always read as 0. This register is initialized to H'0000 0000 by a power-on reset or a manual reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AREA SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. To write to this register, the upper 16-bit data should be H'A5A5.
0	AREASEL	0	R/W	Memory Area Select for Areas 4 and 5 0: Areas 4 and 5 (H'1000 0000 to H'17FF FFFF) are controlled by the DBSC (DRAM area). 1: Areas 4 and 5 (H'1000 0000 to H'17FF FFFF) are controlled by the BSC ($\overline{CS4}$, $\overline{CS5A}$, and $\overline{CS5B}$).

This register should be written by the CPU. The instruction to write to this register should be allocated to the uncacheable area P2 and to the area that is not affected by writing to this register.

This register should be written before enabling the instruction cache, operand cache, and MMU address translation and should not be rewritten until after a power-on reset or manual reset is executed.

14.4.2 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the common items for each area.

CMNCR is initialized by a power-on reset. However, CMNCR is not initialized but retains the previous value by a manual reset or in standby mode. Do not access external memory other than area 0 until the CMNCR initialization is completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	BROM MD1	BROM MD0	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MAP[1:0]		—	—	—	—	—	—	—	—	ENDIAN	—	HIZMEM	HIZCNT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0/1*	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22	BROMMD1	0	R/W	Data Acquisition Timing for Clock-Synchronous Burst ROM When clock-synchronous burst ROM is in use, set both this bit and the PH1RD bit in CSnBCR to 1. 0: Sampling of values on the D0 to D31 pins is synchronized with the operating clock (Bϕ) for the BSC. 1. Values on the D0 to D31 pins are sampled on rising edges of the CKO clock signal.
21	BROMMD0	0	R/W	CKO Clock Output Inversion for Clock-Synchronous Burst ROM Setting this bit to 1 inverts the CKO signal, securing hold time relative to CKO for output control signals. Set this bit to 1 when clock-synchronous burst ROM is in use. 0: CKO output is in phase with Bϕ. 0: CKO output is 180° out of phase with Bϕ.
20 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13, 12	MAP[1:0]	00	R/W	Space Specification Selects the address map for the external address space. The address maps to be selected are shown in tables 14.2 and 14.3. 00: Selects address map 1 01: Selects address map 2 1x: Reserved (setting prohibited)
11 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
3	ENDIAN	0/1*	R	Endian Flag Samples the value of the external pin for specifying endian (MD5) at a power-on reset. The endian of all address spaces is defined by this bit. This is a read-only bit. 0: The external pin for specifying endian (MD5) goes low at a power-on reset. This LSI is being operated as big endian. 1: The external pin for specifying endian (MD5) goes high at a power-on reset. This LSI is being operated as little endian.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	HIZMEM	0	R/W	High-Z Memory Control Specifies the pin states in standby mode for A25 to A0, $\overline{BS}$, $\overline{CSn}$, RDWR, $\overline{WEn}$ ($\overline{BEn}$), and $\overline{RD}$. 0: High impedance in standby mode 1: Driven in standby mode
0	HIZCNT	0	R/W	High-Z Control Specifies the states in standby mode for CKO. 0: High impedance in standby mode. 1: Driven in standby mode.

Note: * The external pin for specifying endian (MD5) is sampled at a power-on reset. When big endian is specified, this bit is read as 0 and when little endian is specified, this bit is read as 1.

14.4.3 CSn Space Bus Control Register (CSnBCR) (n = 0, 4, 5A, 5B, 6A, 6B)

CSnBCR is a 32-bit readable/writable register that specifies the function of each area, the number of idle cycles between bus cycles, and the bus width.

Do not access external memory other than area 0 until the CSnBCR initialization is completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	IWW[2:0]			IWRWD[2:0]			IWRWS[2:0]			IWRRD[2:0]			IWRRS[2:0]		
Initial value:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE[3:0]				—	BSZ[1:0]		—	—	—	—	—	—	—	PH1 RD	—
Initial value:	0	0	0	0	0	1*	1*	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
30 to 28	IWW[2:0]	011	R/W	Specification for Idle Cycles between Write-Read/Write-Write Cycles Specify the number of idle cycles to be inserted after access to memory that is connected to the space is completed. The target cycles are write-read cycles and write-write cycles. 000: Reserved 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted

Bit	Bit Name	Initial Value	R/W	Description
27 to 25	IWRWD[2:0]	011	R/W	Specification for Idle Cycles between Read-Write Cycles in Different Spaces Specify the number of idle cycles to be inserted after access to memory that is connected to the space is completed. The target cycles are read-write cycles in which continuous accesses are switched between different spaces. 000: No idle cycle 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
24 to 22	IWRWS[2:0]	011	R/W	Specification for Idle Cycles between Read-Write Cycles in the Same Space Specify the number of idle cycles to be inserted after access to memory that is connected to the space is completed. The target cycles are read-write cycles in which continuous accesses are performed for the same space. 000: No idle cycle 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted

Bit	Bit Name	Initial Value	R/W	Description
21 to 19	IWRRD[2:0]	011	R/W	Specification for Idle Cycles between Read-Read Cycles in Different Spaces Specify the number of idle cycles to be inserted after access to memory that is connected to the space is completed. The target cycles are read-read cycles in which continuous accesses are switched between different spaces. 000: No idle cycle 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
18 to 16	IWRRS[2:0]	011	R/W	Specification for Idle Cycles between Read-Read Cycles in the Same Space Specify the number of idle cycles to be inserted after access to memory that is connected to the space is completed. The target cycles are read-read cycles in which continuous accesses are performed for the same space. 000: No idle cycle 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	TYPE[3:0]	0000	R/W	Memory Type Specification Specify the type of memory connected to the space. 0000: Normal space 0001: Burst ROM (clock asynchronous) 0010: Reserved (setting prohibited) 0011: SRAM with byte selection 0100: Reserved (setting prohibited) 0101: PCMCIA 0110: Reserved (setting prohibited) 0111: Burst ROM (clock synchronous) 1000 to 1111: Reserved (setting prohibited) Note: The memory type of area 0 immediately after a reset is normal space. Use these bits to select either normal memory or burst ROM (clock asynchronous). Refer to tables 14.2 and 14.3 for the memory type of each area.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10, 9	BSZ[1:0]	11*	R/W	Data Bus Size Specification Specify the data bus sizes of spaces. 00: Reserved (setting prohibited) 01: 8-bit size 10: 16-bit size 11: 32-bit size Notes: 1. The data bus width of area 0 is specified by an external input pin. The settings of bits BSZ[1:0] in CS0BCR are ignored. 2. When PCMCIA space is specified for area 5 or area 6, the bus width of 8 or 16 bits is selectable.
8 to 2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	PH1RD	0	R/W	Data Acquisition Timing for Clock-Synchronous Burst ROM Set this bit to 1 when clock-synchronous burst ROM is in use. 0: Data are sampled on falling edges of the operating clock (Bϕ) for the BSC. $\overline{RD}$ output is on falling edges of Bϕ. 1: Data are sampled on rising edges of the operating clock (Bϕ) for the BSC. $\overline{RD}$ output is on rising edges of Bϕ.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Note: * CS0BCR samples the value of the external pin (MD3), that specifies the bus width, at a power-on reset.

14.4.4 CSn Space Wait Control Register (CSnWCR) (n = 0, 4, 5A, 5B, 6A, 6B)

CSnWCR specifies various wait cycles for memory accesses. The bit configuration of this register varies as shown below according to the memory type (TYPE[3:0]) specified by the CSn space bus control register (CSnBCR). Specify CSnWCR before accessing the target area. Specify CSnBCR first, then specify CSnWCR.

(1) Normal Space, SRAM with Byte Selection

- CS0WCR, CS6AWCR, CS6BWCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR SFIX	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0/1*	0	0	0	0	0	0
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	Byte Access Selection when SRAM with Byte Selection is Used Specifies the $\overline{\text{WE}}$n and RDWR signal timing when SRAM interface with byte selection is used. 0: Asserts the $\overline{\text{WE}}$n signal at the read/write timing and asserts the RDWR signal during the write access cycle 1: Asserts the $\overline{\text{WE}}$n signal during the read/write access cycle and asserts the RDWR signal at the write timing
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Wait Cycles in Write Access Specify the number of cycles necessary for write access. 000: The same cycles as WR[3:0] settings (read access wait) 001: 0 cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15	ADRSFIX	0	R/W	Address Updating Inhibited (only valid for CS6A) 0: Outputs address normally 1: Inhibits address updating of the second and subsequent burst access.
14, 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{\text{CS}}$n Assertion to $\overline{\text{RD}}$, $\overline{\text{WE}}$n Assertion Specify the number of delay cycles from address and $\overline{\text{CS}}$n assertion to $\overline{\text{RD}}$ and $\overline{\text{WE}}$n assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Wait Cycles in Read Access Specify the number of wait cycles necessary for read access. 0000: 0 cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
6	WM	0/1*	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait is valid 1: External wait is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$, $\overline{WEn}$ Negation to Address, $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WEn}$ negation to address and $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Note: * The initial value of this bit is 0 in CS0WCR, and 1 in CS6AWCR and CS6BWCR.

- CS4WCR, CS5AWCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	Byte Access Selection when SRAM with Byte Selection is Used Specifies the $\overline{WEn}$ and RDWR signal timing when SRAM interface with byte selection is used. 0: Asserts the $\overline{WEn}$ signal at the read/write timing and asserts the RDWR signal during the write access cycle 1: Asserts the $\overline{WEn}$ signal during the read/write access cycle and asserts the RDWR signal at the write timing
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles necessary for write access. 000: The same cycles as WR[3:0] settings (read access wait) 001: 0 cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{\text{CSn}}$ Assertion to $\overline{\text{RD}}$, $\overline{\text{WEn}}$ Assertion Specify the number of delay cycles from address and $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles
10 to 7	WR[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of cycles necessary for read/write access. 0000: 0 cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited
6	WM	1	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{\text{RD}}$, $\overline{\text{WEn}}$ Negation to Address, $\overline{\text{CSn}}$ Negation Specify the number of delay cycles from $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ negation to address and $\overline{\text{CSn}}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

- CS5BWCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	BAS	—	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		WR[3:0]				WM	—	—	—	—	HM[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	Byte Access Selection when SRAM with Byte Selection is Used Specifies the $\overline{WEn}$ and RDWR signal timing when SRAM interface with byte selection is used. 0: Asserts the $\overline{WEn}$ signal at the read/write timing and asserts the RDWR signal during the write access cycle 1: Asserts the $\overline{WEn}$ signal during the read/write access cycle and asserts the RDWR signal at the write timing
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles necessary for write access. 000: The same cycles as WR[3:0] settings (read access wait) 001: 0 cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{\text{CSn}}$ Assertion to $\overline{\text{RD}}$, $\overline{\text{WEn}}$ Assertion Specify the number of delay cycles from address and $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ assertion. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles
10 to 7	WR[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of cycles necessary for read/write access. 0000: 0 cycle 1000: 10 cycles 0001: 1 cycle 1001: 12 cycles 0010: 2 cycles 1010: 14 cycles 0011: 3 cycles 1011: 18 cycles 0100: 4 cycles 1100: 24 cycles 0101: 5 cycles 1101: Setting prohibited 0110: 6 cycles 1110: Setting prohibited 0111: 8 cycles 1111: Setting prohibited
6	WM	1	R/W	External Wait Mask Specification Specify whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{\text{RD}}$, $\overline{\text{WEn}}$ Negation to Address, $\overline{\text{CSn}}$ Negation Specify the number of delay cycles from $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ negation to address and $\overline{\text{CSn}}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(2) Burst ROM

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		W[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or later access cycles in burst read access. 00: 0 cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{\text{CSn}}$ Assertion to $\overline{\text{RD}}$, $\overline{\text{WEn}}$ Assertion Specify the number of delay cycles from address and $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ assertion. $\overline{\text{WEn}}$, and $\overline{\text{RD}}$ when CSnBCR.PH1RD = 0 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles $\overline{\text{RD}}$ when CSnBCR.PH1RD = 1 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the write and first read access cycles. 0000: 0 cycle 1000: 10 cycles 0001: 1 cycle 1001: 12 cycles 0010: 2 cycles 1010: 14 cycles 0011: 3 cycles 1011: 18 cycles 0100: 4 cycles 1100: 24 cycles 0101: 5 cycles 1101: Setting prohibited 0110: 6 cycles 1110: Setting prohibited 0111: 8 cycles 1111: Setting prohibited
6	WM	0	R/W	External Wait Mask Specification Specify whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bit are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$, $\overline{WEn}$ Negation to Address, $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WEn}$ negation to address and $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

- CS4WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SW[1:0]		W[3:0]				WM	—	—	—	—	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or later access cycles in burst read access. 00: 0 cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{\text{CSn}}$ Assertion to $\overline{\text{RD}}$, $\overline{\text{WEn}}$ Assertion Specify the number of delay cycles from address and $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ assertion. $\overline{\text{WEn}}$, and $\overline{\text{RD}}$ when CSnBCR.PH1RD = 0 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles $\overline{\text{RD}}$ when CSnBCR.PH1RD = 1 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the write and first read access cycles. 0000: 0 cycle 1000: 10 cycles 0001: 1 cycle 1001: 12 cycles 0010: 2 cycles 1010: 14 cycles 0011: 3 cycles 1011: 18 cycles 0100: 4 cycles 1100: 24 cycles 0101: 5 cycles 1101: Setting prohibited 0110: 6 cycles 1110: Setting prohibited 0111: 8 cycles 1111: Setting prohibited
6	WM	1	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Number of Delay Cycles from $\overline{RD}$, $\overline{WEn}$ Negation to Address, $\overline{CSn}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WEn}$ negation to address and $\overline{CSn}$ negation. 00: 0.5 cycle 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

(3) PCMCIA

- CS5BWCR, CS6BWCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	SA[1:0]		—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TED[3:0]				PCW[3:0]				WM	—	—	TEH[3:0]			
Initial value:	0	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21 20	SA1 SA0	0 0	R/W	Space Attribute Specification When PCMCIA interface is selected, these bits select either the memory card interface or I/O card interface. SA1 0: Specifies the memory card interface for the space with A25 = 1. 1: Specifies the I/O card interface for the space with A25 = 1. SA0 0: Specifies the memory card interface for the space with A25 = 0. 1: Specifies the I/O card interface for the space with A25 = 0.
19 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14 to 11	TED[3:0]	0000	R/W	Number of Delay Cycles from Address to $\overline{RD}$, $\overline{WE}$ Assertion Specify the number of delay cycles from address output on the PCMCIA interface to $\overline{RD}$ and $\overline{WE}$ assertion. 0000: 0.5 cycle 1000: 8.5 cycles 0001: 1.5 cycles 1001: 9.5 cycles 0010: 2.5 cycles 1010: 10.5 cycles 0011: 3.5 cycles 1011: 11.5 cycles 0100: 4.5 cycles 1100: 12.5 cycles 0101: 5.5 cycles 1101: 13.5 cycles 0110: 6.5 cycles 1110: 14.5 cycles 0111: 7.5 cycles 1111: 15.5 cycles
10 to 7	PCW[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted. 0000: 3 cycles 1000: 30 cycles 0001: 6 cycles 1001: 33 cycles 0010: 9 cycles 1010: 36 cycles 0011: 12 cycles 1011: 38 cycles 0100: 15 cycles 1100: 52 cycles 0101: 18 cycles 1101: 60 cycles 0110: 22 cycles 1110: 64 cycles 0111: 26 cycles 1111: 80 cycles
6	WM	1	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycles is 0. 0: External wait input is valid 1: External wait input is ignored
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	THE[3:0]	0000	R/W	Delay Cycles from $\overline{RD}$, $\overline{WE}$ Negation to Address Specify the number of delay cycles from $\overline{RD}$ and $\overline{WE}$ negation to address on the PCMCIA interface. 0000: 0.5 cycle 1000: 8.5 cycles 0001: 1.5 cycles 1001: 9.5 cycles 0010: 2.5 cycles 1010: 10.5 cycles 0011: 3.5 cycles 1011: 11.5 cycles 0100: 4.5 cycles 1100: 12.5 cycles 0101: 5.5 cycles 1101: 13.5 cycles 0110: 6.5 cycles 1110: 14.5 cycles 0111: 7.5 cycles 1111: 15.5 cycles

14.4.5 Reset Bus Wait Counter (RBWTCNT)

RBWTCNT is a 7-bit counter. This counter starts to increment in synchronization with the CKO pin after a power-on reset is cleared, and stops when the value reaches H'7F. External bus access is suspended while the counter is operating. This counter is provided to assume the minimum time from clearing a reset for flash memory and the like to the first access. This counter cannot be read from or written to.

14.5 Operation

14.5.1 Endian/Access Size and Data Alignment

This LSI supports big endian, in which the 0 address is the most significant byte (MSByte) in the byte data and little endian, in which the 0 address is the least significant byte (LSByte) in the byte data. Endian is specified at a power-on reset by the external pin (MD5). When the MD5 pin is low level at a power-on reset, the endian will become big endian and when the MD5 pin is high level at a power-on reset, the endian will become little endian.

Three data bus widths (8, 16, or 32 bits) are available for normal memory and SRAM with byte selection. Two data bus widths (8 or 16 bits) are available for PCMCIA interface. Data alignment is performed in accordance with the data bus width of the device and endian. This also means that when longword data is read from an 8-bit width device, the read operation must be done four times. In this LSI, data alignment and conversion of the data length are performed automatically between the respective interfaces.

Tables 14.8 to 14.13 show the relationship between endian, device data width, and access unit.

Table 14.8 32-Bit External Device/Big Endian Access and Data Alignment

	Data Bus				Strobe Signals			
Operation	D31 to D24	D23 to D16	D15 to D8	D7 to D0	$\overline{WE3}$ ($\overline{BE3}$)	$\overline{WE2}$ ($\overline{BE2}$)	$\overline{WE1}$ ($\overline{BE1}$)	$\overline{WE0}$ ($\overline{BE0}$)
Byte access at 0	Data 7 to data 0	—	—	—	Assert	—	—	—
Byte access at 1	—	Data 7 to data 0	—	—	—	Assert	—	—
Byte access at 2	—	—	Data 7 to data 0	—	—	—	Assert	—
Byte access at 3	—	—	—	Data 7 to data 0	—	—	—	Assert
Word access at 0	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert	—	—
Word access at 2	—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
Longword access at 0	Data 31 to data 24	Data 23 to data 16	Data 15 to data 8	Data 7 to data 0	Assert	Assert	Assert	Assert

Table 14.9 16-Bit External Device/Big Endian Access and Data Alignment

		Data Bus				Strobe Signals			
Operation		**D31 to D24**	**D23 to D16**	**D15 to D8**	**D7 to D0**	**$\overline{WE3}$ ($\overline{BE3}$)**	**$\overline{WE2}$ ($\overline{BE2}$)**	**$\overline{WE1}$ ($\overline{BE1}$)**	**$\overline{WE0}$ ($\overline{BE0}$)**
Byte access at 0		—	—	Data 7 to data 0	—	—	—	Assert	—
Byte access at 1		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 2		—	—	Data 7 to data 0	—	—	—	Assert	—
Byte access at 3		—	—	—	Data 7 to data 0	—	—	—	Assert
Word access at 0		—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
Word access at 2		—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
Longword access at 0	1st time at 0	—	—	Data 31 to data 24	Data 23 to data 16	—	—	Assert	Assert
	2nd time at 2	—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert

Table 14.10 8-Bit External Device/Big Endian Access and Data Alignment

Operation		Data Bus				Strobe Signals			
		D31 to D24	D23 to D16	D15 to D8	D7 to D0	$\overline{WE3}$ ($\overline{BE3}$)	$\overline{WE2}$ ($\overline{BE2}$)	$\overline{WE1}$ ($\overline{BE1}$)	$\overline{WE0}$ ($\overline{BE0}$)
Byte access at 0		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 1		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 2		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 3		—	—	—	Data 7 to data 0	—	—	—	Assert
Word access at 0	1st time at 0	—	—	—	Data 15 to data 8	—	—	—	Assert
	2nd time at 1	—	—	—	Data 7 to data 0	—	—	—	Assert
Word access at 2	1st time at 2	—	—	—	Data 15 to data 8	—	—	—	Assert
	2nd time at 3	—	—	—	Data 7 to data 0	—	—	—	Assert
Longword access at 0	1st time at 0	—	—	—	Data 31 to data 24	—	—	—	Assert
	2nd time at 1	—	—	—	Data 23 to data 16	—	—	—	Assert
	3rd time at 2	—	—	—	Data 15 to data 8	—	—	—	Assert
	4th time at 3	—	—	—	Data 7 to data 0	—	—	—	Assert

Table 14.11 32-Bit External Device/Little Endian Access and Data Alignment

	Data Bus				Strobe Signals			
Operation	**D31 to D24**	**D23 to D16**	**D15 to D8**	**D7 to D0**	**$\overline{WE3}$ ($\overline{BE3}$)**	**$\overline{WE2}$ ($\overline{BE2}$)**	**$\overline{WE1}$ ($\overline{BE1}$)**	**$\overline{WE0}$ ($\overline{BE0}$)**
Byte access at 0	—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 1	—	—	Data 7 to data 0	—	—	—	Assert	—
Byte access at 2	—	Data 7 to data 0	—	—	—	Assert	—	—
Byte access at 3	Data 7 to data 0	—	—	—	Assert	—	—	—
Word access at 0	—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
Word access at 2	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert	—	—
Longword access at 0	Data 31 to data 24	Data 23 to data 16	Data 15 to data 8	Data 7 to data 0	Assert	Assert	Assert	Assert

Table 14.12 16-Bit External Device/Little Endian Access and Data Alignment

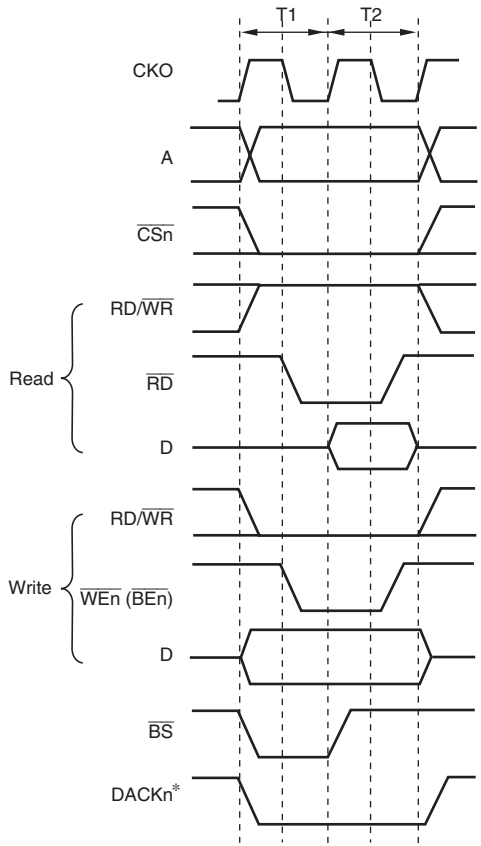
		Data Bus				Strobe Signals			
Operation		**D31 to D24**	**D23 to D16**	**D15 to D8**	**D7 to D0**	**$\overline{WE3}$ ($\overline{BE3}$)**	**$\overline{WE2}$ ($\overline{BE2}$)**	**$\overline{WE1}$ ($\overline{BE1}$)**	**$\overline{WE0}$ ($\overline{BE0}$)**
Byte access at 0		—		—	Data 7 to data 0	—	—	—	Assert
Byte access at 1		—	—	Data 7 to data 0	—	—	—	Assert	—
Byte access at 2		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 3		—	—	Data 7 to data 0	—	—	—	Assert	—
Word access at 0		—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
Word access at 2		—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
Longword access at 0	1st time at 0	—	—	Data 15 to data 8	Data 7 to data 0	—	—	Assert	Assert
	2nd time at 2	—	—	Data 31 to data 24	Data 23 to data 16	—	—	Assert	Assert

Table 14.13 8-Bit External Device/Little Endian Access and Data Alignment

		Data Bus				Strobe Signals			
Operation		**D31 to D24**	**D23 to D16**	**D15 to D8**	**D7 to D0**	**$\overline{WE3}$ ($\overline{BE3}$)**	**$\overline{WE2}$ ($\overline{BE2}$)**	**$\overline{WE1}$ ($\overline{BE1}$)**	**$\overline{WE0}$ ($\overline{BE0}$)**
Byte access at 0		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 1		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 2		—	—	—	Data 7 to data 0	—	—	—	Assert
Byte access at 3		—	—	—	Data 7 to data 0	—	—	—	Assert
Word access at 0	1st time at 0	—	—	—	Data 7 to data 0	—	—	—	Assert
	2nd time at 1	—	—	—	Data 15 to data 8	—	—	—	Assert
Word access at 2	1st time at 2	—	—	—	Data 7 to data 0	—	—	—	Assert
	2nd time at 3	—	—	—	Data 15 to data 8	—	—	—	Assert
Longword access at 0	1st time at 0	—	—	—	Data 7 to data 0	—	—	—	Assert
	2nd time at 1	—	—	—	Data 15 to data 8	—	—	—	Assert
	3rd time at 2	—	—	—	Data 23 to data 16	—	—	—	Assert
	4th time at 3	—	—	—	Data 31 to data 24	—	—	—	Assert

14.5.2 Normal Space Interface

Basic Timing: For access to a normal space, this LSI uses strobe signal output in consideration of the fact that mainly SRAM without a byte-selection pin will be directly connected. When using SRAM with a byte-selection pin, see section 14.5.6, SRAM Interface with Byte Selection. Figure 14.2 shows the basic timing of normal space access. A no-wait normal access is completed in two cycles. The $\overline{BS}$ signal indicates the start of bus cycles and is asserted for one cycle.



Note: * The waveform for DACKn is when active low is specified.

Figure 14.2 Normal Space Basic Access Timing (Access Wait 0)

There is no access size specification when reading from the external bus. The correct access start address is output in the least significant bit of the address, but since there is no access size specification, 16 bits are always read in case of a 16-bit device. When writing, only the $\overline{\text{WE}}$n ($\overline{\text{BE}}$n) signal for the byte to be written is asserted.

It is necessary to output the data that has been read using $\overline{\text{RD}}$ when a buffer is established in the data bus. The RDWR signal is in a read state (high output) when no access has been carried out. Therefore, care must be taken to avoid collision when controlling the external data buffer using RDWR.

Figures 14.3 and 14.4 show the basic timings of continuous accesses to a normal space. If the WM bit in CSnWCR is cleared to 0, a Tnop cycle is inserted to evaluate the external wait (Figure 14.3). However, if the WM bit in CSnWCR is set to 1, external waits are ignored and no Tnop cycle is inserted (Figure 14.4).

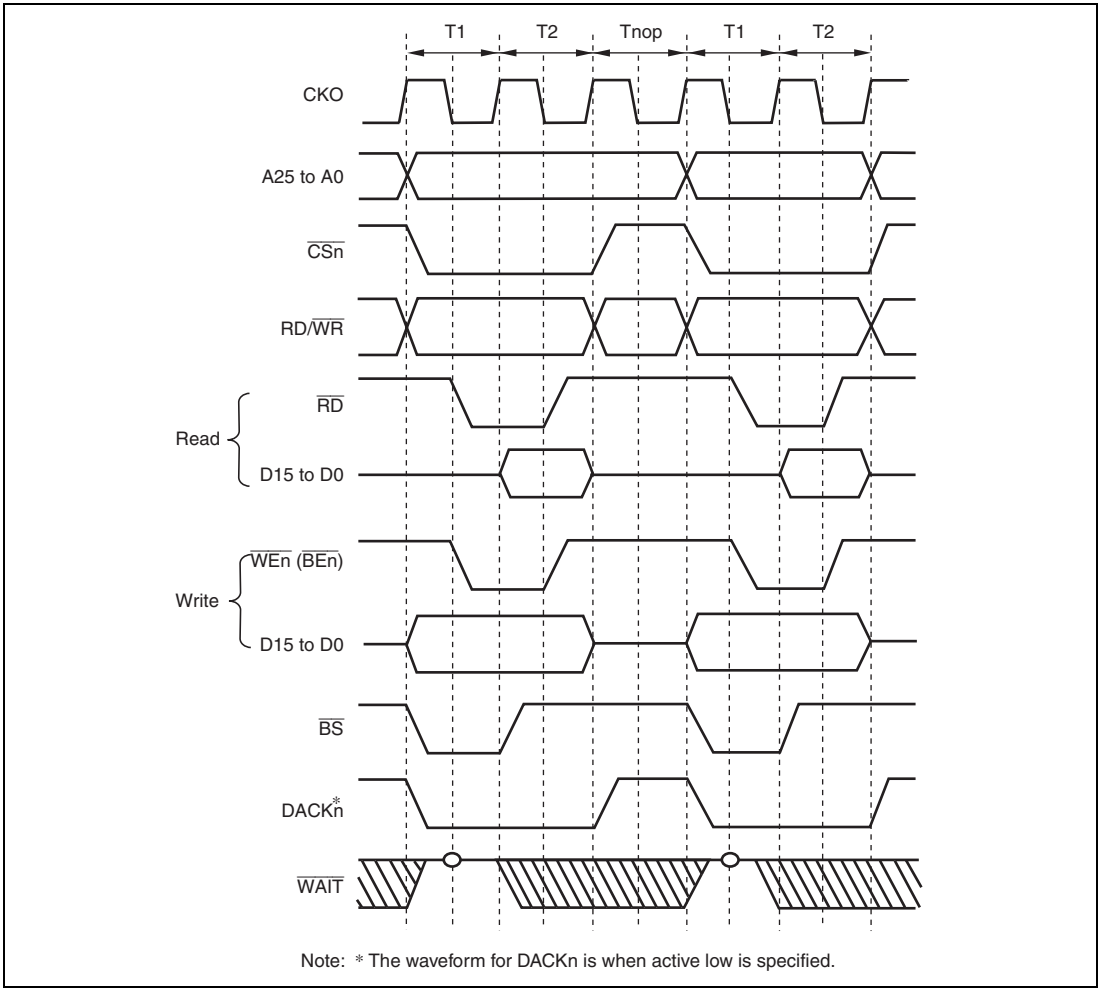


Figure 14.3 Continuous Access for Normal Space 1
Bus Width = 16 Bits, Longword Access, WM Bit in CSnWCR = 0
(Access Wait = 0, Cycle Wait = 0)

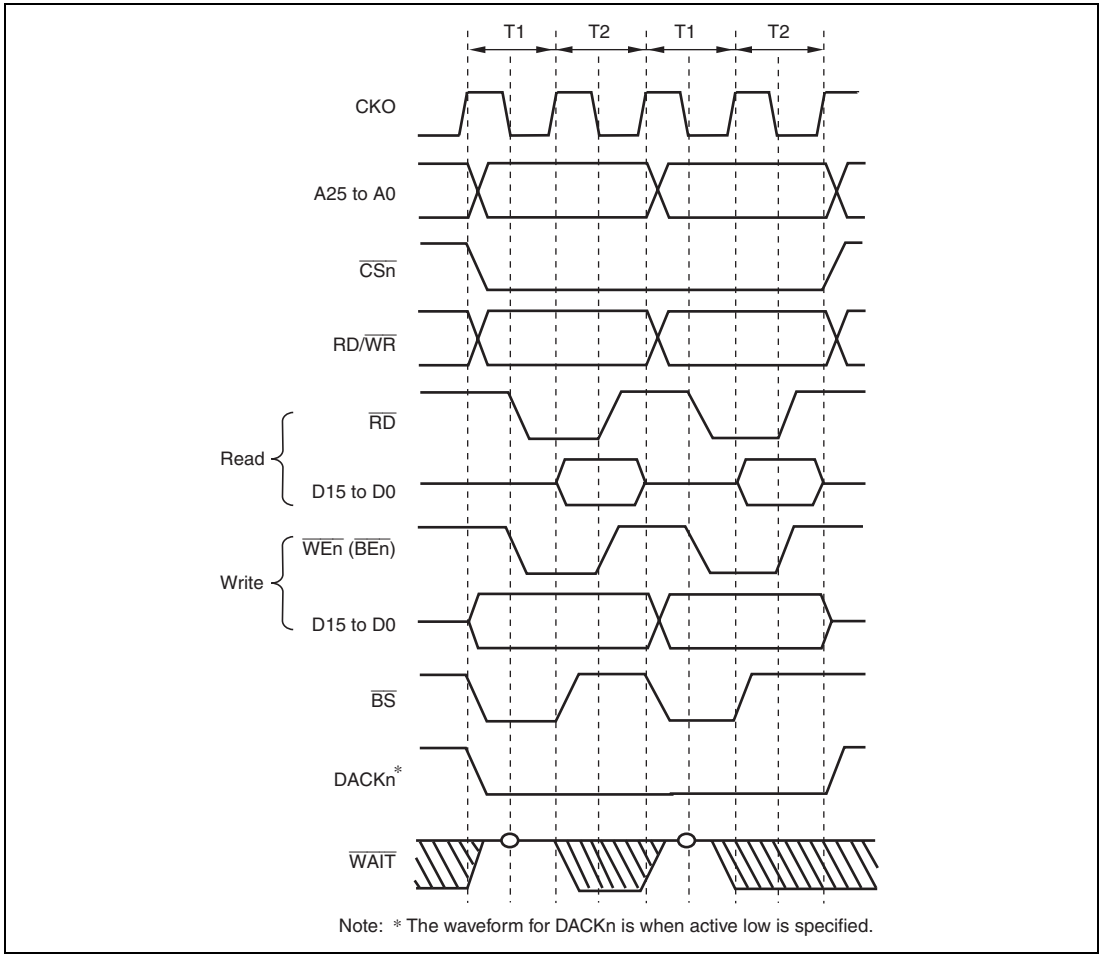


Figure 14.4 Continuous Access for Normal Space 2
Bus Width = 16 Bits, Longword Access, WM Bit in CSnWCR = 1
(Access Wait = 0, Cycle Wait = 0)

Figure 14.5 shows an example of connection to 32-bit data-width SRAM.

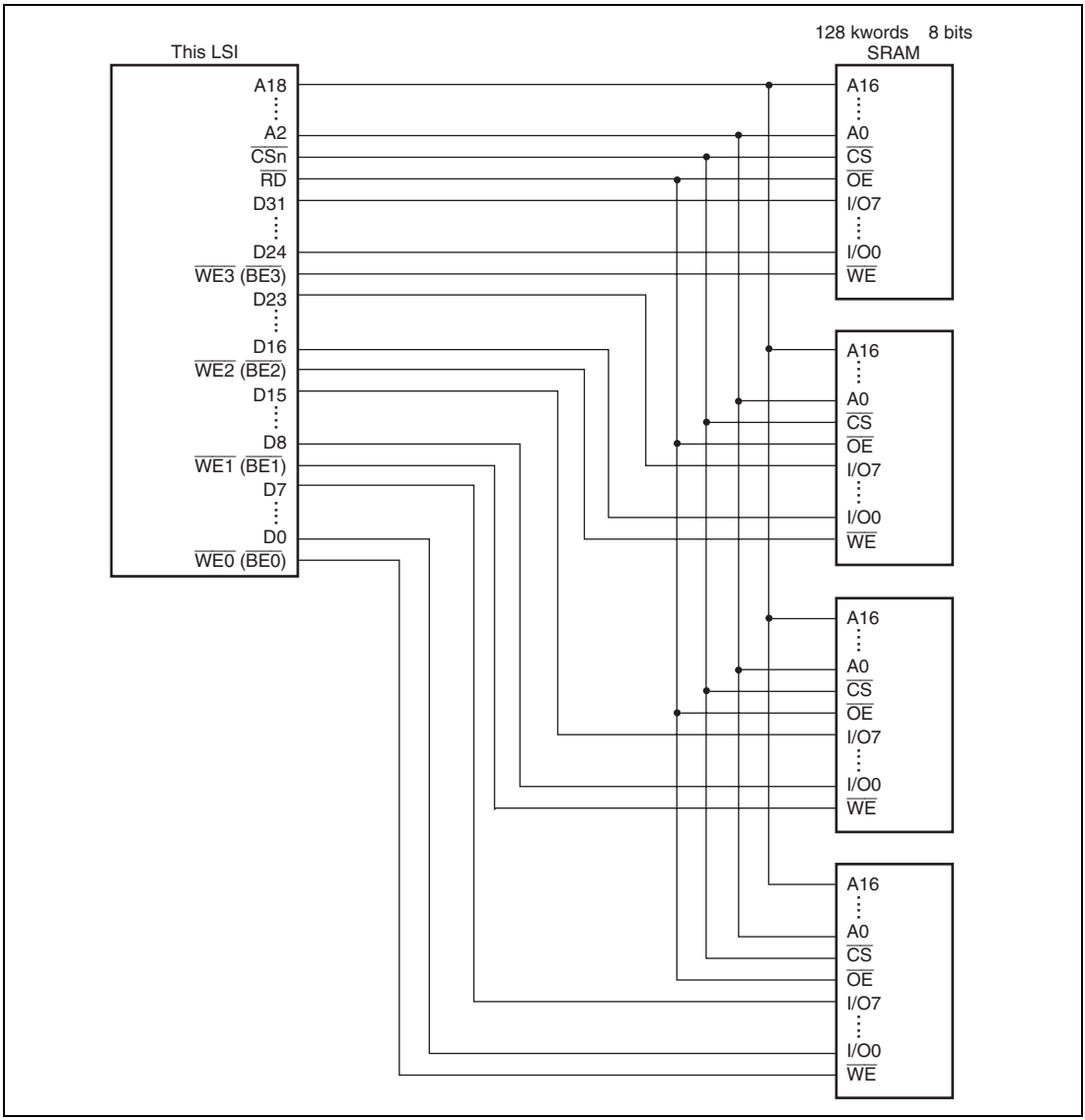


Figure 14.5 Example of 32-Bit Data-Width SRAM Connection

Figure 14.6 shows an example of connection to 16-bit data-width SRAM.

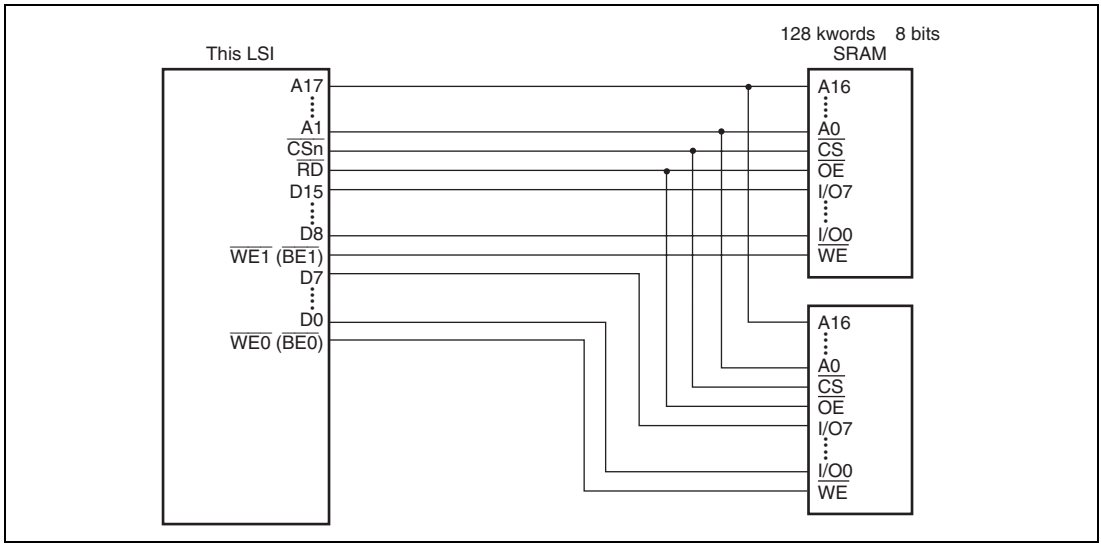


Figure 14.6 Example of 16-Bit Data-Width SRAM Connection

Figure 14.7 shows an example of connection to 8-bit data-width SRAM.

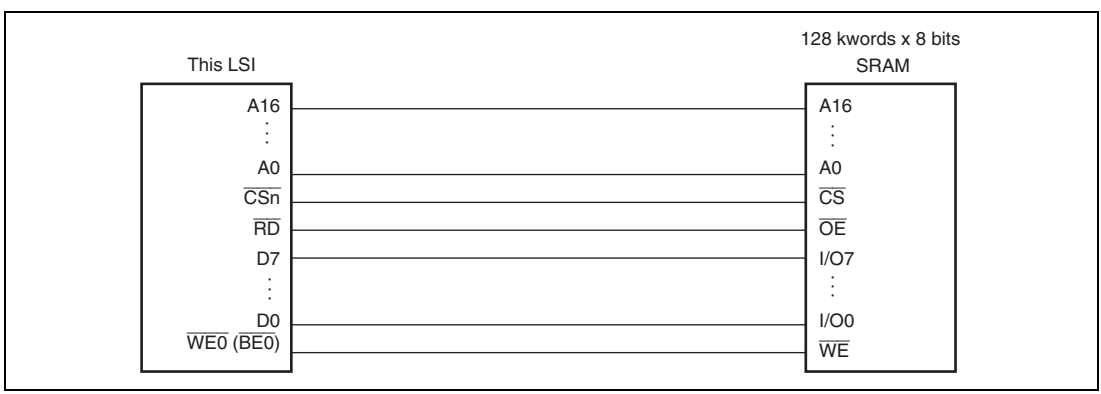


Figure 14.7 Example of 8-Bit Data-Width SRAM Connection

14.5.3 Access Wait Control

Wait cycle insertion on a normal space access can be controlled by the settings of the WR[3:0] bits in CSnWCR. By setting the WW[2:0] bits, it is possible to insert wait cycles independently in read access and in write access. The other areas have common access wait cycles for read and write cycles. The specified number of Tw cycles are inserted as wait cycles in a normal space access shown in Figure 14.8.

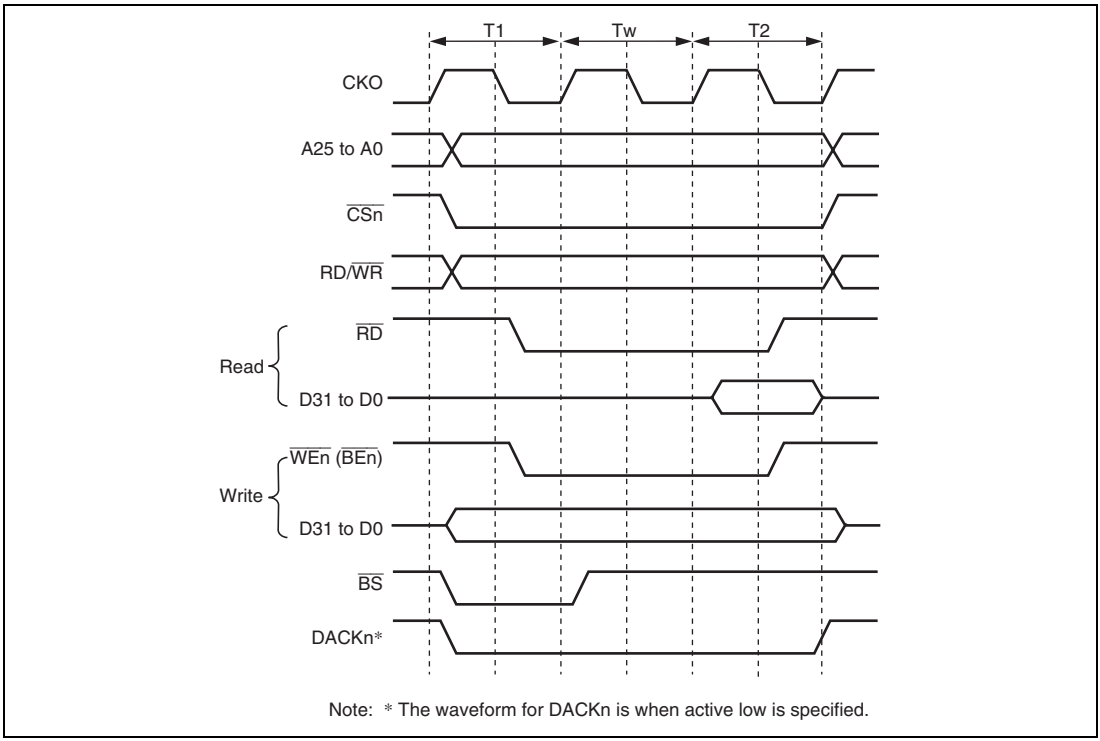


Figure 14.8 Wait Timing for Normal Space Access (Software Wait Only)

When the WM bit in CSnWCR is cleared to 0, the external wait input $\overline{\text{WAIT}}$ signal is also sampled. $\overline{\text{WAIT}}$ pin sampling is shown in Figure 14.9. A 2-cycle wait is specified as a software wait. The $\overline{\text{WAIT}}$ signal is sampled at the falling edge of CKO at the transition from the T1 or Tw cycle to the T2 cycle.

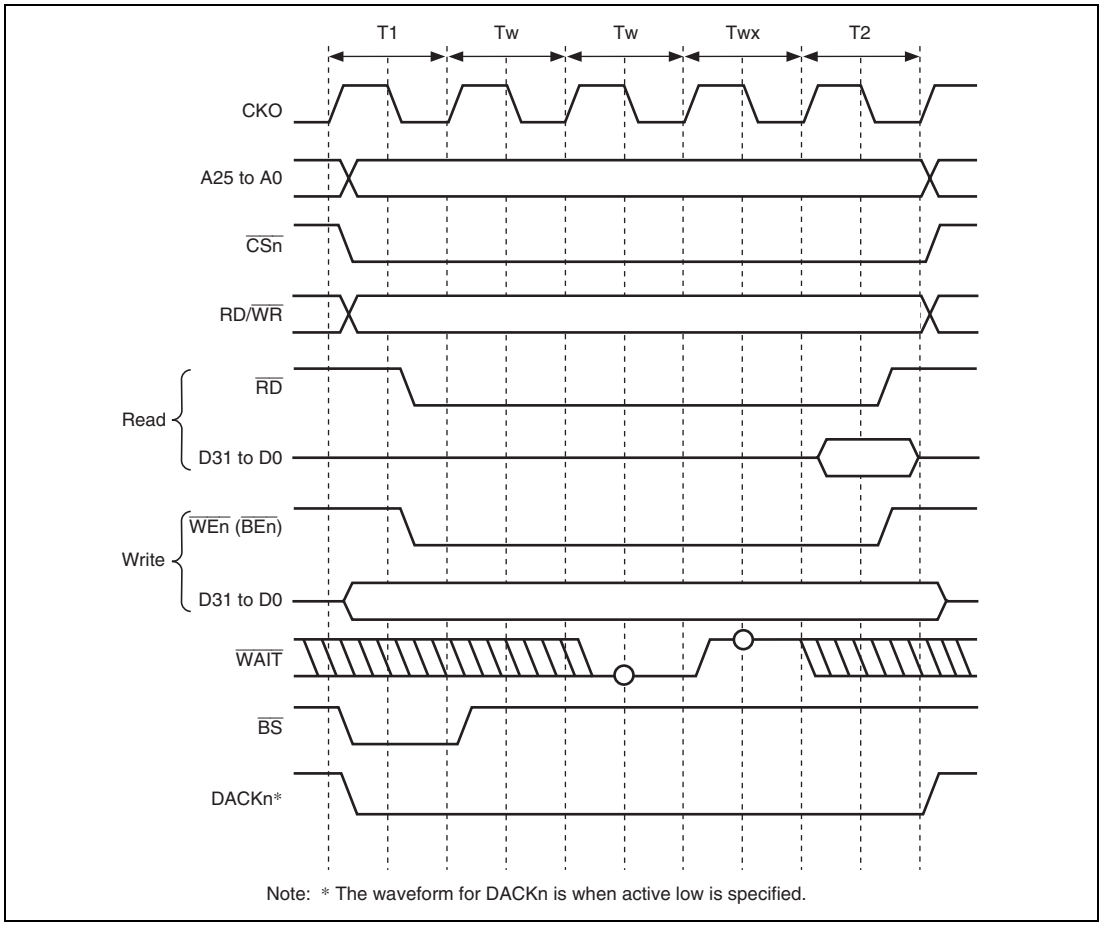


Figure 14.9 Wait Cycle Timing for Normal Space Access (Wait Cycle Insertion Using $\overline{\text{WAIT}}$ Signal)

14.5.4 $\overline{\text{CSn}}$ Assert Period Extension

The number of cycles from $\overline{\text{CSn}}$ assertion to $\overline{\text{RD}}$, $\overline{\text{WEn}}$ ($\overline{\text{BEn}}$) assertion can be specified by setting the SW[1:0] bits in CSnWCR. The number of cycles from $\overline{\text{RD}}$, $\overline{\text{WEn}}$ ($\overline{\text{BEn}}$) negation to $\overline{\text{CSn}}$ negation can be specified by setting the HW[1:0] bits. Therefore, a flexible interface to an external device can be obtained. Figure 14.10 shows an example. A Th cycle and a Tf cycle are added before and after an ordinary cycle, respectively. In these cycles, $\overline{\text{RD}}$ and $\overline{\text{WEn}}$ ($\overline{\text{BEn}}$) are not asserted, while other signals are asserted. The data output is prolonged to the Tf cycle, and this prolongation is useful for devices with slow write operations.

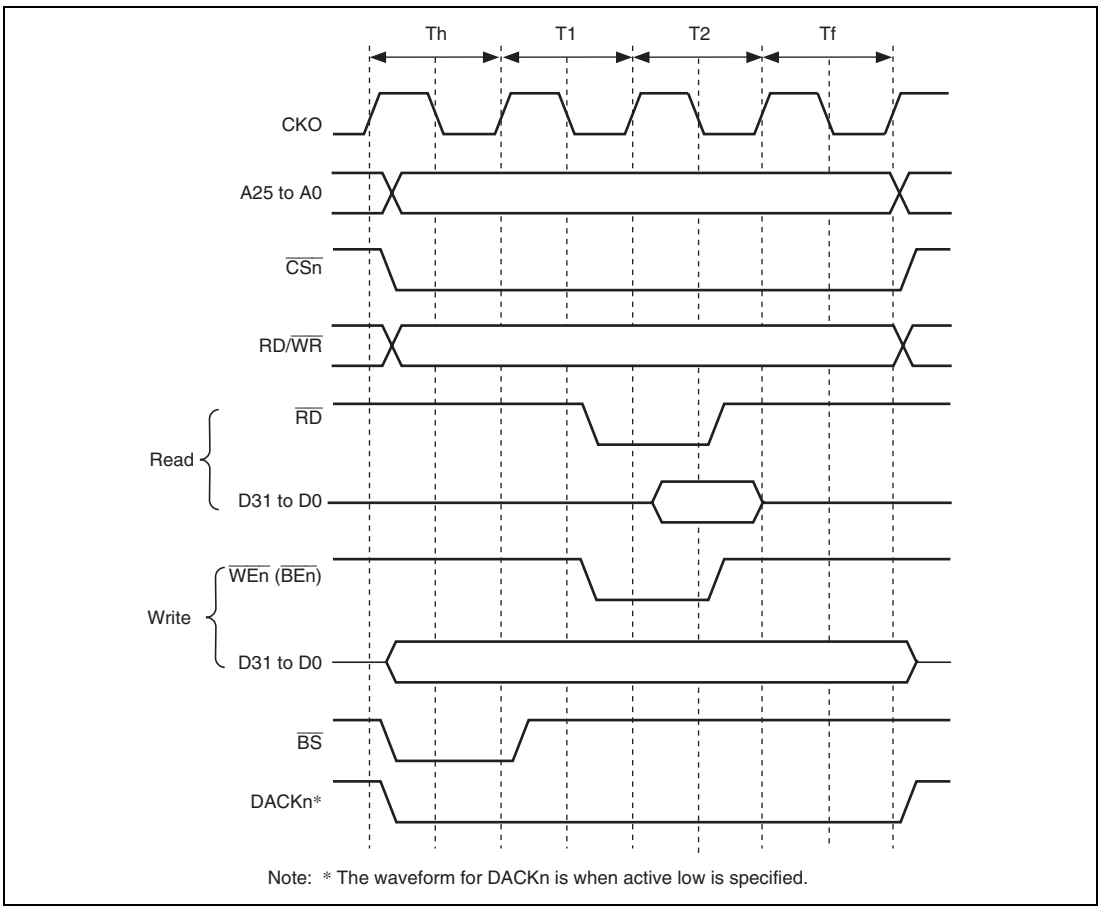


Figure 14.10 $\overline{\text{CSn}}$ Assert Period Extension

14.5.5 Burst ROM Interface

When the CS0 space is being used to connect burst ROM, set the CS0WCR bits correctly so that subsequently access will be generated as burst access.

(1) Burst ROM Interface

The burst ROM (clock asynchronous) interface is used to access a memory with a high-speed read function using a method of address switching called burst mode or page mode. In the burst ROM (clock asynchronous) interface, basically the same access as the normal space is performed, but the 2nd and subsequent accesses are performed by changing only the address, without negating the $\overline{RD}$ signal at the end of the first cycle. In the second and subsequent accesses, addresses are changed at the falling edge of CKO.

For the first access cycle, the number of wait cycles specified by the W[3:0] bits in CSnWCR is inserted. For the second and subsequent access cycles, the number of wait cycles specified by the BW[1:0] bits in CSnWCR is inserted.

In the burst ROM (clock asynchronous) access, $\overline{BS}$ signal is asserted only for the first access cycle. An external wait input is valid only in the first access cycle.

In the single access or write access that does not perform the burst operation in the burst ROM (clock asynchronous) interface, access timing is the same as a normal space.

Table 14.14 lists a relationship between the bus width, access size, and number of bursts. Figure 14.11 shows a timing chart.

Table 14.14 Relationship between Bus Width, Access Size, and Number of Bursts

Bus Width	Access Size	Number of Bursts	Number of Accesses
8 bits	8 bits	1	1
	16 bits	2	1
	32 bits	4	1
	16 bytes	16	1
	32 bytes	16	2
16 bits	8 bits	1	1
	16 bits	1	1
	32 bits	2	1
	16 bytes	8	1
	32 bytes	8	2
32 bits	8 bits	1	1
	16 bits	1	1
	32 bits	1	1
	16 bytes	4	1
	32 bytes	4	2

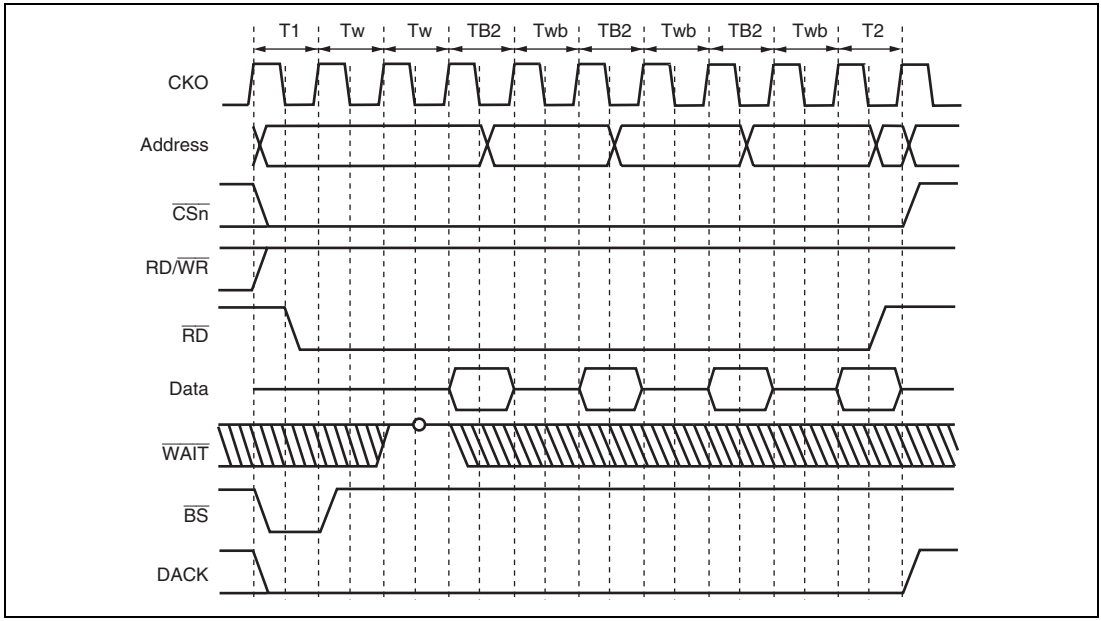


Figure 14.11 Burst ROM (Clock Asynchronous) Access (Bus Width = 32 Bits, 16-byte Transfer (Number of Bursts = 4), Access Wait for 1st Time = 2, Access Wait for 2nd Time and After = 1)

(2) Clock Synchronous Burst Mode

This interface also supports connection to memory that has a clock-synchronous burst mode. If memory of this type is to be connected, connect the BS signal of this LSI chip to the address-valid bit of the memory device.

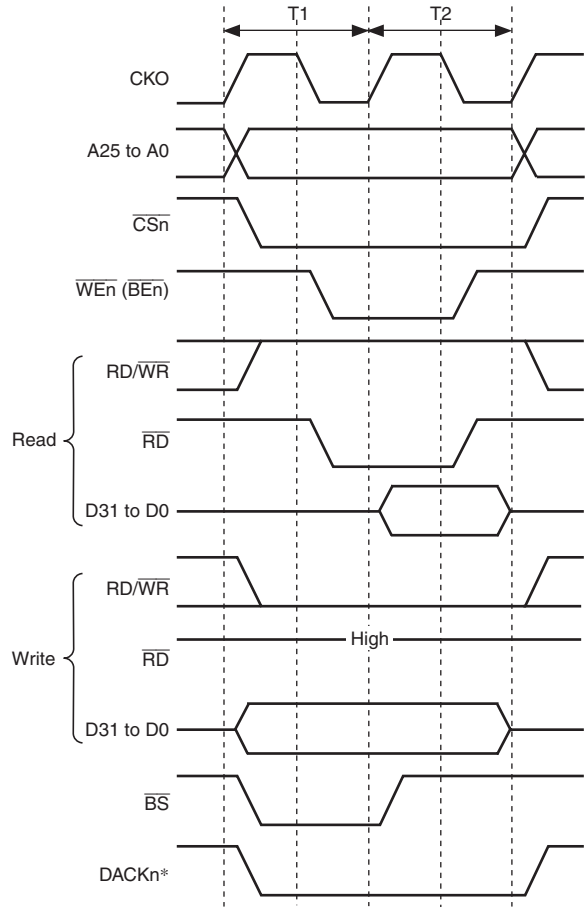
To secure the hold time for supply of the clock signal to the memory, set the BROMM0 bit in the common control register (CMNCR) to 1. Furthermore, to secure the read-data setup time for this chip, set the PH1RD bit in the CSn bus control register (CSnBCR) to 1. Memory settings should be for a burst length of eight with wrap-around.

14.5.6 SRAM Interface with Byte Selection

The SRAM interface with byte selection is a memory interface which outputs a byte-selection pin ($\overline{\text{WEn}}$ ($\overline{\text{BEn}}$)) in a read/write bus cycle. This interface has 16-bit data pins and accesses SRAMs having upper and lower byte-selection pins, such as UB and LB.

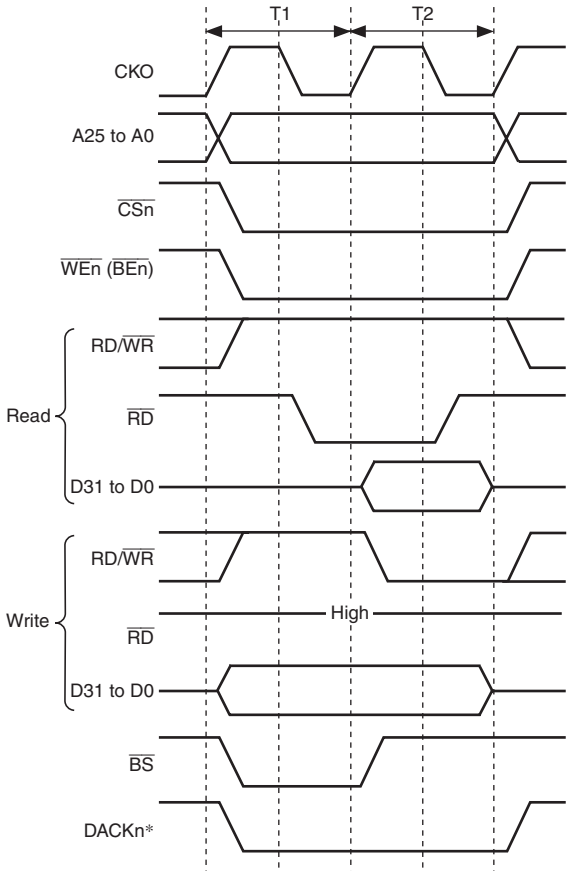
When the BAS bit in CSnWCR is cleared to 0 (initial value), the write access timing of the SRAM interface with byte selection is the same as that for the normal space interface. While in read access of the SRAM interface with byte selection, the byte-selection signal is output from the $\overline{\text{WEn}}$ ($\overline{\text{BEn}}$) pin, which is different from that for the normal space interface. The basic access timing is shown in Figure 14.12. In write access, data is written to memory according to the timing of the byte-selection pin ($\overline{\text{WEn}}$ ($\overline{\text{BEn}}$)). For details, refer to the data sheet for the corresponding memory.

If the BAS bit in CSnWCR is set to 1, the $\overline{\text{WEn}}$ ($\overline{\text{BEn}}$) pin and RDWR pin timings change. Figure 14.12 shows the basic access timing. In write access, data is written to memory according to the timing of the write enable pin (RDWR). Secure the holding time from the RDWR negation to the data write by setting bits HW[1:0] of CSnWCR. Figure 14.13 shows the access timing for the software wait setting.



Note: * The waveform for DACKn is when active low is specified.

Figure 14.12 Basic Access Timing for SRAM with Byte Selection (BAS = 0)



Note: * The waveform for DACKn is when active low is specified.

Figure 14.13 Basic Access Timing for SRAM with Byte Selection (BAS = 1)

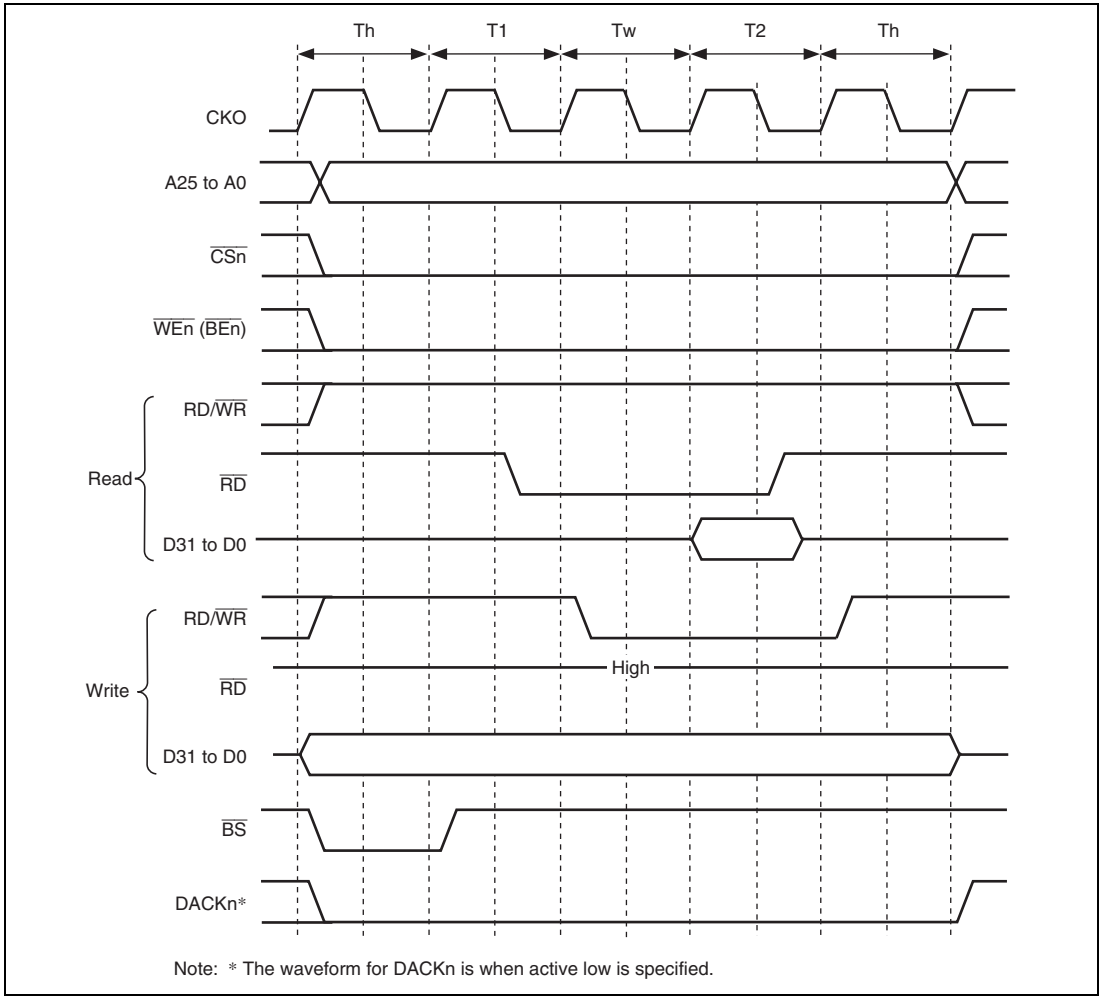


Figure 14.14 Wait Timing for SRAM with Byte Selection (BAS = 1) (for Software Waiting only)

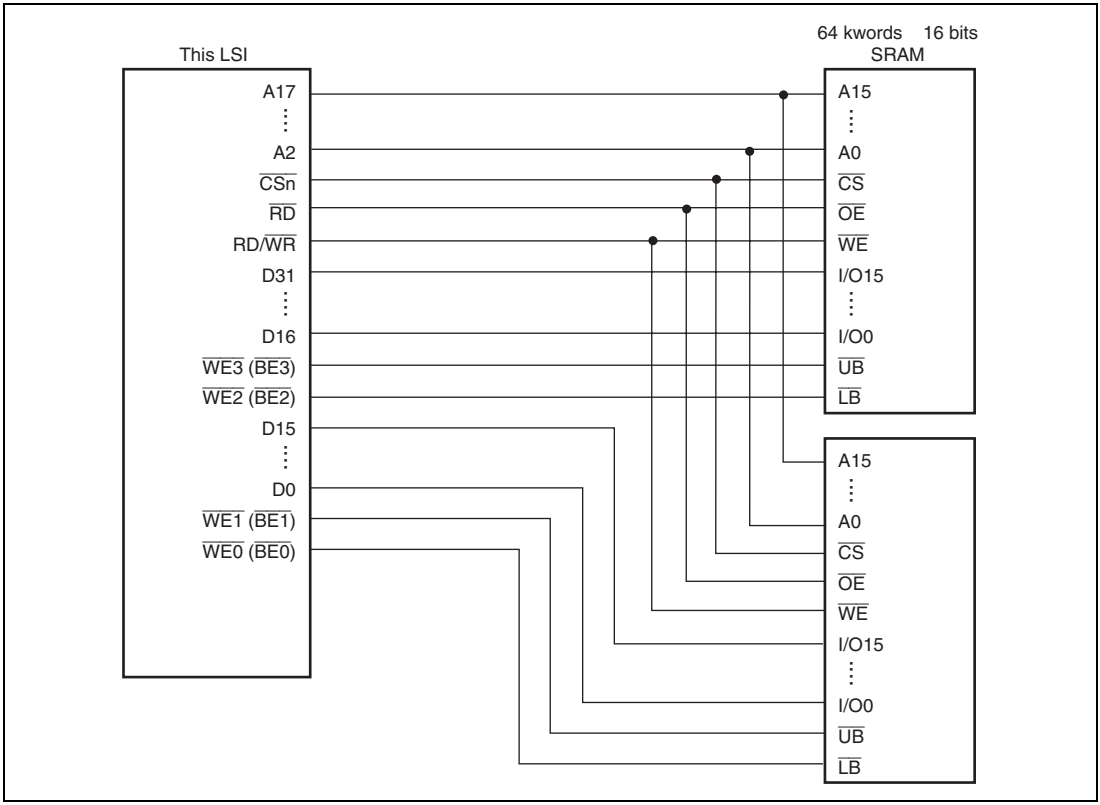


Figure 14.15 Example of Connection with SRAM with Byte Selection (32-Bit Data Bus)

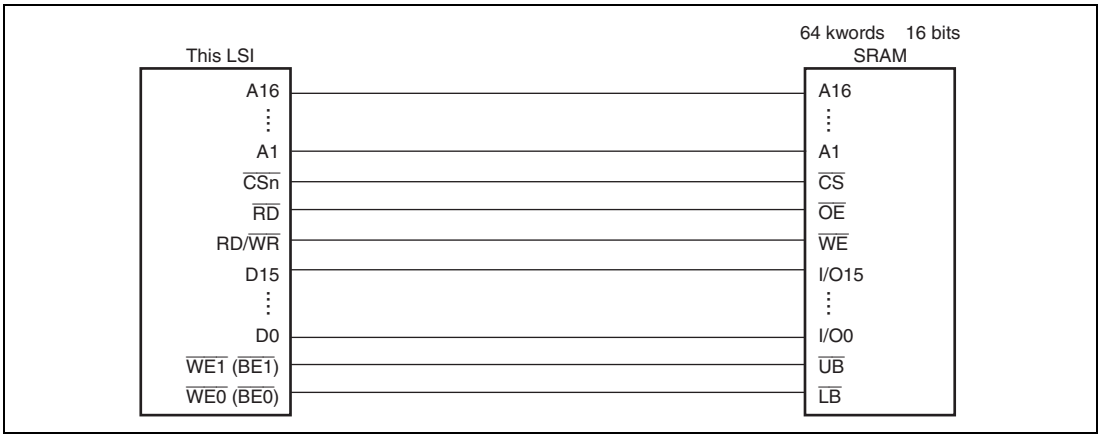


Figure 14.16 Example of Connection with SRAM with Byte Selection (16-Bit Data Bus)

14.5.7 PCMCIA Interface

This LSI can set the PCMCIA interface in area 5 and area 6, when address map 2 is selected by the MAP bit in CMNCR. When bits TYPE[3:0] of CSnBCR (n = 5B, 6B) are set to B'0101, area 5 and area 6 of the physical space operate as "IC memory card and I/O card interface" provided with JEIDA Ver4.2 (PCMCIA2.1). Moreover, with the settings by bits SA[1:0] of CSnWCR (n = 5B, 6B), the first-half space (32 Mbytes) of each area and the second-half space (32 Mbytes) are independently set for the IC memory card or I/O card interface. For instance, when the SA1 bit in CS5BWCR is set to 1 and the SA0 bit in CS5BWCR is cleared to 0, the first-half space (32 Mbytes) operate as IC memory card interface and the second-half space (32 Mbytes) as the I/O card interface.

When the PCMCIA interface is used, make sure to set the bus size to 8 or 16 bits by bits BSZ[1:0] of CS5BBCR or bits BSZ[1:0] of CS6BBCR.

Figure 14.17 shows a connection example with the PCMCIA card and this LSI. To enable the activated insertion of the PCMCIA card (that is, to connect or disconnect the card while the power is supplied to the system), three-state buffer should be connected between the bus interface of this LSI and the PCMCIA card.

Since the JEIDA or PCMCIA does not clearly specify the operation of PCMCIA interface in big endian mode, this LSI specifies it originally.

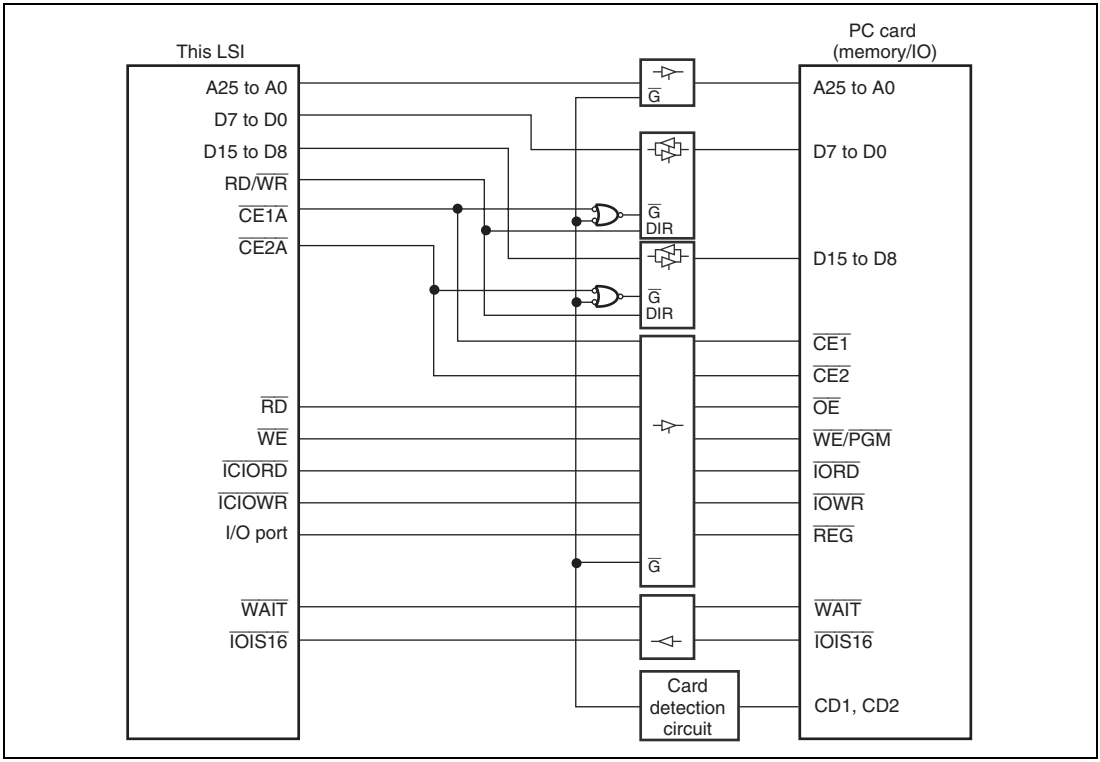


Figure 14.17 Connection Example with PCMCIA Interface

(1) Basic Timing of Memory Card Interface

Figure 14.18 shows the basic timing of "IC memory card interface" of PCMCIA. When area 5 and area 6 of the physical space are set as the PCMCIA interface, access to the common memory space of each area is automatically handled as the access to the "IC memory card interface". If the external bus frequency (CKO) rises, the time for setting and holding will not be secured enough for address signals (A25 to A0) for $\overline{RD}$ and $\overline{WE}$, card enable signals ($\overline{CE1A}$, $\overline{CE2A}$, $\overline{CE1B}$, and $\overline{CE2B}$), and write data signals (D15 to D0) in write cycles. To prevent this, the setting or holding time can be set independently in this LSI for each area in area 5 and area 6 of the physical space, by the CS5BWCR or CS6BWCR register. In addition, as the same in the normal space interface, the software waiting or hardware waiting with the $\overline{WAIT}$ pin is enabled. Figure 14.19 shows the wait timing of PCMCIA memory bus.

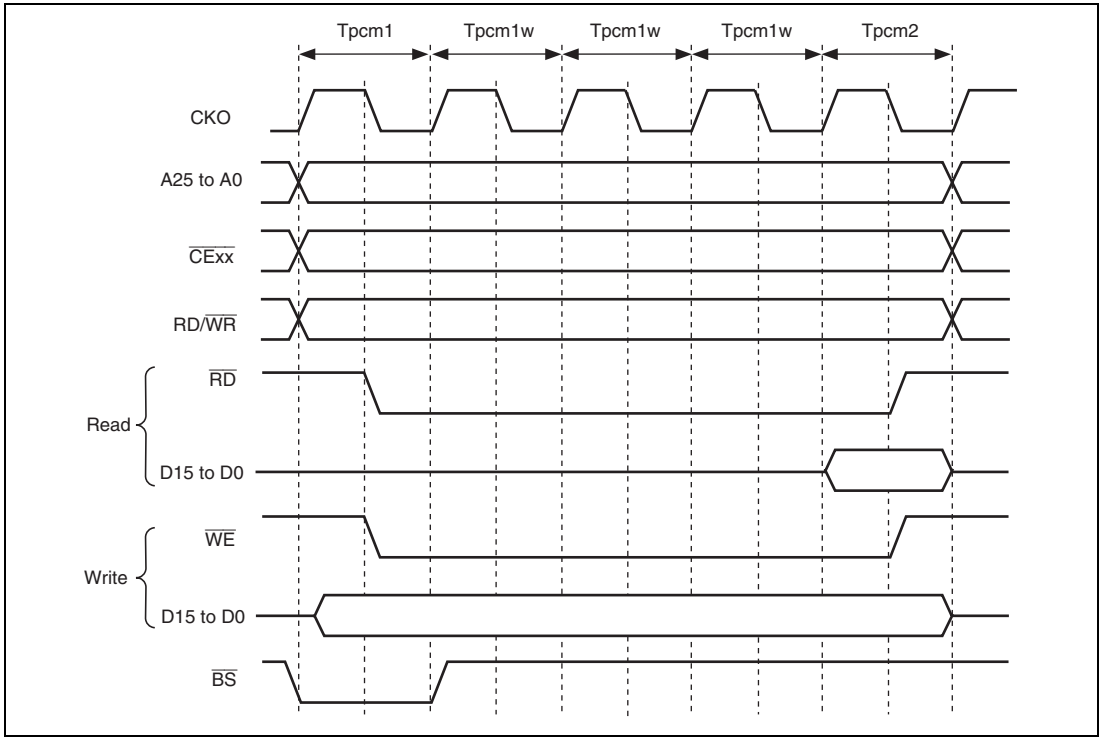


Figure 14.18 Basic Timing of PCMCIA Memory Card Interface

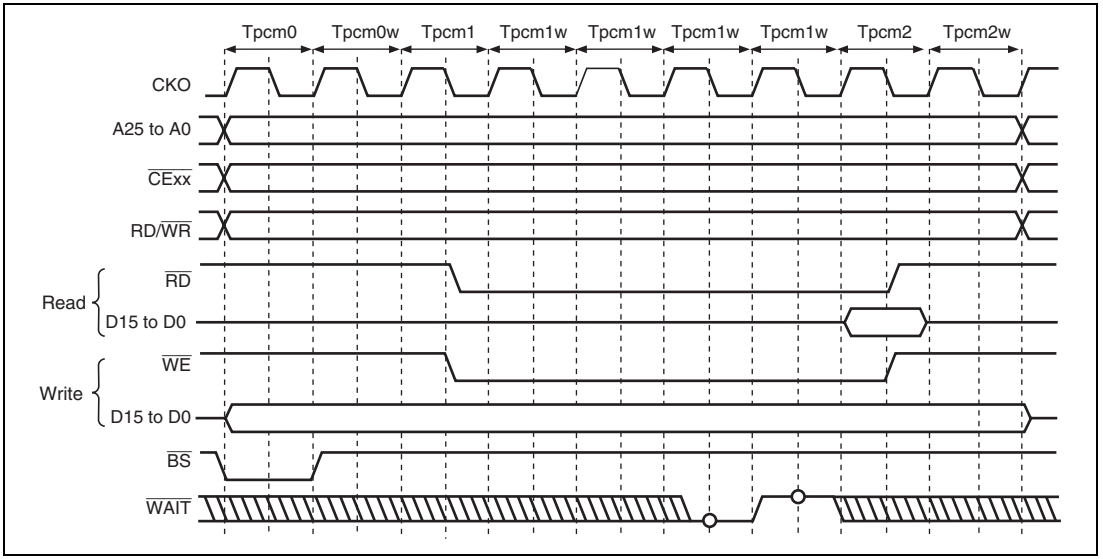


Figure 14.19 Wait Timing of PCMCIA Memory Card Interface (TED[3:0] = B'0010, TEH[3:0] = B'0001, One Software Wait, One Hardware Wait)

When all memory space of 32 Mbytes is used as the IC memory card interface, the $\overline{\text{REG}}$ signal to switch between common memory and attribute memory is provided by a port. In addition, when the memory space of 16 Mbytes or less is enough for use, the memory space is divided into two 16-Mbyte areas. Each is used as common memory or attribute memory, and thus the A24 pin is used for the $\overline{\text{REG}}$ signal.

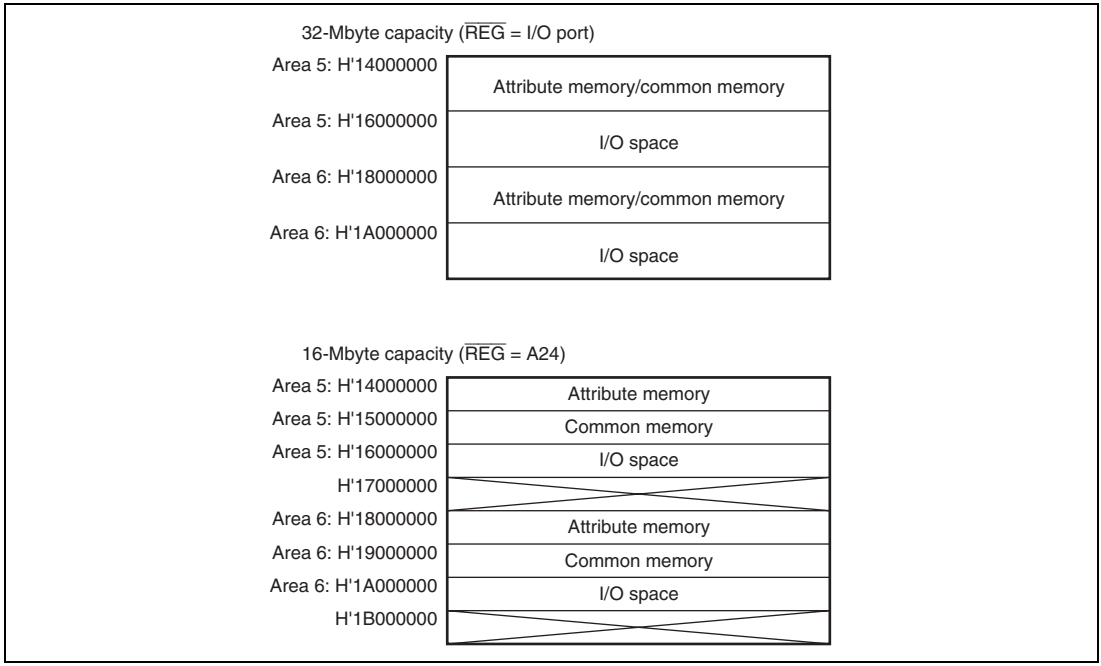


Figure 14.20 Allocation Example of PCMCIA Space (CS5BWCR.SA[1:0] = B'10, CS6BWCR. SA[1:0] = B'10)

(2) Basic Timing of I/O Card Interface

Figures 14.21 and 14.22 show the basic timing of " I/O card interface" of PCMCIA.

The address accessed switches between the I/O card interface and the IC memory card interface. When area 5 of the physical space is set as the PCMCIA interface, access to the physical address of H'16000000 to H'17FFFFFF is automatically handled as the access to the "I/O card interface". And when area 6 of the physical space is set as the PCMCIA interface, access to the physical address of H'1A000000 to H'1BFFFFFF is automatically handled as the access to the "I/O card interface".

When the I/O card of PCMCIA is accessed, use the cache not-targeted area in the logical space (P2 or P3 space) or the area specified as the cache not-targeted area by the MMU.

In little endian mode, if the PCMCIA card is accessed as the I/O card interface, dynamic bus sizing of the I/O bus width is enabled with the $\overline{\text{IOIS16}}$ pin. When the $\overline{\text{IOIS16}}$ signal is driven high in I/O bus cycles of word units, while 16 bits are specified as the bus width of area 6, the bus width is recognized as 8 bits. The data of 8 bits is only accessed in the current I/O bus cycle, and remaining data of 8 bits is automatically accessed after the previous access.

The $\overline{\text{IOIS16}}$ signal is sampled at the falling edge of CKO in the Tpci0, Tpci0w, and Tpci1 cycles where bits TED[3:0] specify 1.5 cycles or more. When 1.5 cycles of CKO have elapsed after the sampling point, the state is reflected to the $\overline{\text{CE2}}$ signal. Set bits TED[3:0] so as to satisfy the $\overline{\text{CEn}}$ setup specification, according to the $\overline{\text{ICIORD}}$ and $\overline{\text{ICIOWR}}$ signals of the PC card for use.

Figure 14.21 shows the basic timing of dynamic bus sizing.

In big endian mode, the $\overline{\text{IOIS16}}$ signal is not supported. Fix the $\overline{\text{IOIS16}}$ signal low in big endian mode.

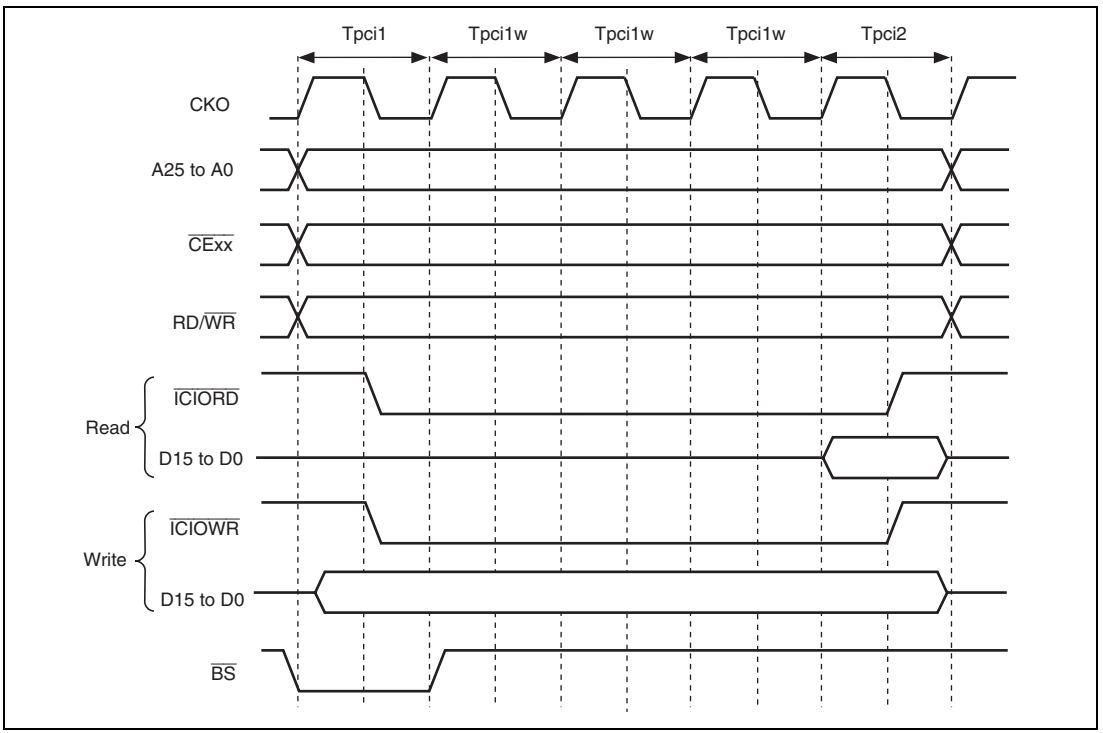


Figure 14.21 Basic Timing of PCMCIA I/O Card Interface

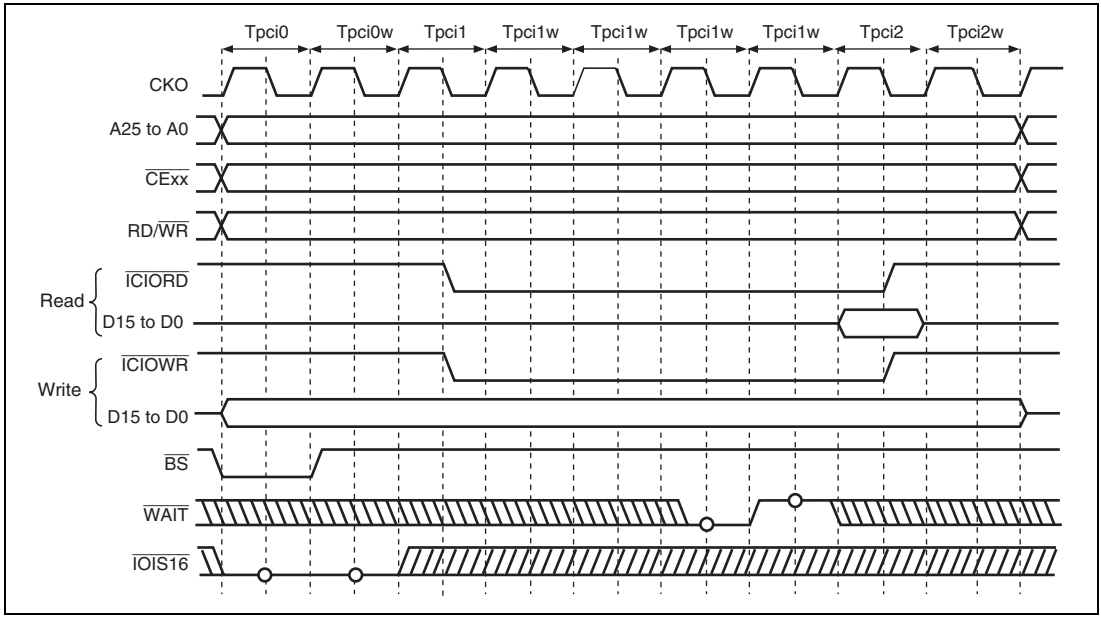


Figure 14.22 Wait Timing of PCMCIA I/O Card Interface (TED[3:0] = B'0010, TEH[3:0] = B'0001, One Software Wait, One Hardware Wait)

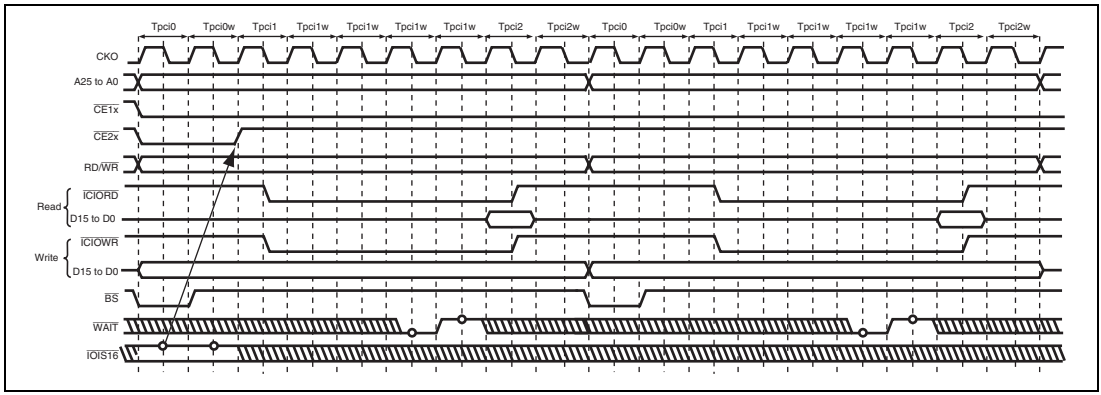


Figure 14.23 Dynamic Bus Sizing Timing of PCMCIA I/O Card Interface (TED[3:0] = B'0010, TEH[3:0] = B'0001, Software Wait = 3)

14.5.8 Wait between Access Cycles

As the operating frequency of LSIs becomes higher, the off-operation of the data buffer often collides with the next data output when the data output from devices with slow access speed is completed. As a result of these collisions, the reliability of the device is low and malfunctions may occur. A function that avoids data collisions by inserting wait cycles between continuous access cycles has been newly added.

The number of wait cycles between access cycles can be set by bits IWW[2:0], IWRWD[2:0], IWRWS[2:0], IWRRD[2:0], and IWRRS[2:0] in CSnBCR. The conditions for inserting the wait cycles (idle cycles) between access cycles are shown below.

1. Continuous accesses are write-read or write-write
2. Continuous accesses are read-write for different spaces
3. Continuous accesses are read-write for the same space
4. Continuous accesses are read-read for different spaces
5. Continuous accesses are read-read for the same space

14.6 Others

The BSC can be initialized completely only by a power-on reset. At a power-on reset, all signals are negated and output buffers are turned off or output a fixed value immediately regardless of the bus cycle state. All control registers are initialized. In standby mode or sleep mode, control registers of the BSC are not initialized.

Some flash memories may specify a minimum time from reset cancel to the first access. To ensure this minimum time, the BSC supports a 7-bit counter (RBWTCNT). At a power-on reset, RBWTCNT is cleared to 0. After clearing a power-on reset, RBWTCNT is counted up synchronized with CKO and an external access will not be generated until RBWTCNT is counted up to H'007F. At a manual reset, RBWTCNT is not cleared.

Section 15 DDR-SDRAM Bus State Controller (DBSC)

The DDR-SDRAM bus state controller (DBSC) supports the SDRAMs including DDR2-SDRAM (referred to as DDR2, hereinafter) or mobile-DDR SDRAM (referred to as mobile-DDR2, hereinafter). The DBSC performs read-accesses and write-accesses to the SDRAM and refreshes the SDRAM.

15.1 Features

The DBSC enables the maximum use of SDRAM bus bandwidth using the following functions:

- Multibank operation that improves the page hit rate.
- SDRAM operation with burst length 4 that precedes execution of the bank precharge and activate commands in empty cycles between read and write commands.
- Supports 16-bit or 32-bit external bus width

Tables 15.1 and 15.2 show the main functions of the DBSC.

Table 15.1 Features of DBSC

Item	Function
Multi-bank operation	Supports 4-bank multibank operation (maximum of 4 banks can be opened at the same time when an 8-bank configuration is used.)
External bus width	16/32 bits
Preceding PRE/ACT command	Determines the content of subsequent requests in a request queue and performs preceding precharge and activate processing for the bank to be accessed during an empty command cycle upon page-miss.
Timing setting	The following timing settings can be specified: Minimum ACT-PRE period, minimum REF-ACT/REF period, minimum ACT-READ/WRITE period, PRE period, minimum ACT(A)-ACT(B) period, write recovery period, minimum ACT-ACT/REF period, minimum READ-WRITE period, minimum WRITE-READ period, minimum READ-PRE period, and minimum CKEH-Command period Only 0 is supported for additive latency (AL). Only 3 is supported for CAS latency (CL).
Address order	The order is Row-Bank-Column. Register setting can change the position of bank address.
Standard of SDRAM	DDR2 JESD79-2A LPDDR JESD209
Auto refresh	The average interval of auto refresh can be set by register. Preceding refresh during cycles empty of requests

Table 15.2 Features of DBSC (DDR2/Mobile-DDR-SDRAM function)

Item			DDR2	Mobile-DDR*[1]
Burst	Burst length		4	4
	Burst type		Sequential	Sequential
	Burst cancel		—	√*[2]
CL	CAS latency		3	3
Bank	Number of banks		4 or 8 banks*[3]	4 banks
Power-down	Power-down		√	√
	Deep power-down		—	√
Self-refresh	Self-refresh		√	√
	Partial self-refresh		—	√
Available memory	32-bit bus	One 64M × 32 bits	—	√
		One 32M × 32 bits	—	√
		One 16M × 32 bits	√	√
		One 8M × 32 bits	—	√
		One 4M × 32 bits	—	√
		Two 128M × 16 bits	√	—
		Two 64M × 16 bits	√	√
		Two 32M × 16 bits	√	√
		Two 16M × 16 bits	√	√
		Two 8M × 16 bits	—	√
	16-bit bus	One 128M × 16 bits	√	√
		One 64M × 16 bits	√	√
		One 32M × 16 bits	√	√
		One 16M × 16 bits	√	√
		One 8M × 16 bits	—	√
		Two 128M × 8 bits	√	—
		Two 64M × 8 bits	√	—
		Two 32M × 8 bits	√	—

Notes: Although this module supports connection to multiple SDRAMs, some chips may not provide multiple SDRAM connection due to the electrical characteristics.

1. The DBSC supports mobile-DDR-compatible products from Elpida, Samsung, and Micron.

2. The DBSC supports burst cancellation between Read and Read and between Write and Write commands.

   It does not support burst cancellation between Read and Write or Write and Read commands, nor does it support burst stopping.

3. The specification of operations when an 8-bank configuration is used conforms with the JEDEC standard for DDR2 (JESD79-2A).

Figure 15.1 shows a block diagram of the DBSC.

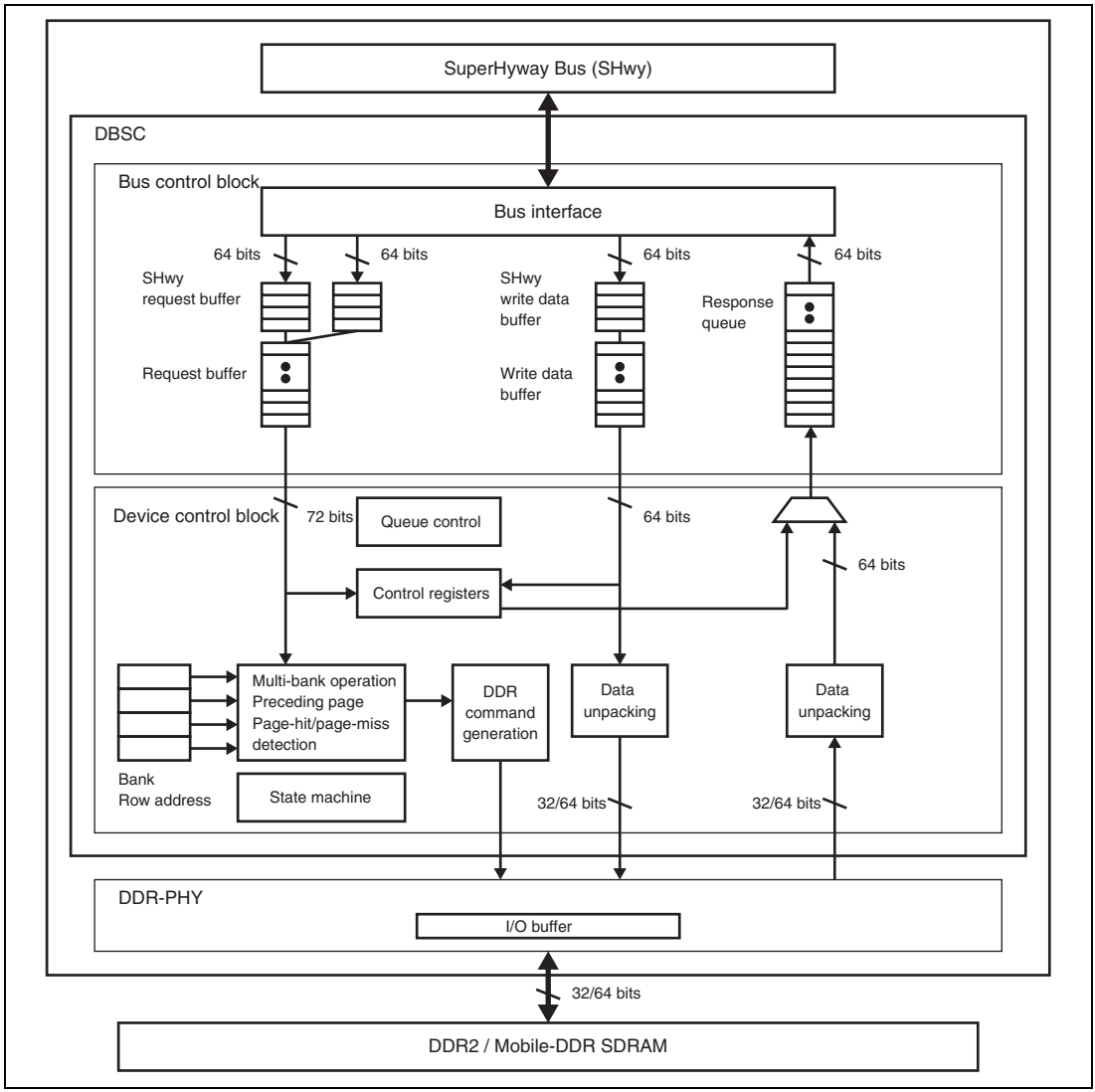


Figure 15.1 Block Diagram of the DBSC

15.2 Input/Output Pins

Table 15.3 shows the pin configuration of the DBSC.

Table 15.3 Pin Configuration of the DBSC

Pin Name	Function	I/O	Description
MCLK	Clock	Output	Clock output for the DDR2-SDRAM/mobile-DDR
$\overline{\text{MCLK}}$	Clock	Output	Clock output for the DDR2-SDRAM/mobile-DDR MCLK inverted clock output
MCKE	Clock enable	Output	CKE output signal
$\overline{\text{MCS}}$	Chip select	Output	Chip select output signal
$\overline{\text{MWE}}$	Write enable	Output	Write enable output signal
$\overline{\text{MRAS}}$	Row address strobe	Output	Row address strobe output signal
$\overline{\text{MCAS}}$	Column address strobe	Output	Column address strobe output signal
MA13 to MA0	Addresses	Output	Address output signals
MBA2, MBA1, MBA0	Bank active	Output	Bank address output signal
MDQ31 to MDQ0	Data	I/O	Data I/O signals
MDQS3 to MDQS0	I/O data strobe	I/O	Data strobe I/O signals
$\overline{\text{MDQS3}}$ to $\overline{\text{MDQS0}}$	I/O data strobe	I/O	Data strobe I/O signals or MDQS3 to MDQS0 inverted signals
MDQM3 to MDQM0	Data mask	Output	Data mask output signals
MSLD	Memory selection	Input	Memory type selection Fix this pin to low or high when the DDR2-SDRAM or Mobile-DDR-SDRAM is used, respectively.
MODT	ODT enable	Output	ODT enable output signal to the SDRAM
MVREF1, MVREF0	Reference voltage input	Input	Reference voltage got the DDR2-SDRAM This pin should be open when the Mobile-DDR-SDRAM is used.

15.3 Register Descriptions

Table 15.4 shows the DBSC register configuration and table 15.5 shows register states in each processing mode of operation.

The register bit width is 32 bits, and access to registers must be made in longword units (32 bits). Operation cannot be guaranteed if access to registers in units other than longword is attempted.

Table 15.4 DBSC Register Configuration

Register Name	Abbreviation	R/W	Area 7 Address	Access Size (Bits)
DBSC SDRAM kind select register	DBKIND	R/W	H'FD00 0008	32
DBSC status register	DBSTATE	R	H'FD00 000C	32
SDRAM operation enable register	DBEN	R/W	H'FD00 0010	32
SDRAM command control register	DBCMDCNT	R/W	H'FD00 0014	32
SDRAM CKE control register	DBCKECNT	R/W	H'FD00 0018	32
SDRAM configuration setting register	DBCONF	R/W	H'FD00 0020	32
SDRAM timing register 0	DBTR0	R/W	H'FD00 0030	32
SDRAM timing register 1	DBTR1	R/W	H'FD00 0034	32
SDRAM timing register 2	DBTR2	R/W	H'FD00 0038	32
SDRAM timing register 3	DBTR3	R/W	H'FD00 003C	32
SDRAM refresh/power-down control register 0	DBRFPDN0	R/W	H'FD00 0040	32
SDRAM refresh/power-down control register 1	DBRFPDN1	R/W	H'FD00 0044	32
SDRAM refresh/power-down control register 2	DBRFPDN2	R/W	H'FD00 0048	32
SDRAM refresh/power-down status register	DBRFSTS	R/W	H'FD00 004C	32
SDRAM mode setting register	DBMRCNT	W	H'FD00 0060	32
DDR-PHY control register	DBPDCNT0	R/W	H'FD00 0108	32

Table 15.5 Register Status in each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
DBKIND	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBSTATE	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBEN	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBCMDCNT	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBCKECNT	Initialized	Initialized	Retained	—	Retained	Initialized	Retained
DBCONF	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBTR0	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBTR1	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBTR2	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBTR3	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBRFPDN0	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBRFPDN1	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBRFPDN2	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBRFSTS	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBMRCNT	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained
DBPDCNT0	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained

15.3.1 DBSC SDRAM kind select register (DBKIND)

The SDRAM kind setting register specifies a kind of SDRAM. Set the designated value according to the type.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DDCG[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0.
2 to 0	DDCG[2:0]	All 0	R/W	DDCG (SDRAM kind select) bit 000 to 011: Setting prohibited 100: Mobile-DDR-SDRAM 101: DDR2-SDRAM 110 to 111: Setting prohibited

15.3.2 DBSC Status Register (DBSTATE)

The DBSC status register (DBSTATE) is a read-only register. Writing is invalid.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENDN	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ENDN	0	R	Endian Indication Indicates the current endian type. The set value is specific to the product type. 0: Big endian 1: Little endian
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

15.3.3 SDRAM Operation Enable Register (DBEN)

The SDRAM operation enable register (DBEN) is a readable/writable register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ACEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation when a value other than 0 is written is not guaranteed.
0	ACEN	0	R/W	SDRAM Access Enable Bit By setting this bit, data accessing of SDRAM is enabled. When set to 0, access is disabled; when set to 1, access is enabled. When access is disabled, DBSC return an error response. 0: Disables access 1: Enables access

15.3.4 SDRAM Command Control Register (DBCMDCNT)

The SDRAM command control register (DBCMDCNT) is a readable/writable register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CMD[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	CMD[2:0]	000	R/W	SDRAM Command Issue Bit These bits are used to issue commands necessary to execute the SDRAM initialization sequence and self-refresh transition/cancellation. When these bits are written, the command corresponding to the written value is issued once. For example, in order to issue the auto-refresh command twice, it would be necessary to write 100 to these bits twice. The precharge interval, minimum interval between auto-refresh and the next command, and other intervals are values set in the SDRAM timing register, described below. When read, these bits are always read as 000. 000: Normal operation (power-on reset) 001: Setting prohibited (Correct operation cannot be guaranteed.) 010: Precharge (PALL) command issued 011:. Setting prohibited (Correct operation cannot be guaranteed.) 100: Auto-refresh (REF) command issued 101 to 111: Setting prohibited (Correct operation cannot be guaranteed.)

Note: This register can be written only when automatic issue of auto-refresh is disabled (the ARFEN bit in the DBRFPDN0 register is cleared to 0).

15.3.5 SDRAM CKE Control Register (DBCKECNT)

The SDRAM CKE control register (DBCKECNT) is a readable/writable register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CKE EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation when a value other than 0 is written is not guaranteed.
0	CKEEN	0	R/W	CKE Setting Bit This bit specifies the value of the MCKE pin when the initialization of SDRAM. The initial value of MCKE varies depending on whether the memory is mobile-DDR-SDRAM or DDR2-SDRAM. Set this bit during the initialization sequence. When set to 0 (at power-on reset), the value of the MCKE initialization signal which is input to the DBSC from the MSLD pin is output to the MCKE signal which is output to DDR-PAD area from the DBSC. When set to 1, the CKE value is output to the MCKE signal in DDR-PAD area from DBSC. The value of MCKE generated from the internal DBSC is H in power-on reset, and it changes according to the state of LSI, self-refresh, power-down and deep-power-down mode. 0: The input value to the MSLD pin is output from the MCKE pin. 1: The MCKE pin is controlled by the DBSC.

Note: This register can be written only when all of these conditions are satisfied:

- SDRAM access is disabled (the ACEN bit in DBEN is cleared to 0).
- Automatic issue of auto-refresh is disabled (the ARFEN bit in the DBRFPDN0 register is cleared to 0).

15.3.6 SDRAM Configuration Setting Register (DBCONF)

The SDRAM configuration setting register (DBCONF) is a readable/writable register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SPLIT[8:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BKADM[1:0]		BKADP[5:0]						BKADB5:0]						BWIDTH[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation when a value other than 0 is written is not guaranteed.
24 to 16	SPLIT[8:0]	All 0	R/W	Memory Configuration Selection These bits select the memory configuration to be used. For the relation between the SDRAM address pins and LSI logical addresses, see section 15.5.9, Relation between SDRAM Address Pins and Logical Addresses. The following shows the value meanings and equations when the SPLIT bit is xx_y_ppp_qqq: x indicates the bit width of memory. When xx = B'10, the memory bit width is 32 bits. y indicates 4 or 8 banks. When y = 0, 4-bank products are supported. When y = 1, 8-bank products are supported. p indicates the row address width. Row address width = B'1010 + B'0ppp q indicates the column address width. Column address width = B'0111 + B'0qqq xx_y_ppp_qqq specifies the bit, bank (bit), row (bit), and column (bit).

Bit	Bit Name	Initial Value	R/W	Description
15 to 14	BKADM[1:0]	00	R/W	Bank Address Usage These bits set whether bank addresses are treated as contiguous addresses or non-contiguous addresses. When they are treated as contiguous addresses, set the bank address location in BKADP. When they are treated as non-contiguous addresses, BA0 and BA1 can be treated as different addresses. Set BA0 in BKADP and BA1 in BKADB. For 8-bank products, BA2 is treated as a contiguous address with BA1 and upper address of BA1. 00: Contiguous addresses (BA0, BA1 and BA2 are set in BKADP.) 01: Non-contiguous addresses (lower one bit) (BA0 is set in BKADP and BA1 and BA2 in BKADB.) 10: Setting prohibited 11: Setting prohibited
13 to 8	BKADP[5:0]	All 0	R/W	Address Location of Bank Address These bits specify the bank address location appropriate for the used memory configuration. Specifically, either the upper bits of the column address or the desired address location can be specified as follows. For details of treating the upper bits of the column address as a bank address, see section 15.5.9, Relation between SDRAM Address Pins and Logical Addresses. Note that the upper bits of the CAS can be specified only when BKADM = B'00 (contiguous addresses). 000000: Upper level of column address (CAS) 001010: 1 Kbytes (Set address 10 to BA0) 001011: 2 Kbytes (Set address 11 to BA0) 001100: 4 Kbytes (Set address 12 to BA0) Others: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	BKADB[5:0]	All 0	R/W	Address Location of Upper Bank Address (Valid when BKADM = B'01.) These bits specify the address location of the upper bank address. These bits are valid when the BKADM bit is set so that the bank addresses should be treated as non-contiguous addresses. The BA1 address location is set as the upper bank address. For 8-bank products, BA2 is treated as the upper address of BA1.When BKADP = B'000000, these bits are used as the upper address of BA0 and cannot be set. 000000: No setting 001101: Set address 13 to BA1 (Set address 14 to BA2) 001110: Set address 14 to BA1 (Set address 15 to BA2) 001111: Set address 15 to BA1 (Set address 16 to BA2) 010000: Set address 16 to BA1 (Set address 17 to BA2) Others: Setting prohibited
1, 0	BWIDTH[1:0]	00	R/W	SDRAM Bus Width Setting These bits set the external data bus width. 00: Setting prohibited 01: 16 bits 10: 32 bits 11: Setting prohibited

Note: 1. Supported memory configuration

(1) DDR2-SDRAM

- 16-bit bus configuration in which one 16-bit width SDRAM is connected, or two 8-bit SDRAMs are connected in parallel.
- 32-bit bus configuration in which one 32-bit width SDRAM is connected, or two 16-bit SDRAMs connected in parallel.

(2) Mobile-DDR-SDRAM

- 16-bit bus configuration in which one 16-bit width SDRAM is connected.
- 32-bit bus configuration in which one 32-bit width SDRAM is connected, or two 16-bit SDRAMs connected in parallel.

2. Writing to this register should be performed only when the following conditions are satisfied.
   - When SDRAM access is disabled (when the ACEN bit in the DBEN register is 0).
   - When automatic issue of auto-refresh is disabled (when the ARFEN bit in the DBRFPDN0 register is cleared to 0).
3. Set the designated value according to the type.

Table 15.6 Setting of SDRAM Configuration Setting Register (DDR2-SDRAM)

SPLIT bits	Memory Configuration	External Bus [bit]	Bank [bit]	Row [bit]	Column [bit]
10_0_011_010	256-Mbit product, Two 16M × 16 bits	32	2	13	9
10_0_011_011	512-Mbit product, Two 32M × 16 bits	32	2	13	10
10_1_011_011	1-Gbit product, Two 64M × 16 bits	32	3	13	10
10_1_100_011	2-Gbit product, Two 128M × 16 bits	32	3	14	10
01_0_011_010	256-Mbit product, One 16M × 16 bits	16	2	13	9
01_0_011_011	512-Mbit product, One 32M × 16 bits	16	2	13	10
01_0_011_011	512-Mbit product, Two 32M × 8 bits	16	2	13	10
01_1_011_011	1-Gbit product, One 64M × 16 bits	16	3	13	10
01_0_100_011	1-Gbit product, Two 64M × 8 bits	16	2	14	10
01_1_100_011	2-Gbit product, One 128M × 16 bits	16	3	14	10
01_1_100_011	2-Gbit product, Two 128M × 8 bits	16	3	14	10
01_1_101_011	4-Gbit product, Two 256M × 8 bits	16	3	15	10

Table 15.7 Setting of SDRAM Configuration Setting Register (Mobile-DDR-SDRAM)

SPLIT bits	Memory Configuration	External Bus [bit]	Bank [bit]	Row [bit]	Column [bit]
10_0_010_001	128-Mbit product, 4M × 32 bits	32	2	12	8
10_0_010_010	256-Mbit product, 8M × 32 bits	32	2	12	9
10_0_100_001	512-Mbit product, 16M × 32 bits	32	2	14	8
10_0_011_010	512-Mbit product, 16M × 32 bits	32	2	13	9
10_0_011_011	1-Gbit product, 32M × 32 bits	32	2	13	10
10_0_100_010	1-Gbit product, 32 M × 32 bits	32	2	14	9
10_0_100_011	2-Gbit product, 64M × 32 bits	32	2	14	10
10_0_100_011	2-Gbit product, 64M × 32 bits	32	2	14	10
01_0_010_010	128-Mbit product, 8M × 16 bits	16	2	12	9
01_0_011_001	128-Mbit product, 8M × 16 bits	16	2	13	8
01_0_011_010	256-Mbit product, 16M × 16bits	16	2	13	9
01_0_011_011	512-Mbit product, 32M × 16 bits	16	2	13	10
01_0_100_011	1-Gbit product, 64M × 16 bits	16	2	14	10
01_0_100_100	2-Gbit product, 128M × 16 bits	16	2	14	11

15.3.7 SDRAM Timing Register 0 (DBTR0)

The SDRAM timing register 0 (DBTR0) is a readable/writable register that sets the timing parameter of SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CL[2:0]			—	—	—	TRAS[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TRFC[6:0]							—	—	—	—	—	TRCD[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0. Operation when a value other than 0 is written is not guaranteed.
26 to 24	CL[2:0]	000	R/W	CAS Latency Setting Bits These bits set the SDRAM CAS latency. The set value should be identical to the CL value in MRS of the SDRAM. (Only 3 is supported for CAS latency.)
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	TRAS[4:0]	00000	R/W	tRAS (ACT-PRE period) Setting Bits ACT-PRE period Setting These bits specify the minimum interval from an activate command (ACT) to a precharge command (PRE). 00000: 1 cycles 00001: 2 cycles : 10001: 18 cycles 10010 to 1111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	TRFC[6:0]	000 0000	R/W	REF-ACT/REF period Setting These bits specify the minimum interval from a refresh command (REF) to an ACT/REF command. 000 0000: Setting prohibited 000 0001: 2 cycles 000 0010: 3 cycles : 110 1100: 109 cycles 110 1101 to 111 1111: Setting prohibited
7 to 3	—	All 0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TRCD[2:0]	000	R/W	ACT-READ/WRITE period Setting These bits specify the minimum interval from an ACT command to a read/write command. 000: 1 cycle 001: 2 cycles : 101: 6 cycles 110: Setting prohibited 111: Setting prohibited

Notes: 1. The interval should be set in terms of the number of the SDRAM operating clock cycles.
2. Set a parameter within the value range specified for the SDRAM type.
3. Writing to this register should be performed only when the following conditions are met.
   - When SDRAM access is disabled (when the ACEN bit in the DBEN register is 0).
   - When automatic issue of auto-refresh is disabled (when the ARFEN bit in the DBRFPDN0 register is cleared to 0).

15.3.8 SDRAM Timing Register 1 (DBTR1)

The SDRAM timing register 1 (DBTR1) is a readable/writable register that sets the timing parameter of SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CKEH[2:0]			—	—	—	—	—	TRP[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TRRD[2:0]			—	—	—	—	—	TWR[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	CKEH[2:0]	000	R/W	CKEH Period Setting These bits specify the period from when CKE is pulled high to when the next command is issued. The interval should be set in terms of the number of the SDRAM operating clock cycles. 000: 1 cycles 001: 2 cycles : 111: 8 cycles
23 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	TRP[2:0]	000	R/W	PRE Period Setting These bits set the minimum time from PRE command to ACT/REF command. 000: 1 cycle 001: 2 cycles : 101: 6 cycles 110: Setting prohibited 111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	TRRD[2:0]	000	R/W	tRRD (ACT(A)-ACT(B) period) Setting Bits These bits set the ACT-ACT minimum period constraint for the different banks. 000: 1 cycle : 011: 4 cycles 100 to 111: Setting prohibit (If specified, correct operation cannot be guaranteed.)
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	TWR[2:0]	000	R/W	Write Recovery Period Setting These bits set the write recovery minimum period constraint. 000: 1 cycles 001: 2 cycles : 101: 6 cycles 110: Setting prohibited 111: Setting prohibited

Notes: 1. The interval should be set in terms of the number of the SDRAM operating clock cycles.
2. Set a parameter within the value range specified for the SDRAM type.
3. Writing to this register should be performed only when the following conditions are met.
   - When SDRAM access is disabled (when the ACEN bit in the DBEN register is 0).
   - When automatic issue of auto-refresh is disabled (when the ARFEN bit in the DBRFPDN0 register is cleared to 0).

15.3.9 SDRAM Timing Register 2 (DBTR2)

The SDRAM timing register 2 (DBTR2) is a readable/writable register that sets the timing parameter of SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	TRTP[1:0]		—	—	—	TRC[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RDWR[3:0]				—	—	—	—	WRRD[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	TRTP[1:0]	00	R/W	READ-PRE Period Setting These bits set the minimum time from READ command to PRE command. 00: Setting prohibited 01: 2 cycle 10: 3 cycles 11: Setting prohibited
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	TRC[4:0]	0 0000	R/W	ACT-ACT/REF Period Setting These bits set the minimum time from ACT command to ACT/REF command. 00000: 1 cycle 00001: 2 cycles : 10110: 23 cycles 10111 to 11111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	RDWR[3:0]	0000	R/W	READ-WRITE Period Setting These bits set the minimum time from read command to issuing the write command. 0000 to 0010: Setting prohibited 0011: 4 cycle 0100: 5 cycles : 1000: 9 cycles 1001 to 1111: Setting prohibited
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	WRRD[3:0]	0000	R/W	WRITE-READ Period Setting These bits set the minimum time from write command to issuing the read command. 0000: Setting prohibited 0001: 2 cycle 0010: 3 cycles : 1100: 13 cycles 1101 to 1111: Setting prohibited

Notes: 1. The interval should be set in terms of the number of the SDRAM operating clock cycles.
2. Set a parameter within the value range specified for the SDRAM type.
3. Writing to this register should be performed only when the following conditions are met.
   - When SDRAM access is disabled (when the ACEN bit in the DBEN register is 0).
   - When automatic issue of auto-refresh is disabled (when the ARFEN bit in the DBRFPDN0 register is cleared to 0).

15.3.10 SDRAM Timing Register 3 (DBTR3)

The SDRAM timing register 3 (DBTR3) is a readable/writable register that sets the timing parameter of SDRAM.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	ODTL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	ODTL[2:0]	000	R/W	MODT Signal Minimum Period Setting Bits These bits set the number of cycle of the MODT signal to be asserted. The cycle number setting described in here is for every write command. 000: No MODT signal asserted 001: 1 cycle of MODT signal asserted 010: 2 cycles of MODT signal asserted : 111: 7 cycles of MODT signal asserted

Notes: 1. Set a parameter within the value range specified for the SDRAM type.

2. Writing to this register should be performed only when the following conditions are met.
   - When SDRAM access is disabled (when the ACEN bit in the DBEN register is 0).
   - When automatic issue of auto-refresh is disabled (when the ARFEN bit in the DBRFPDN0 register is cleared to 0).

15.3.11 SDRAM Refresh Control Register 0 (DBRFPDN0)

The SDRAM refresh control register 0 (DBRFPDN0) is a readable/writable register that controls the refresh and power-down operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ARFEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PDN	DPDN	SRFEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	ARFEN	0	R/W	Auto-Refresh Enable Bit Enables or disables automatic issue of auto-refresh. The auto-refresh command is issued periodically according to the settings of DBRFPDN1/2. For details on the auto-refresh command issue timing, refer to section 15.5.5, Auto-Refresh Operation. 0: Disables automatic issue of auto-refresh. 1: Enables automatic issue of auto-refresh.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	PDN	0	R/W	Power-Down Mode Bit Performs transition to or cancellation of power-down mode. By writing 1, a transition is made to power-down mode. By writing 0, power-down mode is cancelled. For details on transition to or cancellation of power-down mode, refer to section 15.5.7, Power-Down Operation. 0: Cancels power-down mode. 1: Makes a transition to power-down mode.
1	DPDN	0	R/W	Deep Power-Down Mode Bit Performs transition to or cancellation of deep power-down mode. By writing 1, a transition is made to deep power-down mode. By writing 0, deep power-down mode is cancelled. For details on transition to or cancellation of deep power-down mode, refer to section 15.5.8, Deep Power-Down Operation. 0: Cancels deep power-down mode. 1: Makes a transition to deep power-down mode.
0	SRFEN	0	R/W	Self-Refresh Mode Bit Performs transition to or cancellation of self-refresh mode. By writing 1, a transition is made to self-refresh. By writing 0, self-refresh mode is cancelled. For details on transition to or cancellation of self-refresh, refer to section 15.5.4, Self-Refresh Operation. 0: Cancels self-refresh. 1: Makes a transition to self-refresh.

15.3.12 SDRAM Refresh /Power-down Control Register 1 (DBRFPDN1)

The SDRAM refresh/power-down control register 1 (DBRFPDN1) is a readable/writable register that controls the refresh operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TREFI[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	TREFI[12:0]	H'0000	R/W	Average Refresh Interval Setting These bits set the average interval for auto-refresh operation. Upon refresh execution, this value is added to the refresh counter described in section 15.5.5, Auto-Refresh Operation. The refresh counter is not directly accessible to the user. H'0000 to H'003F: Setting prohibited H'0040: 65 cycles H'0041: 66 cycles : H'1FFF: 8192 cycles

Note: 1. Set the average refresh interval (tREFI) prescribed in the datasheet of the memory manufacturer.
2. The interval should be set in terms of the number of the SDRAM operating clock cycles.
3. Writing to this register should be performed only when the ARFEN bit in the DBRFPDN0 register is cleared to 0.

15.3.13 SDRAM Refresh Control Register 2 (DBRFPDN2)

The SDRAM refresh/power-down control register 2 (DBRFPDN2) is a readable/writable register that controls the refresh operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	LV1TH[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	LV0TH[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
30 to 16	LV1TH[14:0]	H'0000	R/W	Level 1 Threshold Setting These bits set the threshold cycles for executing refresh operation in empty bus request. The number of cycles is the number of SDRAM clock cycles. • Sum of the values of the TREFI bit and LV1TH bit in the DBRFPDN1 register is to be maximum value of the auto-refresh counter, that is the maximum interval of refresh command in issuing an auto-refresh periodically. Set the LV1TH bit to fit this value in a range of ACT-PRE command interval listed in the data sheet of each memory vender. • After issuing of the eighth refresh command for SDRAM, since the refresh command must be issued within an interval greater than or equal to tREFI and equal to or within 9 × tREFI, set LV1TH to 7 × tREFI.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9 to 0	LV0TH[9:0]	H'000	R/W	Level 0 Threshold Setting These bits set the threshold cycles for refresh operation in a break of bus request. The number of cycles is the number of SDRAM clock cycles. To prevent underflows of the refresh counter, set the VL0TH bits to five times the number of cycles for a WRITE-WRITE command interval in the case of a bank hit and page miss, i.e. to (tRP + tRCD + tWR + CL - 1) × 5. However, use the value listed in the data sheet for the WRITE-WRITE command interval.

Note: Writing to this register should be performed only when the ARFEN bit in the DBRFPDN0 register is cleared to 0.

15.3.14 SDRAM Refresh Status Register (DBRFSTS)

The SDRAM refresh status register (DBRFSTS) is a readable/writable register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RFUDF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RFUDF	0	R/W	Refresh Counter Underflow Bit Set to 1 to indicate that the refresh counter has underflows when the refresh counter changes from 1 to 0. This bit is cleared to 0 by writing 0 to it. Underflow may occur because the LV0TH bit value is smaller than the maximum number of command execution cycles, so that refresh cannot be issued until the counter value reaches 0. In this case, the value of the LV0TH bit should be changed. For details on the refresh counter, refer to section 15.5.5, Auto-Refresh Operation. 0: Indicates that no underflow occurs. 1: Indicates that an underflow occurs.

15.3.15 SDRAM Mode Setting Register (DBMRCNT)

By writing to this register, the SDRAM address and bank address pins can be directly manipulated to set the mode and extended mode registers. When this register is written, the mode register setting (MRS)/extended mode register setting (EMRS) command is issued for the SDRAM. Upon command execution, set the address and bank pins as specified. Further, settings should be made such that burst length is 4 and the CAS latency (CL) is equal to the corresponding bits in SDRAM timing registers 0 (DBTR0).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	BA[2:0]		
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	MA[14:0]														
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	Undefined	W	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	BA[2:0]	Undefined	W	Bank Address Output Setting Set the value issued from the bank address pin (BA) at mode register setting. Reading the settings is disabled. Bank address pins BA2, BA1, and BA0 correspond to bit 18, bit 17, and bit 16, respectively.
15	—	Undefined	W	Reserved These bits are always read as 0. The write value should always be 0.
14 to 0	MA[14:0]	Undefined	W	Address Output Setting Set the value issued from the address pin (MA) at mode register setting. Reading the settings is disabled. The address pins MA14, MA13, ..., and MA0 correspond to bit 14, bit 13, ..., and bit 0, respectively.

15.3.16 DDR-PHY Setting Register (DBPDCNT0)

The DDR-PHY setting register (DBPDCNT0) is a readable/writable register that controls the DDR-PHY block.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OFFSET[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DQS	ODTEN	ODT	—	—	—	—	—	—	PLUP
Initial value:	—	—	—	—	—	—	0	0	0	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	OFFSET[1:0]	00	R/W	Offset Value Setting These bits set the offset value of the DDR-PAD area. Set these bits to B'00 or B'01 for the DDR2-SDRAM or Mobile-DDR-SDRAM, respectively.
29 to 10	—	Undefined	R	Reserved Writing is invalid.
9	DQS	0	R/W	DQS Operation Mode Setting This bit sets DQS differential mode. When the single mode is to be used, the DQS enable bit in the EMRS(1) register must be disabled. 0: DQS differential mode 1: DQS single mode
8	ODTEN	0	R/W	ODT Disable Setting this bit to 1 disables the ODT. Do not change this bit value from 0 (initial value) when the Mobile-DDR-SDRAM is to be used. 0: ODT control is enabled. 1: ODT control is disabled.
7	ODT	0	R/W	ODT Resistance Setting Bit This bit sets the resistance of ODT. The setting of this bit has no effect when the Mobile-DDR-SDRAM is used. 0: 150 Ω 1: 75 Ω

Bit	Bit Name	Initial Value	R/W	Description
6 to 1	—	Undefined	R	Reserved Writing is invalid.
0	PLUP	0	R/W	Pull-up Setting Sets pull-up for the DDR-PHY initialization sequence. For the setting method, refer to section 15.5.3, Initialization Sequence. 0: Not pulled up. 1: Pulled up.

Note: Before using this register, read the initial values in a 32-bit width and set all the bits except writable bits to the initial values.

15.4 Data Alignment

This LSI supports big endian and little endian method for data alignment. The data alignment is specified by the setting of the external pin (MD5) at a power-on reset.

Table 15.8 Data Alignment specified by External Pin (MD5)

MD5	Data Alignment
0	Big endian
1	Little endian

The DBSC accesses SDRAM with a fixed burst length of 4. As shown in tables 15.9 and 15.10, invalid read data is discarded during reading, and data mask signals are used to mask invalid data during writing, according to the access size. The access times in tables 15.9 and 15.10 correspond to the burst times during reading/writing shown in figure 15.2. For example, when the external bus width is 32 bits with a little endian, the second access (falling edge of DQS) includes valid data if a byte access of address (8n + 0, 1, 2, 3) occurs.

Tables 15.11 and 15.14 show the correspondence with data on the external data bus for each access size. During 16-byte and 32-byte accesses, quad word (8 bytes) access is combined, and the SDRAM command is issued the necessary number of times according to the size to access the SDRAM as shown in figures 15.3. The SDRAM specification stipulates sequential address changes (0→1→2→3, 1→2→3→0, 2→3→0→1, 3→0→1→2), so that the address provided as a command is different for reading and for writing.

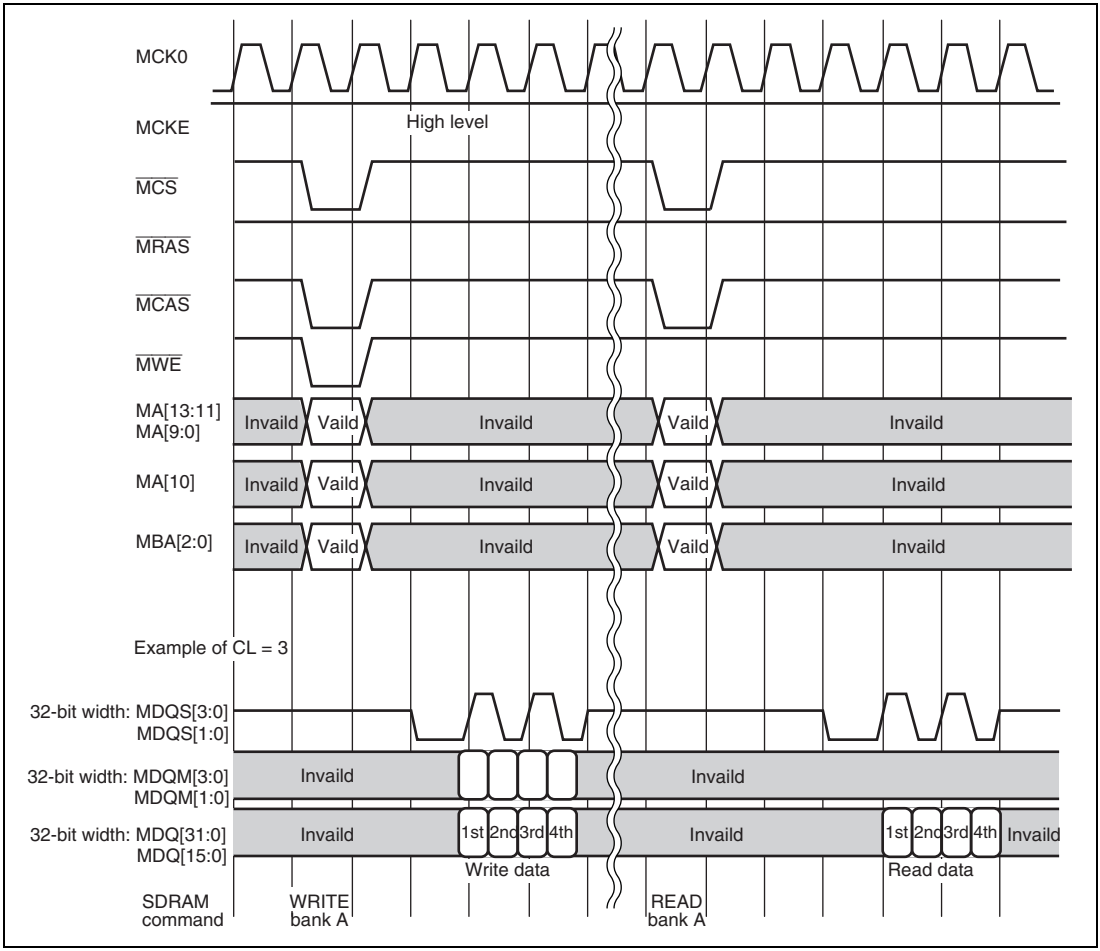


Figure 15.2 Burst Access Operation

Table 15.9 Positions of Valid Data for Access with Burst Length of 4, when the External Data Bus Width Is Set to 32 Bits

Little Endian	First Access	Second Access	Third Access	Fourth Access
Byte access (address 8n + 0, 1, 2, 3)	Invalid	Valid	Invalid	Invalid
Byte access (address 8n + 4, 5, 6, 7)	Valid	Invalid	Invalid	Invalid
Word access (address 8n + 0, 2)	Invalid	Valid	Invalid	Invalid
Word access (address 8n + 4, 6)	Valid	Invalid	Invalid	Invalid
Longword access (address 8n + 0)	Invalid	Valid	Invalid	Invalid
Longword access (address 8n + 4)	Valid	Invalid	Invalid	Invalid
Quadword access (address 8n + 0)	Valid	Valid	Invalid	Invalid

Big Endian	First Access	Second Access	Third Access	Fourth Access
Byte access (address 8n + 0,1,2,3)	Valid	Invalid	Invalid	Invalid
Byte access (address 8n + 4,5,6,7)	Invalid	Valid	Invalid	Invalid
Word access (address 8n + 0, 2)	Valid	Invalid	Invalid	Invalid
Word access (address 8n + 4, 6)	Invalid	Valid	Invalid	Invalid
Longword access (address 8n + 0)	Valid	Invalid	Invalid	Invalid
Longword access (address 8n + 4)	Invalid	Valid	Invalid	Invalid
Quadword access (address 8n + 0)	Valid	Valid	Invalid	Invalid

Table 15.10 Positions of Valid Data for Access with Burst Length of 4, when the External Data Bus Width Is Set to 16 Bits

Little Endian	First Access	Second Access	Third Access	Fourth Access
Byte access (address 8n + 0,1)	Invalid	Invalid	Invalid	Valid
Byte access (address 8n + 2, 3)	Invalid	Invalid	Valid	Invalid
Byte access (address 8n + 4, 5)	Invalid	Valid	Invalid	Invalid
Byte access (address 8n + 6, 7)	Valid	Invalid	Invalid	Invalid
Word access (address 8n + 0)	Invalid	Invalid	Invalid	Valid
Word access (address 8n + 2)	Invalid	Invalid	Valid	Invalid
Word access (address 8n + 4)	Invalid	Valid	Invalid	Invalid
Word access (address 8n + 6)	Valid	Invalid	Invalid	Invalid
Longword access (address 8n + 0)	Invalid	Invalid	Valid	Valid
Longword access (address 8n + 4)	Valid	Valid	Invalid	Invalid
Quadword access (address 8n + 0)	Valid	Valid	Valid	Valid

Big Endian	First Access	Second Access	Third Access	Fourth Access
Byte access (address 8n + 0,1)	Valid	Invalid	Invalid	Invalid
Byte access (address 8n + 2, 3)	Invalid	Valid	Invalid	Invalid
Byte access (address 8n + 4, 5)	Invalid	Invalid	Valid	Invalid
Byte access (address 8n + 6, 7)	Invalid	Invalid	Invalid	Valid
Word access (address 8n + 0)	Valid	Invalid	Invalid	Invalid
Word access (address 8n + 2)	Invalid	Valid	Invalid	Invalid
Word access (address 8n + 4)	Invalid	Invalid	Valid	Invalid
Word access (address 8n + 6)	Invalid	Invalid	Invalid	Valid
Longword access (address 8n + 0)	Valid	Valid	Invalid	Invalid
Longword access (address 8n + 4)	Invalid	Invalid	Valid	Valid
Quadword access (address 8n + 0)	Valid	Valid	Valid	Valid

Table 15.11 Data Alignment for Access in Little Endian when External Data Bus Width Is Set to 32 Bits

Access Size	Address	MDQ31 to MDQ24	MDQ23 to MDQ16	MDQ15 to MDQ8	MDQ7 to MDQ0
Byte	Address 0	—	—	—	Data 7 to 0
	Address 1	—	—	Data 7 to 0	—
	Address 2	—	Data 7 to 0	—	—
	Address 3	Data 7 to 0	—	—	—
	Address 4	—	—	—	Data 7 to 0
	Address 5	—	—	Data 7 to 0	—
	Address 6	—	Data 7 to 0	—	—
	Address 7	Data 7 to 0	—	—	—
Word	Address 0	—	—	Data 15 to 8	Data 7 to 0
	Address 2	Data 15 to 8	Data 7 to 0	—	—
	Address 4	—	—	Data 15 to 8	Data 7 to 0
	Address 6	Data 15 to 8	Data 7 to 0	—	—
Longword	Address 0	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0
	Address 4	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0
Quadword	Address 0 (First access: address 4)	Data 63 to 56	Data 55 to 48	Data 47 to 40	Data 39 to 32
	Address 0 (Second access: Address 0)	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0

Table 15.12 Data Alignment for Access in Big Endian when External Data Bus Width Is Set to 32 Bits

Access Size	Address	MDQ31 to MDQ24	MDQ23 to MDQ16	MDQ15 to MDQ8	MDQ7 to MDQ0
Byte	Address 0	Data 7 to 0	—	—	—
	Address 1	—	Data 7 to 0	—	—
	Address 2	—	—	Data 7 to 0	—
	Address 3	—	—	—	Data 7 to 0
	Address 4	Data 7 to 0	—	—	—
	Address 5	—	Data 7 to 0	—	—
	Address 6	—	—	Data 7 to 0	—
	Address 7	—	—	—	Data 7 to 0
Word	Address 0	Data 15 to 8	Data 7 to 0	—	—
	Address 2	—	—	Data 15 to 8	Data 7 to 0
	Address 4	Data 15 to 8	Data 7 to 0	—	—
	Address 6	—	—	Data 15 to 8	Data 7 to 0
Longword	Address 0	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0
	Address 4	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0
Quadword	Address 0 (First access: Address 0)	Data 63 to 56	Data 55 to 48	Data 47 to 40	Data 39 to 32
	Address 0 (Second access: Address 4)	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0

Table 15.13 Data Alignment for Access in Little Endian when External Data Bus Width Is Set to 16 Bits

Access Size	Address	MDQ15 to MDQ8	MDQ7 to MDQ0
Byte	Address 0	—	Data 7 to 0
	Address 1	Data 7 to 0	—
	Address 2	—	Data 7 to 0
	Address 3	Data 7 to 0	—
	Address 4	—	Data 7 to 0
	Address 5	Data 7 to 0	—
	Address 6	—	Data 7 to 0
	Address 7	Data 7 to 0	—
Word	Address 0	Data 15 to 8	Data 7 to 0
	Address 2	Data 15 to 8	Data 7 to 0
	Address 4	Data 15 to 8	Data 7 to 0
	Address 6	Data 15 to 8	Data 7 to 0
Longword	Address 0 (First access: address 2)	Data 31 to 24	Data 23 to 16
	Address 0 (Second access: address 0)	Data 15 to 8	Data 7 to 0
	Address 4 (First access: address 6)	Data 31 to 24	Data 23 to 16
	Address 4 (Second access: address 4)	Data 15 to 8	Data 7 to 0
Quadword	Address 0 (First access: address 6)	Data 63 to 56	Data 55 to 48
	Address 0 (Second access: Address 4)	Data 47 to 40	Data 39 to 32
	Address 0 (Third access: address 2)	Data 31 to 24	Data 23 to 16
	Address 0 (Forth access: Address 0)	Data 15 to 8	Data 7 to 0

Table 15.14 Data Alignment for Access in Big Endian when External Data Bus Width Is Set to 16 Bits

Access Size	Address	MDQ15 to MDQ8	MDQ7 to MDQ0
Byte	Address 0	Data 7 to 0	—
	Address 1	—	Data 7 to 0
	Address 2	Data 7 to 0	—
	Address 3	—	Data 7 to 0
	Address 4	Data 7 to 0	—
	Address 5	—	Data 7 to 0
	Address 6	Data 7 to 0	—
	Address 7	—	Data 7 to 0
Word	Address 0	Data 15 to 8	Data 7 to 0
	Address 2	Data 15 to 8	Data 7 to 0
	Address 4	Data 15 to 8	Data 7 to 0
	Address 6	Data 15 to 8	Data 7 to 0
Longword	Address 0 (First access: Address 0)	Data 31 to 24	Data 23 to 16
	Address 0 (Second access: Address 2)	Data 15 to 8	Data 7 to 0
	Address 4 (First access: Address 4)	Data 31 to 24	Data 23 to 16
	Address 4 (Second access: Address 6)	Data 15 to 8	Data 7 to 0
Quadword	Address 0 (First access: Address 0)	Data 63 to 56	Data 55 to 48
	Address 0 (Second access: Address 2)	Data 47 to 40	Data 39 to 32
	Address 0 (Third access: Address 4)	Data 31 to 24	Data 23 to 16
	Address 0 (Forth access: Address 6)	Data 15 to 8	Data 7 to 0

When the external bus width is set to 32 bits

16-byte read/write access (a total of two commands are issued)

	1st access
Address 16n + 0	16n+0
Address 16n + 8	16n+8

32-byte read/write access (a total of four commands are issued)

	1st access	2nd access
Address 32n + 0	32n + 0	32n + 16
Address 32n + 8	32n + 0	32n + 16
Address 32n + 16	32n + 16	32n + 0
Address 32n + 24	32n + 16	32n + 0

32-byte write access (a total of four commands are issued)

	1st access	2nd access
Address 32n+ 0	32n + 0	32n + 16
Address 32n + 8	32n + 16	32n+ 0
Address 32n + 16	32n + 16	32n + 0
Address 32n + 24	32n + 0	32n + 16

Figure 15.3 Addresses Generated upon 16-/32-Byte Access when the External Data Bus Width Is 32 Bits

15.5 DBSC Operation

15.5.1 Supported SDRAM Commands

Table 15.15 lists the SDRAM commands issued by the DBSC. These commands are issued to the SDRAM in synchronously with MCK0, $\overline{\text{MCK0}}$. In the table, n-1 indicates the state of the signal applied to SDRAM one cycle before SDRAM command issue; n indicates the state of the signal at the time of command issue.

Table 15.15 Register Status in each Processing Mode

		MCKE						MA	MA10/	MBA	MA
Function	**Symbol**	**n-1**	**n**	**$\overline{\text{MCS}}$**	**$\overline{\text{MRAS}}$**	**$\overline{\text{MCAS}}$**	**$\overline{\text{MWE}}$**	**[13:11]**	**AP**	**[2:0]**	**[9:0]**
Device deslect	DSEL	H	H	H	X	X	X	X	X	X	X
Read	READ	H	H	L	H	L	H	V	L	V	V
Write	WRITE	H	H	L	H	L	L	V	L	V	V
Bank activate	ACT	H	H	L	L	H	H	V	V	V	V
Precharge select bank	PRE	H	H	L	L	H	L	X	L	V	X
Precharge all banks	PALL	H	H	L	L	H	L	X	H	X	X
Auto-refresh	REF	H	H	L	L	L	H	X	X	X	X
Self-refresh entry from IDLE	SLFRSH	H	L	L	L	L	H	X	X	X	X
Self-refresh exit	SLFRSHX	L	H	H	X	X	X	X	X	X	X
Mode register set	MRS/EMRS	H	H	L	L	L	L	V	V	V	V
Power down mode entry	PDEN	H	L	L	H	H	X	X	X	X	X
Deep power down mode entry	DPDEN	H	L	L	H	H	L	X	X	X	X
Power down mode exit	PDEX	L	H	H	X	X	X	X	X	X	X

Legend:
H: High level
L: Low level
X: High or low level (don't care)
V: Valid data

The above DSEL command is issued when SDRAM is not accessed, and so need not be explicitly issued by the user.

15.5.2 SDRAM Command Issue

(1) Basic Access

The DBSC stores in a queue the requests received via the SuperHyway bus. Request processing is begun around the time of preceding precharge/activate processing, but processing completion is in the order received in the queue.

When SDRAM initialization is completed, upon receiving a read/write request, a page miss occurs with all banks in the closed state. Hence the DBSC first issues an activate (ACT) command, to open the corresponding bank. After opening the bank, the read/write command of the SDRAM corresponding to the read/write request is issued. At this time, the number of issued read/write commands differs depending on the bus width and the request size (1/2/4/8/16/32 bytes), as indicated in figure 15.4. For example, when performing 32-byte reading from the SuperHyway bus with an external data bus width of 32 bytes, two read commands are executed. When issuing the read command in the first cycle, data is read with a burst length of 4 (two DDR clock cycles), so that it is necessary to wait until the third cycle to issue the second read command.

When access ends, the DBSC leaves the bank open, without using a precharge (PRE) command. The bank is closed when (1) the following request is for the same bank with a different row address; (2) there is an auto-refresh request; or (3) the user issues a precharge-all (PALL) command using the SDRAM command control register, for self-refresh processing.

Thus in normal access other than self-refresh, the DBSC uses hardware for bank management, so that except for the register settings upon initialization, the user need not execute control.

Further, the DBSC performs multi-bank operation of four banks. Hence the maximum number of banks which can be opened simultaneously is four. Refer to section 15.5.9, Relation between SDRAM Address Pins and Logical Addresses, for the correspondence between access addresses from the SuperHyway bus and SDRAM bank/row addresses.

When using the SDRAM with a memory capacity of 1 Gbit or greater, refer to section 15.6.1, Important Information Regarding Use of 8-Bank DDR2-SDRAM Products.

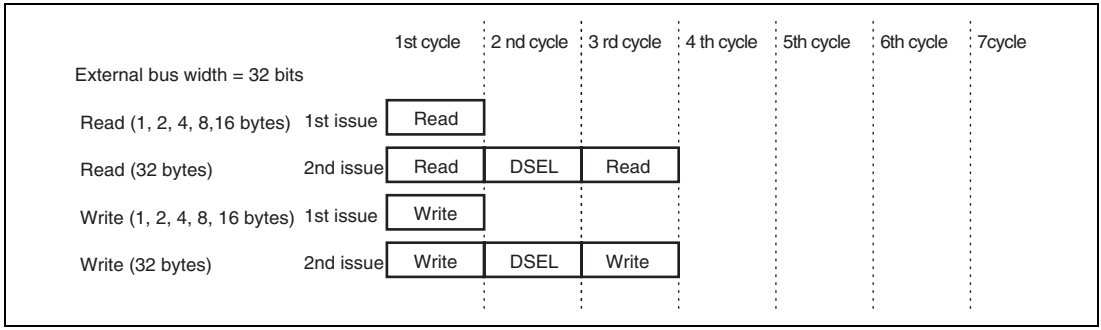


Figure 15.4 Read/Write Command Issued to the SDRAM in Response to the Request from the SuperHyway Bus

(2) Preceding Precharge/Activate Processing

In order to utilize DDR2-SDRAM multibank functions to reduce SDRAM command vacant cycles insofar as possible and improve the efficiency of bus use, the DBSC issues in advance a PRE/ACT command corresponding to the following request queue page miss processing. Only the PRE/ACT command is issued in advance, so there is no change in the read/write order. A PRE/ACT command is issued in advance only when the following request (1) results in a page miss, and moreover (2) entails access of a bank different from that of the request currently being processed. Figure 15.5 shows an example of execution of preceding precharge/activate processing. This is an example of a command issued to the SDRAM when the external data bus width is 32 bits, the PRE/ACT minimum time constraint is 3 cycles, the ACT-READ/WRITE minimum time constraint is 3 cycles, and the ACT (A)-ACT (B) minimum time constraint is 2 cycles. In this example, the first through fourth requests are accumulated, and the first request is the request initially provided to the queue.

First, at time 1 the DBSC issues to the SDRAM a PRE command for the first read (16-byte) request processing. Then, when determining the command to be issued at time 2, due to timing constraints it is not possible to issue at time 2 the ACT command necessary as request processing for the first read (16-byte) request, which has higher priority. Hence the DBSC searches for a command to be issued at time 2 from the following request queue. From the search results it is seen that advance precharge processing can be executed for the third read (8-byte) request and the fourth read (16-byte) request. Because the DBSC gives priority to preceding requests, it decides to perform advance precharge processing for the third read (8-byte) request, and issues a PRE command to the SDRAM.

When the time advances to time 3, the ACT command cannot be issued for the first read (16-byte) request at time 3 either, and so a search of the following queue is performed for a command which can be issued. Due to timing constraints, the ACT command cannot be issued for the third read (8-byte) request, and as a result, issuance of the PRE command corresponding to the fourth read (16-byte) request is selected.

At time 4, it is possible to execute request processing for the first read (16-byte) request, and an ACT command is issued to the SDRAM.

Thereafter, the processing described above is repeated.

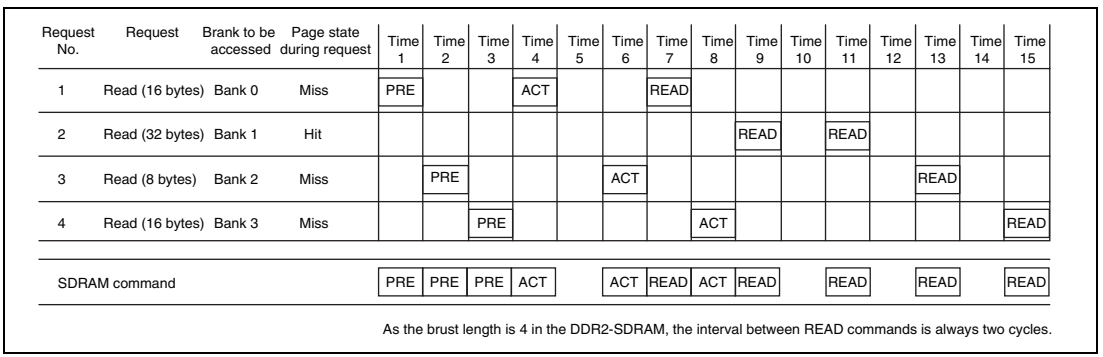


Figure 15.5 Example of Preceding Precharge/Activate Processing

15.5.3 Initialization Sequence

Before permitting accesses to the SDRAM after a reset, the SDRAM should be initialized according to the appropriate sequence as shown below. Since the shown wait time between steps is merely an example, provide the necessary wait time given in the datasheet supplied by the memory vendor.

(1) DDR2-SDRAM

1. After the power supply, reference voltage, and clock supplied to SDRAM are stabilized, wait for at least 200 μs.
2. Set the pull-up setting bit (PLUP) in the DDR-PHY control register 0 (DBPDCNT0) to 1.
3. Enter the settings for the SDRAM configuration setting register (DBCONF), SDRAM timing register 0 (DBTR0), SDRAM timing register 1 (DBTR1), SDRAM timing register 2 (DBTR2), and SDRAM timing register 3 (DBTR3).
4. Set the memory type to DDR2-SDRAM with the DDCG bits in the SDRAM type setting register (DBKIND).
5. Set the CKEEN bit in the SDRAM CKE control register (DBCKECNT) to 1.
6. Wait for 400 ns.
7. Write to the SDRAM command control register (DBCMDCNT) to issue the PALL command.
8. Write to the SDRAM mode setting register (DBMRCNT) to issue the EMRS(2) command to the SDRAM. After that, issue the EMRS(3) command.
9. Write to DBMRCNT to issue the EMRS command to the SDRAM, set various parameters, and enable the DLL.
10. Write to DBMRCNT to issue the MRS command to the SDRAM and set various parameters. The settings should be made such that operating mode is normal, DLL reset is provided, burst length is 4, burst type is sequential, and additive latency is 0. CAS latency should be identical to the CL bit value in the SDRAM timing register 0 (DBTR0).
11. Write DBCMDCNT to issue the PALL command. Then, write DBCMDCNT to issue the REF command at least twice.
12. Write to DBMRCNT to issue the MRS command to the SDRAM and set various parameters. The DLL reset is not performed for this setting.
13. After the MRS command issued in step 12, wait for at least 200 SDRAM clock cycles. For example, dummy-read 40 times the DBSC status register (DBSTATE) of the DBSC.
14. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 1 (access enabled).
15. Set the SDRAM refresh/power-down control registers 1 and 2 (DBRFPDN1 and DBRFPDN2).
16. Read any data in the SDRAM space.

17. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) to issue the PALL command, and then, issue the REF command.
18. Clear the PLUP bit in the DDR-PHY control register 0 (DBPDCNT0) to 0.
19. Set the ARFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 (automatic issue of auto-refresh enabled). Normal access is subsequently enabled.

(2) Mobile-DDR-SDRAM

1. Wait until the power supply and the clocks that supply to the SDRAM become stable.
2. Set the pull-up setting bit (PLUP) in the DDR-PHY control register 0 (DBPDCNT0) to 1.
3. Enter the settings for the SDRAM configuration setting register (DBCONF), SDRAM timing register 0 (DBTR0), SDRAM timing register 1 (DBTR1), SDRAM timing register 2 (DBTR2), and SDRAM timing register 3 (DBTR3).
4. Set the memory type to Mobile-DDR-SDRAM with the DDCG bits in the SDRAM type setting register (DBKIND).
5. Set the CKEEN bit in the SDRAM CKE control register (DBCKECNT) to 1.
6. Wait for 200 μs.
7. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) to issue the PALL command.
8. Write to the SDRAM mode setting register (DBMRCNT) to issue the MRS command to the SDRAM and set various parameters. The settings should be made such that burst length is 4, and burst type is sequential. CAS latency should be identical to the CL bit value in the SDRAM timing register 0 (DBTR0).
9. Write to the SDRAM mode setting register (DBMRCNT) to issue the EMRS command to the SDRAM, and set various parameters.
   When partial self-refresh mode is to be used, settings should be made at this point.
10. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) twice to issue the PALL command twice.
11. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 1 (access enabled).
12. Set the SDRAM refresh/power-down control registers 1 and 2 (DBRFPDN1 and DBRFPDN2).
13. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) to issue the PALL command, and then, issue the REF command.
14. Read any data in the SDRAM space.
15. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) to issue the PALL command, and then, issue the REF command.
16. Clear the PLUP bit in the DDR-PHY control register 0 (DBPDCNT0) to 0.
17. Set the ARFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 (automatic issue of auto-refresh enabled). Normal access is subsequently enabled.

15.5.4 Self-Refresh Operation

If it is not necessary to access the SDRAM, the SDRAM can be put in self-refresh mode to reduce power consumption while still retaining data contents. Before stopping clocks or changing the clock frequency for the DBSC, be sure to shift the SDRAM to the self-refresh mode.

Shifting to self-refresh mode is done by writing 1 to the self-refresh enable bit (SRFEN) in the SDRAM refresh/power-down control register 0 (DBRFPDN0). Self-refresh mode can be cancelled by writing 0 to the SRFEN bit.

Because access is disabled in self-refresh mode, any attempt to access data in the SDRAM will be ignored.

The following procedure is used to make a transition to self-refresh mode, and is an example of the initialization sequence. For detailed information, please refer to the datasheet for the DDR2-SDRAM being used.

(1) DDR2-SDRAM

(a) Shifting to Self-Refresh Mode

1. Make sure that the DBSC is not being accessed. Disable the CPU interrupts to prevent the CPU from processing the interrupts and accessing data in SDRAM.
2. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 0 (access disabled).
3. Set the ARFEN bit in the SDRAM refresh control register 0 (DBRFPDN0) to 0 (automatic issue of auto-refresh disabled).
4. Use the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
5. Use the CMD bits in DBCMDCNT to issue the REF (auto-refresh) command.
6. Set the SRFEN bit in DBRFPDN0 to 1 to make a transition to self-refresh mode.
7. Enable the CPU interrupts.

(b) Canceling Self-Refresh Mode

Use the following procedure to cancel self-refresh mode.

1. Set the pull-up setting bit (PLUP) in the DDR-PHY control register 0 (DBPDCNT0) to 1.
2. Enter the settings for the SDRAM configuration setting register (DBCONF), SDRAM timing register 0 (DBTR0), SDRAM timing register 1 (DBTR1), SDRAM timing register 2 (DBTR2), and SDRAM timing register 3 (DBTR3).

3. Set the memory type to DDR2-SDRAM with the DDCG bits in the SDRAM type setting register (DBKIND).
4. Set the SRFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1.
5. Cancel self-refresh mode by setting the SRFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 0.
6. Set the CKEEN bit in the SDRAM CKE control register (DBCKECNT) to 1.
7. Wait for 400 μs.
8. Write to the SDRAM command control register (DBCMDCNT) to issue the PALL command.
9. Write to the SDRAM mode setting register (DBMRCNT) to issue the EMRS(2) command to the SDRAM. After that, issue the EMRS(3) command.
10. Write to the SDRAM mode setting register (DBMRCNT) to issue the EMRS command to the SDRAM, set various parameters, and enable the DLL.
11. Write to the SDRAM mode setting register (DBMRCNT) to issue the MRS command to the SDRAM and set various parameters. The settings should be made such that operating mode is normal, DLL reset is provided, burst length is 4, burst type is sequential, and additive latency is 0. CAS latency should be identical to the CL bit value in the SDRAM timing register 0 (DBTR0).
12. Write to the command control register (DBCMDCNT) to issue the PALL command. Then, write to DBCMDCNT to issue the REF command at least twice.
13. Write to the SDRAM mode setting register (DBMRCNT) to issue the MRS command to the SDRAM and set various parameters. The DLL reset is not performed for this setting.
14. After the MRS command issued in step 13, wait for at least 200 SDRAM clock cycles. For example, dummy-read 40 times the DBSC status register (DBSTATE) of the DBSC.
15. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 1 (access enabled).
16. Set the SDRAM refresh/power-down control registers 1 and 2 (DBRFPDN1 and DBRFPDN2).
17. Read any data in the SDRAM space.
18. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) to issue the PALL command, and then, issue the REF command.
19. Clear the PLUP bit in the DDR-PHY control register 0 (DBPDCNT0) to 0.
20. Set the ARFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 (automatic issue of auto-refresh enabled).

(2) Mobile-DDR-SDRAM

(a) Shifting to Self-Refresh Mode

1. Make sure that the DBSC is not being accessed.
2. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 0 (access disabled).
3. Set the ARFEN bit in the SDRAM refresh control register 0 (DBRFPDN0) to 0 (automatic issue of auto-refresh disabled).
4. Use the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
5. Use the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the REF (auto-refresh) command.
6. Set the SRFEN bit in the SDRAM refresh control register 0 (DBRFPDN0) to 1 to make a transition to self-refresh mode.

(b) Canceling Self-Refresh Mode

Use the following procedure to cancel self-refresh mode.

1. Set the PLUP bit in the DDR-PHY control register 0 (DBPDCNT0) to 1.
2. Enter the settings for the SDRAM configuration setting register (DBCONF), SDRAM timing register 0 (DBTR0), SDRAM timing register 1 (DBTR1), SDRAM timing register 2 (DBTR2), and SDRAM timing register 3 (DBTR3).
3. Set the memory type to Mobile-DDR-SDRAM with the DDCG bits in the SDRAM type setting register (DBKIND).
4. Set the SRFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1.
5. Cancel self-refresh mode by setting the SRFEN bit in in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 0.
6. Set the CKEEN bit in the SDRAM CKE control register (DBCKECNT) to 1.
7. Wait for 200 μs.
8. Write to the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
9. Write to the SDRAM mode setting register (DBMRCNT) to issue the MRS command to the SDRAM and set various parameters. The settings should be made such that burst length is 4, and burst type is sequential. CAS latency should be identical to the CL bit value in the SDRAM timing register 0 (DBTR0).
10. Write to the SDRAM mode setting register (DBMRCNT) to issue the EMRS command to the SDRAM, and set various parameters. Settings for partial self-refresh mode should be made at this point.

11. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) twice to issue the REF command twice.
12. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 1 (access enabled).
13. Set the SDRAM refresh/power-down control registers 1 and 2 (DBRFPDN1 and DBRFPDN2).
14. Read any data in the SDRAM space once. The read data is invalid. (Issue commands including ACT command before issuing the READ command according to the memory specifications as required.)
15. Write to the CMD bit in the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
16. Set the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the REF (refresh) command.
17. Clear the PLUP bit in the DDR-PHY control register 0 (DBPDCNT0) to 0.
18. Set the ARFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 (automatic issue of auto-refresh enabled).

15.5.5 Auto-Refresh Operation

When the auto-refresh enable bit (ARFEN) in the SDRAM refresh control register 0 (DBRFPDN0) is 1, the auto-refresh command is issued periodically. If accessing data in the SDRAM, always make sure this is set.

The average refresh interval is set in the TREFI bits in the SDRAM refresh control register 1 (DBRFCNT1).

In order to minimize reductions in the data transfer capability caused by auto-refreshing, the timing at which auto-refreshing is carried out can be divided into three levels and controlled:

- Level 0: Refreshing is done in vacant periods between commands being received from the SuperHyway bus.
- Level 1: Refreshes are issued during request empty cycles.
- Level 2: Refreshing is not done.

The threshold values for level 0 and level 1 are set using the LV0TH bit in the SDRAM refresh control register 2 (DBRFCNT2), and the threshold values for level 1 and level 2 are set using the LV1TH bit.

The refresh timing is controlled using a 14-bit refresh counter. The refresh counter counts down based on the DDR clock, until a refresh is carried out. When a refresh is carried out, the counter value increments by the amount of the average refresh interval set with the TREFI bits in DBRFCNT1. Figure 15.6 shows an example of the refresh operation and the update of the refresh counter.

If there is a bank that is open before the auto-refresh is carried out, the DBSC automatically uses the PALL (precharge all banks) command to precharge all of the banks, and then issues the REF (auto-refresh) command. Consequently, after the refresh takes place, data access for all of the banks will be in the missed page state.

Note: Even if the refresh counter becomes Level 0, refresh is executed in the break of a 16-byte data transfer. If the counter value in this break point is less than Level 0, an auto-refresh is executed. Set the Level 0 value in consideration of the clock cycle that handles a 16-byte data access (Pre-Act-Wr-Wr command interval). Note that a refresh is executed during Lock.

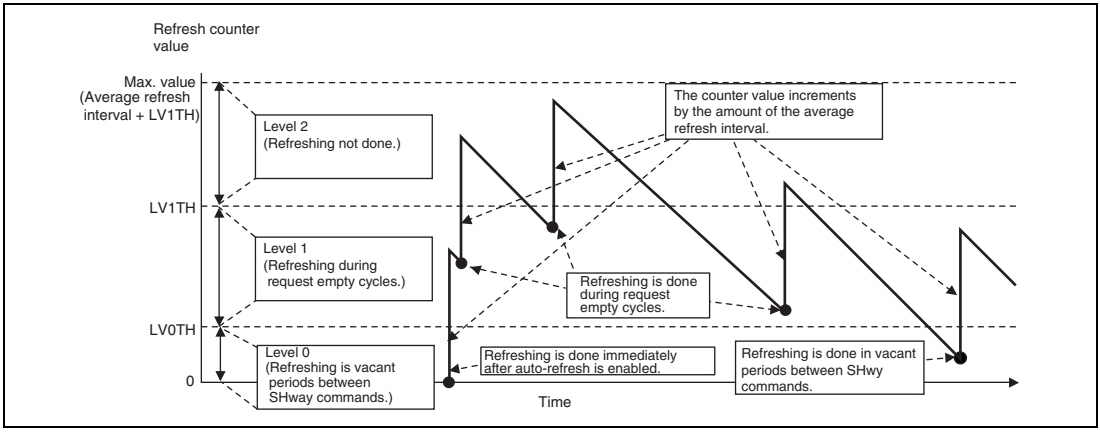


Figure 15.6 Relation between Auto-Refresh Operation and Threshold Values

15.5.6 Partial Self-Refresh Operation

If it is not necessary to access the SDRAM, the SDRAM can be put in partial self-refresh mode to reduce power consumption while still retaining data contents.

In partial self-refresh operation, only a half or a quarter of the number of banks is able to transit to self-refresh. Writing to DBMRCNT issues the EMRS command to the SDRAM and sets partial self-refresh.

15.5.7 Power-Down Operation

If it is not necessary to access the SDRAM, the SDRAM can be put in power-down mode to reduce power consumption by making the SDRAM internal clock inactive.

Shifting to power-down mode is done by writing 1 to the power-down bit (PDN) in the SDRAM refresh/power-down control register 0 (DBRFPDN0). Self-refresh mode can be cancelled by writing 0 to the PDN bit. Auto-refresh makes SDRAM data contents retain during power-down mode.

The following shows an example of the power-down sequence. For detailed information, please refer to the datasheet for the SDRAM being used.

1. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 0 (access disabled).
2. Use the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
3. Set the PDN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 to make a transition to power-down mode.

Use the following procedure to cancel power-down mode.

1. Set the PDN bit in DBRFPDN0 to 0, to cancel power-down mode.
2. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 1 (access enabled).

15.5.8 Deep Power-Down Operation

If it is not necessary to access the SDRAM, the SDRAM can be put in deep power-down mode to reduce power consumption by making the SDRAM internal clock inactive.

Shifting to deep power-down mode is done by writing 1 to deep power-down bit (DPDN) in the SDRAM refresh/power-down control register 0 (DBRFPDN0). Deep power-down mode can be cancelled by writing 0 to the DPDN bit.

The following shows an example of the initialization sequence. For detailed information, please refer to the datasheet for the mobile-DDR-SDRAM being used.

1. Make sure the DBSC is not being accessed.
2. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 0 (access disabled).
3. Set the ARFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 0 (automatic issue of auto-refresh disabled).
4. Use the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
5. Set the DPDN bit in SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 to make a transition to deep power-down mode.

During this period, changing the clock frequency or stopping the clock is available.

Use the following procedure to cancel deep power-down mode.

1. Set the DPDN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 0, to cancel deep power-down mode.
2. Wait for at least 200μs until access to SDRAM becomes available.
3. Use the CMD bits in the SDRAM command control register (DBCMDCNT) to issue the PALL (precharge all banks) command.
4. Write to the SDRAM mode setting register (DBMRCNT) to issue the MRS command and EMRS command to the SDRAM and sets the various parameters.
5. Write to the CMD bits in the SDRAM command control register (DBCMDCNT) twice to issue the REF command twice.
6. Set the ACEN bit in the SDRAM operation enable register (DBEN) to 1 (access enabled).
7. Set the ARFEN bit in the SDRAM refresh/power-down control register 0 (DBRFPDN0) to 1 (automatic issue of auto-refresh enabled).
8. Access to SDRAM becomes available.

15.5.9 Relation between SDRAM Address Pins and Logical Addresses

The following shows the relation between SDRAM address pins and the logical address on SHwy transaction.

- DDR2-SDRAM

Table 15.16 Relation between SDRAM Address Pins and Logical Addresses when the External Data Bus Width is Set to 32 Bits (BWIDTH = B'10)

Memory		BA2	BA1	BA0	MA14	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A12	A11	—	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	—	AP	—	A10	A9	A8	A7	A6	A5	A4	A3	A2
32M × 16b	ROW	—	A13	A12	—	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A13	A12	—	—	—	—	AP	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2
64M × 16b	ROW	A14	A13	A12	—	—	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15
	COL	A14	A13	A12	—	—	—	—	AP	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2
128M × 16b	ROW	A14	A13	A12	—	A28	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15
	COL	A14	A13	A12	—	—	—	—	AP	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2

Notes: 1. A31 to A0 are the logical address bits of a byte unit. A31 is MSB and A0 is LSB.
2. AP stands for Auto Precharge Option.

Table 15.17 Relation between SDRAM Address Pins and Logical Addresses when the External Data Bus Width is Set to 16 Bits (BWIDTH = B'01)

Memory		BA2	BA1	BA0	MA14	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M×16b	ROW	—	A11	A10	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1
32M×16b	ROW	—	A12	A11	—	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
32M×8b	ROW	—	A12	A11	—	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64M×16b	ROW	A13	A12	A11	—	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
64M×8b	ROW	—	A12	A11	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
128M×16b	ROW	A13	A12	A11	—	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1
128M×8b	ROW	A13	A12	A11	—	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	A13	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2	A1

Note: 1. A31 to A0 are the logical address bits of a byte unit. A31 is MSB and A0 is LSB.
2. AP stands for Auto Precharge Option.

- Mobile-DDR-SDRAM

Table 15.18 Relation between SDRAM Address Pins and Logical Addresses when the External Data Bus Width is Set to 32 Bits (BWIDTH = B'10)

Memory		BA2	BA1	BA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
4M × 32b	ROW	—	A11	A10	—	—	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	AP	—	—	A9	A8	A7	A6	A5	A4	A3	A2
8M × 32b	ROW	—	A12	A11	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	—	A10	A9	A8	A7	A6	A5	A4	A3	A2
16M × 32b	ROW	—	A11	A10	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	AP	—	—	A9	A8	A7	A6	A5	A4	A3	A2
16M × 32b	ROW	—	A12	A11	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	—	A10	A9	A8	A7	A6	A5	A4	A3	A2
32M × 32b	ROW	—	A13	A12		A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A13	A12	—	—	—	AP	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2
32M × 32b	ROW	—	A12	A11	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	—	A10	A9	A8	A7	A6	A5	A4	A3	A2
64M × 32b	ROW	—	A13	A12	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A13	A12	—	—	—	AP	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2
8M × 16b	ROW	—	A12	A11	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	—	A10	A9	A8	A7	A6	A5	A4	A3	A2
8M × 16b	ROW	—	A11	A10	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	AP	—	—	A9	A8	A7	A6	A5	A4	A3	A2
16M × 16b	ROW	—	A12	A11	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A12	A11	—	—	—	AP	—	A10	A9	A8	A7	A6	A5	A4	A3	A2
32M × 16b	ROW	—	A13	A12	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A13	A12	—	—	—	AP	A1	A1	A9	A8	A7	A6	A5	A4	A3	A2
64M × 16b	ROW	—	A13	A12	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A13	A12	—	—	—	AP	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2

Notes: 1. A31 to A0 are the logical address bits of a byte unit. A31 is MSB and A0 is LSB.
2. AP stands for Auto Precharge Option.

Table 15.19 Relation between SDRAM Address Pins and Logical Addresses when the External Data Bus Width is Set to 16 Bits (BWIDTH = B'01)

Memory		BA2	BA1	BA0	MA13	MA12	MA11	MA10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M×16b	ROW	—	A11	A10	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13
	COL	—	A11	A10	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2
32M×16b	ROW	—	A12	A11	—	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2
32M×8b	ROW	—	A12	A11	—	—	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2
64M×16b	ROW	A13	A12	A11	—	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15
	COL	A13	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2
64M×8b	ROW	—	A12	A11	—	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14
	COL	—	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2
128M×16b	ROW	A13	A12	A11	—	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15
	COL	A13	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2
128M×8b	ROW	A13	A12	A11	—	A27	A26	A25	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15
	COL	A13	A12	A11	—	—	—	—	AP	A10	A9	A8	A7	A6	A5	A4	A3	A2

Notes: 1. A31 to A0 are the logical address bits of a byte unit. A31 is MSB and A0 is LSB.
2. AP stands for Auto Precharge Option.

15.5.10 Alignment of the Position of Bank Address

The address location for the bank address can be specified without SDRAM configuration dependency. The bank address can be used for contiguous address and non-contiguous address.

(1) Settings for the Contiguous Address

Set BKADM to 00 for the contiguous address. In this case, the bank address location is specified by BKADP. The bank address location settings can be selected either setting in the upper column address or specifying the address location. In a case that the address location is specified, specifies the lower bit location of the bank address (BA0).

Describes a case that one SDRAM (16-bit width 32 M × 16 bits) connection with DDR2-SDRAM when the external bus width is set to 16-bit width.

Table 15.20 BKADM = "B'00", BKADP = "B'000000"

Memory Type		BA2	BA1	BA0	MA 14	MA 13	MA 12	MA 11	MA 10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A11	A10	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 15.21 BKADM = "B'00", BKADP = "B'001010"

Memory Type		BA2	BA1	BA0	MA 14	MA 13	MA 12	MA 11	MA 10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A11	A10	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 15.22 BKADM = "B'00", BKADP = "B'001100"

Memory Type		BA2	BA1	BA0	MA 14	MA 13	MA 12	MA 11	MA 10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A13	A12	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A11	A10
	COL	—	A13	A12	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1

(2) Settings for the Non-Contiguous Address

Set BKADM to 01 for the non-contiguous address. In this case, the lower bank address location is specified by BKADP and the upper bank address location is specified by BKADB. The lower bank address location settings can be selected either setting in the upper column address or specifying the address location. In a case that the address location is specified, specifies the lower bit location of the bank address (BA0). The address location for the upper bank address location can be specified. In a case that the address location is specified, specifies the upper bit location of the bank address (BA1). BA2 is located the upper of BA1 for the 8-bank product.

Describes a case that one SDRAM (16-bit width 64 M × 16 bits) connection with DDR2-SDRAM when the external bus width is set to 16-bit width.

Table 15.23 BKADM = "B'00", BKADP = "B'000000", BKADB = "B'000000"

Memory Type		BA2	BA1	BA0	MA 14	MA 13	MA 12	MA 11	MA 10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A11	A10	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A13	A12
	COL	—	A11	A10	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 15.24 BKADM = "B'01", BKADP = "B'000000", BKADB = "B'001101"

Memory Type		BA2	BA1	BA0	MA 14	MA 13	MA 12	MA 11	MA 10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A13	A10	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A16	A15	A14	A12	A11
	COL	—	A13	A10	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1

Table 15.25 BKADM = "B'01", BKADP = "B'001100", BKADB = "B'010000"

Memory Type		BA2	BA1	BA0	MA 14	MA 13	MA 12	MA 11	MA 10	MA9	MA8	MA7	MA6	MA5	MA4	MA3	MA2	MA1	MA0
16M × 16b	ROW	—	A16	A12	—	—	A24	A23	A22	A21	A20	A19	A18	A17	A15	A14	A13	A11	A10
	COL	—	A16	A12	—	—	—	—	AP	—	A9	A8	A7	A6	A5	A4	A3	A2	A1

(3) Combination of the bank address setting

The following shows the bank address position in DBCONF. The setting other than the following is prohibited.

Table 15.26 Combination of Bank Address Setting

BKADM	BKADP	BKADB
00	000000	000000
00	001010	000000
00	001011	000000
00	001100	000000
01	001010	001101
01	001010	001110
01	001010	001111
01	001010	010000
01	001011	001101
01	001011	001110
01	001011	001111
01	001011	010000
01	001100	001101
01	001100	001110
01	001100	001111
01	001100	010000

15.5.11 SDRAM Access and Timing Constraints

In this section, waveforms at the various pins during basic SDRAM access are explained first and then the relation between SDRAM access and the CAS latency (CL), tRAS, tRFC, tRCD, tRP, tRRD, tWR, tRTP, tRC, READ-WRITE minimum interval, WRITE-READ minimum interval set using the SDRAM timing registers 0 to 2 (DBTR0 to DBTR2) is explained.

(1) Basic SDRAM Access

In this section, waveforms at the various pins during basic SDRAM access, including reading, writing, auto-refresh, and self-refresh operations, are explained.

Figure 15.7 shows waveforms for 1-/2-/4-/8-/16-byte reading when the bus width is set to 32 bits. In this case, single-reading is performed in which the READ command is issued once. In this example, read access processing is executed for bank A after the ACT command is issued, but when there is a page hit, access begins with the issue of the READ command.

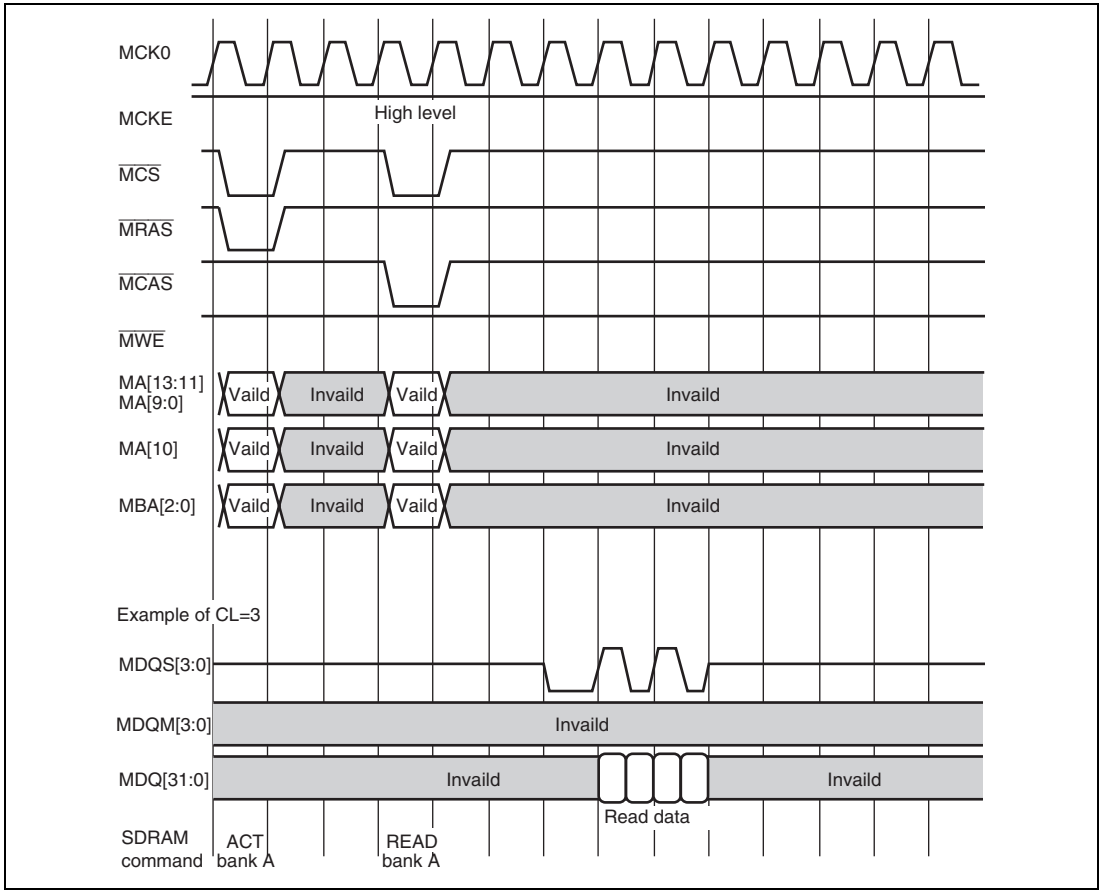


Figure 15.7 Waveforms for 1-/2-/4-/8-/16-Byte Reading (When the Bus Width Is Set to 32 Bits)

Figure 15.8 shows waveforms for 32-byte reading when the bus width is set to 32 bits. In this case, the READ command is issued twice. In this example, read access processing is executed for bank A after the ACT command is issued, but when there is a page hit, access begins with the issue of the READ command.

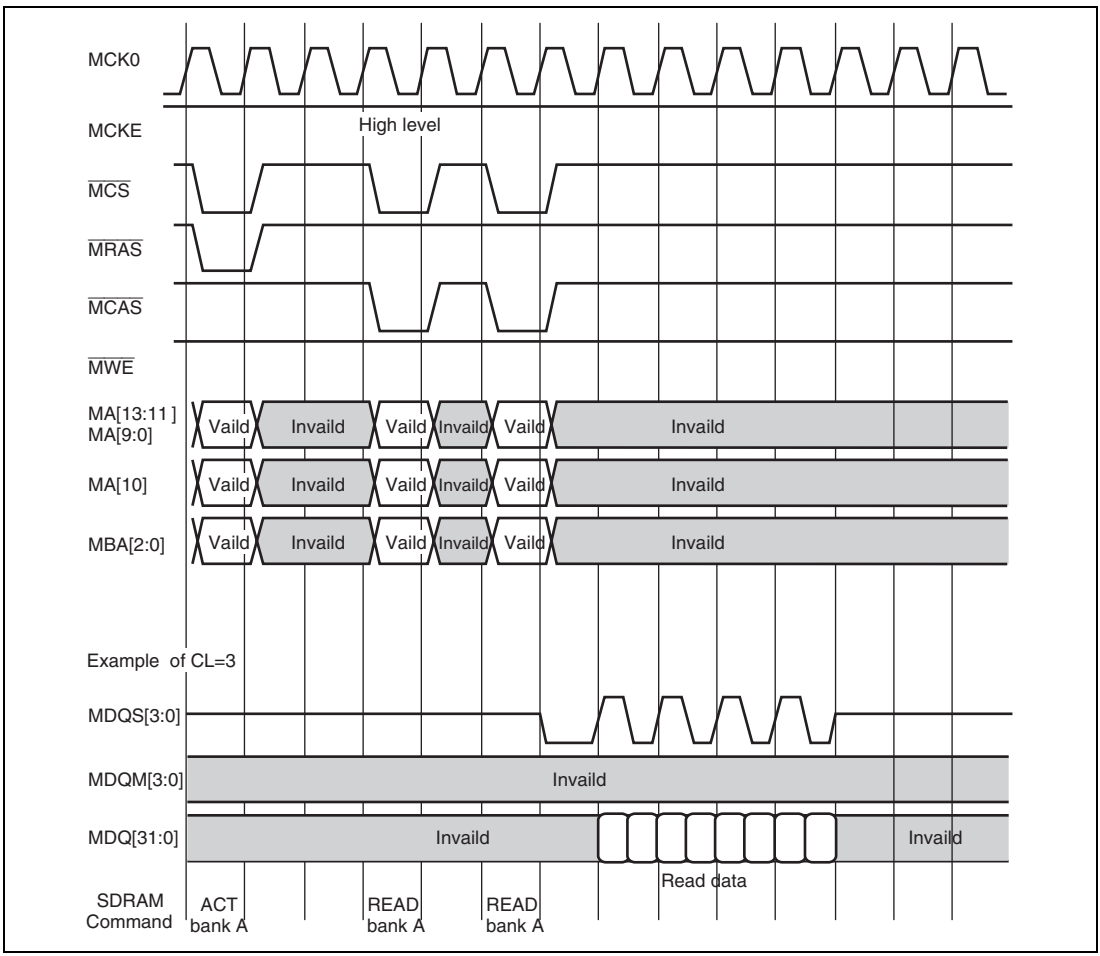


Figure 15.8 Waveforms for 32-Byte Reading (When the Bus Width Is Set to 32 Bits)

Figure 15.9 shows waveforms for 1-/2-/4-/8-/16-byte writing when the bus width is set to 32 bits. In this case, single-writing is performed in which the WRITE command is issued once. In this example, write access processing is executed for bank A after the ACT command is issued, but when there is a page hit, access begins with the issue of the WRITE command.

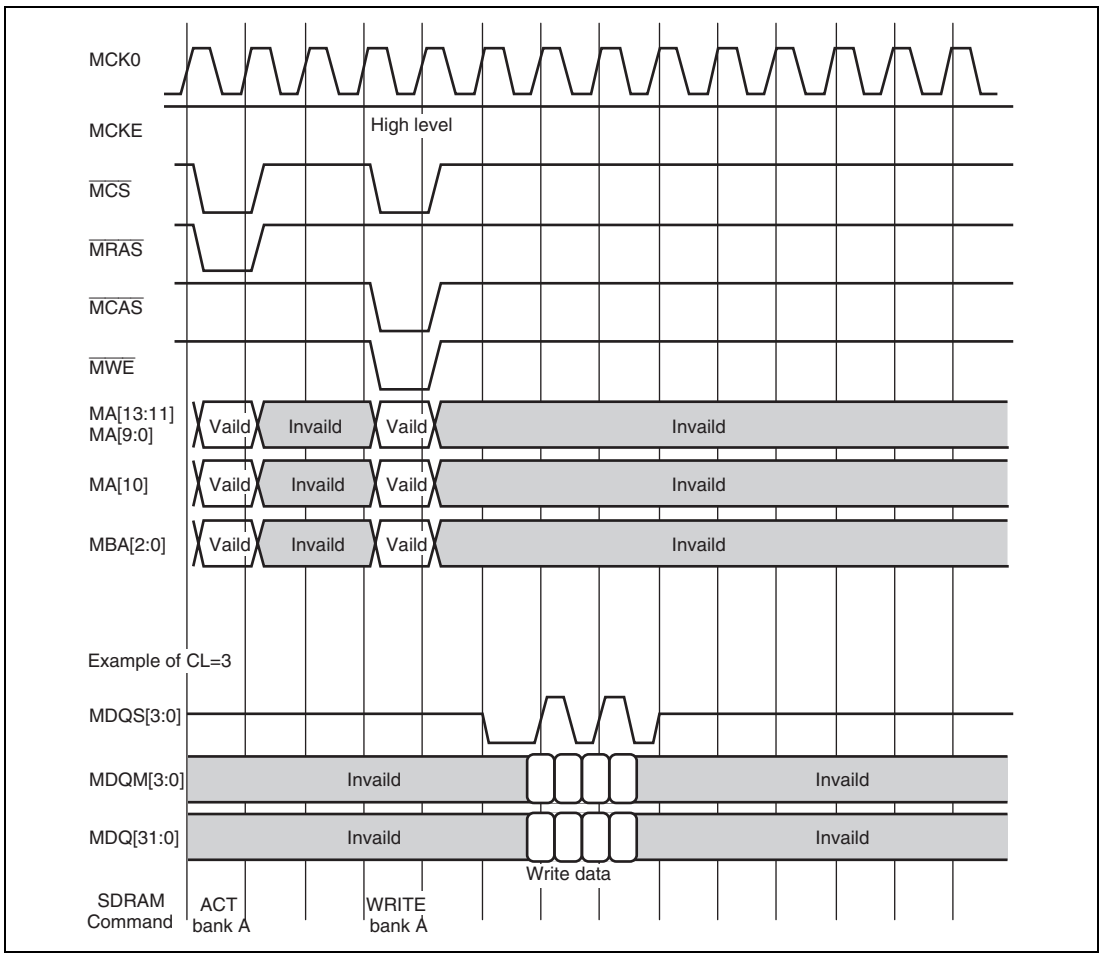


Figure 15.9 Waveforms for 1-/2-/4-/8-/16-Byte Writing (When the Bus Width Is Set to 32 Bits)

Figure 15.10 shows waveforms for 32-byte writing when the bus width is set to 32 bits. In this case, the WRITE command is issued twice. In this example, write access processing is executed for bank A after the ACT command is issued, but when there is a page hit, access begins with the issue of the WRITE command.

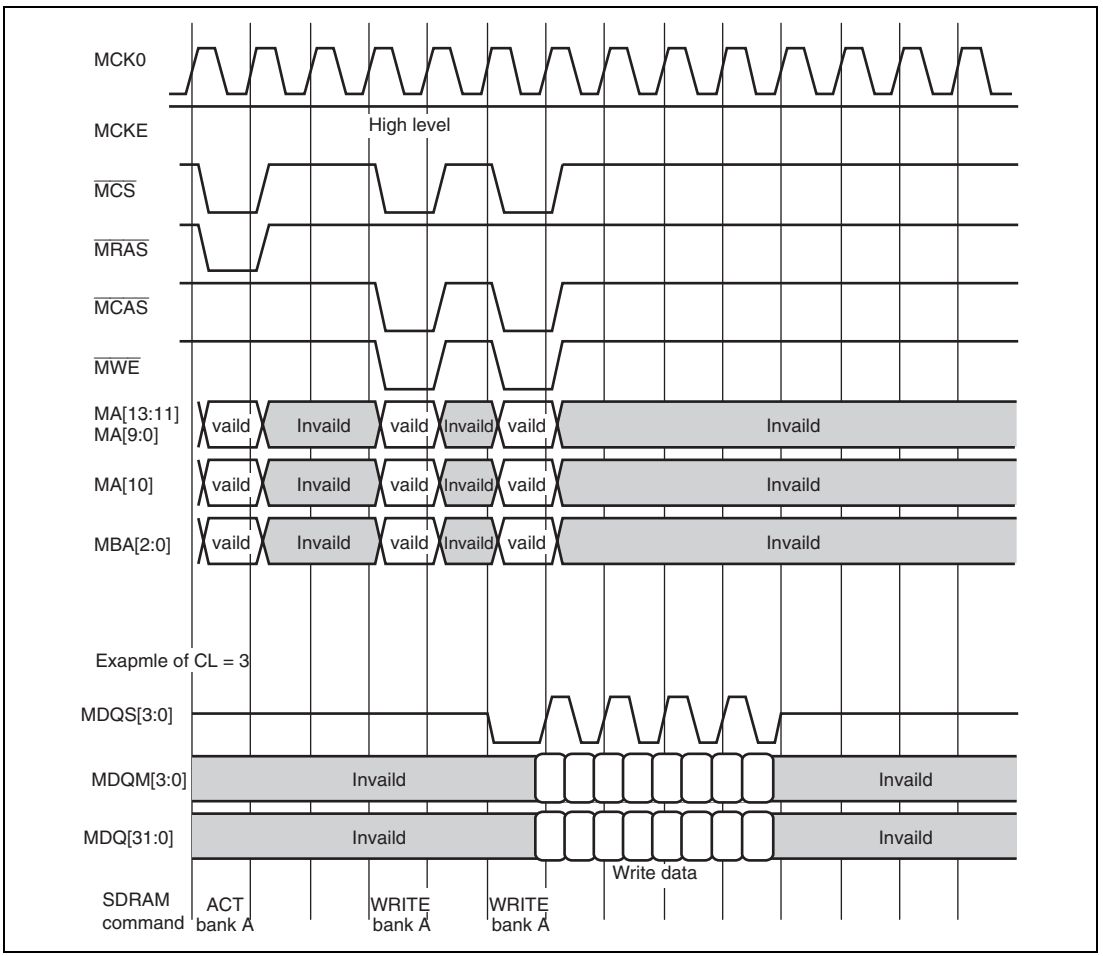


Figure 15.10 Waveforms for 32-Byte Writing (When the Bus Width is Set to 32 Bits)

Figure 15.11 shows waveforms during auto-refresh operation resulting from settings of the SDRAM refresh control registers 0, 1, and 2. The DBSC issues a REF command automatically after the PALL command is issued when at least one SDRAM bank is activated before the REF command. Consequently, there is no need to use software to manage precharging of all the banks for the auto-refresh operation.

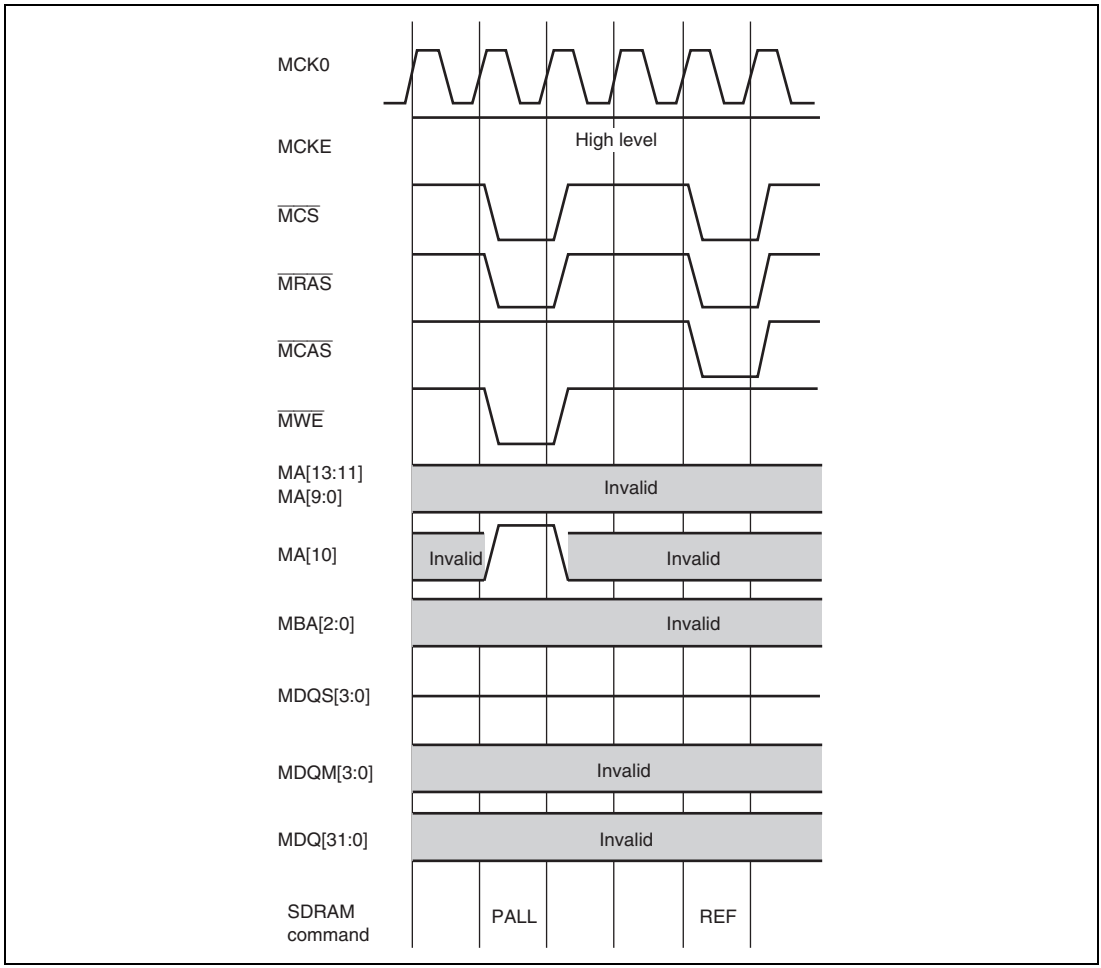


Figure 15.11 Auto-Refresh Operation

Figure 15.12 shows the self-refresh operation. In order to perform self-refresh operation, the sequence must be observed. For details, refer to section 15.5.4, Self-Refresh Operation.

When performing processing according to the sequence in section 15.5.4, Self-Refresh Operation, commands to be issued to the SDRAM are those shown in figure 15.12. Before the transition to self-refresh, the PALL command is issued in software. Then, software is used to issue the REF command, and the SLFRSH (self-refresh entry from IDLE) command is issued. The SDRAM continues in self-refresh mode until self-refresh is cancelled in software. After issuing the SLFRSHX (self-refresh exit) command in software, it is necessary to wait for the time (tXSNR) specified in the datasheet for the SDRAM being used until issuing a REF command.

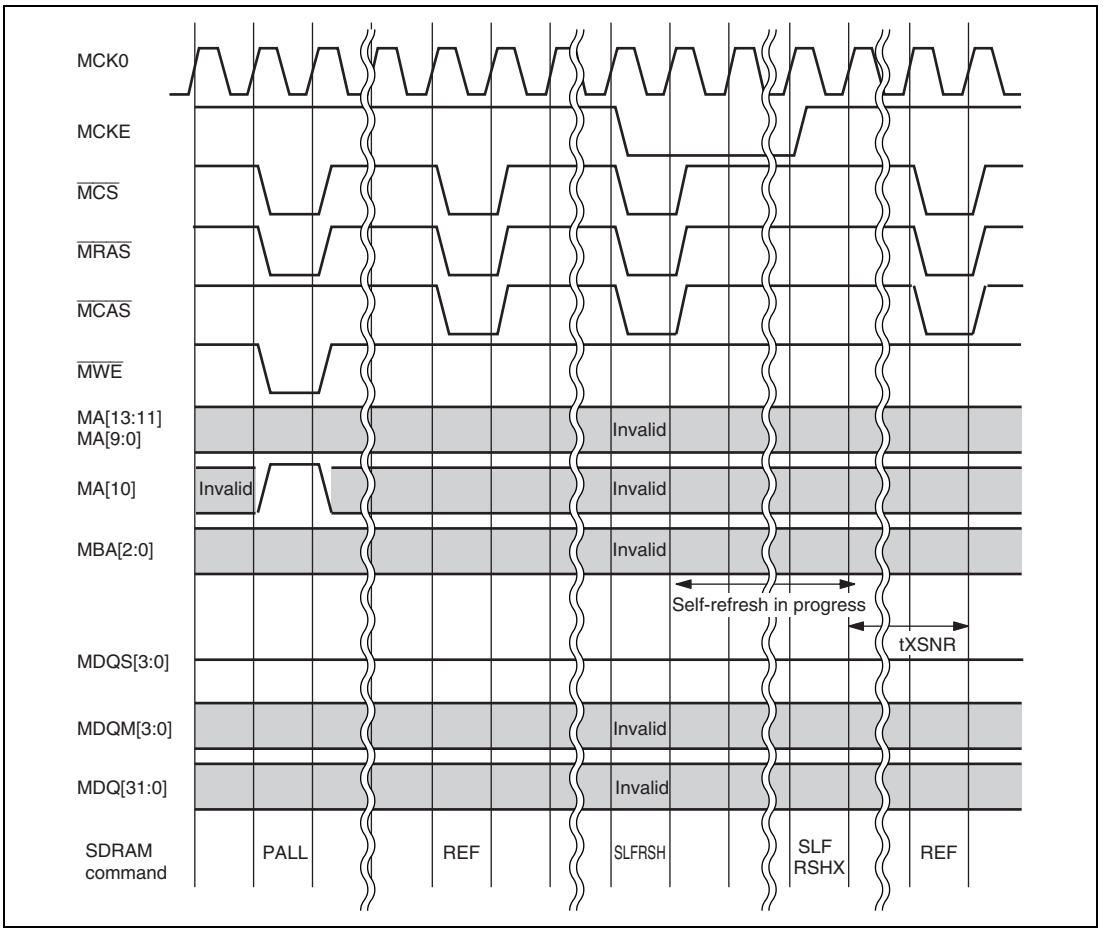


Figure 15.12 Self-Refresh Operation

(2) Timing Constraints

Figure 15.13 shows the relation among the settings of CL, tRAS, tRCD and tRP, and the issuing of commands. Figure 15.14 shows the relation between tRRD and tRTP, figure 15.15 shows the relation of tWR, figure 15.16 shows the relation of tRC, figure 15.17 shows the relation of READ-WRITE, figure 15.18 shows the relation of WRITE-READ, and figure 15.19 shows the relation of tRFC.

Figure 15.13 corresponds to operation in a case in which, with bank A open, there is a read access of bank A and a page miss occurs. The constraint tRP between the PRE command and ACT command, the constraint tRCD between the ACT command and READ command, and the constraint tRAS between the ACT command and the PRE command are involved. The DBSC waits to issue commands until each of the constraints is satisfied.

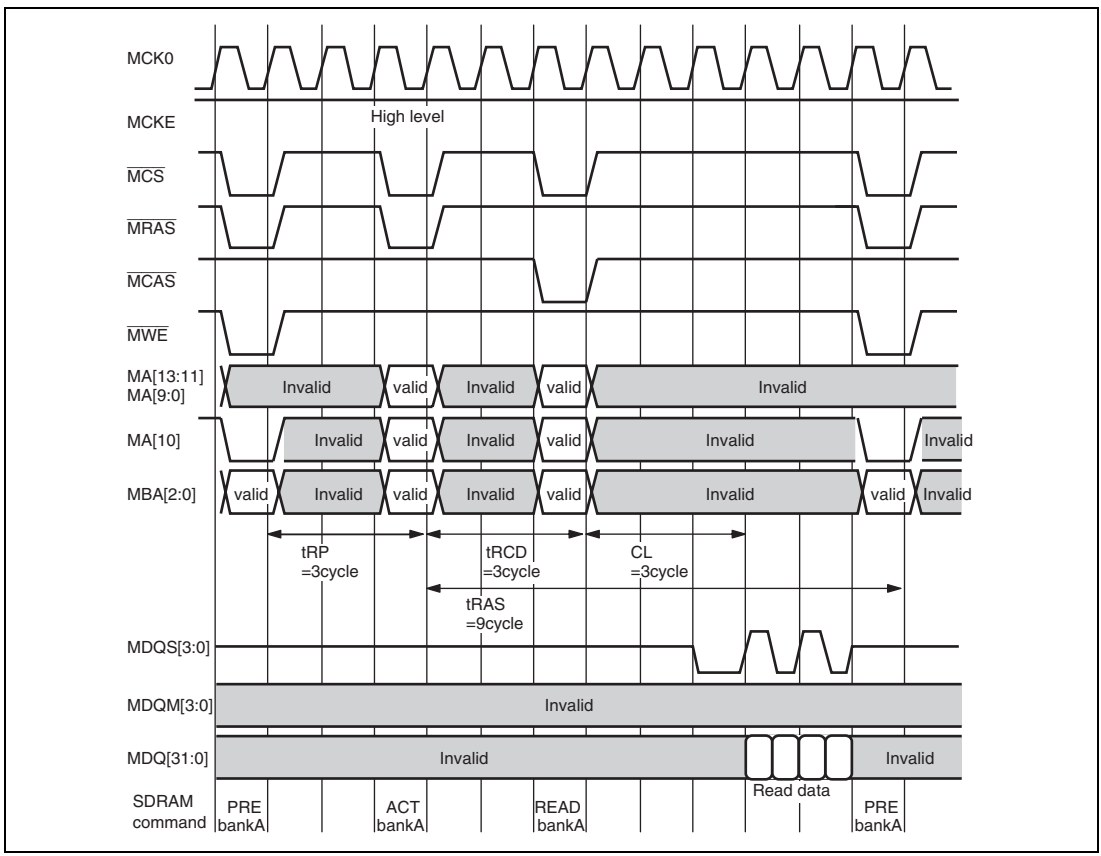


Figure 15.13 tRP, tRCD, CL, and tRAS

Figure 15.14 shows a case in which the pages for both of banks A and B are closed, the page for bank C is open, and a page hit has occurred. When the tRRD time constraint has been satisfied starting from issue of the ACT command for bank A, the ACT command for bank B is issued. Because time tRCD has elapsed from the issue of the ACT command for bank A, a READ command can be used. The READ command has a burst length of 4, so after two cycles a READ command for bank B can be issued. A further two cycles later, a READ command for bank C can be issued. However, the next request is an access for which bank C must be closed, and so after the elapse of time tRTP a PRE command is issued.

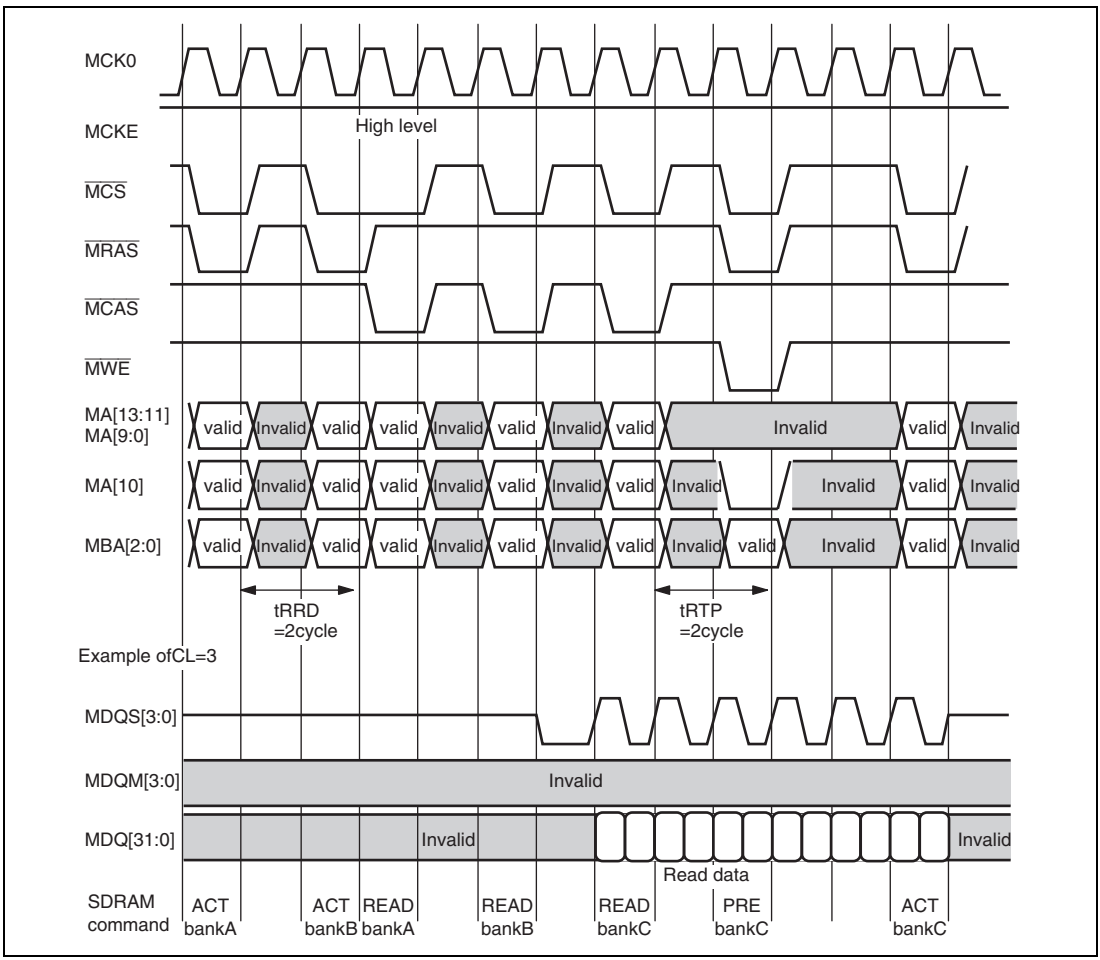


Figure 15.14 tRRD and tRTP

Figure 15.15 shows a case in which, after a write request, access occurs requiring that bank B be closed. After the issue of a WRITE command, it is necessary to wait for time tWR or longer after output of the write data before issuing a PRE command.

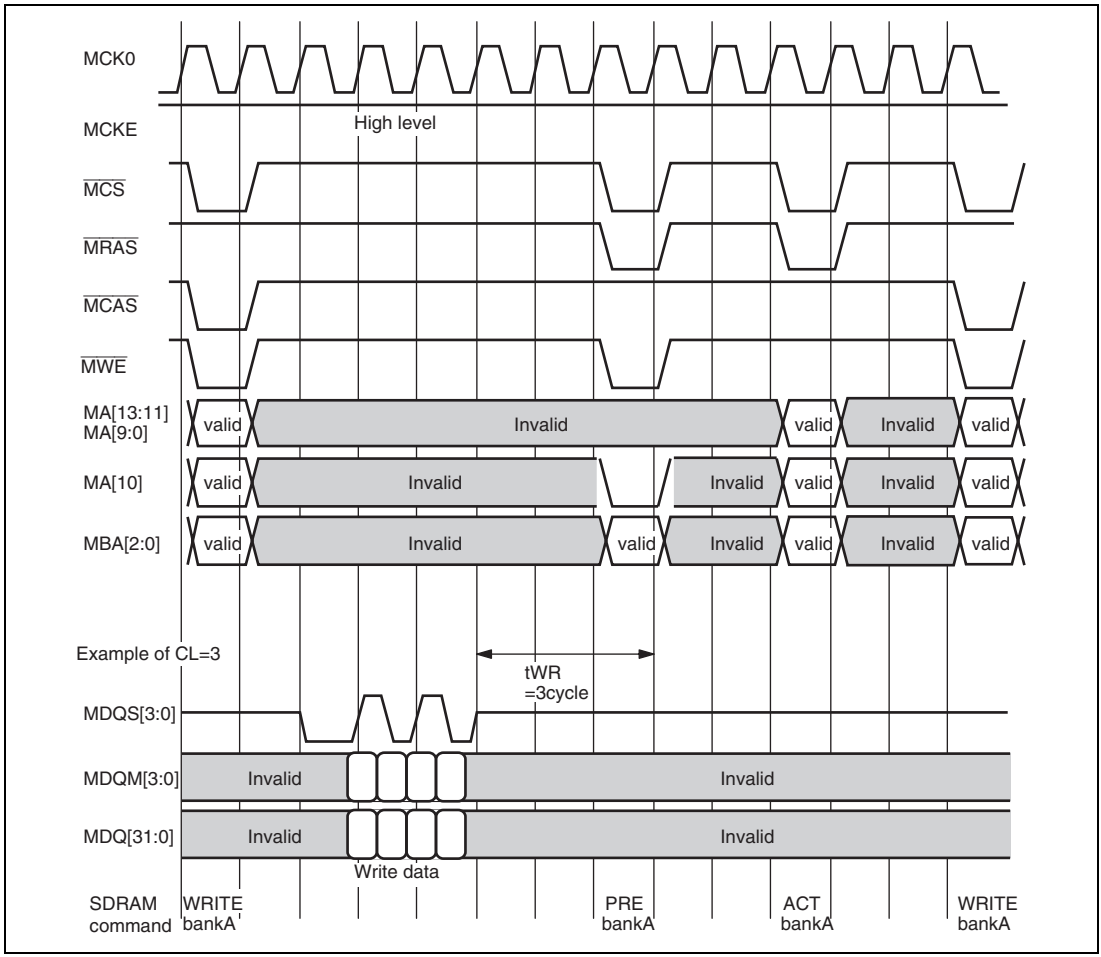


Figure 15.15 tWR

Figure 15.16 shows an example of performing auto-refresh after read access of bank A, the page for which had been closed. After issuing an ACT command and READ command for bank A and performing data reading, a PALL command must be used to close all banks in order to perform auto-refresh. In order to issue the PALL command, the tRAS time constraint must be satisfied, and issuing of the PALL command is delayed until this time. Then, when issuing the REF command, both of time constraints tRP and tRC must be satisfied simultaneously. When these constraints are both satisfied, the REF command is issued and auto-refresh is performed.

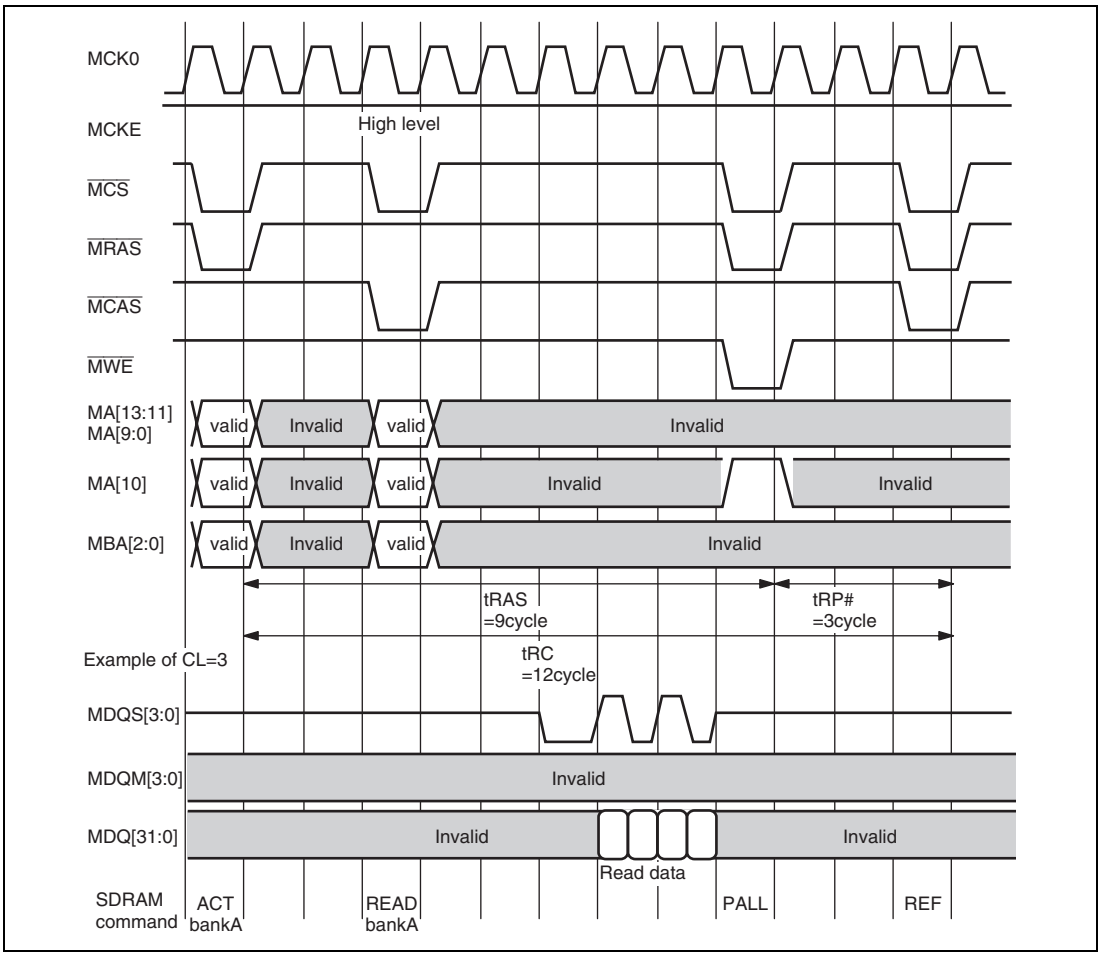


Figure 15.16 tRC

Figure 15.17 is an example of a case in which, after issuing a READ command, a WRITE command is issued. In order to issue the WRITE command after issuing the READ command, the DBSC waits for a minimum time stipulated by the RDWR bits.

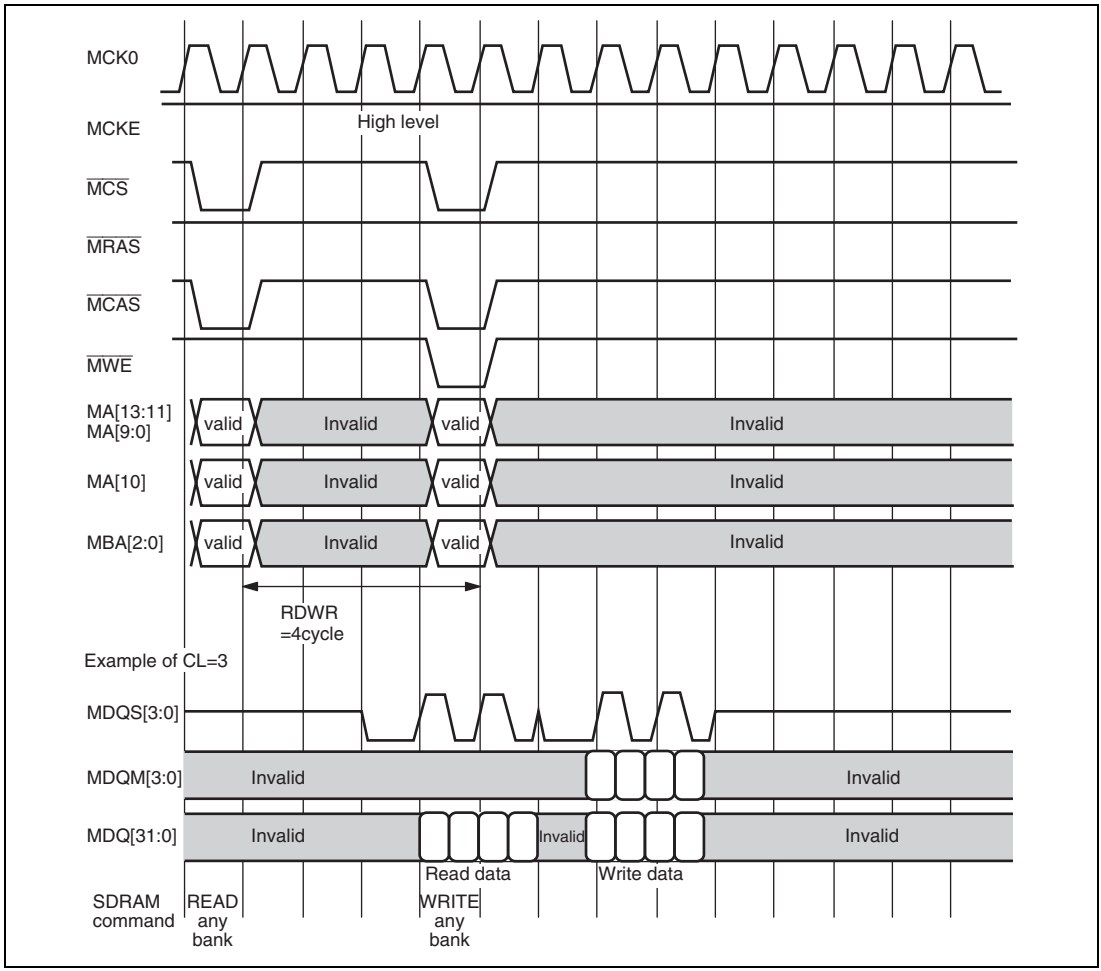


Figure 15.17 READ-WRITE Minimum Time

Figure 15.18 is an example of a case in which, after issuing a WRITE command, a READ command is issued. In order to issue the READ command after issuing the WRITE command, the DBSC waits for a minimum time stipulated by the WRRD bits.

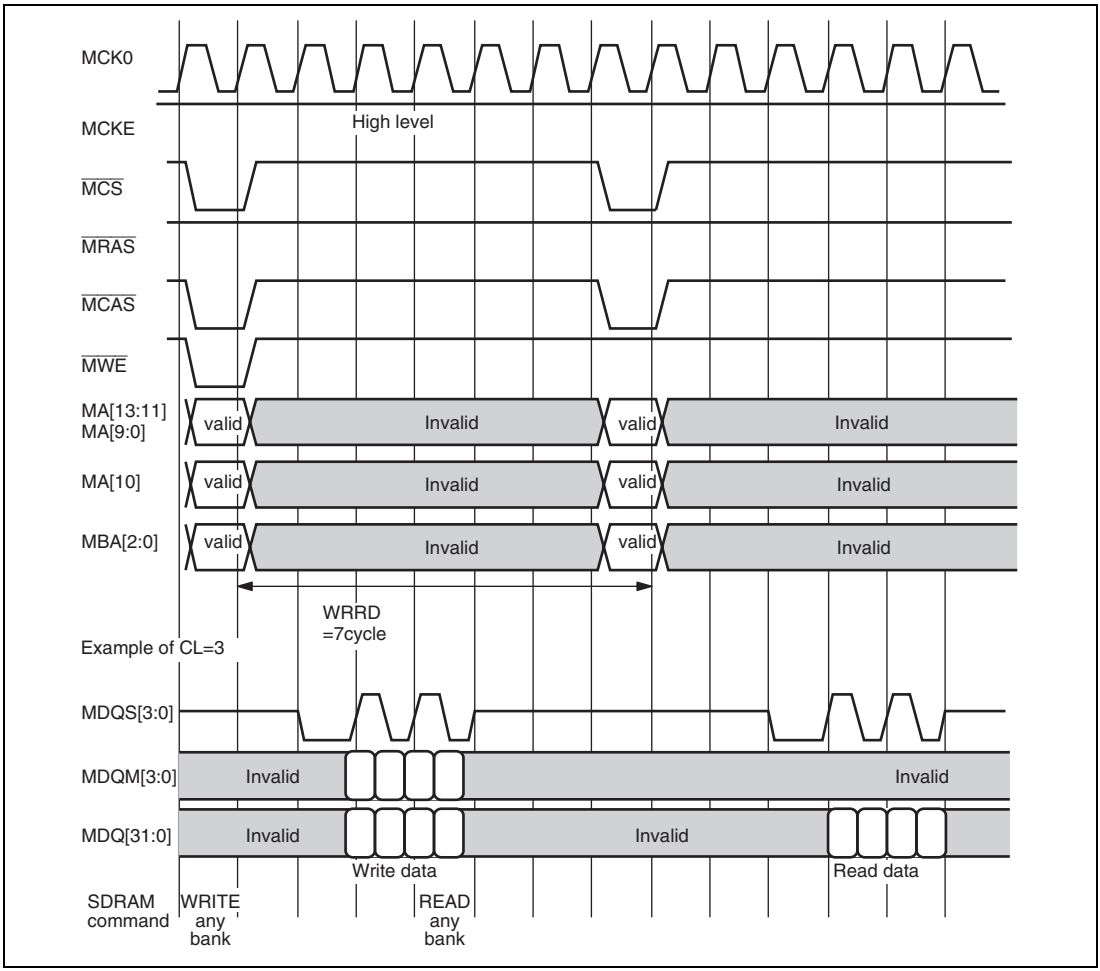


Figure 15.18 WRITE-READ Minimum Time

Figure 15.19 is an example of a case in which, after issuing a REF command, a READ request is issued. In order to issue the ACT commend after issuing the REF command, the DBSC waits for a time stipulated by tRFC.

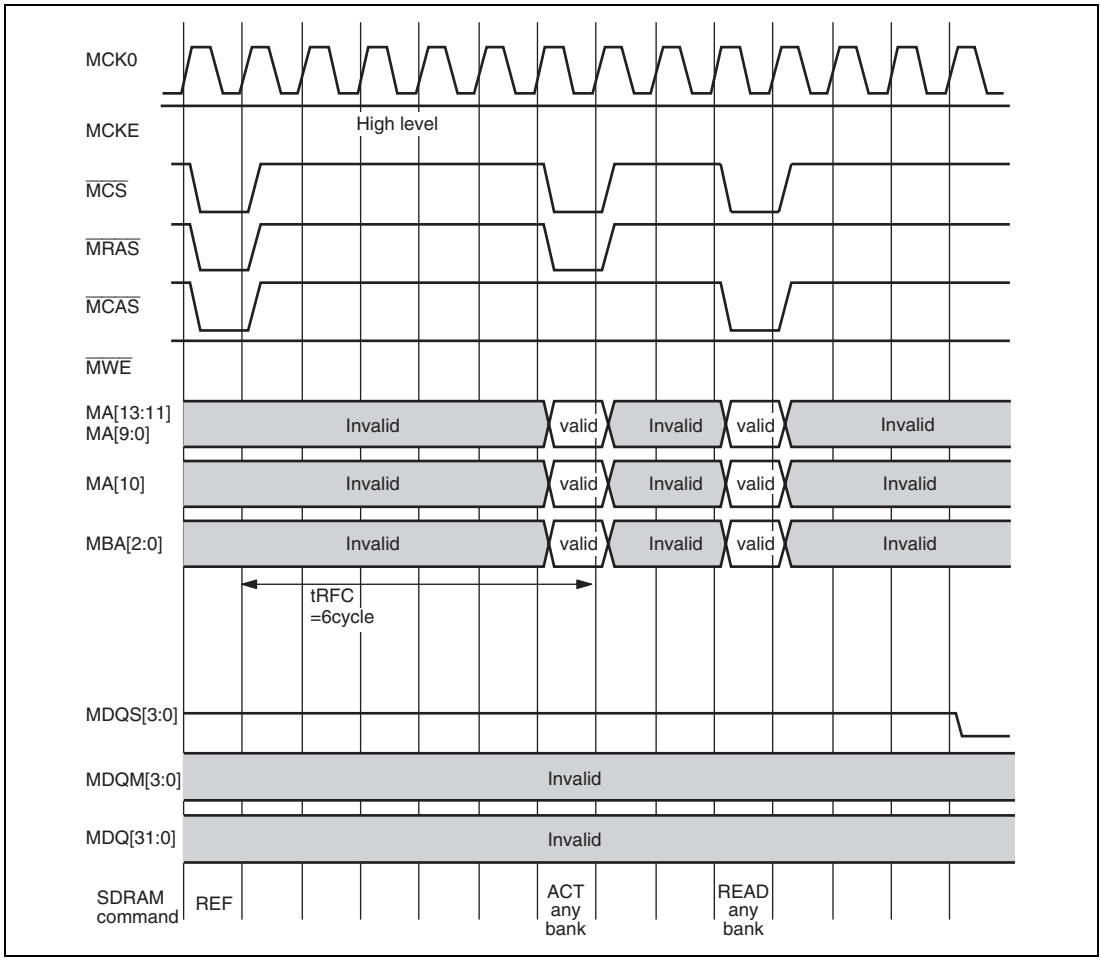


Figure 15.19 tRFC

15.5.12 Method for Securing Time Required for Operations Including Initialization and Self-Refresh Cancellation

When using DBSC register settings to set initialization, cancel self-refresh and the like, it is necessary to wait a time stipulated by the SDRAM specifications. One example of this waiting is the method used to read the DBSC status register (DBSTATE). Upon executing reading of the DBSC status register (DBSTATE), a minimum of eight cycles of the memory clock elapse. If operation is at 166 MHz, then approximately 48 ns elapses in a single DBSTATE read operation. This can be utilized to secure the required time, by repeating read access the necessary number of times.

15.5.13 Operation of MCKE Signal

Operation of the MCKE signal is explained with the aid of figure 15.20. By fixing the MSLD pin to the low level when DDR2-SDRAM is in use, or to the high level when mobile-DDR-SDRAM is in use, levels as requested by the SDRAM are output on the MCKE pin.

After release from the power-on reset state, writing B'1 to the CKEEN bits in the CKE setting register (DBCKECNT) leads to output of the MCKE signal at the high level, which corresponds to the enabled state. After the MCKE signal has been placed at the high level in this way, the level of the MCKE signal changes in accord with state transitions of the DBSC, becoming low during self-refreshing and periods in the power-down or deep power-down state.

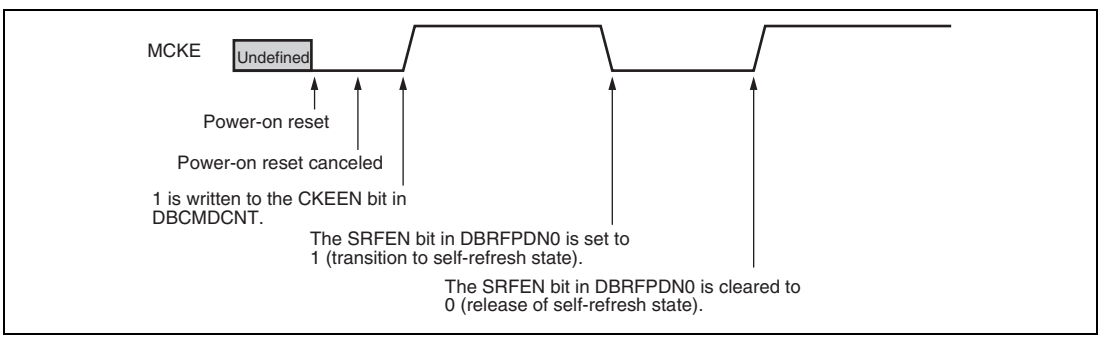


Figure 15.20 Operation of MCKE Signal (When MSLD is at Low Level)

15.6 Usage Notes

15.6.1 Important Information Regarding Use of 8-Bank DDR2-SDRAM Products

The DDR2-SDRAM specifications limit the number of banks in an 8-bank product which can be activated simultaneously. Control must be executed so that the number of activated banks never exceeds four banks. Hence the DBSC handles (BA2, BA1, BA0) = (1, X, Y) and (0, X, Y) as access to the same banks. Through this handling, no more than four banks can be activated simultaneously. As an operation example, consider a case in which the page corresponding to bank (BA2, BA1, BA0) = (0, 0, 0) is opened, and then access of (BA2, BA1, BA0) = (1, 0, 0) occurs. After using a PRE command to close the page of the bank corresponding to (BA2, BA1, BA0) = (0, 0, 0), the DBSC issues an ACT command for the bank corresponding to (BA2, BA1, BA0) = (1, 0, 0) to open the page, and accesses the memory. Because the DBSC executes the above control, if a program which is activated simultaneously is placed in an address area such that (BA2, BA1, BA0) = (1, X, Y) and (0, X, Y), frequent page misses may result.

15.6.2 Important Information regarding ODT Control Signal Output to SDRAM

The following should be noted when having the DBSC output an ODT control signal to the SDRAM.

- When an ODT control signal is output to the SDRAM, a CAS latency of at least three DDR clock cycles is necessary and set the db_odt_out assert period setting bit (ODTL) in the SDRAM timing resister 3 to 2 clock cycles. As figure 15.21 shows, it will assert MODT signals.

The DBSC supports only the memory for which tAOND is 2 cycles and tAOFD is 2.5 cycles.

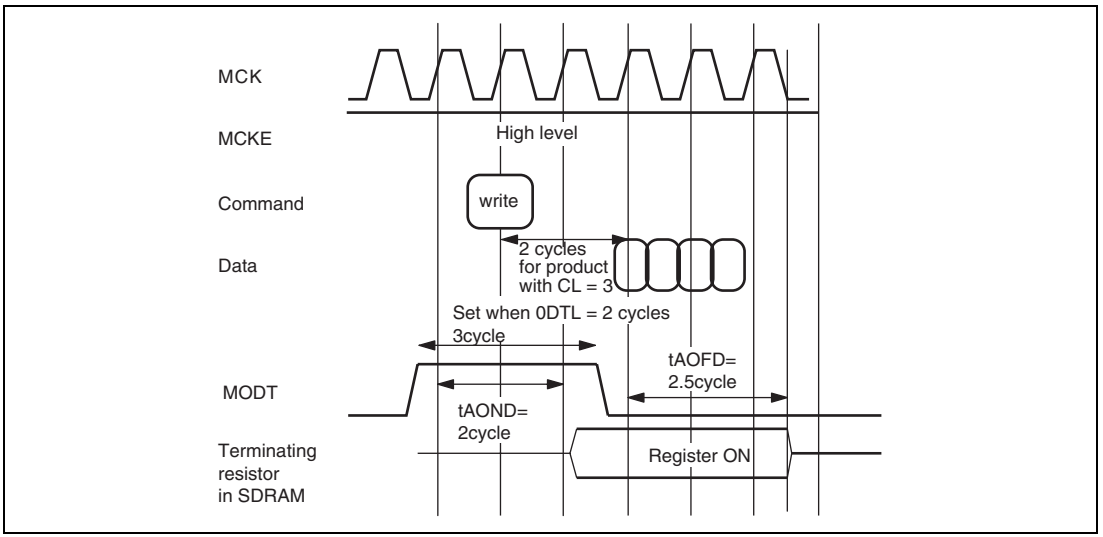


Figure 15.21 ODT Control Signal when CL = 3

15.6.3 Point to Note when Stopping Clocks or Changing the Clock Frequency for the DBSC

Before stopping clocks or changing the clock frequency for the DBSC, be sure to shift the SDRAM to the self-refresh mode. After this has been done, the DDR-PHY block must be initialized before release from self-refreshing. For transitions to the self-refresh mode and for release from self-refreshing, follow the procedures in section 15.5.4, Self-Refresh Operation.

The above procedures are required in the following cases:

- Software standby, R-standby, U-standby; see section 18, Reset and Power-Down Modes, for details.
- Changes to the PLL multiplication ratio or Sϕ clock division ratio; see section 17, Clock Pulse Generator (CPG), for details.

Section 16 Direct Memory Access Controller (DMAC)

This LSI includes the direct memory access controller of two modules (DMAC0/1).

The DMAC0 and DMAC1 operate as independent bus-master, and can be used in place of the CPU to perform high-speed transfers between external devices that have DACK (transfer request acknowledge signal), external memory, on-chip memory, memory-mapped external devices, and on-chip peripheral modules.

16.1 Features

- Six channels for each module (12 channels in total)
- 4-Gbyte physical address space
- Data transfer unit is selectable: Byte, word (2 bytes), longword (4 bytes), 8 bytes, 16 bytes, and 32 bytes
- Maximum transfer count: 16,777,216 transfers
- Address mode: Dual address mode
- Transfer requests:

  External request, on-chip peripheral module request, or auto request can be selected.

  The following modules can issue an on-chip peripheral module request.

  — SCIF0 to SCIF2, SCIFA3 to SCIFA5, MSIOF0, MSIOF1, SDHI0, SDHI1, MMCIF, TSIF, IrDA, USB0, and USB1
- Selectable bus modes:

  Cycle steal mode (normal mode and intermittent mode) or burst mode can be selected.
- Selectable channel priority levels:

  The channel priority levels are selectable between fixed mode and round-robin mode.
- Interrupt request: An interrupt request can be generated to the CPU after half of the transfers ended, all transfers ended, or an address error occurred.

- External request detection: There are following four types of DREQ input detection (channel 0 and channel 1 of DMAC0).
  - Low-level detection, high-level detection
  - Rising-edge detection, falling-edge detection
- Transfer request acknowledge signal:
  Active levels for DACK can be set independently (channel 0 and channel 1 of DMAC0).
- Two channels can receive an external request (channel 0 and channel 1 of DMAC0).

Figure 16.1 shows the block diagram of the DMAC.

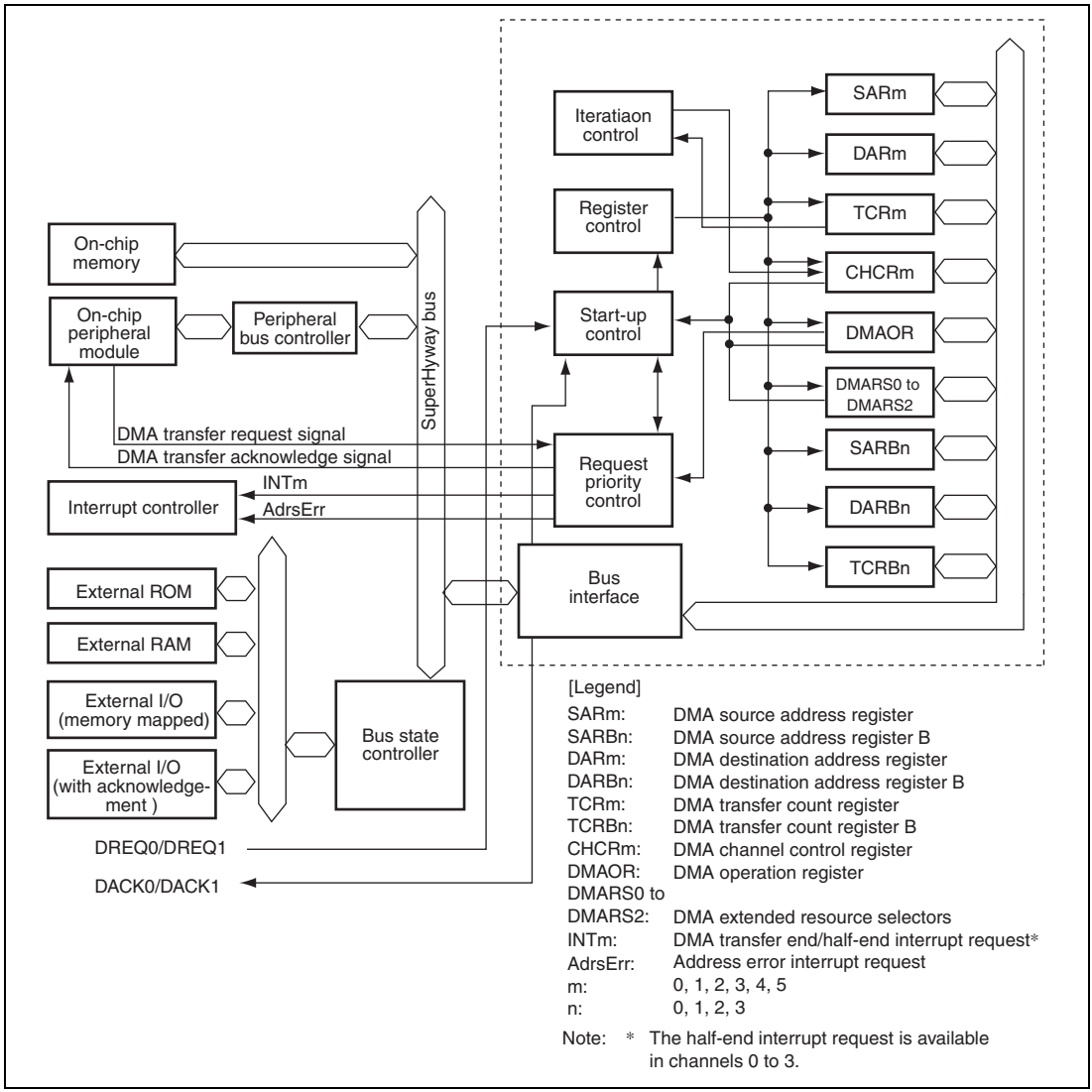


Figure 16.1 Block Diagram of DMAC

16.2 Input/Output Pins

The external pins for the DMAC0 are described below. Table 16.1 lists the configuration of the pins that are connected to external bus. The DMAC0 has pins for one channel (channel 0) for external bus use. The DMAC1 doesn't have external pins.

Table 16.1 Pin Configuration

Channel	Pin Name	Function	I/O	Description
0	DREQ0	DMA transfer request	Input	DMA transfer request input from external device to channel 0 of DMAC0
	DACK0	DMA transfer request acknowledge	Output	DMA transfer request acknowledge output from channel 0 of DMAC0 to external device
1	DREQ1	DMA transfer request	Input	DMA transfer request input from external device to channel 1 of DMAC0
	DACK1	DMA transfer request acknowledge	Output	DMA transfer request acknowledge output from channel 1 of DMAC0 to external device

16.3 Register Descriptions

Table 16.2 shows the configuration of registers of the DMAC0/1. Table 16.3 shows the state of registers in each processing mode. The SAR for channel 0 is expressed such as SAR_0.

Table 16.2 Register Configuration of DMAC

Name	Abbreviation	R/W	Address	Access Size
DMA0 source address register_0	DMA0_SAR_0	R/W	H'FE008020	32
DMA0 destination address register_0	DMA0_DAR_0	R/W	H'FE008024	32
DMA0 transfer count register_0	DMA0_TCR_0	R/W	H'FE008028	32
DMA0 channel control register_0	DMA0_CHCR_0	R/W	H'FE00802C	32
DMA0 source address register_1	DMA0_SAR_1	R/W	H'FE008030	32
DMA0 destination address register_1	DMA0_DAR_1	R/W	H'FE008034	32
DMA0 transfer count register_1	DMA0_TCR_1	R/W	H'FE008038	32
DMA0 channel control register_1	DMA0_CHCR_1	R/W	H'FE00803C	32
DMA0 source address register_2	DMA0_SAR_2	R/W	H'FE008040	32
DMA0 destination address register_2	DMA0_DAR_2	R/W	H'FE008044	32
DMA0 transfer count register_2	DMA0_TCR_2	R/W	H'FE008048	32
DMA0 channel control register_2	DMA0_CHCR_2	R/W	H'FE00804C	32
DMA0 source address register_3	DMA0_SAR_3	R/W	H'FE008050	32
DMA0 destination address register_3	DMA0_DAR_3	R/W	H'FE008054	32
DMA0 transfer count register_3	DMA0_TCR_3	R/W	H'FE008058	32
DMA0 channel control register_3	DMA0_CHCR_3	R/W	H'FE00805C	32
DMA0 operation register	DMA0_DMAOR	R/W	H'FE008060	16
DMA0 source address register_4	DMA0_SAR_4	R/W	H'FE008070	32
DMA0 destination address register_4	DMA0_DAR_4	R/W	H'FE008074	32
DMA0 transfer count register_4	DMA0_TCR_4	R/W	H'FE008078	32
DMA0 channel control register_4	DMA0_CHCR_4	R/W	H'FE00807C	32
DMA0 source address register_5	DMA0_SAR_5	R/W	H'FE008080	32
DMA0 destination address register_5	DMA0_DAR_5	R/W	H'FE008084	32
DMA0 transfer count register_5	DMA0_TCR_5	R/W	H'FE008088	32
DMA0 channel control register_5	DMA0_CHCR_5	R/W	H'FE00808C	32
DMA0 source address register B_0	DMA0_SARB_0	R/W	H'FE008120	32
DMA0 destination address register B_0	DMA0_DARB_0	R/W	H'FE008124	32

Name	Abbreviation	R/W	Address	Access Size
DMA0 transfer count register B_0	DMA0_TCRB_0	R/W	H'FE008128	32
DMA0 source address register B_1	DMA0_SARB_1	R/W	H'FE008130	32
DMA0 destination address register B_1	DMA0_DARB_1	R/W	H'FE008134	32
DMA0 transfer count register B_1	DMA0_TCRB_1	R/W	H'FE008138	32
DMA0 source address register B_2	DMA0_SARB_2	R/W	H'FE008140	32
DMA0 destination address register B_2	DMA0_DARB_2	R/W	H'FE008144	32
DMA0 transfer count register B_2	DMA0_TCRB_2	R/W	H'FE008148	32
DMA0 source address register B_3	DMA0_SARB_3	R/W	H'FE008150	32
DMA0 destination address register B_3	DMA0_DARB_3	R/W	H'FE008154	32
DMA0 transfer count register B_3	DMA0_TCRB_3	R/W	H'FE008158	32
DMA0 extended resource selector 0	DMA0_DMARS0	R/W	H'FE009000	16
DMA0 extended resource selector 1	DMA0_DMARS1	R/W	H'FE009004	16
DMA0 extended resource selector 2	DMA0_DMARS2	R/W	H'FE009008	16
DMA1 source address register_0	DMA1_SAR_0	R/W	H'FDC08020	32
DMA1 destination address register_0	DMA1_DAR_0	R/W	H'FDC08024	32
DMA1 transfer count register_0	DMA1_TCR_0	R/W	H'FDC08028	32
DMA1 channel control register_0	DMA1_CHCR_0	R/W	H'FDC0802C	32
DMA1 source address register_1	DMA1_SAR_1	R/W	H'FDC08030	32
DMA1 destination address register_1	DMA1_DAR_1	R/W	H'FDC08034	32
DMA1 transfer count register_1	DMA1_TCR_1	R/W	H'FDC08038	32
DMA1 channel control register_1	DMA1_CHCR_1	R/W	H'FDC0803C	32
DMA1 source address register_2	DMA1_SAR_2	R/W	H'FDC08040	32
DMA1 destination address register_2	DMA1_DAR_2	R/W	H'FDC08044	32
DMA1 transfer count register_2	DMA1_TCR_2	R/W	H'FDC08048	32
DMA1 channel control register_2	DMA1_CHCR_2	R/W	H'FDC0804C	32
DMA1 source address register_3	DMA1_SAR_3	R/W	H'FDC08050	32
DMA1 destination address register_3	DMA1_DAR_3	R/W	H'FDC08054	32
DMA1 transfer count register_3	DMA1_TCR_3	R/W	H'FDC08058	32
DMA1 channel control register_3	DMA1_CHCR_3	R/W	H'FDC0805C	32
DMA1 operation register	DMA1_DMAOR	R/W	H'FDC08060	16
DMA1 source address register_4	DMA1_SAR_4	R/W	H'FDC08070	32
DMA1 destination address register_4	DMA1_DAR_4	R/W	H'FDC08074	32

Name	Abbreviation	R/W	Address	Access Size
DMA1 transfer count register_4	DMA1_TCR_4	R/W	H'FDC08078	32
DMA1 channel control register_4	DMA1_CHCR_4	R/W	H'FDC0807C	32
DMA1 source address register_5	DMA1_SAR_5	R/W	H'FDC08080	32
DMA1 destination address register_5	DMA1_DAR_5	R/W	H'FDC08084	32
DMA1 transfer count register_5	DMA1_TCR_5	R/W	H'FDC08088	32
DMA1 channel control register_5	DMA1_CHCR_5	R/W	H'FDC0808C	32
DMA1 source address register B_0	DMA1_SARB_0	R/W	H'FDC08120	32
DMA1 destination address register B_0	DMA1_DARB_0	R/W	H'FDC08124	32
DMA1 transfer count register B_0	DMA1_TCRB_0	R/W	H'FDC08128	32
DMA1 source address register B_1	DMA1_SARB_1	R/W	H'FDC08130	32
DMA1 destination address register B_1	DMA1_DARB_1	R/W	H'FDC08134	32
DMA1 transfer count register B_1	DMA1_TCRB_1	R/W	H'FDC08138	32
DMA1 source address register B_2	DMA1_SARB_2	R/W	H'FDC08140	32
DMA1 destination address register B_2	DMA1_DARB_2	R/W	H'FDC08144	32
DMA1 transfer count register B_2	DMA1_TCRB_2	R/W	H'FDC08148	32
DMA1 source address register B_3	DMA1_SARB_3	R/W	H'FDC08150	32
DMA1 destination address register B_3	DMA1_DARB_3	R/W	H'FDC08154	32
DMA1 transfer count register B_3	DMA1_TCRB_3	R/W	H'FDC08158	32
DMA1 extended resource selector 0	DMA1_DMARS0	R/W	H'FDC09000	16
DMA1 extended resource selector 1	DMA1_DMARS1	R/W	H'FDC09004	16
DMA1 extended resource selector 2	DMA1_DMARS2	R/W	H'FDC09008	16

Table 16.3 State of Registers in Each Operating Mode

Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
DMA0_SAR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DAR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_CHCR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SAR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DAR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_CHCR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SAR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DAR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_CHCR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SAR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DAR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_CHCR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DMAOR	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SAR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DAR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_CHCR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SAR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DAR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_CHCR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SARB_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DARB_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCRB_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SARB_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DARB_1	Initialized	Initialized	Retained	Retained	Initialized	Retained

Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
DMA0_TCRB_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SARB_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DARB_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCRB_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_SARB_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DARB_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_TCRB_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DMARS0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DMARS1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA0_DMARS2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SAR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DAR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_CHCR_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SAR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DAR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_CHCR_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SAR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DAR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_CHCR_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SAR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DAR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_CHCR_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DMAOR	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SAR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DAR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_CHCR_4	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SAR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained

Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
DMA1_DAR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_CHCR_5	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SARB_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DARB_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCRB_0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SARB_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DARB_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCRB_1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SARB_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DARB_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCRB_2	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_SARB_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DARB_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_TCRB_3	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DMARS0	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DMARS1	Initialized	Initialized	Retained	Retained	Initialized	Retained
DMA1_DMARS2	Initialized	Initialized	Retained	Retained	Initialized	Retained

16.3.1 DMA Source Address Registers (DMA0_SAR_0 to DMA0_SAR_5, DMA1_SAR_0 to DMA1_SAR_5,)

SAR are 32-bit readable/writable registers that specify the source address of a DMA transfer. During a DMA transfer, these registers indicate the next source address.

To transfer data in word or in longword units, specify the address with word or longword address boundary. When transferring data in 8-byte, 16-byte, or 32-byte units, an 8-byte, 16-byte, or 32-byte boundary must be set for the source address value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SAR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SAR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16.3.2 DMA Source Address Registers (DMA0_SARB_0 to DMA0_SARB_3, DMA1_SARB_0 to DMA1_SARB_3)

SARB are 32-bit readable/writable registers that specify the source address of a DMA transfer that is set in SAR again in repeat/reload mode. Data to be written from the CPU to SAR is also written to SARB. To set SARB address that differs from SAR address, write data to SARB after SAR.

To transfer data in word or in longword units, specify the address with word or longword address boundary. When transferring data in 8-byte, 16-byte, or 32-byte units, an 8-byte, 16-byte, or 32-byte boundary must be set for the source address value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SARB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SARB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16.3.3 DMA Destination Address Registers (DMA0_DAR_0 to DMA0_DAR_5, DMA1_DAR_0 to DMA1_DAR_5)

DAR are 32-bit readable/writable registers that specify the destination address of a DMA transfer. During a DMA transfer, these registers indicate the next destination address.

To transfer data in word or in longword units, specify the address with word or longword address boundary. When transferring data in 8-byte, 16-byte, or 32-byte units, an 8-byte, 16-byte, or 32-byte boundary must be set for the source address value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DAR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16.3.4 DMA Destination Address Registers (DMA0_DARB_0 to DMA0_DARB_3, DMA1_DARB_0 to DMA1_DARB_3)

DARB are 32-bit readable/writable registers that specify the destination address of a DMA transfer that is set in DAR again in repeat/reload mode. Data to be written from the CPU to DAR is also written to DARB. To set DARB address that differs from DAR address, write data to DARB after DAR.

To transfer data in word or in longword units, specify the address with word or longword address boundary. When transferring data in 8-byte, 16-byte, or 32-byte units, an 8-byte, 16-byte, or 32-byte boundary must be set for the source address value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DARB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DARB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16.3.5 DMA Transfer Count Registers (DMA0_TCR_0 to DMA0_TCR_5, DMA1_TCR_0 to DMA1_TCR_5)

TCR are 32-bit readable/writable registers that specify the DMA transfer count. The number of transfers is 1 when the setting is H'00000001, 16,777,215 when H'00FFFFFF is set, and 16,777,216 (the maximum) when H'00000000 is set. During a DMA transfer, these registers indicate the remaining transfer count.

The upper eight bits of TCR are always read as 0, and the write value should always be 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCR															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16.3.6 DMA Transfer Count Registers (DMA0_TCRB_0 to DMA0_TCRB_3, DMA1_TCRB_0 to DMA1_TCRB_3)

TCRB are 32-bit readable/writable registers. Data to be written from the CPU to TCR is also written to TCRB. While the half end function is used, TCRB are used as the initial value hold registers to detect half end. Also, TCRB specify the number of DMA transfers which are set in TCR in repeat mode. TCRB specify the number of DMA transfers and are used as transfer count counters in reload mode.

In reload mode, the lower 16 bits operate as transfer count counters, values of SAR and DAR are updated after the value of the lower 16 bits became 0, and then the value of the upper 16 bits of TCRB are loaded to the lower 16 bits. In upper 16 bits, set the number of transfers which starts reloading. In reload mode, the same number of transfers should be set in both upper and lower 16 bits. Also, set the HIE bit in CHCR to 0 and do not use the half end function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCRB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCRB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

16.3.7 DMA Channel Control Registers (DMA0_CHCR_0 to DMA0_CHCR_5, DMA1_CHCR_0 to DMA1_CHCR_5)

CHCR are 32-bit readable/writable registers that control the DMA transfer mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	LCKN	—	—	RPT[2:0]			DA	DO	TS[3:2]		—	HE	HIE	AM	AL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/(W)*	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM[1:0]		SM[1:0]		RS[3:0]				DL	DS	TB	TS[2:0]		IE	TE	DE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/(W)*	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	LCKN	0	R/W	Bus Right Release Enable in Cycle Steal Mode Specifies whether enable or disable the bus right for bus masters except the DMAC between reading and writing in cycle steal mode. Initially, the DMAC has the bus right. Setting this bit to 1 accepts the bus request from bus masters except the DMAC, allowing improvement of the bus usage rate in the entire system. This setting is valid in cycle steal mode. Do not set this bit to 1 in burst mode. 0: Disables the release of bus mastership between reading and writing. 1: Enables the release of bus mastership between reading and writing.
29, 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
27 to 25	RPT[2:0]	000	R/W	DMA Setting Renewal Specify These bits are enabled in CHCR_0 to CHCR_3. 000: Normal mode (DMAC operation) 001: Repeat mode SAR/DAR used as repeat area 010: Repeat mode DAR used as repeat area 011: Repeat mode SAR used as repeat mode 100: Reserved (setting prohibited) 101: Reload mode SAR/DAR used as reload area 110: Reload mode DAR used as reload area 111: Reload mode SAR used as reload area
24	DA	0	R/W	DREQ Synchronous Input Specify Selects whether DREQ is sampled as an asynchronous signal or synchronous signal. This bit is valid only in CHCR_0/ CHCR_1 of DMAC0. 0: DREQ is sampled as an asynchronous signal. 1: DREQ is sampled as a synchronous signal.
23	DO	0	R/W	DMA Overrun Selects whether DREQ is detected by overrun 0 or by overrun 1. This bit is valid only in CHCR_0/ CHCR_1 of DMAC0. 0: Detects DREQ by overrun 0 1: Detects DREQ by overrun 1
22	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
21, 20	TS[3:2]	00	R/W	DMA Transfer Size Specify With TS[1:0], these bits specify the DMA transfer size. When the transfer source or transfer destination is a register of an on-chip peripheral module with a transfer size set, a proper transfer size for the register should be set. In 2-division transfer mode, 16/32-byte data is halved and transferred in two operations. When 16-byte data is transferred in a peripheral module, 16-byte 2-division transfer should be selected. For the transfer source or destination address specified by SAR or DAR, an address boundary should be set according to the transfer data size. TS[3:0] 0000: Byte units transfer 0001: Word (2-byte) units transfer 0010: Longword (4-byte) units transfer 0011: 16-byte units transfer 0100: 32-byte units transfer 0111: 8-byte units transfer 1011: 16-byte 2-division (8-byte units × 2) transfer 1100: 32-byte 2-division (16-byte units × 2) transfer Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Descriptions
19	HE	0	R/(W)*	Half End Flag After HIE (bit 18) is set to 1 and the number of transfers become half of TCR (1 bit shift to right) which is set before transfer starts, HE becomes 1. The HE bit is not set when transfers are ended by an NMI interrupt or address error, or by clearing the DE bit and the DME bit in DMAOR before the number of transfers is decreased to half of the TCR value set preceding the transfer. The HE bit is kept set when the transfer ends by an NMI interrupt or address error, or clearing the DE bit (bit 0) or the DME bit in DMAOR after the HE bit is set to 1. To clear the HE bit, write 0 after reading 1 in the HE bit. This bit is valid only in CHCR_0 to CHCR_3. Always write 1 to the HE bit unless it is being cleared. Writing 1 to the HE bit will not affect the flag value. 0: During the DMA transfer or DMA transfer has been interrupted TCR > (TCR set before transfer)/2 [Clearing condition] Writing 0 after HE = 1 is read. 1: TCR ≤ (TCR set before transfer)/2
18	HIE	0	R/W	Half End Interrupt Enable Specifies whether an interrupt request is generated to the CPU when the number of transfers is decreased to half of the TCR value set preceding the transfer. When the HIE bit is set to 1 and the HE bit is set, an interrupt request is generated to the CPU. Set this bit to 0 while reload mode is set. This bit is valid in CHCR_0 to CHCR_3. 0: Interrupt request is disabled when TCR = (TCR set before transfer)/2 1: Interrupt request is enabled when TCR = (TCR set before transfer)/2
17	AM	0	R/W	Acknowledge Mode Selects whether DACK is output in data read cycle or in data write cycle in dual address mode. This bit is valid only in CHCR_0/ CHCR_1 of DMAC0. 0: DACK output in read cycle (dual address mode) 1: DACK output in write cycle (dual address mode)

Bit	Bit Name	Initial Value	R/W	Descriptions
16	AL	0	R/W	Acknowledge Level Specifies whether the DACK signal output is high active or low active. This bit is valid only in CHCR_0/ CHCR_1 of DMAC0. 0: Low-active output of DACK 1: High-active output of DACK
15, 14	DM[1:0]	00	R/W	Destination Address Mode Specify whether the DMA destination address is incremented, decremented, or left fixed. 00: Fixed destination address The address is incremented by the first and second transfers in 16/32-byte division transfer mode. Since the address set in DAR is not modified, the same address is output in subsequent operations. 01: Destination address is incremented +1 in byte units transfer +2 in word units transfer +4 in longword units transfer +8 in 8-byte units transfer +16 in 16-byte units transfer +32 in 32-byte units transfer 10: Destination address is decremented –1 in byte units transfer –2 in word units transfer –4 in longword units transfer Setting prohibited in 8/16/32-byte units transfer 11: Fixed destination address Set to prevent an address from being changed in the objective modules. The address is not changed even in 16/32-byte division transfer mode. Example: When specifying FIFOs in the external devices and peripheral modules.

Bit	Bit Name	Initial Value	R/W	Descriptions
13, 12	SM[1:0]	00	R/W	Source Address Mode Specify whether the DMA source address is incremented, decremented, or left fixed. 00: Fixed source address The address is incremented by the first and second transfers in 16/32-byte division transfer mode. Since the address set in SAR is not modified, the same address is output in subsequent operations. 01: Source address is incremented +1 in byte units transfer +2 in word units transfer +4 in longword units transfer +8 in 8-byte units transfer +16 in 16-byte units transfer +32 in 32-byte units transfer 10: Source address is decremented –1 in byte units transfer –2 in word units transfer –4 in longword units transfer Setting prohibited in 8/16/32-byte units transfer 11: Fixed source address Set to prevent an address from being changed in the objective modules. The address is not changed even in 16/32-byte division transfer mode. Example: When specifying FIFOs in the external devices and peripheral modules.
11 to 8	RS[3:0]	0000	R/W	Resource Select Specify which transfer requests will be sent to the DMAC. The changing of transfer request source should be done in the state that the DMA enable bit (DE) is set to 0. 0000: External request, dual address mode 0100: Auto request 1000: Selected by DMA extended resource selector Other than above: Setting prohibited Note: External request specification is valid only in CHCR_0/ CHCR_1 of DMAC0. None of the external request can be selected in CHCR_2 to CHCR_5 of DMAC0 and CHCR_0 to CHCR_5 of DMAC1.

Bit	Bit Name	Initial Value	R/W	Descriptions
7	DL	0	R/W	DREQ Level and DREQ Edge Select
6	DS	0	R/W	Specify the detecting method of the DREQ pin input and the detecting level. These bits are valid only in CHCR_0 and CHCR_1 of DMAC0. In channels 0 and 1, also the other channels, if the transfer request source is specified as an on-chip peripheral module or if an auto-request is specified, these bits are invalid. 00: DREQ detected in low level 01: DREQ detected at falling edge 10: DREQ detected in high level 11: DREQ detected at rising edge
5	TB	0	R/W	Transfer Bus Mode Specifies the bus mode when DMA transfers data. 0: Cycle steal mode 1: Burst mode
4, 3	TS[1:0]	00	R/W	DMA Transfer Size Specify See the description of TS[3:2] (bits 21 and 20).
2	IE	0	R/W	Interrupt Enable Specifies whether or not an interrupt request is generated to the CPU at the end of the DMA transfer. Setting this bit to 1 generates an interrupt request (DEI) to the CPU when the TE bit is set to 1. 0: Interrupt request is disabled. 1: Interrupt request is enabled.

Bit	Bit Name	Initial Value	R/W	Descriptions
1	TE	0	R/(W)*	Transfer End Flag Shows that DMA transfer ends. The TE bit is set to 1 when data transfer ends when TCR becomes to 0. The TE bit is not set to 1 in the following cases. • DMA transfer ends due to an NMI interrupt or DMA address error before TCR is cleared to 0. • DMA transfer is ended by clearing the DE bit and DME bit in DMAOR. To clear the TE bit, the TE bit should be written to 0 after reading 1. Even if the DE bit is set to 1 while this bit is set to 1, transfer is not enabled. Always write 1 to the TE bit unless it is being cleared. Writing 1 to the TE bit will not affect the flag value. 0: During the DMA transfer or DMA transfer has been interrupted [Clearing condition] Writing 0 after TE = 1 read 1: DMA transfer ends by the specified count (DMATCR = 0)
0	DE	0	R/W	DMA Enable Enables or disables the DMA transfer. In auto request mode, DMA transfer starts by setting the DE bit and DME bit in DMAOR to 1. In this time, all of the bits TE, NMIF, and AE in DMAOR must be 0. In an external request or peripheral module request, DMA transfer starts if DMA transfer request is generated by the devices or peripheral modules after setting the bits DE and DME to 1. In this case, however, all of the bits TE, NMIF, and AE must be 0, which is the same as in the case of auto request mode. Clearing the DE bit to 0 can terminate the DMA transfer. 0: DMA transfer disabled 1: DMA transfer enabled

Note: * The only value that can be written is 0 to clear the flag.

16.3.8 DMA Operation Register (DMA0_DMAOR, DMA1_DMAOR)

DMAOR is a 16-bit readable/writable register that specifies the priority level of channels at the DMA transfer. This register shows the DMA transfer status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMS[3:0]				—	—	PR[1:0]		—	—	—	—	—	AE	NMIF	DME
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R/(W)*	R/(W)*	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	CMS[3:0]	0000	R/W	Cycle Steal Mode Select 1, 0 Select either normal mode or intermittent mode in cycle steal mode. It is necessary that all channel's bus modes are set to cycle steal mode to make valid intermittent mode. 0000: Normal mode 0010: Intermittent mode 16 Executes one DMA transfer in each of 16 clocks of an external bus clock. 0011: Intermittent mode 64 Executes one DMA transfer in each of 64 clocks of an external bus clock. 0100: Intermittent mode 256 Executes one DMA transfer in each of 256 clocks of an external bus clock. Other than above: Setting prohibited
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	PR[1:0]	00	R/W	Priority Mode Select the priority level between channels when there are transfer requests for multiple channels simultaneously. 00: CH0 > CH1 > CH2 > CH3 > CH4 > CH5 01: CH0 > CH2 > CH3 > CH1 > CH4 > CH5 10: Setting prohibited 11: Round-robin mode

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	AE	0	R/(W)*	Address Error Flag Indicates that an address error interrupt occurred during DMA transfer. This bit is set under following conditions: • The value set in SAR or DAR does not match to the transfer size boundary. • The transfer source or transfer destination is invalid space. • The transfer source or transfer destination is in module stop mode If this bit is set, DMA transfer is disabled even if the DE bit in CHCR and the DME bit in DMAOR are set to 1. Always write 1 to the AE bit unless it is being cleared. Writing 1 to the AE bit will not affect the flag value. 0: No DMAC address error interrupt [Clearing condition] Writing AE = 0 after AE = 1 read 1: DMAC address error interrupt occurs
1	NMIF	0	R/(W)*	NMI Flag Indicates that an NMI interrupt occurred. If this bit is set, DMA transfer is disabled even if the DE bit in CHCR and the DME bit in DMAOR are set to 1. When the NMI is input, the DMA transfer in progress can be done in at least one transfer unit. When the DMAC is not in operational, the NMIF bit is set to 1 even if the NMI interrupt was input. Always write 1 to the NMIF bit unless it is being cleared. Writing 1 to the NMIF bit will not affect the flag value. 0: No NMI interrupt [Clearing condition] Writing NMIF = 0 after NMIF = 1 read 1: NMI interrupt occurs

Bit	Bit Name	Initial Value	R/W	Description
0	DME	0	R/W	DMA Master Enable Enables or disables DMA transfers on all channels. If the DME bit and the DE bit in CHCR are set to 1, transfer is enabled. In this time, all of the bits TE in CHCR, NMIF, and AE in DMAOR must be 0. If this bit is cleared during transfer, transfers in all channels are terminated. 0: Disables DMA transfers on all channels 1: Enables DMA transfers on all channels

Note: * The only value that can be written is 0 to clear the flag.

16.3.9 DMA Extended Resource Selectors 0 to 2 (DMA0_DMARS0 to DMA0_DMARS2, DMA1_DMARS0 to DMA1_DMARS2)

DMARS are 16-bit readable/writable registers that specify the DMA transfer sources from peripheral modules in each channel. DMARS0 specifies for channels 0 and 1, DMARS1 specifies for channels 2 and 3, and DMARS2 specifies for channels 4 and 5. This register can set the transfer request of peripheral modules.

When MID/RID other than the values listed in Table 16.4 is set, the operation of this LSI is not guaranteed. The transfer request from DMARS is valid only when the resource select bits RS[3:0] have been set to B'1000 for CHCR_0 to CHCR_5 registers. Otherwise, even if DMARS has been set, transfer request source is not accepted.

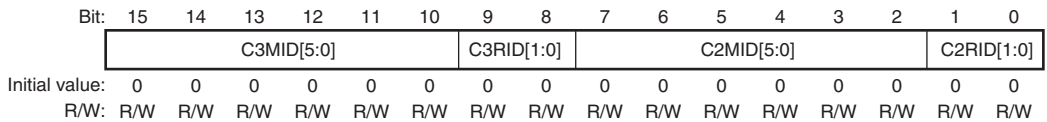
- DMARS0

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C1MID[5:0]						C1RID[1:0]		C0MID[5:0]						C0RID[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	C1MID[5:0]	000000	R/W	Transfer request module ID5 to ID0 for DMA channel 1 (MID) See Table 16.4.
9, 8	C1RID[1:0]	00	R/W	Transfer request register ID1 and ID0 for DMA channel 1 (RID) See Table 16.4.

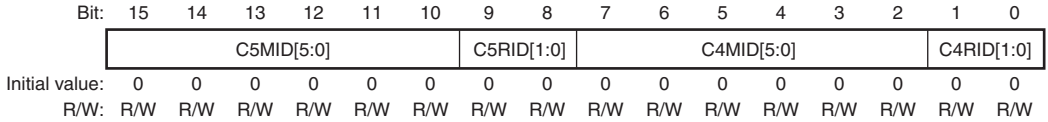
Bit	Bit Name	Initial Value	R/W	Description
7 to 2	C0MID[5:0]	000000	R/W	Transfer request module ID5 to ID0 for DMA channel 0 (MID) See Table 16.4.
1, 0	C0RID[1:0]	00	R/W	Transfer request register ID1 and ID0 for DMA channel 0 (RID) See Table 16.4.

- DMARS1



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	C3MID[5:0]	000000	R/W	Transfer request module ID5 to ID0 for DMA channel 3 (MID) See Table 16.4.
9, 8	C3RID[1:0]	00	R/W	Transfer request register ID1 and ID0 for DMA channel 3 (RID) See Table 16.4.
7 to 2	C2MID[5:0]	000000	R/W	Transfer request module ID5 to ID0 for DMA channel 2 (MID) See Table 16.4.
1, 0	C2RID[1:0]	00	R/W	Transfer request register ID1 and ID0 for DMA channel 2 (RID) See Table 16.4.

- DMARS2



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	C5MID[5:0]	000000	R/W	Transfer request module ID5 to ID0 for DMA channel 5 (MID) See Table 16.4.
9, 8	C5RID[1:0]	00	R/W	Transfer request register ID1 and ID0 for DMA channel 5 (RID) See Table 16.4.
7 to 2	C4MID[5:0]	000000	R/W	Transfer request module ID5 to ID0 for DMA channel 4 (MID) See Table 16.4.
1, 0	C4RID[1:0]	00	R/W	Transfer request register ID1 and ID0 for DMA channel 4 (RID) See Table 16.4.

16.4 Operation

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request. In bus mode, burst mode or cycle steal mode can be selected.

16.4.1 DMA Transfer Requests

DMA transfer requests are basically generated in either the data transfer source or destination, but they can also be generated by external devices or on-chip peripheral modules that are neither the source nor the destination. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request. The request mode is selected in the bits RS[3:0] in CHCR_0 to CHCR_3, and DMARS_0 to DMARS_2.

(1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the DMAC to automatically generate a transfer request signal internally. When the DE bits in CHCR_0 to CHCR_3 and the DME bit in DMAOR are set to 1, the transfer begins so long as the AE and NMIF bits in DMAOR are all 0.

(2) External Request Mode

In this mode, a transfer is performed at the request signal (DREQ0, DREQ1) of an external device. This mode is valid only in channel 0 and channel 1 of DMAC0. When this mode is selected, if the DMA transfer is enabled (DE = 1, DME = 1, TE = 0, AE = 0, NMIF = 0), a transfer is performed upon a request at the DREQ input.

Choose to detect DREQ by either the edge or level of the signal input with the DL bit and DS bit in CHCR_0 or CHCR_1. The source of the transfer request does not have to be the data transfer source or destination.

When DREQ is accepted, the DREQ pin becomes request accept disabled state. After issuing acknowledge signal DACK for the accepted DREQ, the DREQ pin again becomes request accept enabled state.

When DREQ is used by level detection, there are following two cases by the timing to detect the next DREQ after outputting DACK.

- Overrun 0: Transfer is aborted after the same number of transfer has been performed as requests.
- Overrun 1: Transfer is aborted after transfers have been performed for (the number of requests plus 1) times.

The DO bit in CHCR selects this overrun 0 or overrun 1.

(3) On-Chip Peripheral Module Request Mode

In this mode, a transfer is performed at the transfer request signal of an on-chip peripheral module. Transfer request signals are shown in Table 16.4.

When this mode is selected, if the DMA transfer is enabled (DE = 1, DME = 1, TE = 0, AE = 0, NMIF = 0), a transfer is performed upon the input of a transfer request signal.

When a transmit data empty transfer request of the SCIF is set as the transfer request, the transfer destination must be the SCIF's transmit data register. Likewise, when receive data full transfer request of the SCIF is set as the transfer request, the transfer source must be the SCIF's receive data register. These conditions also apply to the other modules listed in table 16.4.

The number of the receive FIFO triggers can be set as a transfer request depending on an on-chip peripheral module. Data needs to be read after the DMA transfer is ended, because data may be remained in the receive FIFO when the receive FIFO trigger condition is not satisfied.

The cycle steal mode is the only available bus mode that is associated with the on-chip peripheral request mode.

Table 16.4 Selecting On-Chip Peripheral Module Request

DMARS						
MID	RID	Setting Value*	DMA Transfer Request Source	DMA Transfer Request Signal	Source	Destination
001000	01	H'21	SCIF0 transmitter	TXI (transmit FIFO data empty)	Any	SCFTDR0
	10	H'22	SCIF0 receiver	RXI (receive FIFO data full)	SCFRDR0	Any
001001	01	H'25	SCIF1 transmitter	TXI (transmit FIFO data empty)	Any	SCFTDR1
	10	H'26	SCIF1 receiver	RXI (receive FIFO data full)	SCFRDR1	Any
001010	01	H'29	SCIF2 transmitter	TXI (transmit FIFO data empty)	Any	SCFTDR2
	10	H'2A	SCIF2 receiver	RXI (receive FIFO data full)	SCFRDR2	Any
001011	01	H'2D	SCIFA3 transmitter	TXI (transmit FIFO data empty)	Any	SCAFTDR0
	10	H'2E	SCIFA3 receiver	RXI (receive FIFO data full)	SCAFRDR0	Any
001100	01	H'31	SCIFA4 transmitter	TXI (transmit FIFO data empty)	Any	SCAFTDR1
	10	H'32	SCIFA4 receiver	RXI (receive FIFO data full)	SCAFRDR1	Any
001101	01	H'35	SCIFA5 transmitter	TXI (transmit FIFO data empty)	Any	SCAFTDR2
	10	H'36	SCIFA5 receiver	RXI (receive FIFO data full)	SCAFRDR2	Any
001110	01	H'39	IrDA transmitter	TBEI (transmit empty transfer request)	Any	IRIF_UART3
	10	H'3A	IrDA receiver	RBEI (receive full transfer request)	IRIF_UART4	Any
010100	01	H'51	MSIOF0 transmitter	TXI (transmit FIFO data empty)	Any	SITFDR0
	10	H'52	MSIOF0 receiver	RXI (receive FIFO data full)	SIRFDR0	Any
010101	01	H'55	MSIOF1 transmitter	TXI (transmit FIFO data empty)	Any	SITFDR1
	10	H'56	MSIOF1 receiver	RXI (receive FIFO data full)	SIRFDR1	Any
011100	11	H'73	USB0	USB0 transmitter 0	Any	D0FIFO
				USB0 receiver 0	D0FIFO	Any
011101	11	H'77		USB0 transmitter 1	Any	D1FIFO
				USB0 receiver 1	D1FIFO	Any

DMARS

MID	RID	Setting Value*	DMA Transfer Request Source	DMA Transfer Request Signal	Source	Destination
101010	11	H'AB	USB1	USB1 transmitter 0	Any	D0FIFO
				USB1 receiver 0	D0FIFO	Any
101011	11	H'AF		USB1 transmitter 1	Any	D1FIFO
				USB1 receiver 1	D1FIFO	Any
110000	01	H'C1	SDHI0 channel 1	Transmit empty transfer request	Any	Data register
	10	H'C2	SDHI0 channel 0	Receive full transfer request	Data register	Any
110001	01	H'C9	SDHI1 channel 1	Transmit empty transfer request	Any	Data register
	10	H'CA	SDHI1 channel 0	Receive full transfer request	Data register	Any
110011	01	H'CD	MMCIF transmitter	Transmit empty transfer request	Any	Data register
	10	H'CE	MMCIF receiver	Receive full transfer request	Data register	Any
111010	10	H'EA	TSIF	Receive full transfer request	TSTSDR	Any

Note: * A value which indicates the transfer source (MID + RID)

16.4.2 Channel Priority

When the DMAC receives simultaneous transfer requests on two or more channels, it transfers data according to a predetermined priority. Two modes (fixed mode and round-robin mode) are selected by the bits PR[1:0] in DMAOR.

(1) Fixed Mode

In this mode, the priority levels among the channels remain fixed. There are two kinds of fixed modes as follows:

- CH0 > CH1 > CH2 > CH3 > CH4 > CH5
- CH0 > CH2 > CH3 > CH1 > CH4 > CH5

These are selected by the bits PR[1:0] in DMAOR.

(2) Round-Robin Mode

In round-robin mode each time data of one transfer unit (byte, word, longword, 8-byte, 16-byte, or 32-byte unit) is transferred on one channel, the priority is rotated. The channel on which the transfer was just finished rotates to the bottom of the priority. The round-robin mode operation is shown in Figure 16.2. The priority of round-robin mode is CH0 > CH1 > CH2 > CH3 > CH4 > CH5 immediately after reset.

When round-robin mode is specified, do not mix the cycle steal mode and the burst mode in multiple channels' bus modes.

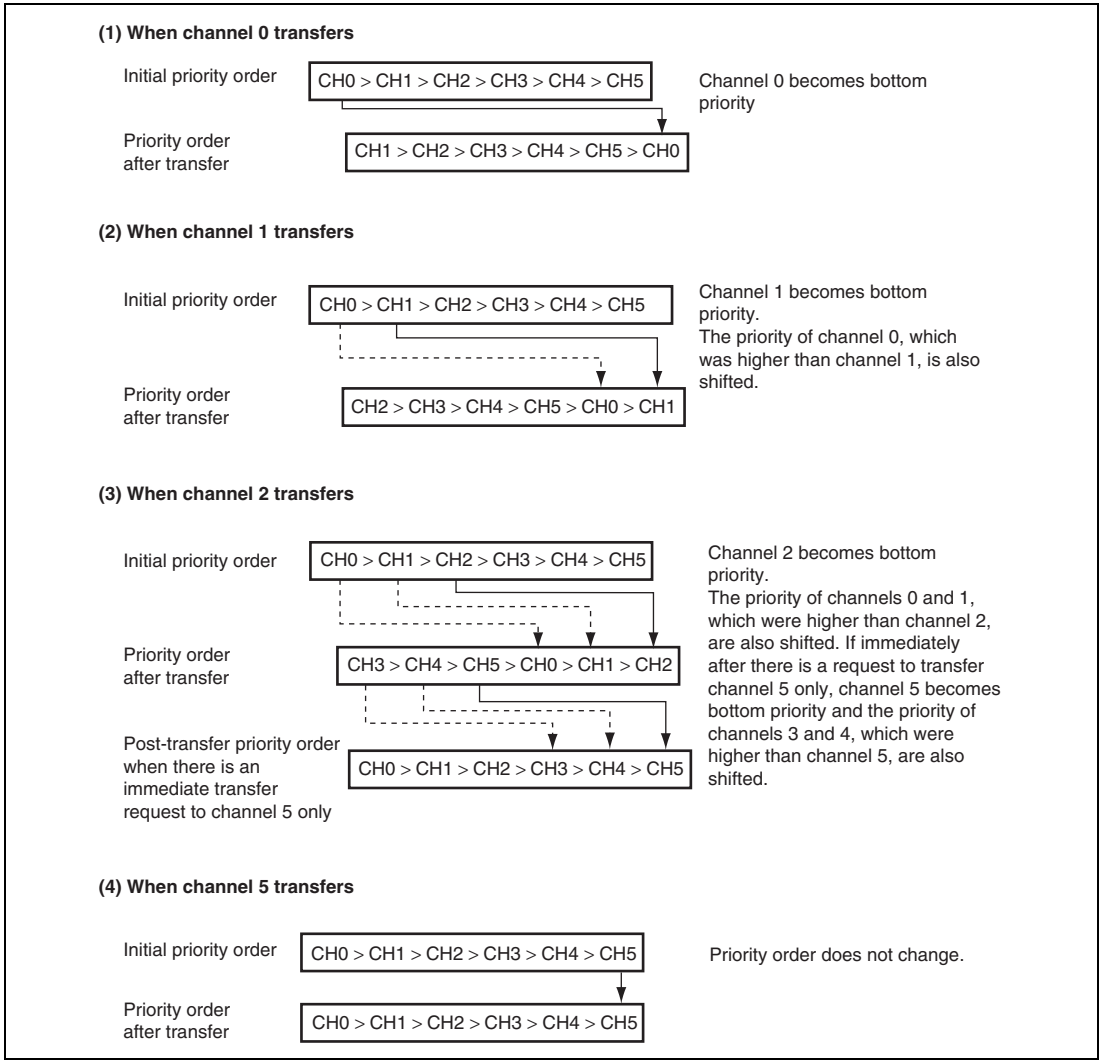


Figure 16.2 Round-Robin Mode

Figure 16.3 shows how the priority changes when channel 0 and channel 3 transfers are requested simultaneously and a channel 1 transfer is requested during the channel 0 transfer. The DMAC operates as follows:

1. Transfer requests are generated simultaneously to channels 0 and 3.
2. Channel 0 has a higher priority, so the channel 0 transfer begins first (channel 3 waits for transfer).
3. A channel 1 transfer request occurs during the channel 0 transfer (channels 1 and 3 are both waiting)
4. When the channel 0 transfer ends, channel 0 becomes lowest priority.
5. At this point, channel 1 has a higher priority than channel 3, so the channel 1 transfer begins (channel 3 waits for transfer).
6. When the channel 1 transfer ends, channel 1 becomes lowest priority.
7. The channel 3 transfer begins.
8. When the channel 3 transfer ends, channels 3 and 2 shift downward in priority so that channel 3 becomes the lowest priority.

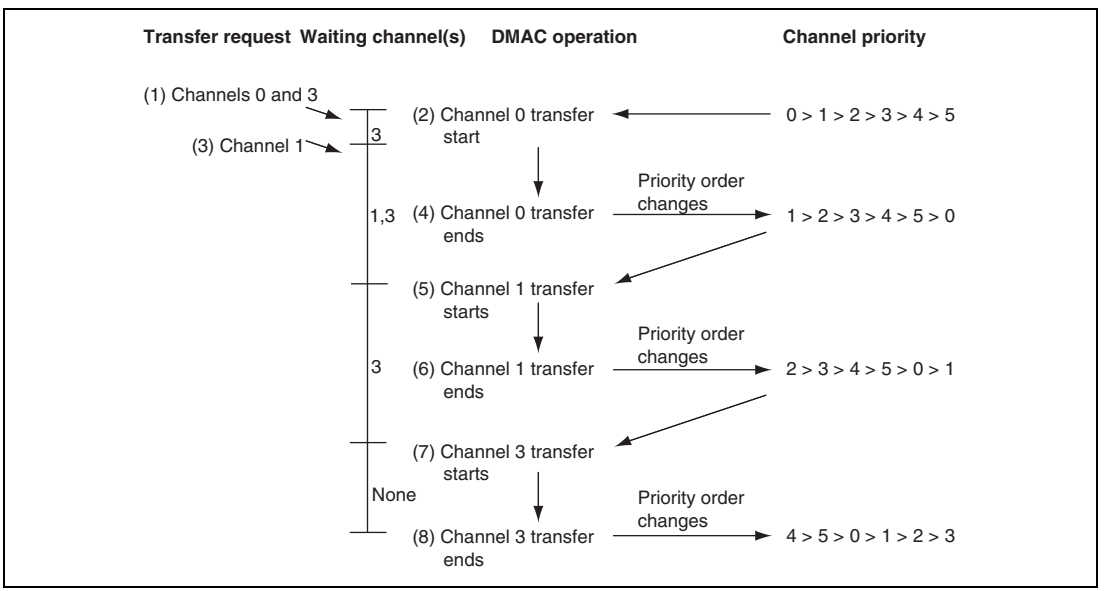


Figure 16.3 Changes in Channel Priority in Round-Robin Mode

16.4.3 DMA Transfer Types

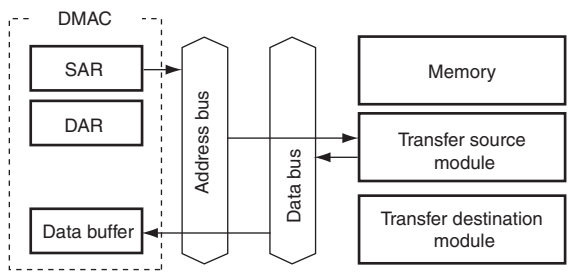
DMA transfer type is dual address mode transfer. A data transfer timing depends on the bus mode, which has cycle steal mode and burst mode.

(1) Address Modes

- Dual Address Mode

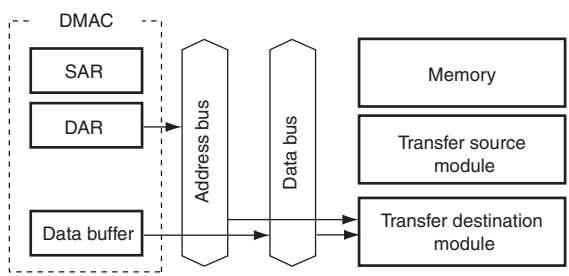
  In dual address mode, both the transfer source and destination are accessed by an address. The source and destination can be located externally or internally.

  DMA transfer requires two bus cycles because data is read from the transfer source in a data read cycle and written to the transfer destination in a data write cycle. At this time, transfer data is temporarily stored in the DMAC. In the transfer between on-chip peripheral modules as shown in Figure 16.4, data is read to the DMAC from the transfer source module in a data read cycle, and then that data is written to the transfer destination module in a write cycle.



The SAR value is an address, data is read from the transfer source module, and the data is temporarily stored in the DMAC.

First bus cycle



The DAR value is an address and the value stored in the data buffer in the DMAC is written to the transfer destination module.

Second bus cycle

Figure 16.4 Data Flow of Dual Address Mode

Auto request, external request, and on-chip peripheral module request are available for the transfer request. DACK can be output in read cycle or write cycle in dual address mode. CHCR can specify whether the DACK is output in read cycle or write cycle.

Figure 16.5 shows an example of DMA transfer timing in dual address mode.

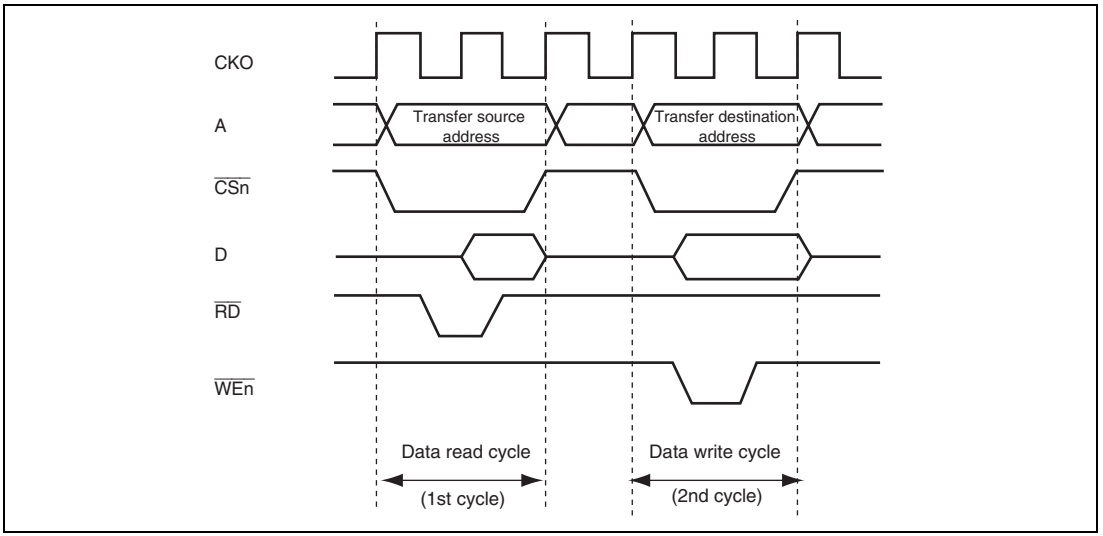


Figure 16.5 Example of DMA Transfer Timing in Dual Mode (Source: Ordinary Memory, Destination: Ordinary Memory)

(2) Bus Modes

There are two bus modes: cycle steal mode and burst mode. Select the mode in the TB bits in CHCR.

- Cycle-Steal Mode
  - — Normal mode

    In cycle-steal normal mode, the bus mastership is given to another bus master after a one-transfer-unit (byte, word, longword, 8-byte, 16-byte, or 32-byte unit) DMA transfer. When another transfer request occurs, the bus mastership is obtained from the other bus master and a transfer is performed for one transfer unit. When that transfer ends, the bus mastership is passed to the other bus master. This is repeated until the transfer end conditions are satisfied.

    In cycle-steal normal mode, transfer areas are not affected regardless of settings of the transfer request source, transfer source, and transfer destination.

  Figure 16.6 shows an example of DMA transfer timing in cycle-steal normal mode. Transfer conditions shown in the figure are:

  — Dual address mode

  — DREQ low level detection

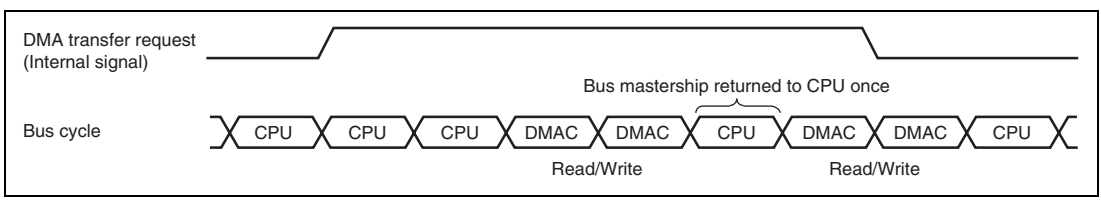


Figure 16.6 DMA Transfer Example in Cycle-Steal Normal Mode (Dual Address, DREQ Low Level Detection)

  — Intermittent mode 16 and intermittent mode 64

    In intermittent mode of cycle steal, the DMAC returns the bus mastership to other bus master whenever a unit of transfer (byte, word, longword, 8-byte, 16-byte, or 32-byte unit) is complete. If the next transfer request occurs after that, the DMAC gets the bus mastership from other bus master after waiting for 16 or 64 clocks in Bϕ count. The DMAC then transfers data of one unit and returns the bus mastership to other bus master. These operations are repeated until the transfer end condition is satisfied. It is thus possible to make lower the ratio of bus occupation by DMA transfer than cycle-steal normal mode.

    When the DMAC gets again the bus mastership, DMA transfer can be postponed in case of entry updating due to cache miss.

This intermittent mode can be used for all transfer section; transfer request source, transfer source, and transfer destination. The bus modes, however, must be cycle steal mode in all channels.

Figure 16.7 shows an example of DMA transfer timing in cycle steal intermittent mode. Transfer conditions shown in the figure are:

— Dual address mode

— DREQ low level detection

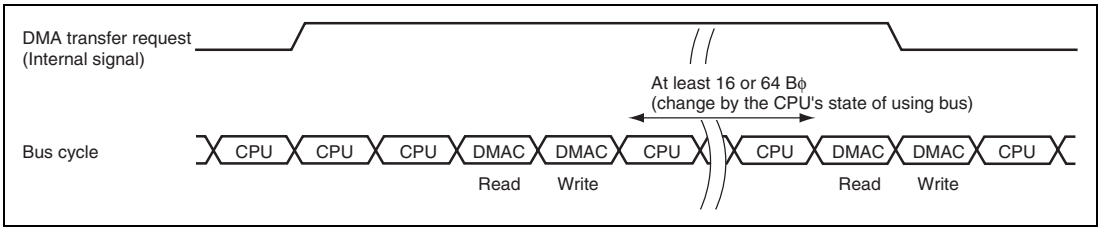


Figure 16.7 Example of DMA Transfer in Cycle Steal Intermittent Mode (Dual Address, DREQ Low Level Detection)

- Burst Mode

  In burst mode, once the DMAC obtains the bus mastership, the transfer is performed continuously without releasing the bus mastership until the transfer end condition is satisfied. In external request mode with level detection of the DREQ pin, however, when the DREQ pin is not active, the bus mastership passes to the other bus master after the DMAC transfer request that has already been accepted ends, even if the transfer end conditions have not been satisfied.

  Burst mode cannot be used when the on-chip peripheral module is the transfer request source.

Figure 16.8 shows DMA transfer timing in burst mode.

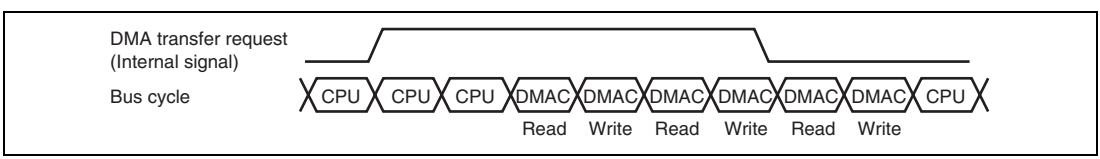


Figure 16.8 DMA Transfer Example in Burst Mode (Dual Address, DREQ Low Level Detection)

(3) Relationship between Request Modes and Bus Modes by DMA Transfer Category

Table 16.5 shows the relationship between request modes and bus modes by DMA transfer category.

Table 16.5 Relationship between Request Modes and Bus Modes by DMA Transfer Category

(4) DMA Transfer Category by Auto Requests

	Transfer Destination			
Transfer Source	**BSC space**	**DBSC space**	**Peripheral Module***	**IL memory**
BSC space	√	√	√	√
DBSC space	√	√	√	√
Peripheral Module*	√	√	√	√
IL memory	√	√	√	√

[Legend]

√: Transferable

Note: * Access size permitted for the on-chip peripheral module register functioning as the transfer source or transfer destination.

(5) DMA Transfer Category by External Requests (DMAC0, Channel 0, or Channel 1 is Only Available)

	Transfer Destination			
Transfer Source	**BSC space**	**DBSC space**	**Peripheral Module*[1]**	**IL memory**
BSC space	√	√*[2]	√	√
DBSC space	√*[3]	—	√*[3]	√*[3]
Peripheral Module*[1]	√	√*[2]	√	√
IL memory	√	√*[2]	√	√

[Legend]

√: Transferable

—: Nontransferable

Notes: 1. Access size permitted for the on-chip peripheral module register functioning as the transfer source or transfer destination.

2. Transferable if the AM bit in CHCR is set to 0.

3. Transferable if the AM bit in CHCR is set to 1.

(6) DMA Transfer Category by Peripheral Module Requests

	Transfer Destination			
Transfer Source	**BSC space**	**DBSC space**	**Peripheral Module***	**IL memory**
BSC space	—	—	√	—
DBSC space	—	—	√	—
Peripheral Module*	√	√	√	√
IL memory	—	—	√	—

[Legend]

√: Transferable

—: Nontransferable

Note: * Access size permitted for the on-chip peripheral module register functioning as the transfer source or the transfer destination.
The request source register must be designated as the transfer source or the transfer destination. Also, The cycle steal mode is the only available bus mode that is associated with the on-chip peripheral request mode.

(7) Bus Mode and Channel Priority

When the priority is set in fixed mode (CH0 > CH1) and channel 1 is transferring in burst mode, if there is a transfer request to channel 0 with a higher priority, the transfer of channel 0 will begin immediately.

At this time, if channel 0 is also operating in burst mode, the channel 1 transfer will continue after the channel 0 transfer has completely finished.

When channel 0 is in cycle steal mode, channel 0 with a higher priority performs the transfer of one transfer unit and the channel 1 transfer is continuously performed without releasing the bus mastership. The bus mastership will then switch between the two in the order channel 0, channel 1, channel 0, and channel 1. Therefore, the bus state is such that the CPU cycle after the completion of cycle steal mode transfer has been replaced with the channel 1 burst mode transfer. (Hereinafter referred to as burst mode priority execution.)

This example is shown in Figure 16.9. When multiple channels are operating in burst modes, the channel with the highest priority is executed first.

When DMA transfer is executed in the multiple channels, the bus mastership will not be given to the bus master until all competing burst transfers are complete.

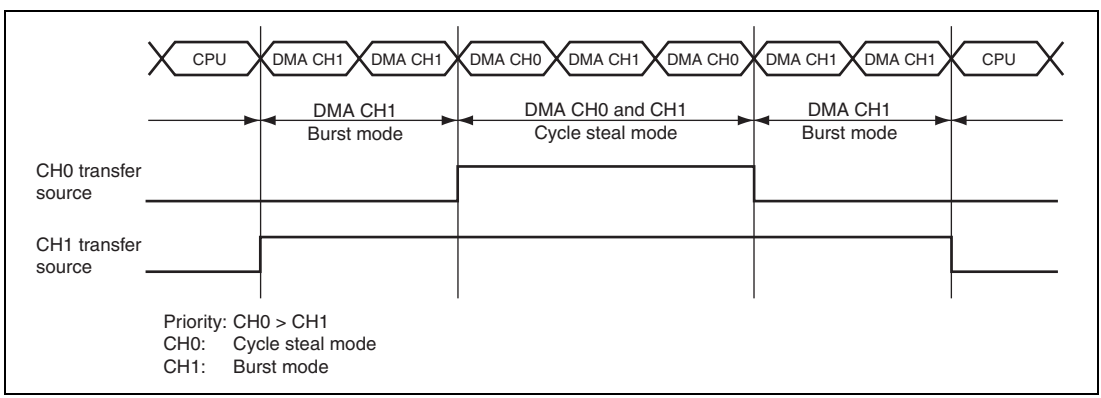


Figure 16.9 Bus State when Multiple Channels are Operating

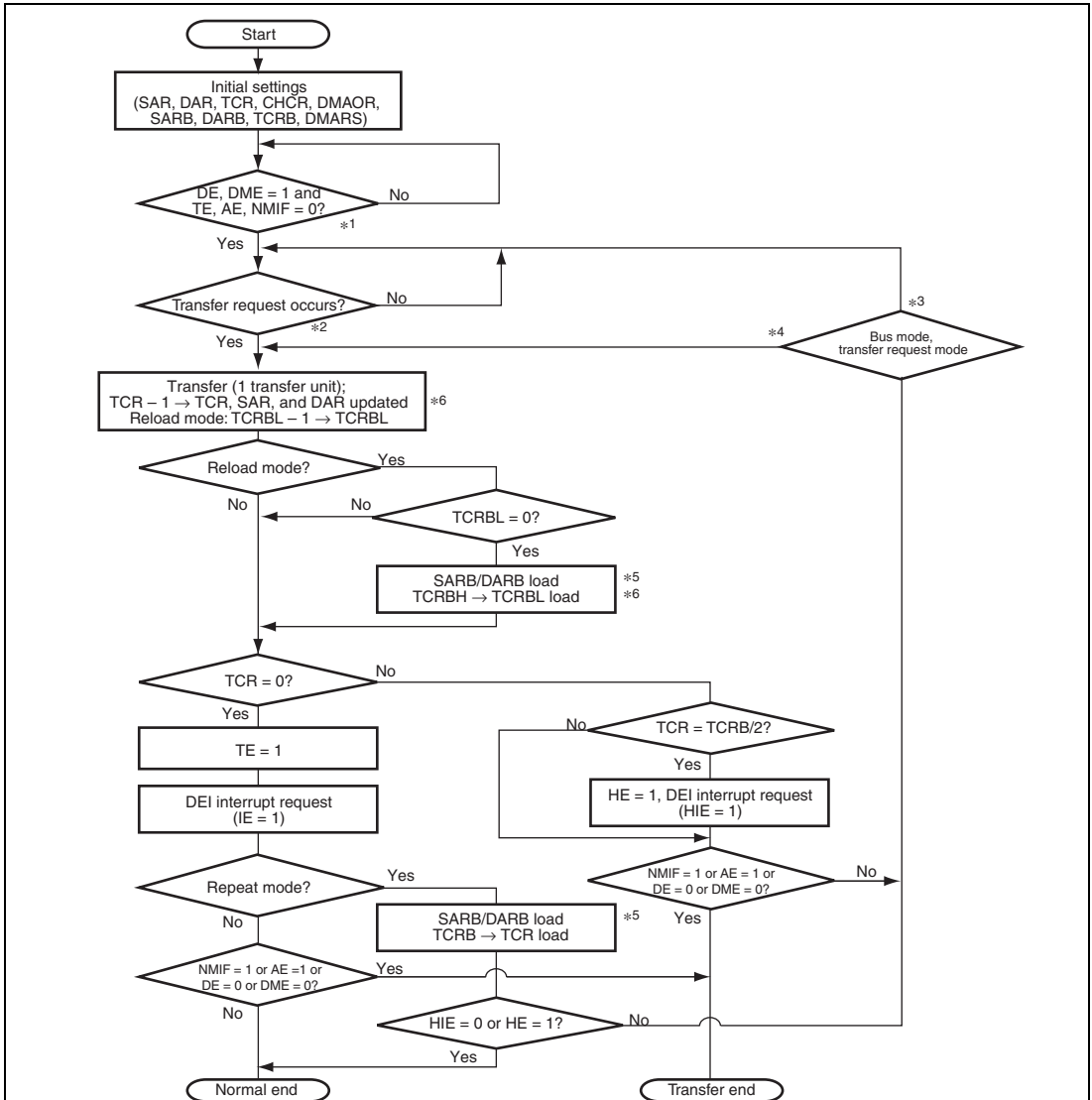
In round-robin mode, the priority changes according to the specification shown in Figure 16.2. However, the channel in cycle steal mode cannot be mixed with the channel in burst mode.

16.4.4 DMA Transfer Flow

After the DMA source address registers (SAR), DMA destination address registers (DAR), DMA transfer count registers (DMATCR), DMA channel control registers (CHCR), DMA operation register (DMAOR), and DMA extended resource selectors (DMARS) are set, the DMAC transfers data according to the following procedure:

1. Checks to see if transfer is enabled (DE = 1, DME = 1, TE = 0, AE = 0, NMIF = 0)
2. When a transfer request occurs while transfer is enabled, the DMAC transfers one transfer unit of data (depending on the TS3 to TS0 settings). In auto request mode, the transfer begins automatically when the DE bit and DME bit are set to 1. The DMATCR value will be decremented for each transfer. The actual transfer flows vary by address mode and bus mode.
3. When the specified number of transfer have been completed (when DMATCR reaches 0), the transfer ends normally. If the IE bit in CHCR is set to 1 at this time, a DEI interrupt is sent to the CPU.
4. When an address error or an NMI interrupt is generated, the transfer is aborted. Transfers are also aborted when the DE bit in CHCR or the DME bit in DMAOR is changed to 0.

Figure 16.10 shows a flowchart of this procedure.



Notes:
1. In repeat mode, a transfer request is acceptted with TE =1 when HIE = 1 and HE = 0.
2. In auto-request mode, transfer starts when bits NMIF, AE, and TE are all 0 or bits TE and HIE are 1 and HE is 0 (in repeat mode), and bits DE and DME are set to 1.
3. In cycle-steal mode.
4. Auto request in burst mode.
5. Loading to SAR and DAR differs according to the operating conditions in each mode.
6. TCRBH represents bits TCRB[13:16] and TCRBL represents bits TCRB [15 to 0].

Figure 16.10 DMA Transfer Flowchart

16.4.5 Repeat Mode Transfer

In a repeat mode transfer, a DMA transfer is repeated without specifying the transfer settings every time before executing a transfer.

Using a repeat mode transfer with the half end function allows a double buffer transfer executed virtually. Following processings can be executed effectively by using a repeat mode transfer. As an example, operation of receiving voice data from the VOICE CODEC and compressing it is explained.

In the following example, processing of compressing 40-word voice data every data reception is explained. In this case, it is assumed that voice data is received by means of MSIOF.

1. DMAC settings
- Set address of the MSIOF receive data register in SAR
- Set address of an internal memory data store area in DAR
- Set TCR to 80 (H'50)
- Satisfy the following settings of CHCR

  Bits RPT[2:0] = B'010: Repeat mode (use DAR as a repeat area)

  Bit HIE = B'1: TCR/2 interrupt generated

  Bits DM[1:0] = B'01: DAR incremented

  Bits SM[1:0] = B'00: SAR fixed

  Bit IE = B'1: Interrupt enabled

  Bit DE = B'1: DMA transfer enabled
- Set such as bits TB and TS[3:0] according to use conditions
- Set bits CMS[1:0] and PR[1:0] in DMAOR according to use conditions and set the DME bit to B'1

2. Voice data is received and then transferred by MSIOF/DMAC
3. TCR is decreased to half of its initial value and an interrupt is generated

   Read CHCR to confirm that the HE bit is set to 1 by an interrupt processing, and compress 40-word voice data from the address set in DAR.
4. TCR is cleared to 0 and an interrupt is generated

   Read CHCR to confirm that the TE bit is set to 1 by an interrupt processing, and compress 40-word voice data from the address set in DAR + 40. After this operation, the value of DARB is copied to DAR in DMAC and initialized, and the value of TCRB is copied to TCR and initialized to 80.

5. Hereafter, steps 2 and 3 are repeated until DME or DE is set to B'0, or an NMI interrupt is generated.

As explained above, a repeat mode transfer enables sequential voice compression by changing buffer for storing data received consequentially and a data buffer for processing signals alternately.

16.4.6 Reload Mode Transfer

In a reload mode transfer, according to the settings of bits RPT[2:0] in CHCR, the value set in SARB/DARB is set to SAR/DAR and the value of bits TCRB[31:16] is set in bits TCRB[15:0] at each transfer set in the bits TCRB[15:0], and the transfer is repeated until TCR becomes 0 without specifying the transfer settings again. A reload mode transfer is effective when repeating data transfer with specific area. Figure 16.11 shows the operation of reload mode transfer.

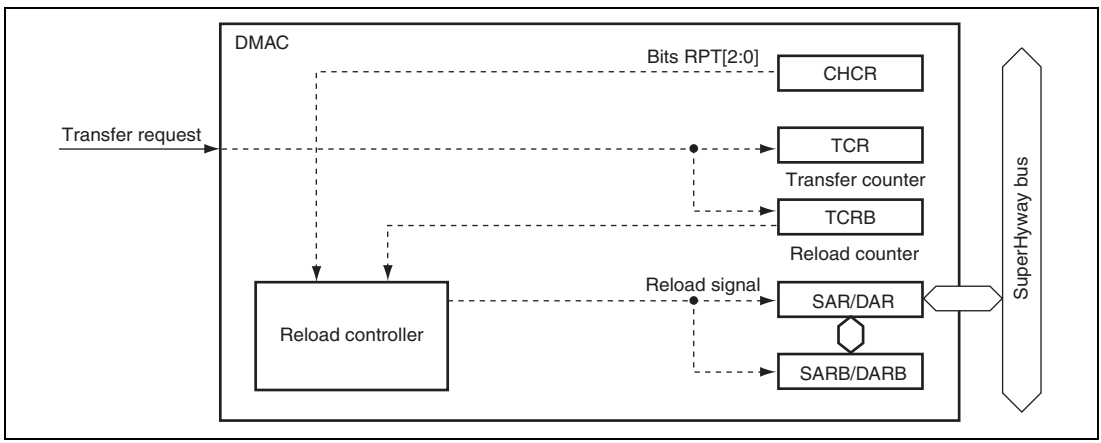


Figure 16.11 Reload Mode Transfer

When a reload mode transfer is executed, TCRB is used as a reload counter. Set TCRB according to section 16.3.6, DMA Transfer Count Registers (DMA0_TCRB_0 to DMA0_TCRB_3, DMA1_TCRB_0 to DMA1_TCRB_3).

16.4.7 DREQ Pin Sampling Timing

Figures 16.12 to 16.15 show the sample timing of the DREQ input in each bus mode, respectively.

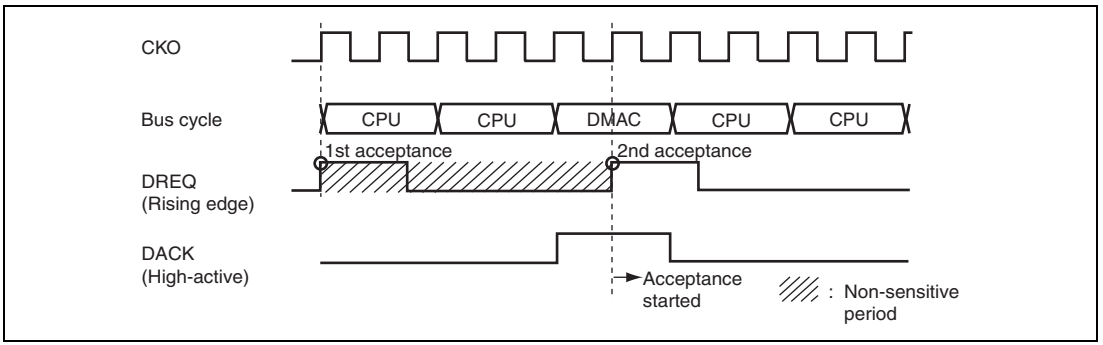


Figure 16.12 Example of DREQ Input Detection in Cycle Steal Mode Edge Detection

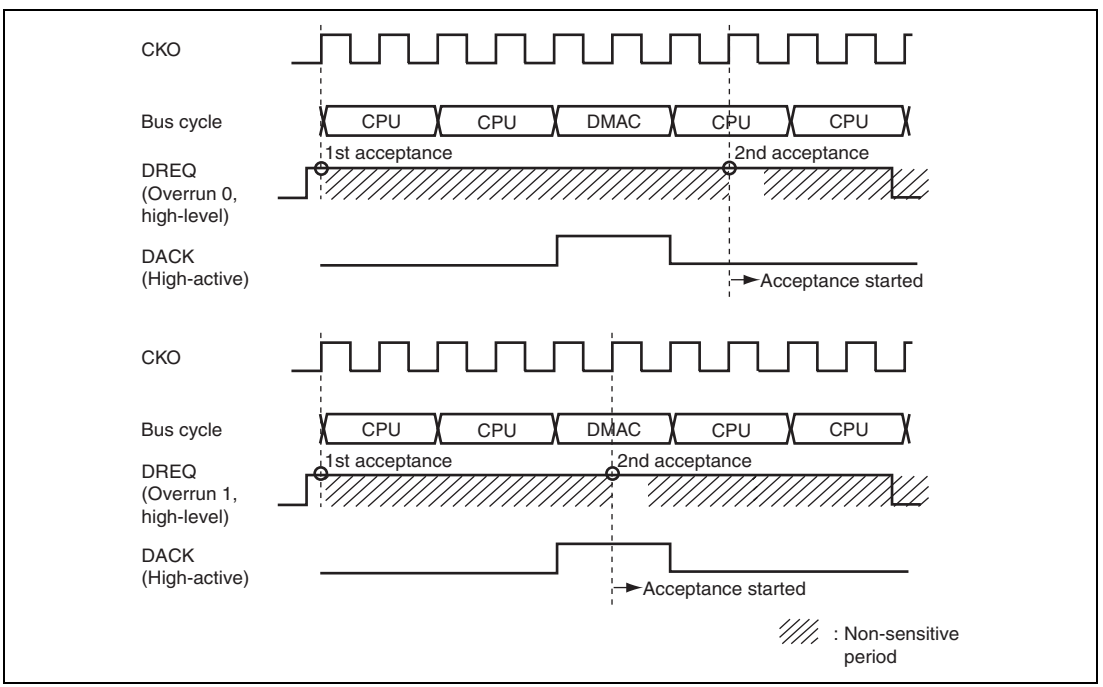


Figure 16.13 Example of DREQ Input Detection in Cycle Steal Mode Level Detection

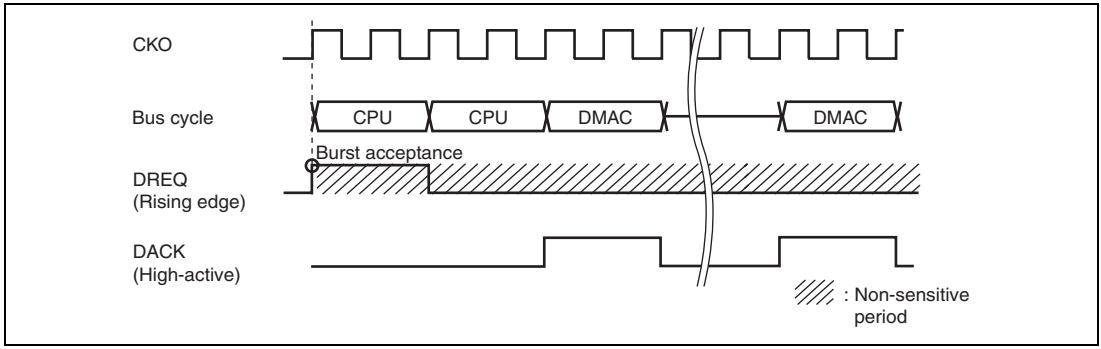


Figure 16.14 Example of DREQ Input Detection in Burst Mode Edge Detection

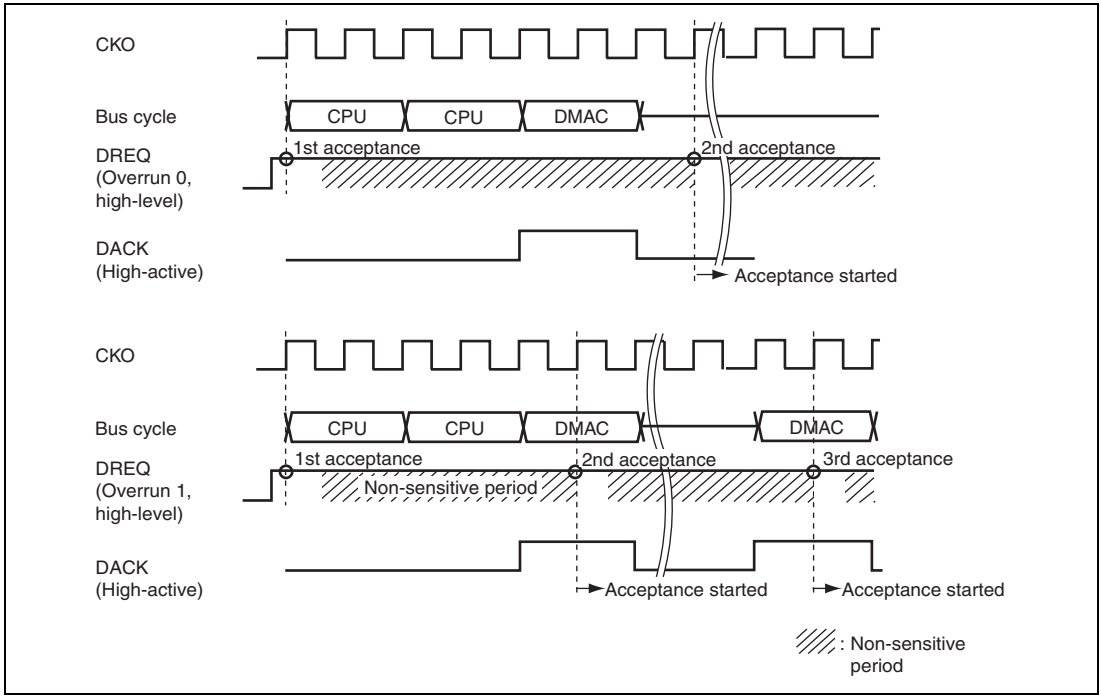
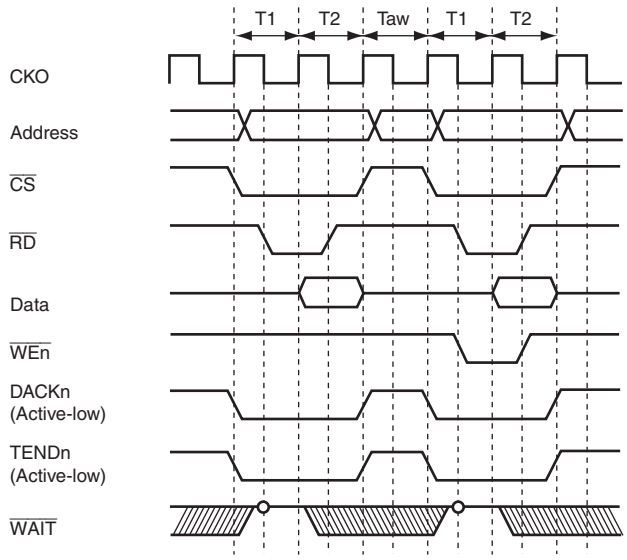


Figure 16.15 Example of DREQ Input Detection in Burst Mode Level Detection

When a 16-bit external device is accessed in longword units, the DACK output is divided because of the data alignment. This example is shown in Figure 16.16.



Note: TEND is asserted for the last transfer unit of the DMA transfer. When the transfer unit is divided into several bus cycles and CS is negated between bus cycles, TEND is also divided.

Figure 16.16 BSC Ordinary Memory Access (No Wait, Idle Cycle 1, Longword Access to 16-Bit Device)

16.5 Usage Notes

Pay attentions to the following notes when the DMAC is used.

16.5.1 DMA Transfer for Peripheral Modules

When executing a 16-byte DMA transfer for peripheral modules, set the TS[3:0] bits in CHCR to B'1011 and execute in 16-byte 2-division transfer mode. This DMA transfer can only be executed when a 16-byte boundary can be set in SAR or DAR as a transfer source address or transfer destination address. When a transfer source address or transfer destination address is not a 16-byte boundary, data cannot be transferred successfully.

16.5.2 Module Stop

While DMAC is in operation, the DMAC should not be stopped by the module stop register (MSTPCR0). While DMAC for the peripheral modules is in operation, DMA transfer should not be forcibly suspended. When modules are stopped and DMA transfer is forcibly suspended, transfer contents cannot be guaranteed.

16.5.3 Address Error

When a DMA address error is occurred, set registers of all channels again and then start a transfer.

16.5.4 Notes on Burst Mode Transfer

During a burst mode transfer, following operation should not be executed until the transfer of corresponding channel has completed.

1. Frequency should not be changed.
2. Transition to sleep mode should not be made.
3. Transition to various standby modes should not be made.

16.5.5 Notes on Setting of DMA Extended Resource Selectors

Do not set the same transfer requests to DMA extended resource selectors (DMA0_DMARS0 to DMA0_DMARS2, DMA1_DMARS0 to DMA1_DMARS2). When the same transfer requests are set, transfer contents cannot be guaranteed.

Section 17 Clock Pulse Generator (CPG)

The clock pulse generator generates the clocks used in this LSI and consists of a PLL circuit, a FLL circuit, dividers, and the associated control circuit.

17.1 Features

- Generation of the various clocks for LSI internal operations

  CPU clock (Iϕ): Operating clock for the CPU core

  S clock (Sϕ): Operating clock for the SuperHyway bus and DBSC

  Bus clock (Bϕ): Operating clock for the BSC. Operating clock for peripheral modules on the SuperHyway bus

  VPU clock (M1ϕ): Operating clock for the VPU

  Peripheral clock (Pϕ): Operating clock for peripheral modules on the HPB (peripheral bus).

  RCLK (Rϕ): Operating clock only for peripheral modules and clock source for FLL circuit.

  SPU clock (SPUϕ): Operating clock for SPU.

  IrDA clock (IrDAϕ): Operating clock for IrDA.
- Generation of clocks for external interfaces

  Bus clock (CKO): Clock for the BSC bus interface (same as Bϕ)

  SDRAM clock (MCLK): Clock for the SDRAM interface (same as Sϕ)

  Video clock (VIO_CKO): Clock output for cameras

  FSI clock A (FSICKA): Clock for the FSI external interface

  FSI clock B (FSICKB): Clock for the FSI external interface
- Frequency-change function

  The frequency of each clock can be changed independently by using the PLL circuit, FLL circuit, or dividers within the CPG. Frequencies are changed under software control by register settings.
- Clock mode

  The clock mode pin setting selects external inputs (EXTAL or RCLK) or crystal oscillator as the clock source. In addition, the PLL and FLL can be turned on or off by the clock mode pin setting after a power-on reset.
- Power-down mode control

  The clocks are stopped in sleep mode, software standby mode, R-standby mode, and U-standby mode; clocks for specific modules can be stopped by using the module standby function. For details, see section 18, Reset and Power-Down Modes.

17.2 Block Diagram

A block diagram of the CPG is shown in figure 17.1.

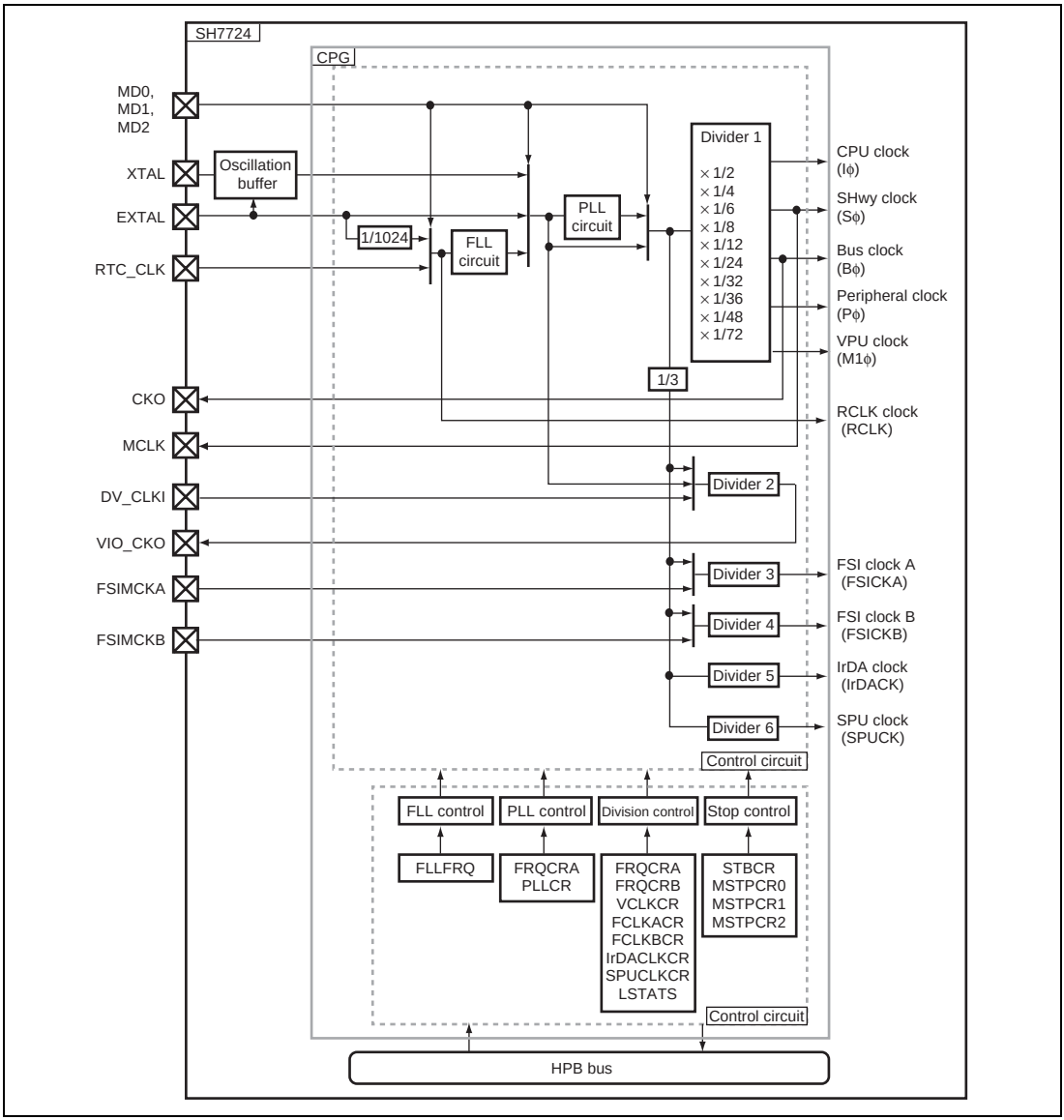


Figure 17.1 Block Diagram of CPG

The CPG blocks function as follows:

(1) FLL Circuit

The FLL circuit multiples the clock frequency (32.768 kHz) input from the RTC_CLK pin or the divided clock by 1024 input from EXTAL pin. This circuit is only enabled in clock mode 3. The multiplication rate is set in the FLL multiplication register (FLLFRQ). The initial value of the multiplication rate is 1017 and the generated clock is at 32.768 kHz × 1017 = 33.33 MHz.

The output clock frequency for the FLL circuit is in the range from 20 MHz to 33.4 MHz

(2) PLL Circuit

The PLL circuit multiples, by factors from 12 to 48, the frequency of the clock input from the EXTAL pin or of the multiplied clock signal produced by the FLL circuit. The multiplication rate is set in the frequency control register (FRQCRA). The PLL circuit is turned on or off by the settings of the clock mode pins or the PLL control register (PLLCR).

The input clock frequency for the PLL circuit is in the range from 10 to 66.7 MHz. The output clock frequency is in the range from 500 to 1014 MHz.

(3) Divider 1

Divider 1 divides the frequency of the clock input from the EXTAL pin or of the clock input from the clock buffer for crystal oscillator, or of the multiplied clock produced by the FLL circuit.

(4) Divider 2 to 6

Divider 2 to 6 divides the frequency of the PLL circuit output divided by 3 or of the external clocks. The division ratio is set in the relevant frequency control register.

(5) Control Circuits

The control circuit controls the clock frequency according to the settings of the MD0, MD1, MD2 pins and the frequency control registers, and controls clock stop modes for each modules.

17.3 Input/Output Pins

Table 17.1 lists the CPG pin configuration.

Table 17.1 Pin Configuration and Functions of CPG

Pin Name	Function	I/O	Description
MD0	Clock mode control pins	Input	Sets the clock operating mode.
MD1		Input	Sets the clock operating mode.
MD2		Input	Sets the clock operating mode.
EXTAL	System Clock pins	Input	Connects the crystal oscillator. Or used as an external clock input pin.
XTAL		Output	Connects the crystal oscillator.
RTC_CLK	RTC clock input pin	Input	Inputs the RTC clock (32.768 kHz) *[1]
FSIMCKA	FSI clock input pin	Input	Clock input for FSI
FSIMCKB		Input	Clock input for FSI
DV_CLKI	Video clock input pin	Input	Video clock for VOU
CKO	Bus clock output pin	Output	Used as a BSC interface clock output pin.
MCLK	SDRAM clock output pin	Output	Used as a SDRAM interface clock output pin.
VIO_CKO	Video clock output pin	Output	Used as a clock output pin for cameras.

Note: 1. Always input RTC_CLK in this LSI when any of clock operating modes 3 to 7 is selected, even if RTC is not used. Even when any of clock operating modes 0 to 2 is selected, input RTC_CLK when the RTC module is used.

17.4 Clock Operating Modes

Table 17.2 shows the relationship between the mode control pin (MD2, MD1, MD0) combinations and the initial clock settings after a power-on reset. The frequency ratios in the table are relative to a clock source frequency of 1.

Table 17.2 Clock Operating Modes

Clock Mode	Pin Setting			RCLK Clock Source	Register Initial Value		PLL Clock Source	PLL	FLL	Iϕ	Sϕ	Bϕ	M1ϕ	Pϕ
	MD2	MD1	MD0		FRQCR	PLLCR								
0	0	0	0	EXTAL x 1/1024	H'1155 5558	H'0000 4000	EXTAL	ON (× 36)	OFF	1/12	1/12	1/12	1/12	1/24
1	0	0	1	EXTAL x 1/1024	H'0E55 5558	H'0000 4000	EXTAL	ON (× 30)	OFF	1/12	1/12	1/12	1/12	1/24
2	0	1	0	EXTAL x 1/1024	H'0E55 5558	H'0000 4000	EXTAL (crystal)	ON (× 30)	OFF	1/12	1/12	1/12	1/12	1/24
3	0	1	1	RTC_CLK	H'0E55 5558	H'0000 5000	FLL Output	ON (× 30)	ON (× 1017)	1/12	1/12	1/12	1/12	1/24
4	1	0	0	RTC_CLK	H'1155 5558	H'0000 4000	EXTAL	ON (× 36)	OFF	1/12	1/12	1/12	1/12	1/24
5	1	0	1	RTC_CLK	H'0E55 5558	H'0000 4000	EXTAL	ON (× 30)	OFF	1/12	1/12	1/12	1/12	1/24
6	1	1	0	RTC_CLK	H'0E55 5558	H'0000 4000	EXTAL (crystal)	ON (× 30)	OFF	1/12	1/12	1/12	1/12	1/24
7	1	1	1	RTC_CLK	H'0E22 2222	H'0000 0000	EXTAL	OFF	OFF	1/4	1/4	1/4	1/4	1/4

17.5 Register Descriptions

Table 17.3 shows the CPG register configuration. Table 17.4 shows the register states in each operating mode.

Table 17.3 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Frequency control register A	FRQCRA	R/W	H'A415 0000	32
Frequency control register B	FRQCRB	R/W	H'A415 0004	32
Video clock frequency control register	VCLKCR	R/W	H'A415 0048	32
FSI clock A frequency control register	FCLKACR	R/W	H'A415 0008	32
FSI clock B frequency control register	FCLKBCR	R/W	H'A415 000C	32
IrDA clock frequency control register	IrDACLKCR	R/W	H'A415 0018	32
SPU clock frequency control register	SPUCLKCR	R/W	H'A415 003C	32
PLL control register	PLLCR	R/W	H'A415 0024	32
FLL multiplication register	FLLFRQ	R/W	H'A415 0050	32
Frequency change status register	LSTATUS	R	H'A415 0060	32

Table 17.4 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
FRQCRA	Initialized	Retained	Retained	—	Retained	Initialized	Retained
FRQCRB	Initialized	Retained	Retained	—	Retained	Initialized	Retained
VCLKCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained
FCLKACR	Initialized	Retained	Retained	—	Retained	Initialized	Retained
FCLKBCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained
IrDACLKCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
SPUCLKR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained
PLLCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained
FLLFRQ	Initialized	Retained	Retained	—	Retained	Initialized	Retained
LSTATUS	Initialized	Retained	Retained	—	Initialized	Initialized	Retained

17.5.1 Frequency Control Register A (FRQCRA)

FRQCRA is a 32-bit readable/writable register used to specify the frequency multiplication ratio of the PLL circuit, and the frequency division ratio of the CPU clock, S clock, bus clock, SDRAM clock, and peripheral clock. FRQCRA can be accessed only in longwords. The initial values of the frequency multiplication rate and division ratio are determined by the clock mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	KICK	—	STC[5:0]						IFC[3:0]				—	—	—	—
Initial value:	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SFC[3:0]				BFC[3:0]				—	—	—	—	P1FC[3:0]			
Initial value:	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	KICK	0	R/W	Kick bit for FRQCRA and FRQCRB. 0: this bit is cleared automatically after “1” writing. 1: validate the values of FRQCRA and FRQCRB
30	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
29 to 24	STC[5:0]	See table 17.2.	R/W	PLL Circuit Multiplication Ratio 000101: × 12 000111: × 16 001011: × 24 001110: × 30 001111: × 32 010001: × 36 010111: × 48 Other settings are prohibited Note: The setting of these bits has no effect while the PLL is turned off.

Bit	Bit Name	Initial Value	R/W	Description	
23 to 20	IFC[3:0]	See table 17.2.	R/W	CPU Clock (Iϕ) Frequency Division Ratio	
				0000: × 1/2	1000: × 1/24
				0001: × 1/3	1001: × 1/32
				0011: × 1/6	1010: × 1/36
				0100: × 1/8	1011: × 1/48
				0101: × 1/12	1101: × 1/72
				0110: × 1/16	Other settings are prohibited
19 to 16	—	See table 17.2.	R	Reserved This bit is read as initial value in table 17.2. The write value should always be the same as read value.	
15 to 12	SFC[3:0]	See table 17.2.	R/W	S Clock (Sϕ) Frequency Division Ratio	
				0000: × 1/4	1000: × 1/24
				0011: × 1/6	1001: × 1/32
				0100: × 1/8	1010: × 1/36
				0101: × 1/12	1011: × 1/48
				0110: × 1/16	1101: × 1/72
					Other settings are prohibited
11 to 8	BFC[3:0]	See table 17.2.	R/W	Bus Clock (Bϕ) Frequency Division Ratio	
				0010: × 1/4	1001: × 1/32
				0011: × 1/6	1010: × 1/36
				0100: × 1/8	1010: × 1/12
				0101: × 1/12	1011: × 1/48
				0110: × 1/16	1101: × 1/72
				1000: × 1/24	Other settings are prohibited
7 to 4	—	See table 17.2.	R	Reserved These bits are read the same value as SFC[3:0]. To write these bits is invalid.	
3 to 0	P1FC[3:0]	See table 17.2.	R/W	Peripheral Clock (Pϕ) Frequency Division Ratio	
				0010: × 1/4	1001: × 1/32
				0011: × 1/6	1010: × 1/36
				0100: × 1/8	1011: × 1/48
				0101: × 1/12	1101: × 1/72
				0110: × 1/16	Other settings are prohibited
				1000: × 1/24	

Notes: * The initial values depend on the clock mode.

17.5.2 Frequency Control Register B (FRQCRB)

FRQCRB is a 32-bit readable/writable register used to specify the frequency division ratio of the M1 clock. FRQCRB can be accessed only in longwords. The initial values of the frequency division ratio are determined by the clock mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	M1FC[3:0]				—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	*	*	*	*	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
7 to 4	M1FC[3:0]	See table 17.2.	R/W	VPU Clock (M1ϕ) Frequency Division Ratio 0010: × 1/4 0011: × 1/6 0100: × 1/8 0101: × 1/12 0110: × 1/16 1000: × 1/24 1001: × 1/32 1010: × 1/36 1011: × 1/48 1101: × 1/72 Other settings are prohibited
3 to 0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Notes: * The initial values depend on the clock mode.

The following conditions must be satisfied when setting the values in FRQCRA and FRQCRB (N is an integer).

1. Iϕ : Sϕ

   Integer clock ratio Iϕ : Sϕ = N : 1

2. Sϕ : Bϕ

   Integer clock ratio Sϕ : Bϕ = N : 1

3. Bϕ : Pϕ

   Integer clock ratio Bϕ : Pϕ = N : 1

4. Iϕ : Pϕ

   Integer clock ratio Iϕ : Pϕ = N : 1

5. Sϕ : M1ϕ : Bϕ

   Integer clock ratio Sϕ : M1ϕ : Bϕ = 2 : 1 : 1, 2 :2 :1, 1

17.5.3 PLL Control Register (PLLCR)

PLLCR is a 32-bit readable/writable register used to turn on or off the PLL circuit and FLL circuit, and also enable or disable clock output from the CKO pin. PLLCR can be accessed only in longwords.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PLLE	—	DLLE	—	—	—	—	—	—	—	—	—	—	CKOFF	—
Initial value:	0	*	0	*	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	PLLE	See table 17.2.	R/W	PLL Enable Turns on or off the PLL circuit. 0: PLL circuit is off 1: PLL circuit is on
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	FLLE	See table 17.2.	R/W	FLL Enable Turns on or off the FLL circuit. 0: FLL circuit is off 1: FLL circuit is on Note: Do not change the initial value of this bit.
11 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CKOFF	0	R/W	CKO Output Stop Controls the CKO output. 0: Clock is output from the CKO pin 1: No clock is output from the CKO pin CKO: Fixed low

Bit	Bit Name	Initial Value	R/W	Description
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Note: * The initial values of bits 14 and 12 depend on the clock mode.

17.5.4 Video Clock Frequency Control Register (VCLKCR)

VCLKCR is a 32-bit readable/writable register that controls the video clock frequency. VCLKCR can be accessed only in longwords.

Specify the value of DIV[5:0] in VCLKCR not to exceed 83.4 MHz.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SRC[2:0]			—	—	—	CKSTP	—	—	DIV[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W:	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	SRC[2:0]	000	R/W	Clock source selection 000: PLL circuit output clock × 1/3 010: External clock input from DV_CLKI pin 100: EXTAL input (clock mode 0,1,4,5,7) Crystal oscillator (clock mode 2,6) FLL output (clock mode 3) Other settings are prohibited.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CKSTP	0	R/W	Clock Stop 0: Video clock is supplied 1: Video clock is halted
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	VDIV[5:0]	111111	R/W	Division Ratio These bits set the frequency division ratio of the video clock. The specified division ratio is × 1/(setting + 1).

17.5.5 FSI Clock A Frequency Control Register (FCLKACR)

FCLKACR is a 32-bit readable/writable register that controls the FSI clock A frequency. FCLKACR can be accessed only in longwords.

Specify the value of DIV[5:0] in FCLKACR not to exceed 41.7 MHz.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CKSTP	SRC[1:0]		DIV[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CKSTP	0	R/W	Clock Stop 0: FSI clock A is supplied 1: FSI clock A is halted
7, 6	SRC[1:0]	00	R/W	Clock Source Select Selects the FSICKA clock source. 00: PLL circuit output clock × 1/3 10: External input clock (FSIMCKA) Other settings are prohibited.
5 to 0	DIV[5:0]	111111	R/W	Division Ratio These bits set the frequency division ratio of the FSICKA clock. The specified division ratio is × 1/(setting + 1).

17.5.6 FSI Clock B Frequency Control Register (FCLKBCR)

FCLKBCR is a 32-bit readable/writable register that controls the FSI clock B frequency. FCLKBCR can be accessed only in longwords.

Specify the value of DIV[5:0] in FCLKBCR not to exceed 41.7 MHz.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CKSTP	SRC[1:0]		DIV[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CKSTP	0	R/W	Clock Stop 0: FSI clock B is supplied 1: FSI clock B is halted
7, 6	SRC[1:0]	00	R/W	Clock Source Select Selects the FSICKB clock source. 00: PLL circuit output clock × 1/3 10: External input clock (FSIMCKB) Other settings are prohibited.
5 to 0	DIV[5:0]	111111	R/W	Division Ratio These bits set the frequency division ratio of the FSICKB clock. The specified division ratio is × 1/(setting + 1).

17.5.7 IrDA Clock Frequency Control Register (IrDACLKCR)

IrDACLKCR is a 32-bit readable/writable register that controls the IrDA clock frequency. IrDACLKCR can be accessed only in longwords.

Specify the value of DIV[5:0] in IrDACLKCR not to exceed 41.7 MHz.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CKSTP	—	—	DIV[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	*	*	*	*	*	*
R/W:	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CKSTP	0	R/W	Clock Stop 0: IrDA clock is supplied 1: IrDA clock is halted
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	DIV[5:0]	111111	R/W	Division Ratio These bits set the frequency division ratio for the PLL circuit output × 1/3. The division ratio is set to × 1/(setting + 1).

17.5.8 SPU clock frequency control register (SPUCLKCR)

SPUCLKCR is a 32-bit readable/writable register that controls the SPU clock frequency. SPUCLKCR can be accessed only in longwords.

Specify the value of DIV[5:0] in SPUCLKCR not to exceed 83.4 MHz.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CKSTP	—	—	DIV[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CKSTP	0	R/W	Clock Stop 0: SPU clock is supplied 1: SPU clock is halted
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	DIV[5:0]	111111	R/W	Division Ratio These bits set the frequency division ratio for the PLL circuit output × 1/3. The division ratio is set to × 1/(setting + 1).

17.5.9 FLL Multiplication Register (FLLFRQ)

FLLFRQ is a 32-bit readable/writable register used to specify the frequency multiplication ratio of the FLL circuit. FLLFRQ can be accessed only in longwords. Specify the value of DLF[10:0] in the range from 20 to 33.4 MHz.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SELXM[1:0]		—	—	—	FLF[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	1
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	SELXM[1:0]	00	R/W	FLL output division 00: FLL output × 1 01: FLL output × 1/2 Other settings are prohibited.
13 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	FLF[10:0]	H'3F9	R/W	FLL Multiplication Ratio These bits specify the frequency multiplication ratio of the FLL circuit. Multiplication is performed with a ratio of setting. H'3F9: Approximately 33 MHz* H'338: Approximately 27 MHz* H'2FB: Approximately 25 MHz* H'2DC: Approximately 24 MHz* H'262: Approximately 20 MHz* Other settings are prohibited Note: * The value is indicated when RCLK = 32.768 kHz.

17.5.10 Frequency change status register (LSTATS)

LSTATS is a 32-bit read only register to indicate the status of frequency changing. LSTATS can be accessed only in longwords.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FRQF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FRQF	0	R	Frequency changing status flag 0: Not on frequency changing status 1: On frequency changing status

17.6 Changing Frequency

The clock controlled by the frequency control register can be changed either by changing the multiplication ratio of the PLL circuit or by changing the division ratio of the divider. All of these are controlled by software through the frequency control register. The methods are described below.

17.6.1 Changing Multiplication Ratio of PLL Circuit

Changing the multiplication ratio of the PLL circuit can be achieved by simply rewriting the STC[5:0] bits in FRQCRA and writing 1 to the KICK bit in FRQCRA because the PLL oscillation settling time is internally detected automatically. The PLL reference clock is supplied within the LSI and clock output to the CKO pin is halted until the oscillation of the PLL has settled.

An invalid clock may be output through the CKO pin when the clock output to the CKO pin is stopped or started.

Before changing the PLL multiplication ratio, complete all peripheral module operations so that they enter idle state. While the multiplication ratio is being changed, no activation trigger should be applied to any peripheral module through external pins.

When 1 is written to the KICK bit in FRQCRA, the CPG issues stop requests to all initiators and targets. After the initiators and targets have stopped, the CPG changes the multiplication ratio for the PLL circuit.

Procedure for changing the multiplication ratio of the PLL

1. Set the FRQCRA.STC[5:0] bits.
2. Write 1 to the FRQCRA.KICK bit.
3. Wait until LSTATS.FRQF is cleared.

17.6.2 Changing Division Ratio

Changing the division ratio can be achieved by rewriting each set of bits for setting the division ratio in FRQCRA and FRQCRB and writing 1 to the KICK bit in FRQCRA.

When 1 is written to the KICK bit in FRQCRA, the CPG issues a stop request only to the CPU if only the CPU clock (Iϕ) division ratio is to be changed. After the CPU has stopped, the CPG changes the division ratio of the CPU clock (Iϕ). If the frequency of the S clock (Sϕ), bus clock (Bϕ), VPU clock (M1ϕ), or peripheral clock (Pϕ) is to be changed, the CPG issues stop requests to all initiators and targets. After the initiators and targets have stopped, the CPG changes the division ratio.

Except when changing the CPU clock (Iϕ) only, complete all peripheral module operations so that they enter idle state before changing the division ratio. While the division ratio is being changed, no activation trigger should be applied to any peripheral module through external pins.

After the bus clock (Bϕ) is changed, an invalid clock may be output through the CKO pin when the clock output to the CKO pin is stopped or started.

Procedure for changing the division ratio

1. Set each set of bits for setting the division ratio in FRQCRA and FRQCRB
2. Write 1 to the FRQCRA.KICK bit.
3. Wait until LSTATS. FRQF is cleared.

17.6.3 Changing Clock Operating Mode

The values of the mode control pins (MD2, MD1, MD0) that define the clock operating mode are reflected at a power-on reset. Do not change the MD2, MD1, and MD0 pin settings during operation.

17.6.4 Turning On/Off PLL Circuit

The PLL circuit can be turned on or off by rewriting the PLLE bit in PLLCR.

In the same way as when changing the multiplication ratio of the PLL circuit, the oscillation settling time of the PLL circuit is internally detected automatically.

17.6.5 Changing Multiplication Ratio of FLL Circuit

Changing the multiplication ratio of the FLL circuit can be achieved by simply rewriting the FLF[10:0] bits in FLLFRQ and the SELXM bits in FLLFRQ because the oscillation settling times of the FLL circuit and PLL circuit are internally detected automatically. The RWDT setting is not required.

17.7 Notes on Board Design

(1) Bypass Capacitor

Insert laminated ceramic capacitors as bypass capacitors for each V_{SS}/V_{CC} pair. Mount the bypass capacitor near the power supply pins of the LSI. Use components with a frequency characteristic suitable for the operating frequency of the LSI, as well as a suitable capacitance value.

(2) Usage Note on PLL Oscillator Circuit

Keep the wiring from the PLL V_{DD} and V_{SS} connection pattern to the power supply pins short, and make the pattern width large to minimize the inductance component.

The analog power supply system of the PLL circuit is sensitive to noise. Therefore system malfunction may occur by the intervention with another power supply. Do not supply the analog power supply with the same resource as the digital power supply of V_{DD} and $V_{CC}Q$.

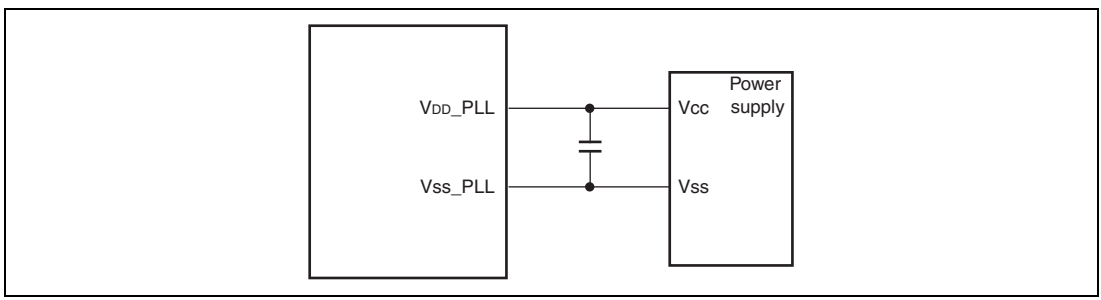


Figure 17.2 Point to Note when Using the PLL Oscillator Circuit

(3) Usage Note on FLL Oscillator Circuit

Keep the wiring from the FLL V_{DD} and V_{SS} connection pattern to the power supply pins short, and make the pattern width large to minimize the inductance component.

Adding a bypass capacitor and CR filter to the FLL power supply is recommended to reduce the noise. Connect a bypass capacitor of approximately 0.1 μF in the vicinity of the FLL power supply pin. A frequency of approximately 1/20 of the RCLK is recommended for the cutoff frequency of the CR filter.

The FLL power supply is sensitive to a noise. Therefore system malfunction may occur by the intervention with another power supply. Do not supply the power supply with the same resource as the digital power supply of V_{DD} and $V_{CC}Q$.

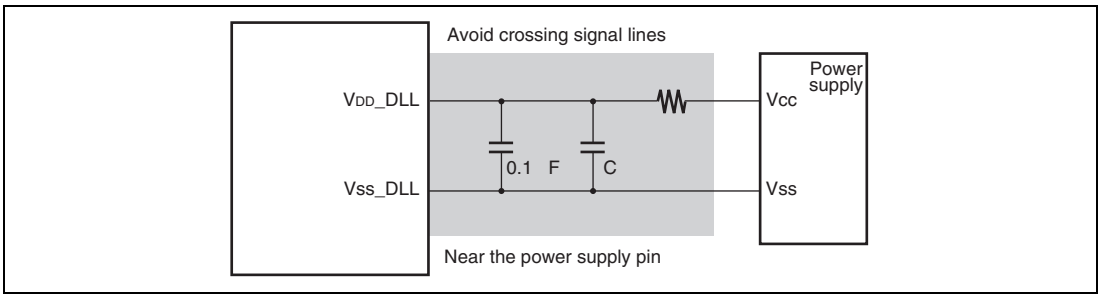


Figure 17.3 Point to Note when Using the FLL Oscillator Circuit

Section 18 Reset and Power-Down Modes

This LSI supports R-standby, U-standby modes, in which low power consumption is achieved by turning off the internal power-supply to part of the chip. This LSI also supports sleep mode, software standby mode, and module standby function, in which clock supply to the LSI is controlled optimally.

18.1 Features

- Supports a variety of power-down modes, i.e. sleep, software standby, module standby, and U-standby and R-standby modes.
- In U-standby and R-standby modes, the RWDT, CMT, KEYSC, and RTC that operate on RCLK are operational.

18.1.1 Division of Power-Supply Areas

To realize power-down modes, this LSI is divided into the following four power-supply areas.

- Core area

  This area is operated by the V_{DD} power supply and encompasses all areas other than the following three. Power consumption on standby is greatly reduced in U-standby and R-standby modes by turning off the power to this area.
- Back-up area

  This area is operated by the V_{DD} power supply. This area includes RS memory and registers back-up area for some modules.

  On waking up from R-standby mode, the register contents backed up in this area are automatically rewritten to the respective registers.
- Sub area

  This area is operated by the V_{DD} power supply and encompasses the RWDT, CMT, KEYSC, and RTC.
- I/O area

  This area is operated by the V_{CC} power supply and encompasses the I/O buffer.

18.1.2 Types of Resets and Power-Down Modes

This LSI has the following types of power-down modes. Table 18.1 shows the state in each mode and methods for making transitions and canceling each mode.

- Sleep mode: Supply of the clock to the CPU core is stopped.
- Software standby mode: Supply of the clock is stopped throughout the LSI.
- Module standby function: The operation of modules that are not in use can be stopped under software control.
- U-standby mode: The supply of power to core area and back-up area are stopped. (A power is supplied to I/O area and sub area.)
- R-standby mode: The supply of power to core area is stopped. (A power is supplied to I/O area, sub area and back-up area.)

Table 18.1 States of Resets and Power-Down Modes

		State						
Power-Down Mode	**Transition Conditions**	**CPG**	**CPU Core**	**CPU Registers**	**On-Chip Memory**	**On-Chip Peripheral Modules*1**	**External SDRAM**	**Canceling Method**
Sleep mode	Execute the SLEEP instruction with STBY = 0 and USTBY = 0 in STBCR.	Operating	Stopped	Retained	Stopped (contents retained)	Operating	Auto-refreshing	• Interrupt • Power-on reset • System reset
Software standby mode	Execute the SLEEP instruction with STBY = 1 and USTBY = 0 in STBCR.	Stopped	Stopped	Retained	Stopped (contents retained)	Stopped*2	Self-refreshing	• IRQ, NMI, CMT, KEYSC, RTC • Power-on reset • System reset
Module standby function	Set the MSTP bit of the respective module to 1 in MSTPCR.	Operating	Operating or stopped	Retained	Specified module stopped (contents retained)	Specified module stopped	Auto-refreshing	• Clear the MSTP bit to 0.

Power-Down Mode	Transition Conditions	State						Canceling Method
		CPG	CPU Core	CPU Registers	On-Chip Memory	On-Chip Peripheral Modules*1	External SDRAM	
R-standby mode	Execute the SLEEP instruction with RSTBY = 1, USTBY = 0 and STBY = 0 in STBCR.	Stopped	Stopped	Not retained	RS memory is retained, Others are Not retained.	Stopped*2	Self-refreshing	• IRQ, NMI, CMT, KEYSC, RTC • Power-on reset • System reset
U-standby mode	Execute the SLEEP instruction with USTBY = 1 and STBY = 0 in STBCR.	Stopped	Stopped	Not retained	Not retained	Stopped*2	Self-refreshing	• CMT, KEYSC, RTC*4 • Power-on reset • System reset
Power-on reset	Drive the $\overline{\text{RESETP}}$ pin low.	Initial state	Initial state	Initial state	Initial state	Initial state	Initial state	—
System reset	Drive the $\overline{\text{RESETA}}$ pin low. RWDT overflows.	Initial state	Initial state	Initial state	Initial state	Initial state	Initial state	—
Manual reset	Generate an exception other than a user break while SR.BL = 1.	Retained	Initial state	Initial state	Initial state/retained*3	Initial state/retained*3	Auto-refreshing	—

Notes: 1 The on-chip peripheral modules refer to modules that are directly connected to the SuperHyway bus or peripheral bus.

2 Modules with RCLK operation (RWDT, CMT, KEYSC, and RTC) continue to operate.

3. This depends on the module. See the sections on the individual modules.

4. In clock mode 3 (FLL is used), canceling U-standby mode by CMT, KEYSC, or RTC is prohibited.

18.2 Input/Output Pins

Table 18.2 lists the pin configuration related to resets and power-down modes.

Table 18.2 Pin Configuration

Pin Name	Function	I/O	Description
STATUS0	Processing state 0	Output	Becomes high level in various standby modes (software standby mode and R/U-standby mode).
STATUS2	Processing state 2	Output	Becomes high level in R-standby mode.
$\overline{\text{RESETP}}$	Reset input pin	Input	This LSI enters the power-on reset state when this pin becomes low level.
$\overline{\text{RESETA}}$	Reset input pin	Input	This LSI enters the system reset state when this pin becomes low level.
$\overline{\text{RESETOUT}}$	Reset output signal	Output	Becomes low level while this LSI is being reset.
PDSTATUS	Power-down state signal	Output	Becomes high level when the power-supply separating region is turned off. PDSTATUS can control the supply current to the regulator.

18.3 Register Descriptions

Table 18.3 shows the register configuration for power-down modes. Table 18.4 shows the register states in each operating mode.

Table 18.3 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Standby control register	STBCR	R/W	H'A415 0020	32
Module stop register 0	MSTPCR0	R/W	H'A415 0030	32
Module stop register 1	MSTPCR1	R/W	H'A415 0034	32
Module stop register 2	MSTPCR2	R/W	H'A415 0038	32
Boot address register	BAR	R/W	H'A415 0040	32

Table 18.4 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
STBCR	Initialized	Retained	Retained	—	Retained	Retained	Retained
MSTPCR0	Initialized	Retained	Retained	—	Retained	Initialized	Retained
MSTPCR1	Initialized	Retained	Retained	—	Retained	Initialized	Retained
MSTPCR2	Initialized	Retained	Retained	—	Retained	Initialized	Retained
BAR	Initialized	Retained	Retained	—	Retained	Initialized	Retained

18.3.1 Standby Control Register (STBCR)

STBCR is a 32-bit readable/writable register that can select sleep mode, standby mode, or R/U-standby mode.

STBCR can be accessed only in longwords.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	STBY	—	RSTBY	USTBY	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	STBY	0	R/W	Standby Executing the SLEEP instruction after this bit is set to 1 makes a transition to software standby mode.
6	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	RSTBY	0	R/W	R-Standby Executing the SLEEP instruction after this bit is set to 1 makes a transition to R-standby mode.
4	USTBY	0	R/W	U-Standby Executing the SLEEP instruction after this bit is set to 1 makes a transition to U-standby mode.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Writing 1 to more than one bit for selecting a standby mode is prohibited; set all bits to 0 or set only one bit to 1.

When all bits for selecting power-down modes are cleared to 0, sleep mode is entered.

Note that the RSTBY bit should be set to 1 before entering R-standby mode, and when R-standby mode is canceled by an interrupt, the RSTBY bit should be returned to 0 after waking up from R-standby mode. Likewise, the USTBY bit should be set to 1 before entering U-standby mode, and when U-standby mode is canceled by a request from the CMU, KEYSC, or RTC, the USTBY bit should be returned to 0 after waking up from U-standby mode.

The boot address when R-standby mode is canceled by an interrupt is the value set in BAR.

18.3.2 Module Stop Register 0 (MSTPCR0)

MSTPCR0 is a 32-bit readable/writable register that can individually start or stop the module assigned to each bit.

MSTPCR0 can be accessed only in longwords.

After canceling the module stop state for the instruction cache (IC), operand cache (OC), TLB, either of the following preprocessing must be performed before accessing these modules. Note that such module access includes instruction fetch from a relevant module and instruction fetch using a relevant module.

- After reading the changed MSTPn bit once, execute the RTE instruction.
- After reading the changed MSTPn bit once, execute the ICBI instruction for any address. The address can be in a non-cacheable area.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSTP031	MSTP030	MSTP029	MSTP028	MSTP027	MSTP026	—	MSTP024	—	MSTP022	MSTP021	MSTP020	MSTP019	MSTP018	MSTP017	MSTP016
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTP015	MSTP014	MSTP013	MSTP012	—	MSTP010	MSTP009	MSTP008	MSTP007	MSTP006	MSTP005	MSTP004	—	MSTP002	MSTP001	—
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31	MSTP031	0	R/W	Module Stop Bit 031 When the MSTP031 bit is set to 1, the supply of the clock to the TLB is stopped. When the CPU core is used in 32-bit address expansion mode, this bit must be fixed at 0 (TLB operates). 0: TLB operates 1: Clock supply to TLB stopped
30	MSTP030	0	R/W	Module Stop Bit 030 When the MSTP030 bit is set to 1, the supply of the clock to the instruction cache (IC) is stopped. 0: IC operates 1: Clock supply to IC stopped
29	MSTP029	0	R/W	Module Stop Bit 029 When the MSTP029 bit is set to 1, the supply of the clock to the operand cache (OC) is stopped. 0: OC operates 1: Clock supply to OC stopped
28	MSTP028	0	R/W	Module Stop Bit 028 When the MSTP028 bit is set to 1, the supply of the clock to the RS memory is stopped. 0: RS memory operates 1: Clock supply to RS memory stopped
27	MSTP027	0	R/W	Module Stop Bit 027 When the MSTP027 bit is set to 1, the supply of the clock to the IL memory is stopped. 0: IL memory operates 1: Clock supply to IL memory stopped
26	MSTP026	0	R/W	Module Stop Bit 026 When the MSTP026 bit is set to 1, the supply of the clock to the secondary cache is stopped. 0: Secondary cache operates 1: Clock supply to secondary cache stopped
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0

Bit	Bit Name	Initial Value	R/W	Description
24	MSTP024	0	R/W	Module Stop Bit 024 When the MSTP024 bit is set to 1, the supply of the clock to the FPU is stopped. 0: FPU operates 1: Clock supply to FPU stopped
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	MSTP022	0	R/W	Module Stop Bit 022 When the MSTP022 bit is set to 1, the supply of the clock to the INTC is stopped. 0: INTC operates 1: Clock supply to INTC stopped
21	MSTP021	0	R/W	Module Stop Bit 021 When the MSTP021 bit is set to 1, the supply of the clock to the DMAC0 is stopped. 0: DMAC0 operates 1: Clock supply to DMAC0 stopped
20	MSTP020	0	R/W	Module Stop Bit 020 When the MSTP020 bit is set to 1, the supply of the clock to the SuperHyway bus is stopped. 0: SuperHyway bus operates 1: Clock supply to SuperHyway bus stopped
19	MSTP019	0	R/W	Module Stop Bit 019 When the MSTP019 bit is set to 1, the supply of the clock to the H-UDI is stopped. 0: H-UDI operates 1: Clock supply to H-UDI stopped
18	MSTP018	0	R/W	Module Stop Bit 018 When the MSTP018 bit is set to 1, the supply of the clock to the debugging function in the LSI (DBG) is stopped. This bit should be set to 0 when the H-UDI, UBC and AUD are used. 0: DBG operates 1: Clock supply to DBG stopped

Bit	Bit Name	Initial Value	R/W	Description
17	MSTP017	0	R/W	Module Stop Bit 017 When the MSTP017 bit is set to 1, the supply of the clock to the UBC is stopped. 0: UBC operates 1: Clock supply to UBC stopped
16	MSTP016	0	R/W	Module Stop Bit 016 The MSTP016 bit is a reserved bit for debugging. Set this bit to 1 in the initializing routine. Normally, set this bit to the immediately preceding read-out.
15	MSTP015	0	R/W	Module Stop Bit 015 When the MSTP015 bit is set to 1, the supply of the clock to the TMU0 is stopped. 0: TMU0 operates 1: Clock supply to TMU0 stopped
14	MSTP014	0	R/W	Module Stop Bit 014 When the MSTP014 bit is set to 1, the supply of the clock to the CMT is stopped 0: CMT operates 1: Clock supply to CMT stopped
13	MSTP013	0	R/W	Module Stop Bit 013 When the MSTP013 bit is set to 1, the supply of the clock to the RWDT is stopped. 0: RWDT operates 1: Clock supply to RWDT stopped
12	MSTP012	1	R/W	Module Stop Bit 012 When the MSTP012 bit is set to 1, the supply of the clock to the DMAC1 is stopped. 0: DMAC1 operates 1: Clock supply to DMAC1 stopped
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	MSTP010	0	R/W	Module Stop Bit 010 When the MSTP010 bit is set to 1, the supply of the clock to the TMU1 is stopped. 0: TMU1 operates 1: Clock supply to TMU1 stopped

Bit	Bit Name	Initial Value	R/W	Description
9	MSTP009	0	R/W	Module Stop Bit 009 When the MSTP009 bit is set to 1, the supply of the clock to the SCIF0 is stopped. 0: SCIF0 operates 1: Clock supply to SCIF0 stopped
8	MSTP008	0	R/W	Module Stop Bit 008 When the MSTP008 bit is set to 1, the supply of the clock to the SCIF0 is stopped. 0: SCIF1 operates 1: Clock supply to SCIF1 stopped
7	MSTP007	0	R/W	Module Stop Bit 007 When the MSTP007 bit is set to 1, the supply of the clock to the SCIF2 is stopped. 0: SCIF2 operates 1: Clock supply to SCIF2 stopped
6	MSTP006	0	R/W	Module Stop Bit 006 When the MSTP006 bit is set to 1, the supply of the clock to the SCIF3 is stopped. 0: SCIF3 operates 1: Clock supply to SCIF3 stopped
5	MSTP005	0	R/W	Module Stop Bit 005 When the MSTP005 bit is set to 1, the supply of the clock to the SCIF4 is stopped. 0: SCIF4 operates 1: Clock supply to SCIF4 stopped
4	MSTP004	0	R/W	Module Stop Bit 004 When the MSTP005 bit is set to 1, the supply of the clock to the SCIF5 is stopped. 0: SCIF5 operates 1: Clock supply to SCIF5 stopped
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	MSTP002	0	R/W	Module Stop Bit 002 When the MSTP002 bit is set to 1, the supply of the clock to the MSIOF0 is stopped. 0: MSIOF0 operates 1: Clock supply to S MSIOF0 stopped
1	MSTP001	0	R/W	Module Stop Bit 001 When the MSTP001 bit is set to 1, the supply of the clock to the MSIOF1 is stopped. 0: MSIOF1 operates 1: Clock supply to S MSIOF1 stopped
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Note: To write to a certain bit in MSTPCR0, read all bit values from MSTPCR0 first and then write back to MSTPCR0 with only the target bit value changed.

18.3.3 Module Stop Register 1 (MSTPCR1)

MSTPCR1 is a 32-bit readable/writable register that can individually start or stop the module assigned to each bit.

MSTPCR1 can be accessed only in longwords.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MSTP 112	MSTP 111	—	MSTP 109	MSTP 108	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	MSTP112	0	R/W	Module Stop Bit 112 When the MSTP112 bit is set to 1, the supply of the clock to the KEYSC is stopped. 0: KEYSC operates 1: Clock supply to KEYSC stopped
11	MSTP111	0	R/W	Module Stop Bit 111 When the MSTP111 bit is set to 1, the supply of the clock to the RTC is stopped. 0: RTC operates 1: Clock supply to RTC stopped
10	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	MSTP109	0	R/W	Module Stop Bit 109 When the MSTP109 bit is set to 1, the supply of the clock to the I^2C0 is stopped. 0: I^2C0 operates 1: Clock supply to I^2C0 stopped

Bit	Bit Name	Initial Value	R/W	Description
8	MSTP108	0	R/W	Module Stop Bit 108 When the MSTP108 bit is set to 1, the supply of the clock to the I^2C1 is stopped. 0: I^2C1 operates 1: Clock supply to I^2C1 stopped
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: To write to a certain bit in MSTPCR1, read all bit values from MSTPCR1 first and then write back to MSTPCR1 with only the target bit value changed.

18.3.4 Module Stop Register 2 (MSTPCR2)

MSTPCR2 is a 32-bit readable/writable register that can individually start or stop the module assigned to each bit.

MSTPCR2 can be accessed only in longwords.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MSTP229	MSTP228	—	MSTP226	MSTP225	MSTP224	—	MSTP222	MSTP221	MSTP220	MSTP219	MSTP218	MSTP217	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSTP215	—	MSTP213	MSTP212	—	MSTP210	MSTP209	—	—	MSTP206	MSTP205	MSTP204	MSTP203	MSTP202	MSTP201	MSTP200
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 30	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
29	MSTP229	1	R/W	Module Stop Bit 229 When the MSTP229 bit is cleared to 0, the supply of the clock to the MMC is started. 0: MMC operates 1: Clock supply to MMC stopped
28	MSTP228	1	R/W	Module Stop Bit 228 When the MSTP228 bit is cleared to 0, the supply of the clock to the EtherMAC is started. 0: EtherMAC operates 1: Clock supply to EtherMAC stopped
27	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
26	MSTP226	1	R/W	Module Stop Bit 226 When the MSTP226 bit is cleared to 0, the supply of the clock to the ATAPI is started. 0: ATAPI operates 1: Clock supply to ATAPI stopped

Bit	Bit Name	Initial Value	R/W	Description
25	MSTP225	1	R/W	Module Stop Bit 225 When the MSTP225 bit is cleared to 0, the supply of the clock to the TPU is started. 0: TPU operates 1: Clock supply to TPU stopped
24	MSTP224	1	R/W	Module Stop Bit 224 When the MSTP224 bit is cleared to 0, the supply of the clock to the IrDA is started. 0: IrDA operates 1: Clock supply to IrDA stopped
23	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
22	MSTP222	1	R/W	Module Stop Bit 222 When the MSTP222 bit is cleared to 0, the supply of the clock to the TSIF is started. 0: TSIF operates 1: Clock supply to TSIF stopped
21	MSTP221	1	R/W	Module Stop Bit 221 When the MSTP221 bit is cleared to 0, the supply of the clock to the USB1 is started. 0: USB1 operates 1: Clock supply to USB1 stopped
20	MSTP220	1	R/W	Module Stop Bit 220 When the MSTP220 bit is cleared to 0, the supply of the clock to the USB0 is started. 0: USB0 operates 1: Clock supply to USB0 stopped
19	MSTP219	1	R/W	Module Stop Bit 219 When the MSTP219 bit is cleared to 0, the supply of the clock to the 2DG is started. 0: 2DG operates 1: Clock supply to 2DG stopped

Bit	Bit Name	Initial Value	R/W	Description
18	MSTP218	1	R/W	Module Stop Bit 218 When the MSTP218 bit is cleared to 0, the supply of the clock to the SDHI0 is started. 0: SDHI0 operates 1: Clock supply to SDHI0 stopped
17	MSTP217	1	R/W	Module Stop Bit 217 When the MSTP218 bit is cleared to 0, the supply of the clock to the SDHI1 is started. 0: SDHI1 operates 1: Clock supply to SDHI1 stopped
16	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
15	MSTP215	1	R/W	Module Stop Bit 215 When the MSTP215 bit is cleared to 0, the supply of the clock to the VEU1 is started. 0: VEU1 operates 1: Clock supply to VEU1 stopped
14	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
13	MSTP213	1	R/W	Module Stop Bit 213 When the MSTP213 bit is cleared to 0, the supply of the clock to the CEU1 is started. 0: CEU1 operates 1: Clock supply to CEU1 stopped
12	MSTP212	1	R/W	Module Stop Bit 212 When the MSTP212 bit is cleared to 0, the supply of the clock to the BEU1 is started. 0: BEU1 operates 1: Clock supply to BEU1 stopped
11	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
10	MSTP210	1	R/W	Module Stop Bit 210 When the MSTP210 bit is cleared to 0, the supply of the clock to the 2DDMAC is started. 0: 2DDMAC operates 1: Clock supply to 2DDMAC stopped
9	MSTP209	1	R/W	Module Stop Bit 209 When the MSTP209 bit is cleared to 0, the supply of the clock to the SPU is started. 0: SPU operates 1: Clock supply to SPU stopped
8	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
7	MSTP207	1	R/W	Module Stop Bit 207 When the MSTP207 bit is cleared to 0, the supply of the clock to the GCU is started. 0: GCU operates 1: Clock supply to GCU stopped
6	MSTP206	1	R/W	Module Stop Bit 206 When the MSTP206 bit is cleared to 0, the supply of the clock to the JPU is started. 0: JPU operates 1: Clock supply to JPU stopped
5	MSTP205	1	R/W	Module Stop Bit 205 When the MSTP205 bit is cleared to 0, the supply of the clock to the VOU is started. 0: VOU operates 1: Clock supply to VOU stopped
4	MSTP204	1	R/W	Module Stop Bit 204 When the MSTP204 bit is cleared to 0, the supply of the clock to the BEU0 is started. 0: BEU0 operates 1: Clock supply to BEU0 stopped

Bit	Bit Name	Initial Value	R/W	Description
3	MSTP203	1	R/W	Module Stop Bit 203 When the MSTP203 bit is cleared to 0, the supply of the clock to the CEU0 is started. 0: CEU0 operates 1: Clock supply to CEU0 stopped
2	MSTP202	1	R/W	Module Stop Bit 202 When the MSTP202 bit is cleared to 0, the supply of the clock to the VEU0 is started. 0: VEU0 operates 1: Clock supply to VEU0 stopped
1	MSTP201	1	R/W	Module Stop Bit 201 When the MSTP201 bit is cleared to 0, the supply of the clock to the VPU is started. 0: VPU operates 1: Clock supply to VPU stopped
0	MSTP200	1	R/W	Module Stop Bit 200 When the MSTP200 bit is cleared to 0, the supply of the clock to the LCDC is started. 0: LCDC operates 1: Clock supply to LCDC stopped

Note: To write to a certain bit in MSTPCR2, read all bit values from MSTPCR2 first and then write back to MSTPCR2 with only the target bit value changed.

18.3.5 Boot Address Register (BAR)

BAR is a 32-bit readable/writable register that specifies the start address after waking up from R-standby mode. BAR can specify an address in RS memory or in area 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BAR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BAR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	BAR[31:0]	H'0000 0000	R/W	These bits set the boot address on waking up from R-standby mode.

18.4 Operation

18.4.1 Reset

This LSI has three types of resets: a power-on reset, system reset, and a manual reset.

(1) Power-on Reset

A power-on reset must always be used when power is supplied. The pin used for a power-on reset is $\overline{\text{RESETP}}$. A power–on reset stops all processing and cancels the processing of all events that are yet to be processed. Then reset processing starts immediately. For the timing sequence in a power-on reset, see section 52, Electrical Characteristics.

The registers initialized exclusively by a power-on reset are listed below. These are not initialized by a system reset.

1. Bits 7 and 4 in RWTCSR (RWDT)
2. Bits 15 and 14 in CMCSR (CMT)

(2) System Reset

A system reset is used for re-execution from the initial state. The pin used for a system reset is $\overline{\text{RESETA}}$. A system reset stops all processing and cancels the processing of all events that are yet to be processed. Then reset processing starts immediately. The conditions for generating a system reset are as follows:

1. Input of a low level on the $\overline{\text{RESETA}}$ pin
2. Overflow of the RWDT after it has started counting
3. Generation of an H-UDI reset (for details on the H-UDI reset, see section 50, User Debugging Interface (H-UDI)).

Exception handling by the CPU in the case of a system reset is the same as that for a power-on reset.

(3) Manual Reset

A manual reset is a reset generated by software. For details on the manual reset, see section 5, Exception Handling.

18.4.2 Sleep Mode

(1) Transition to Sleep Mode

Executing the SLEEP instruction when the STBY, RSTBY, USTBY bits in STBCR are 0 causes a transition from the program execution state to sleep mode. In sleep mode, the supply of the clock to the CPU core is stopped. Although the CPU stops immediately after executing the SLEEP instruction, the contents of the CPU core registers and memory remain unchanged. The on-chip peripheral modules continue to operate in sleep mode and the clock continues to be output to the CKO and MCLK pins.

The procedure for a transition to sleep mode is as follows:

1. Clear the STBY, RSTBY, and USTBY bits in STBCR to 0.
2. Execute the SLEEP instruction.

(2) Canceling Sleep Mode

Sleep mode is canceled by an interrupt (NMI, IRQ, or on-chip peripheral module) or a reset.

Interrupts are accepted in sleep mode even when the BL bit in SR is 1. If necessary, save SPC and SSR in the stack before executing the SLEEP instruction.

(a) Canceling with Interrupt

When an NMI, IRQ, or on-chip peripheral module interrupt occurs, sleep mode is canceled and interrupt exception handling is executed. A code indicating the interrupt source is set in INTEVT.

For the types of on-chip peripheral module interrupts, see section 13, Interrupt Controller (INTC).

(b) Canceling with Reset

Sleep mode is canceled by a power-on reset or a system reset.

18.4.3 Software Standby Mode

(1) Transition to Software Standby Mode

Executing the SLEEP instruction when the STBY bit is 1 and the USTBY bit is 0 in STBCR causes a transition from the program execution state to software standby mode. Since the CPG is stopped in software standby mode, modules other than those operated by RCLK are also stopped (modules operated by RCLK: RWDT, CMT, KEYSC, and RTC). However, functions of the I/O area that do not require the clock (detection of NMI and IRQ interrupts) continue.

The contents of the CPU core registers and memory remain unchanged. For the register states of the on-chip peripheral modules, see the descriptions of registers in individual sections.

In software standby mode, input RCLK and RTC_CLK to operate the RWDT, CMT, KEYSC, and RTC. In clock operating modes 0 to 2, input EXTAL as the source clock of RCLK, and also input RTC_CLK when the RTC is used. In clock modes 3 to 7, input RTC_CLK as the source clock of RCLK.

The procedure for a transition to software standby mode is as follows:

1. Set the STBY bit to 1 and clear the USTBY bit to 0 in STBCR.
2. Execute the SLEEP instruction.
3. Software standby mode is entered and the clocks within the LSI are halted. The output of the STATUS0 pin goes high.

(2) Canceling Software Standby Mode

Software standby mode is canceled by an interrupt (NMI, IRQ, CMT, KEYSC, or RTC), a power-on reset, or a system reset.

(a) Canceling with Interrupt

Operation of the CPG is started in response to an NMI, IRQ, CMT, KEYSC, or RTC interrupt. After oscillation of all the PLL and DLL circuits that should operate has settled, the supply of the clock to the entire LSI starts. Here, the PLL and DLL oscillation settling times are automatically ensured by the LSI. When software standby mode is canceled, the STATUS0 pin goes low. After this, the interrupt exception handling by the CPU is executed. Interrupts are accepted in software standby mode even when the BL bit in SR is 1. If necessary, save SPC and SSR in the stack before executing the SLEEP instruction.

The clock outputs of the CKO and HPCLK pins are halted until software standby mode is canceled.

(b) Canceling with Reset

Software standby mode is canceled by a power-on reset or system reset. The start address after returning from software standby mode is the reset vector address (H'A000 0000).

18.4.4 Module Standby Function

(1) Transition to Module Standby State

Setting the MSTP bits in the module stop registers to 1 halts the supply of clocks to the corresponding on-chip peripheral modules. This function can be used to reduce power consumption in the normal operation of the CPU.

Modules in the module standby state keep the state immediately before the transition to the module standby state. The registers keep the contents before halted, and the external pins keep the functions before halted. On return from the module standby state, operation is restarted from the condition immediately before the registers and external pins have halted.

Note: Make sure to set the MSTP bit to 1 while the modules have completed the operation and are in an the idle state, with no interrupt sources from the external pins or other modules.

(2) Canceling Module Standby State

The module standby state can be canceled by clearing the respective MSTP bit to 0.

18.4.5 U-Standby Mode

(1) Transition to U-Standby Mode

Executing the SLEEP instruction when the USTBY bit is 1 and the STBY bit is 0 in STBCR causes a transition from the program execution state to U-standby mode. In U-standby mode, power to the I/O area and sub area of the power-supply separating region is on, but power to the core area is off. This provides a greater reduction of the leak current than is achieved in software standby mode.

In U-standby mode, the RWDT, CMT, KEYSC, and RTC (modules that can operate from RCLK) continue to operate. NMI and IRQ interrupt detection is not operational.

Since the contents of the registers and memory for all modules in the core area that are turned off are lost on entry to U-standby mode, settings must be re-made on return from U-standby mode.

In U-standby mode, input RCLK and RTC_CLK to operate the RWDT, CMT, KEYSC, and RTC. In clock modes 0 to 2, input EXTAL as the source clock of RCLK, and also input RTC_CLK when the RTC is used. In clock modes 3 to 7, input RTC_CLK as the source clock of RCLK.

The procedure for a transition to U-standby mode is as follows:

1. Set the USTBY bit to 1 and clear the STBY bit to 0 in STBCR.
2. Execute the SLEEP instruction.
3. U-standby mode is entered and the clocks within the LSI are halted. The output of the STATUS0 pin goes high. When the core area is turned off, the output of the PDSTATUS pin goes high.

(2) Canceling U-Standby Mode

U-standby mode is canceled by a power-on reset, system reset, CMT, KEYSC, or RTC and the core area is turned on. The start address after returning from U-standby mode is the reset vector address (H'A0000000). Note that canceling U-standby mode by the CMT, KEYSC, or RTC is prohibited in clock operating mode 3; use a power-on reset or a system reset to cancel U-standby mode in clock mode 3.

18.4.6 R-Standby Mode

(1) Transition to R-Standby Mode

Executing the SLEEP instruction when the RSTBY bit is 1 and the STBY and USTBY bits are 0 in STBCR causes a transition from the program execution state to R-standby mode. In R-standby mode, power to the I/O area, backup area, and sub area of the power-supply separating region is on, but power to the core area is off. This provides a greater reduction of the leak current than is achieved in software standby mode. In addition, since the register contents and the destination address on waking up from R-standby mode can be held in the backup area, high-speed wakeup by an interrupt is possible.

In R-standby mode, modules that can operate on RCLK or RTC_CLK (RWDT, CMT, KEYSC, RTC) continue to operate and the I/O area functions of NMI and IRQ interrupt detection are operational.

In R-standby mode, input RCLK and RTC_CLK to operate the RWDT, CMT, KEYSC, and RTC. In clock modes 0 to 2, input EXTAL as the source clock of RCLK, and also input RTC_CLK when the RTC is used. In clock operating modes 3 to 7, input RTC_CLK as the source clock of RCLK.

In R-standby mode, the contents of the registers and memory for each module in the core area are initialized or held by being saved in the backup area, depending on the module. For the states of each register and memory in R-standby mode, see section 18.4.6 (3), Registers and Memory Held in R-Standby Mode. Since the contents of the registers and memory for initialized modules are lost on entry to R-standby mode, operation should be guaranteed by software.

The procedure for a transition to R-standby mode is as follows:

1. Set the RSTBY bit to 1 and clear the STBY and USTBY bits to 0 in STBCR.
2. Set in BAR the first branch destination address used when making a transition from R-standby mode to the program execution state.
3. Execute the SLEEP instruction.
4. R-standby mode is entered and the clocks within the LSI are halted. The outputs of the STATUS0 and STATUS2 pins go high. When the core area is turned off, the output of the PDSTATUS pin goes high.

(2) Canceling R-Standby Mode

R-standby mode is canceled by an interrupt (NMI, IRQ, or CMT, KEYSC, RTC), a power-on reset, or a system reset.

(a) Canceling with Interrupt

When an NMI, IRQ, or CMT, KEYSC, RTC interrupt is detected, the core area is turned on. The reset is asserted only in the core area*, and the clock will be supplied to the entire LSI after oscillation of all PLL and DLL circuits that should operate has settled. In this case, the PLL and DLL oscillation settling times are automatically ensured by the LSI. When R-standby mode is canceled, the STATUS0 and STATUS2 pins go low. The start address on waking up from R-standby mode is the address set in BAR. After this, the interrupt exception handling by the CPU is executed. Interrupts are accepted in R-standby mode even when the BL bit in SR is 1. If necessary, save SPC and SSR in the stack before executing the SLEEP instruction.

The clock outputs of the CKO and LPCLK pins are halted until R-standby mode is canceled.

Note: * The $\overline{\text{RESETOUT}}$ pin is not asserted.

(b) Canceling with Reset

R-standby mode is canceled by a power-on reset or a system reset, and the core area is turned on. The start address after returning from R-standby mode is the reset vector address (H'A000 0000).

(3) Registers and Memory Held in R-Standby Mode

In R-standby mode, the register contents saved in the backup area and the RS memory remain unchanged. When R-standby mode is canceled by an interrupt, operation is started with the same register values before the transition to R-standby mode. This recovering process after canceling R-standby mode is operated automatically prior to the execution of the instruction specified by BAR. For details on the held registers, refer to the register descriptions in each section.

(4) R-Standby Mode Usage Example

In R-standby mode, only partial data necessary for restarting operation is saved. Data not saved must be guaranteed by software. Procedures for halting operating modules are also processed by software.

(a) Processing before Transition

1. Set the MD, BL, and RB bits in SR to 1.
2. Carry out the processing to halt on-chip peripheral modules. As the FIFO buffer contents are not saved, confirm that the FIFO buffer is empty before executing the processing to halt operation.
3. Save data, such as values set in registers, in external memory whose contents are held and the on-chip RS memory.
4. When cache is enabled, purge (write back to main memory), reset, and disable cache. When the TLB is used, reset and disable the TLB after carrying out the processing necessary for restarting operation.
5. Set the RSTBY bit to 1 and clear the STBY and USTBY bits to 0 in STBCR.
6. Set the address of the instruction to be executed first when transiting from R-standby mode to the program execution state in BAR.
7. Execute the SLEEP instruction.
8. R-standby mode is entered.

(b) Processing after Transition

1. The values set in the BSC, DBSC, INTC, CPG, PFC, and I/O ports and the RS memory contents are saved when waking up from R-standby mode. All other contents need to be re-specified by software to guarantee operation.
2. Start executing the instruction in the address set in BAR. At this time, the interrupt initiating wakeup is pended since SR is reset and the BL bit in SR is set to 1.
3. Set SR and R15. The BL bit in SR should remain set to 1.
4. Use software to process the procedures for restarting on-chip peripheral modules. After respecifying the control registers and others, set the module enable bits to 1.
5. Newly specify the VBR, TLB and cache.
6. Clear the BL bit in SR to 0. Set other bits accordingly.
7. The pended interrupt is accepted, a branch is made to the interrupt handler, and normal interrupt handling is executed.

18.4.7 Mode Transitions

Figure 18.1 shows the mode transitions of power-down modes.

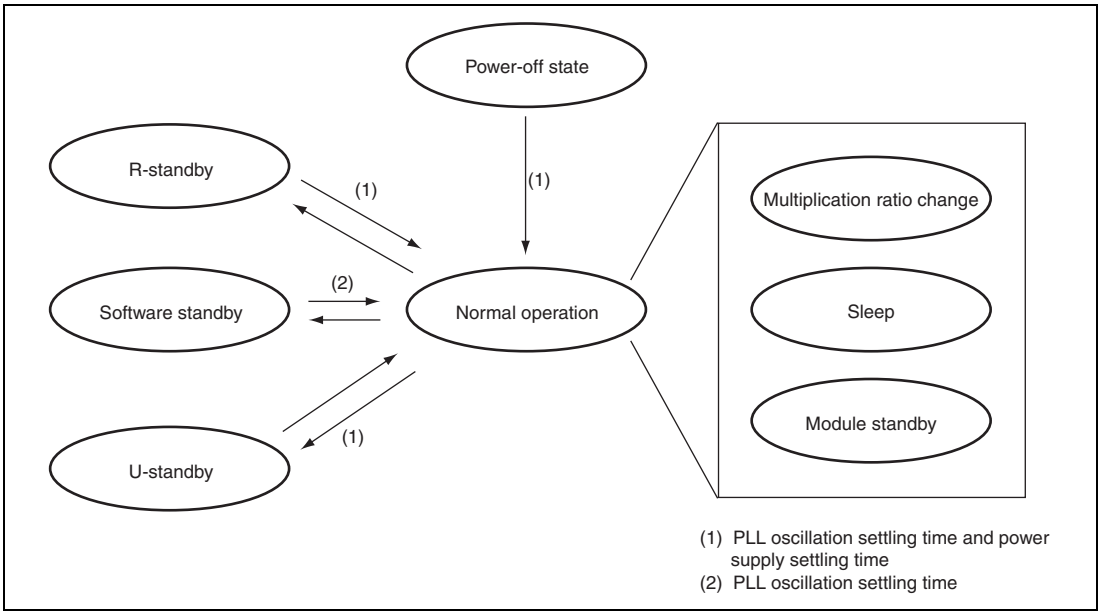
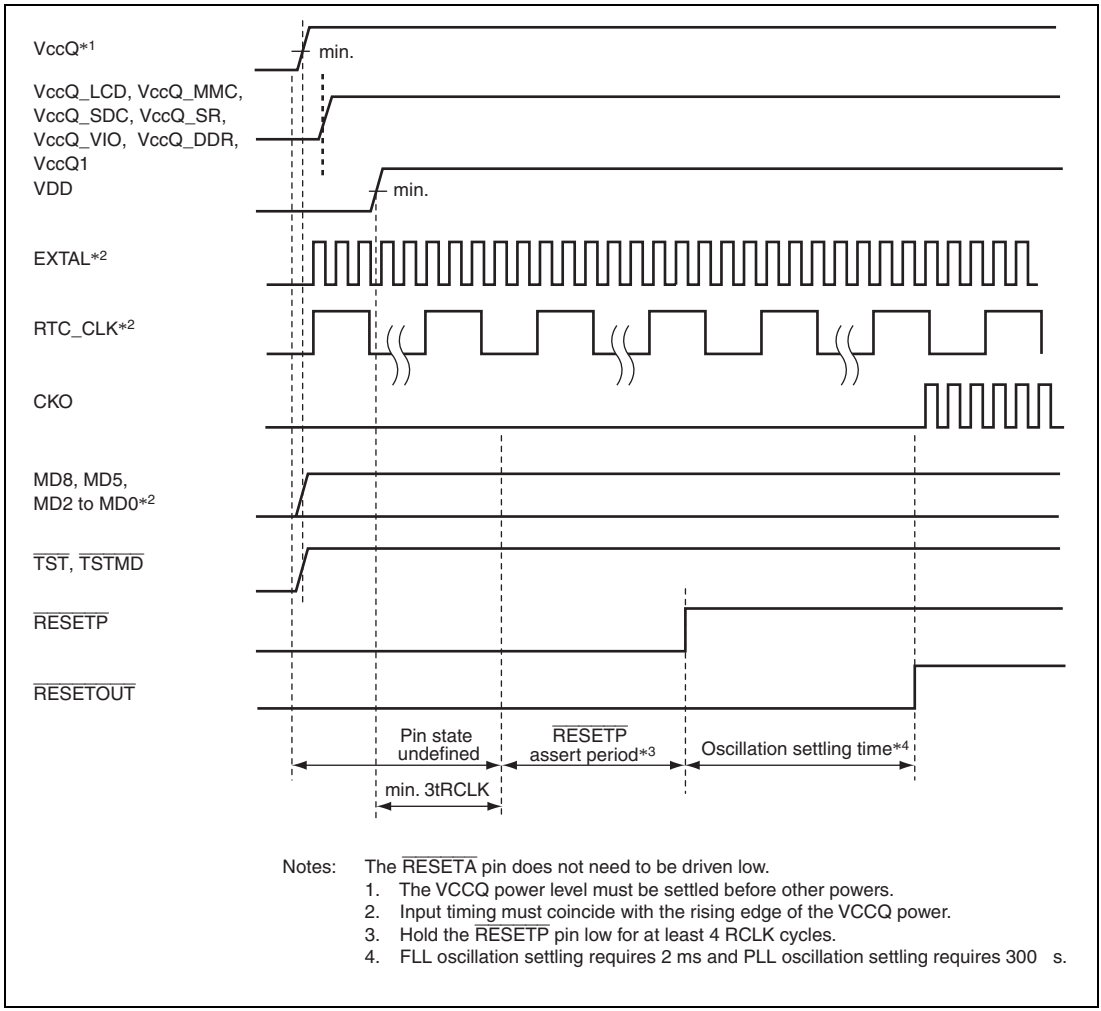


Figure 18.1 Mode Transition Diagram

18.4.8 Power-On Sequence

Figure 18.2 shows the power-on sequence.

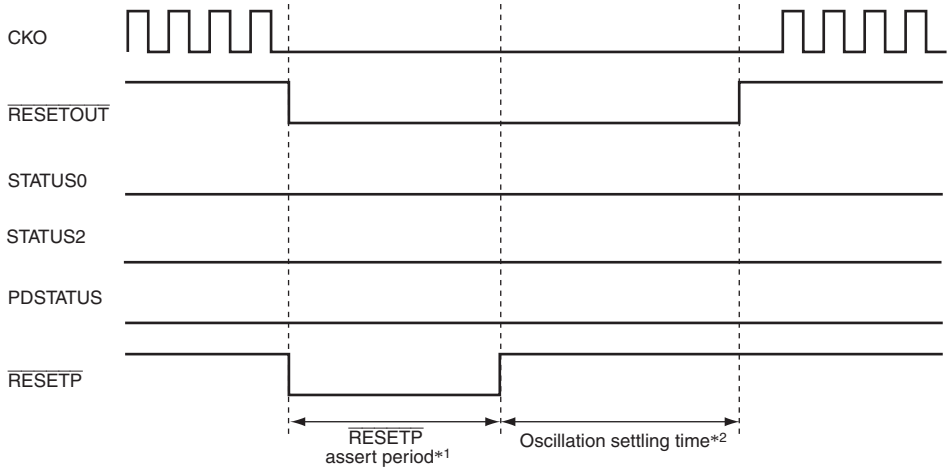


Notes: The RESETA pin does not need to be driven low.
1. The VCCQ power level must be settled before other powers.
2. Input timing must coincide with the rising edge of the VCCQ power.
3. Hold the RESETP pin low for at least 4 RCLK cycles.
4. FLL oscillation settling requires 2 ms and PLL oscillation settling requires 300 s.

Figure 18.2 Power-On Sequence

18.4.9 Output Pins Change Timing

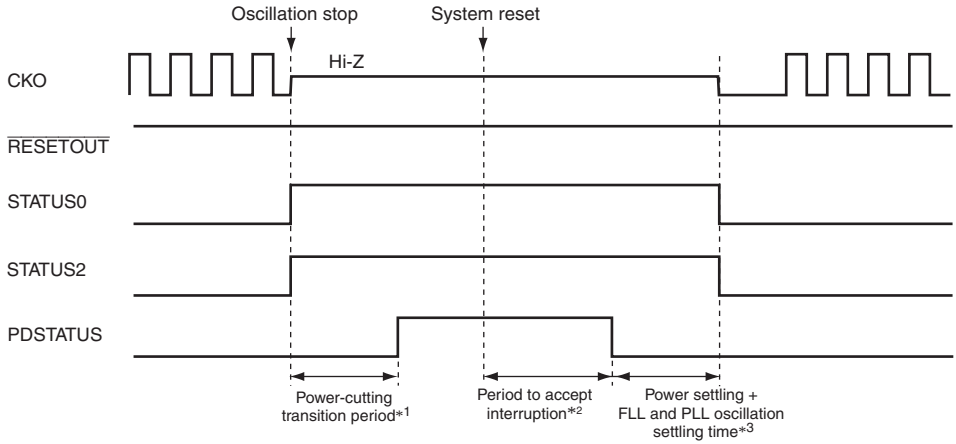
Figure 18.3 shows the state of output pins at a power-on reset.



Notes: The RESETA pin does not need to be driven low (fixed high).
1. Hold the RESETP pin low for at least 4 RCLK cycles.
2. FLL oscillation settling requires 2 ms and PLL oscillation settling requires 300 s.

Figure 18.3 State of Output Pins at Power-On Reset

Figure 18.4 shows the state of output pins in R-standby mode.



Notes: 1. Turning off power of the power-cutting region requires approximately 3 RCLK cycles. After that, approximately 3 RCLK cycles elapse until the PDSTATUS pin is driven high.
2. Starting the operation to cancel R-standby mode after an interrupt request is issued requires approximately 2 RCLK cycles.
3. Settling of the power after the power of the power-cutting region is turned on requires 200 µs. FLL oscillation settling requires 2 ms and PLL oscillation settling requires 300 s.

Figure 18.4 State of Output Pins for R-Standby → R-Standby Cancellation by Interrupt

Figure 18.5 shows the state of output pins in software standby mode.

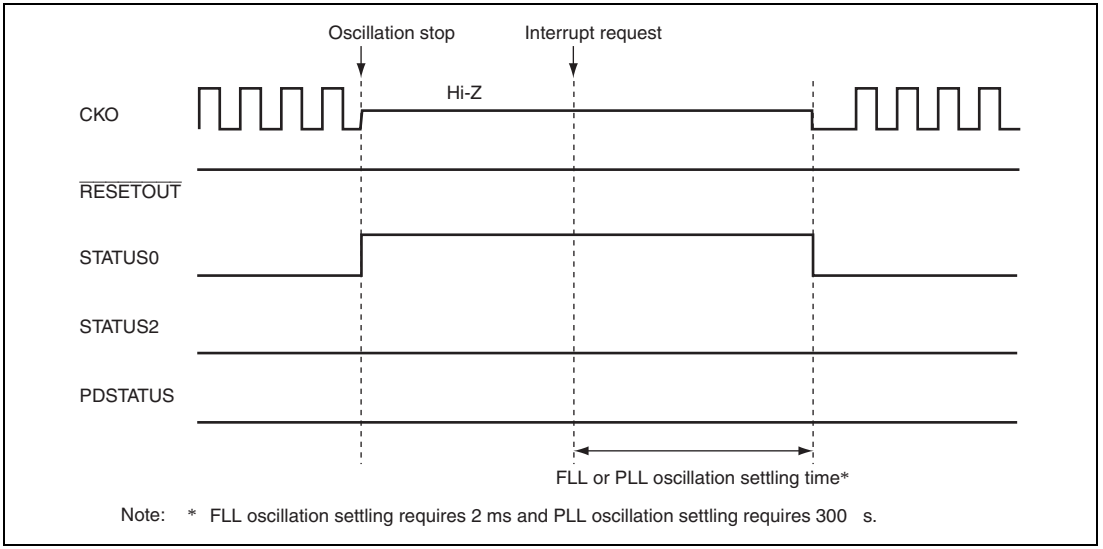
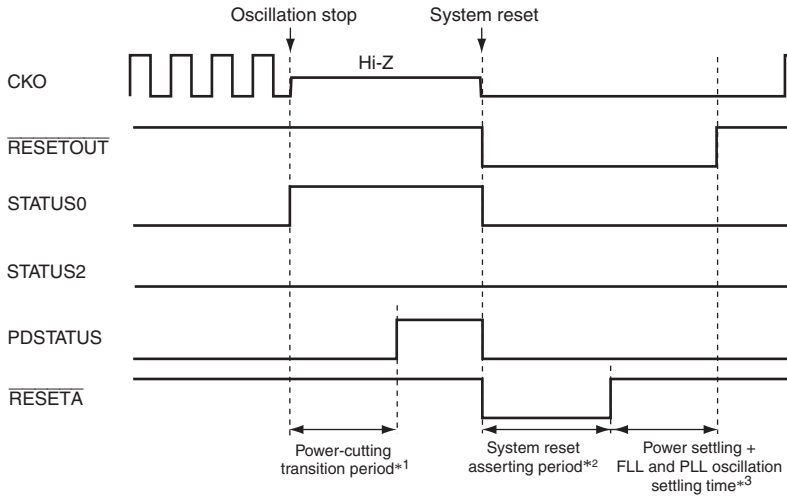


Figure 18.5 State of Output Pins for Software Standby → Software Standby Cancellation by Interrupt

Figure 18.6 shows the state of output pins in U-standby mode.



Notes: 1. Turning off power of the power-cutting region requires approximately 3 RCLK cycles. After that, approximately 3 RCLK cycles elapse until the PDSTATUS pin is driven high.
2. The system reset asserting period requires at least 4 RCLK cycles.
3. Settling of the power after the power of the power-cutting region is turned on requires 200 s. FLL oscillation settling requires 2 ms and PLL oscillation settling requires 300 s.

Figure 18.6 State of Output Pins for U-Standby → U-Standby Cancellation by Reset

Section 19 RCLK Watchdog Timer (RWDT)

This LSI includes the RCLK watchdog timer (RWDT).

The RWDT is a single-channel timer that uses a RTC clock as an input and can be used as a watchdog timer for the system monitoring.

This LSI can be reset by the overflow of the counter when the value of the counter has not been updated because of a system runaway.

19.1 Features

- Can be used as a watchdog timer. A system reset is generated when the counter overflows.
- Choice of eight counter input clocks.
  Eight clocks (RCLK/1 to RCLK/4096) that are obtained by dividing the RCLK.

Figures 19.1 shows block diagrams of the RWDT.

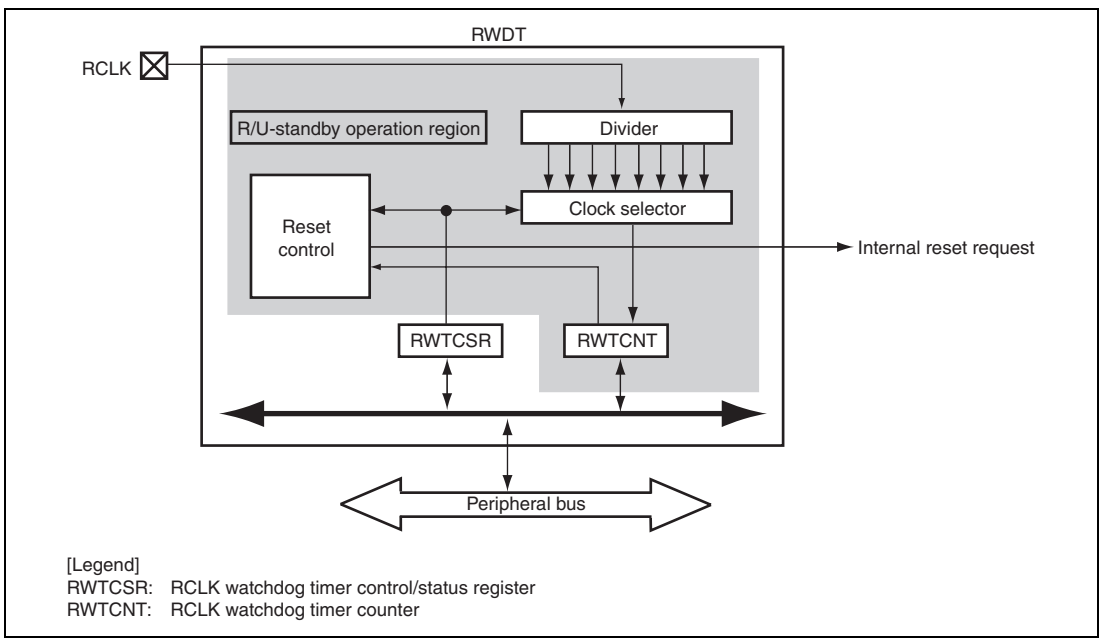


Figure 19.1 Block Diagram of RWDT

19.2 Input/Output Pins for RWDT

Table 19.1 lists the pin configuration and functions of the RWDT.

Table 19.1 RWDT Pin Configuration

Pin Name	Function	I/O	Description
RCLK	RTC clock	Input	Clock input from an external RTC (32.768 kHz)

19.3 Register Descriptions for RWDT

Table 19.2 shows the RWDT register configuration. Table 19.3 shows the register state in each processing mode.

Table 19.2 Register Configuration of RWDT

Name	Abbreviation	R/W	Address	Access Size
RCLK watchdog timer counter	RWTCNT	R/W	H'A4520000	8/16*
RCLK watchdog timer control/status register	RWTCSR	R/W	H'A4520004	8/16*

Note: * Write is performed in 16-bit and read in 8-bit.

Table 19.3 Register State of RWDT in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
RWTCNT	Initialized	Retained	Retained	Retained	Retained	Initialized	Retained
RWTCSR	Initialized	Retained	Retained	Retained	Initialized*	Initialized*	Retained

Note: * Values of the WOVF and SRSTF bits change according to the operation.

19.3.1 RCLK Watchdog Timer Counter (RWTCNT)

RWTCNT is an 8-bit readable/writable register that increment on the selected clock. When an overflow occurs, it generates a system reset.

The RWTCNT counter is initialized to H'00 by a power-on reset and a system reset. Use a word access to write to the RWTCNT counter, with H'5A in the upper byte. Use a byte access to read RWTCNT.

Bit:	7	6	5	4	3	2	1	0
	RWTCNT							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.2 RCLK Watchdog Timer Control/Status Register (RWTCSR)

RWTCSR is an 8-bit readable/writable register composed of bits to select the clock used for the count, overflow flags, and enable bit.

RWTCSR is initialized to H'87 by a power-on reset. Use a word access to write to RWTCSR, with H'A5 in the upper byte. Use a byte access to read RWTCSR.

Bit:	7	6	5	4	3	2	1	0
	TME	—	WRFLG	WOVF	SRSTF	CKS[2:0]*2		
Initial value:	1	0	0	0	0	1	1	1
R/W:	R/W	R	R	R/W	R/(W)*1	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TME	1	R/W	Starts and stops timer operation. 0: Timer disabled: Count-up stops and RWTCNT value is retained 1: Timer enabled
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5	WRFLG	0	R	Write Status Flag The writing to the RWTCNT is disabled during this bit is 1.The writing to the RWTCNT is masked for the prescribed period to synchronize after the writing to the RWTCNT. Confirm that this bit is 0 to write to continuously the RWTCNT.

Bit	Bit Name	Initial Value	R/W	Description
4	WOVF	0	R/W	Indicates that the RWTCNT has overflowed. Write 0 to this bit before using the RWDT. 0: No overflow 1: RWTCNT has overflowed
3	SRSTF	0	R/(W)*[1]	System Reset Input Flag This bit is set to 1 when a system reset is input. Writing 0 to this bit clears the flag. This bit must be cleared to 0 before shifting to U-standby mode. 0: No system reset is input 1: System reset is input
2 to 0	CKS[2:0]*[2]	111	R/W	RTC Clock Select These bits select the clock to be used for the RWTCNT count from the eight types obtainable by dividing the RTC clock. The overflow period that is shown inside the parenthesis in the table is the value when the RTC clock is 32.768 kHz. 000: Rϕ (7.8 ms) 001: Rϕ/4 (31.3 ms) 010: Rϕ/16 (125.0 ms) 011: Rϕ/32 (250.0 ms) 100: Rϕ/64 (500.0 ms) 101: Rϕ/128 (1.0 s) 110: Rϕ/1024 (8.0 s) 111: Rϕ/4096 (32.0 s)

Notes: 1. Only 0 can be written to clear the flag.
2. If bits CKS[2:0] are modified when the RWDT is operating, the up-count may not be performed correctly. Ensure that the bits CKS[2:0] are modified only when the RWDT is not operating.

19.3.3 Notes on Register Access

The writing procedure to RWTCNT and RWTCSR differs from that of other registers with the purpose of preventing an unintended write. The procedure for writing to these registers is given below.

Writing to RWTCNT and RWTCSR:

- These registers must be written by a word transfer instruction. They cannot be written by a byte or longword transfer instruction.
- When writing to RWTCNT, set the upper byte to H'5A and transfer the lower byte as the write data. When writing to RWTCSR, set the upper byte to H'A5 and transfer the lower byte as the write data.

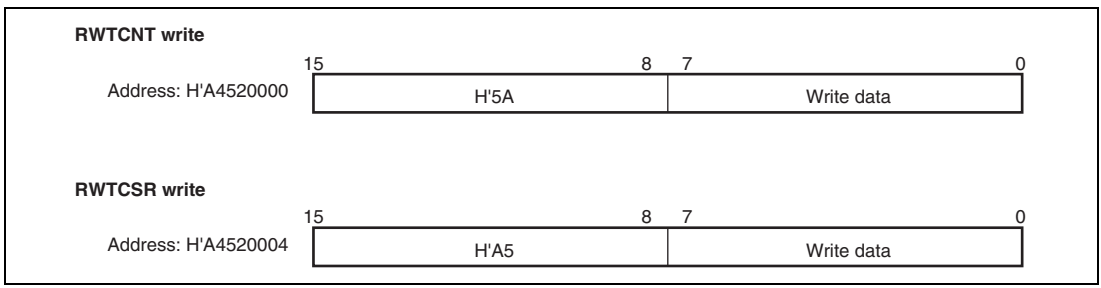


Figure 19.2 Writing to RWTCNT and RWTCSR

19.4 RWDT Usage

19.4.1 Control of System Runaway

The RWDT is activated by a power-on reset, and starts counting up on the RCLK. When the counter overflow occurs, an internal reset is again generated. By this function, an internal reset can be automatically generated even when this LSI has caused a system runaway.

1. After a power-on reset, stop the RWDT by clearing the TME bit in RWTCSR to 0 in the boot routine before RWTCNT overflows.
2. Clear WOVF in RWTCSR to 0.
3. Set the kind of count clock to bits CKS[2: 0] in RWTCSR.
4. Start the counting by setting the TME bit in RWTCSR to 1.
5. Write periodically RWTCNT to H'00 so that RWTCNT does not overflow.
6. When RWTCNT overflows, a system reset is generated because the RWDT sets the WOVF flag in RWTCSR to 1. At this time, RWTCNT and RWTCSR are initialized.

Section 20 Timer Unit (TMU)

This LSI includes two three-channel 32-bit timer units (TMU).

20.1 Features

- Each channel is provided with an auto-reload 32-bit down counter
- All channels are provided with 32-bit constant registers and 32-bit down counters that can be read or written to at any time
- All channels generate interrupt requests when the 32-bit down counter underflows (H'00000000 → H'FFFFFFFF)
- Allows selection among five counter input clocks: Pϕ/4, Pϕ/16, Pϕ/64, Pϕ/256, and Pϕ/1024

Figure 20.1 shows a block diagram of the TMU.

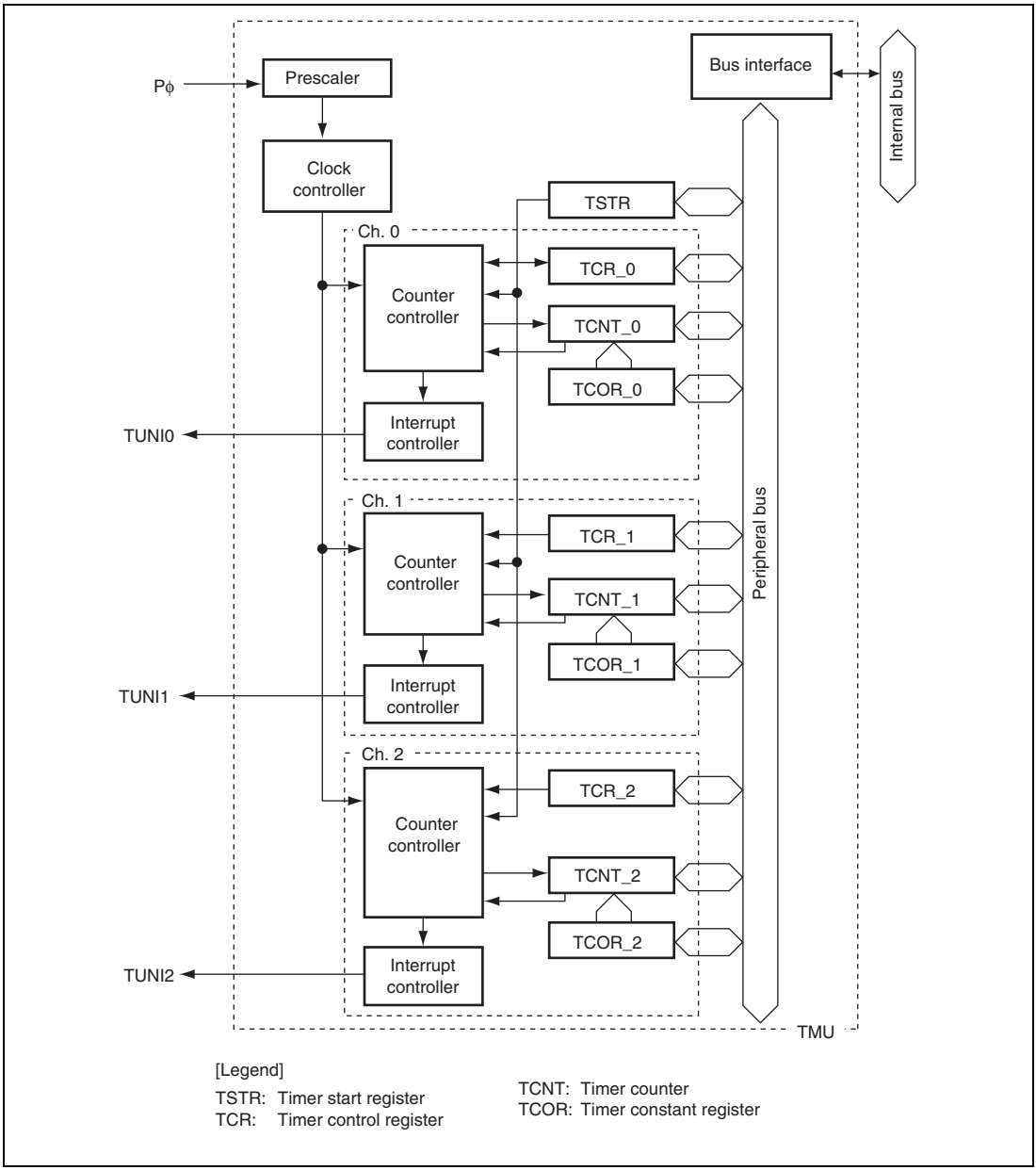


Figure 20.1 Block Diagram of TMU

20.2 Register Descriptions

Tables 20.1 and 20.2 show the TMU0 and TMU1 register configuration, respectively. Table 20.3 shows the register states in each operating mode. In the following descriptions, TCOR for unit 1 and channel 0 is noted as TCOR1_0.

Table 20.1 TMU0 Register Configuration

Register	Abbreviation	R/W	Address	Access Size
Timer start register0	TSTR	R/W	H'FFD80004	8
Timer constant register0_0	TCOR0_0	R/W	H'FFD80008	32
Timer counter0_0	TCNT0_0	R/W	H'FFD8000C	32
Timer control register0_0	TCR0_0	R/W	H'FFD80010	16
Timer constant register0_1	TCOR0_1	R/W	H'FFD80014	32
Timer counter0_1	TCNT0_1	R/W	H'FFD80018	32
Timer control register0_1	TCR0_1	R/W	H'FFD8001C	16
Timer constant register0_2	TCOR0_2	R/W	H'FFD80020	32
Timer counter0_2	TCNT0_2	R/W	H'FFD80024	32
Timer control register0_2	TCR0_2	R/W	H'FFD80028	16

Table 20.2 TMU1 Register Configuration

Register	Abbreviation	R/W	Address	Access Size
Timer start register1	TSTR1	R/W	H'FFD90004	8
Timer constant register1_0	TCOR1_0	R/W	H'FFD90008	32
Timer counter1_0	TCNT1_0	R/W	H'FFD9000C	32
Timer control register1_0	TCR1_0	R/W	H'FFD90010	16
Timer constant register1_1	TCOR1_1	R/W	H'FFD90014	32
Timer counter1_1	TCNT1_1	R/W	H'FFD90018	32
Timer control register1_1	TCR1_1	R/W	H'FFD9001C	16
Timer constant register1_2	TCOR1_2	R/W	H'FFD90020	32
Timer counter1_2	TCNT1_2	R/W	H'FFD90024	32
Timer control register1_2	TCR1_2	R/W	H'FFD90028	16

Table 20.3 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
TSTR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCOR0_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCNT0_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCR0_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCOR0_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCNT0_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCR0_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCOR0_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCNT0_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCR0_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TSTR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCOR1_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCNT1_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCR1_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCOR1_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCNT1_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCR1_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCOR1_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCNT1_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TCR1_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

20.2.1 Timer Start Register 0,1 (TSTR0, TSTR1)

TSTR are 8-bit readable/writable registers that selects whether to operate or halt the timer counters (TCNT).

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	STR2	STR1	STR0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	STR2	0	R/W	Counter Start 2 Selects whether to operate or halt timer counter 2 (TCNT_2). 0: TCNT_2 count halted 1: TCNT_2 counts
1	STR1	0	R/W	Counter Start 1 Selects whether to operate or halt timer counter 1 (TCNT_1). 0: TCNT_1 count halted 1: TCNT_1 counts
0	STR0	0	R/W	Counter Start 0 Selects whether to operate or halt timer counter 0 (TCNT_0). 0: TCNT_0 count halted 1: TCNT_0 counts

20.2.2 Timer Control Registers 0, 1 (TCR0, TCR1)

TCR are 16-bit readable/writable registers that control the timer counters (TCNT) and interrupts.

TCR control the issuance of interrupts when the flag indicating timer counter (TCNT) underflow has been set to 1, and also carry out counter clock selection.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	UNF	—	—	UNIE	—	—	TPSC[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/(W)*	R	R	R/W	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	UNF	0	R/(W)*	Underflow Flag Status flag that indicates occurrence of a TCNT underflow. 0: TCNT has not underflowed [Clearing condition] 0 is written to UNF 1: TCNT has underflowed [Setting condition] TCNT underflows
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	UNIE	0	R/W	Underflow Interrupt Control Controls enabling of interrupt generation when the status flag (UNF) indicating TCNT underflow has been set to 1. 0: Interrupt due to UNF (TUNI) is disabled 1: Interrupt due to UNF (TUNI) is enabled
4, 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	TPSC[2:0]	000	R/W	Timer Prescaler 2, 1, and 0 Select the TCNT count clock. 000: Count on Pϕ/4 001: Count on Pϕ/16 010: Count on Pϕ/64 011: Count on Pϕ/256 100: Count on Pϕ/1024 Others: Setting prohibited

Note: * Only 0 can be written to clear the flag.

20.2.3 Timer Constant Registers 0, 1 (TCOR0, TCOR1)

TCOR are 32-bit readable/writable registers that specify the value to be set in TCNT when TCNT underflows.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCOR															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCOR															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.2.4 Timer Counters 0, 1 (TCNT0, TCNT 1)

TCNT count down upon input of a clock. The clock input is selected using bits TPSC[1:0] in TCR.

When a TCNT countdown results in an underflow (H'00000000 → H'FFFFFFFF), the underflow flag (UNF) in TCR that is corresponding to the channel is set. The TCOR value is simultaneously set in TCNT itself and the countdown continues from that value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNT															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNT															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

20.3 Operation

Each channel has a 32-bit TCNT and a 32-bit TCOR. TCNT counts down. The auto-reload function enables cyclic counting.

20.3.1 Counter Operation

When bits STR[2:0] in TSTR are set to 1, the corresponding TCNT starts counting. When TCNT underflows, the UNF flag in the corresponding TCR is set. At this moment, if the UNIE bit in TCR is 1, an interrupt request is sent to the CPU. Also, the value in TCOR is copied to TCNT and the count down operation continues.

Count Operation Setting Procedure: An example procedure for setting the count operation is shown in Figure 20.2.

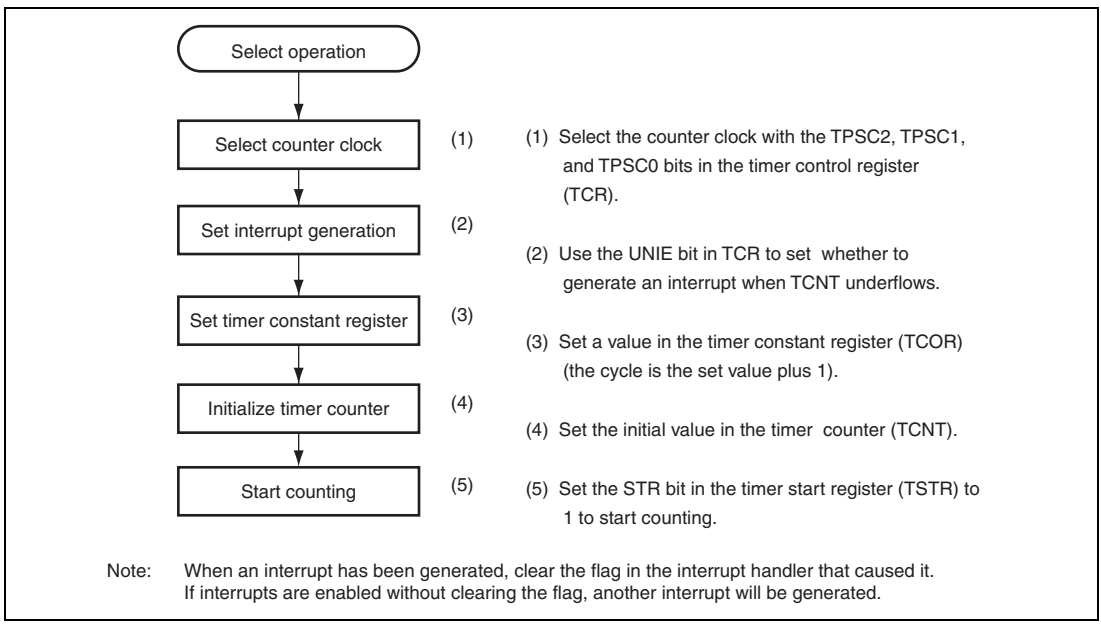


Figure 20.2 Setting Count Operation

Auto-Reload Count Operation: Figure 20.3 shows the TCNT auto-reload operation.

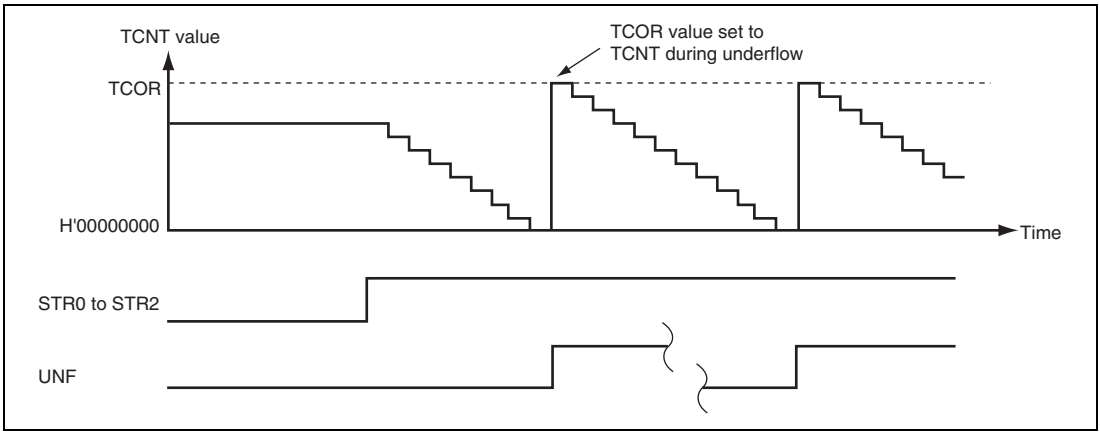


Figure 20.3 Auto-Reload Count Operation

TCNT Count Timing: Setting the bits TPSC[2:0] in TCR allows to select one of the five internal clocks (Pϕ/4, Pϕ/16, Pϕ/64, Pϕ/256, and Pϕ/1024) that are generated by dividing the peripheral module clock. Figure 20.4 shows the timing.

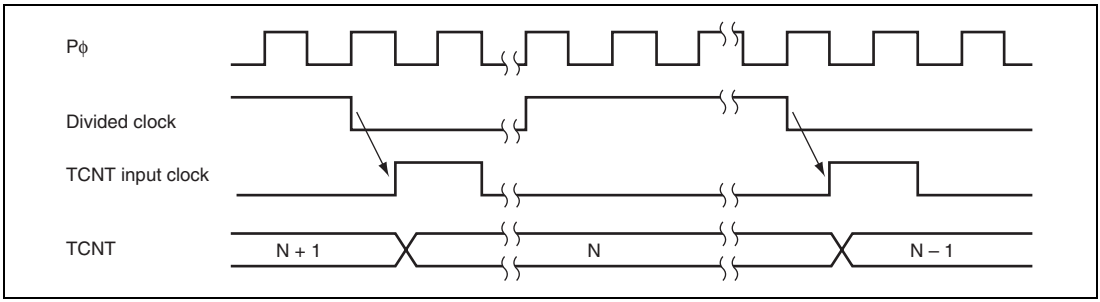


Figure 20.4 Count Timing when Internal Clock is Operating

20.4 Interrupts

There is one source of TMU interrupts: underflow interrupts (TUNI).

20.4.1 Status Flag Set Timing

The UNF bit is set to 1 when TCNT underflows. Figure 20.5 shows the timing.

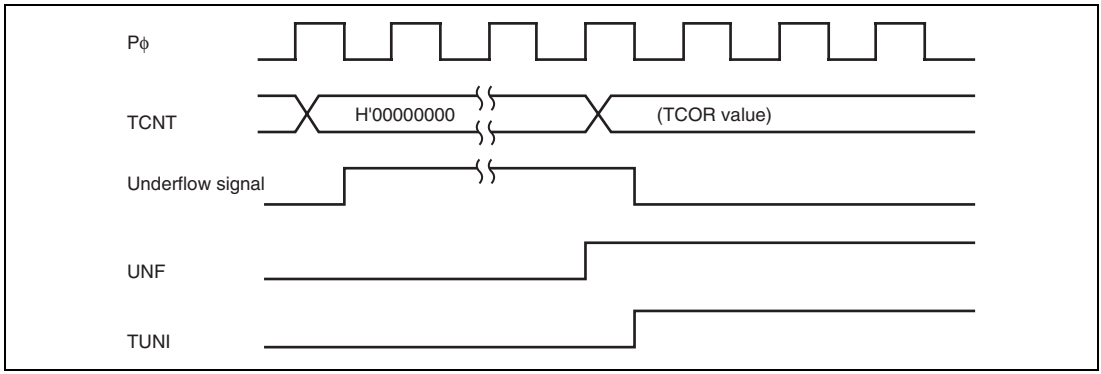


Figure 20.5 UNF Set Timing

20.4.2 Status Flag Clear Timing

The status flag can be cleared by writing 0 from the CPU. Figure 20.6 shows the timing.

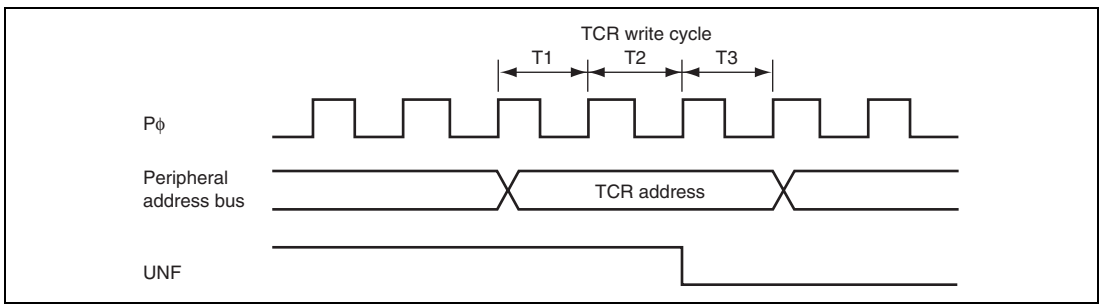


Figure 20.6 Status Flag Clear Timing

20.4.3 Interrupt Sources and Priorities

The TMU generates underflow interrupts for each channel. When the interrupt request flag and interrupt enable bit are both set to 1, the interrupt is requested. A specific code is set in the interrupt event register (INTEVT) for this interrupt and interrupt processing must be executed according to the code.

The priorities between channels are changable by the interrupt controller. For details, see section 5, Exception Handling, and section 13, Interrupt Controller (INTC). Table 20.4 lists TMU interrupt sources.

Table 20.4 TMU Interrupt Sources

Channel	Interrupt Source	Description	Priority
0	TUNI0	Underflow interrupt 0	High
1	TUNI1	Underflow interrupt 1	↕
2	TUNI2	Underflow interrupt 2	Low

20.5 Usage Notes

20.5.1 Writing to Registers

Synchronization processing is not performed for timer counting during register writes. When writing to registers, be sure to clear the start bits (STR2 to STR0) of the channel in TSTR and halt the timer counting.

20.5.2 Reading Registers

Synchronization processing is performed for timer counting during register reads. When the timer counting and register read are performed simultaneously, the register value stored before the TCNT countdown is read through the synchronization processing.

Section 21 16-Bit Timer Pulse Unit (TPU)

This LSI has an on-chip 16-bit timer pulse unit (TPU), which consists of four 16-bit timer channels.

21.1 Features

- Various timer general registers

  TPU has a total of 16 timer general registers provided with four registers (TPU_TGRA to TPU_TGRD) for each channel. TPU_TGRA enables an output compare setting. TPU_TGRB, TPU_TGRC, and TPU_TGRD in each channel can be used as the timer counter clear registers. TPU_TGRC and TPU_TGRD can be used as the buffer registers.
- Counter clock can be selected in four kinds of clocks for channel 0 and 1, in five kind of clocks for channel 2 and 3.
- The following operation can be set for each channel:

  Counter clear operation: Counter clearing possible by compare match
- Buffer operation settable for each channel

  Automatic rewriting of output compare register possible
- One interrupt request

  Enabling or disabling the compare match/overflow interrupt request can be set independently for each interrupt source.
- The following output can be made from every channel.

  Waveform output at compare match: Selection of 0, 1, or toggle output

  PWM mode: Any PWM output duty cycle can be set

Table 21.1 describes the TPU functions.

Table 21.1 TPU Functions

Item		TPU: Channel 0	TPU: Channel 1	TPU: Channel 2	TPU: Channel 3
Count clock		Bϕ/1 Bϕ/4 Bϕ/16 Bϕ/64	Bϕ/1 Bϕ/4 Bϕ/16 Bϕ/64	Bϕ/1 Bϕ/4 Bϕ/16 Bϕ/64 TPUTI2	Bϕ/1 Bϕ/4 Bϕ/16 Bϕ/64 TPUTI3
General register		TPU_TGR0A TPU_TGR0B	TPU_TGR1A TPU_TGR1B	TPU_TGR2A TPU_TGR2B	TPU_TGR3A TPU_TGR3B
General register/ Buffer register		TPU_TGR0C TPU_TGR0D	TPU_TGR1C TPU_TGR1D	TPU_TGR2C TPU_TGR2D	TPU_TGR3C TPU_TGR3D
Output pin		TPUTO	TPUTO1	TPUTO2	TPUTO3
Counter clear function		TPU_TGR compare match	TPU_TGR compare match	TPU_TGR compare match	TPU_TGR compare match
Compare match output	0 output	O	O	O	O
	1 output	O	O	O	O
	Toggle output	O	O	O	O
PWM mode		O	O	O	O
Buffer mode		O	O	O	O
Interrupt request		5 sources • Compare match • Overflow	5 sources • Compare match • Overflow	5 sources • Compare match • Overflow	5 sources • Compare match • Overflow

21.2 Block Diagram

A block diagram of the TPU is shown in Figure 21.1.

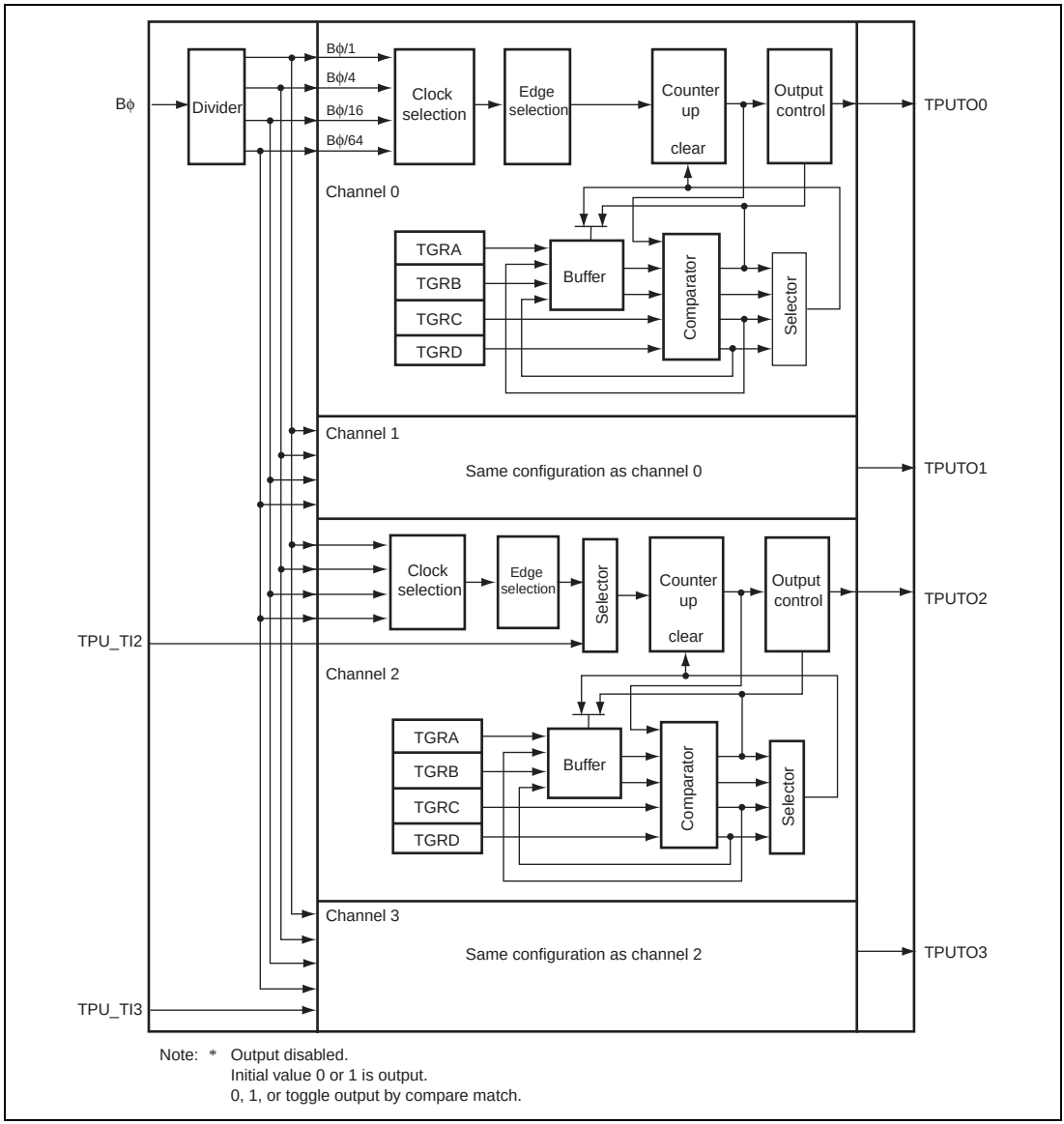


Figure 21.1 TPU Block Diagram

21.3 Input/Output Pin

Table 21.2 shows the pin configuration of the TPU.

Table 21.2 Pin Configuration

Channel	Pin Name	Function	I/O	Description
0	TPUTO0	TPU output compare match 0	Output	TPU_TGR0A output compare output/ PWM output pin
1	TPUTO1	TPU output compare match 1	Output	TPU_TGR1A output compare output/ PWM output pin
2	TPUTO2	TPU output compare match 2	Output	TPU_TGR2A output compare output/ PWM output pin
	TPUTI2	Clock input 2	Input	External clock input for channel 2
3	TPUTO3	TPU output compare match 3	Output	TPU_TGR3A output compare output/ PWM output pin
	TPUTI3	Clock input 3	Input	External clock input for channel 3

21.4 Register Descriptions

Table 21.3 shows the TPU register configuration. Table 21.4 shows the register states in each operating mode.

Table 21.3 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Timer start register	TPU_TSTR	R/W	H'A4C9 0000	16
Timer control register 0	TPU_TCR0	R/W	H'A4C9 0010	16
Timer mode register 0	TPU_TMDR0	R/W	H'A4C9 0014	16
Timer I/O control register 0	TPU_TIOR0	R/W	H'A4C9 0018	16
Timer interrupt enable register 0	TPU_TIER0	R/W	H'A4C9 001C	16
Timer status register 0	TPU_TSR0	R/W	H'A4C9 0020	16
Timer counter 0	TPU_TCNT0	R/W	H'A4C9 0024	16
Timer general register 0A	TPU_TGR0A	R/W	H'A4C9 0028	16
Timer control register 0B	TPU_TGR0B	R/W	H'A4C9 002C	16
Timer control register 0C	TPU_TGR0C	R/W	H'A4C9 0030	16
Timer control register 0D	TPU_TGR0D	R/W	H'A4C9 0034	16
Timer control register 1	TPU_TCR1	R/W	H'A4C9 0050	16

Register Name	Abbreviation	R/W	Address	Access Size
Timer mode register 1	TPU_TMDR1	R/W	H'A4C9 0054	16
Timer I/O control register 1	TPU_TIOR1	R/W	H'A4C9 0058	16
Timer interrupt enable register 1	TPU_TIER1	R/W	H'A4C9 005C	16
Timer status register 1	TPU_TSR1	R/W	H'A4C9 0060	16
Timer counter 1	TPU_TCNT1	R/W	H'A4C9 0064	16
Timer general register 1A	TPU_TGR1A	R/W	H'A4C9 0068	16
Timer control register 1B	TPU_TGR1B	R/W	H'A4C9 006C	16
Timer control register 1C	TPU_TGR1C	R/W	H'A4C9 0070	16
Timer control register 1D	TPU_TGR1D	R/W	H'A4C9 0074	16
Timer control register 2	TPU_TCR2	R/W	H'A4C9 0090	16
Timer mode register 2	TPU_TMDR2	R/W	H'A4C9 0094	16
Timer I/O control register 2	TPU_TIOR2	R/W	H'A4C9 0098	16
Timer interrupt enable register 2	TPU_TIER2	R/W	H'A4C9 009C	16
Timer status register 2	TPU_TSR2	R/W	H'A4C9 00A0	16
Timer counter 2	TPU_TCNT2	R/W	H'A4C9 00A4	16
Timer general register 2A	TPU_TGR2A	R/W	H'A4C9 00A8	16
Timer control register 2B	TPU_TGR2B	R/W	H'A4C9 00AC	16
Timer control register 2C	TPU_TGR2C	R/W	H'A4C9 00B0	16
Timer control register 2D	TPU_TGR2D	R/W	H'A4C9 00B4	16
Timer control register 3	TPU_TCR3	R/W	H'A4C9 00D0	16
Timer mode register 3	TPU_TMDR3	R/W	H'A4C9 00D4	16
Timer I/O control register 3	TPU_TIOR3	R/W	H'A4C9 00D8	16
Timer interrupt enable register 3	TPU_TIER3	R/W	H'A4C9 00DC	16
Timer status register 3	TPU_TSR3	R/W	H'A4C9 00E0	16
Timer counter 3	TPU_TCNT3	R/W	H'A4C9 00E4	16
Timer general register 3A	TPU_TGR3A	R/W	H'A4C9 00E8	16
Timer control register 3B	TPU_TGR3B	R/W	H'A4C9 00EC	16
Timer control register 3C	TPU_TGR3C	R/W	H'A4C9 00F0	16
Timer control register 3D	TPU_TGR3D	R/W	H'A4C9 00F4	16

Table 21.4 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
TPU_TSTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TMDR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIOR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIER0	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TSR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCNT0	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR0A	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR0B	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR0C	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR0D	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TMDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIOR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIER1	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TSR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCNT1	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR1A	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR1B	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR1C	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR1D	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TMDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIOR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIER2	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TSR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCNT2	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR2A	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR2B	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR2C	Initialized	Initialized	Retained	Retained	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
TPU_TGR2D	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TMDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIOR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TIER3	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TSR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TCNT3	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR3A	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR3B	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR3C	Initialized	Initialized	Retained	Retained	Initialized	Retained
TPU_TGR3D	Initialized	Initialized	Retained	Retained	Initialized	Retained

21.4.1 Timer Control Register (TPU_TCR)

TPU_TCR controls the TPU_TCNT for each channel. The TPU has one TPU_TCR register for each channel. TPU_TCR is initialized to H'0000 at a reset.

TPU_TCR register settings should be made only while TPU_TCNT operation is stopped.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CCLR[2:0]			CKEG[1:0]		TPSC[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	CCLR[2:0]	000	R/W	Counter Clear Select the TPU_TCNT clearing source. 000: TPU_TCNT clearing disabled 001: TPU_TCNT cleared by TPU_TGRA compare match 010: TPU_TCNT cleared by TPU_TGRB compare match 011: Setting prohibited 100: TPU_TCNT clearing disabled 101: TPU_TCNT cleared by TPU_TGRC compare match 110: TPU_TCNT cleared by TPU_TGRD compare match 111: Setting prohibited
4, 3	CKEG[1:0]	00	R/W	Clock Edge Select the input clock edge. When the internal clock is counted using both edges, the input clock period is halved (e.g. Bϕ/4 both edges = Bϕ/2 rising edge). When the phase coefficient mode is set, this setting is ignored. 00: Count at rising edge 01: Count at falling edge 1X: Count at both edges* [Legend] X: Don't care Note: * If the input clock is Bϕ/1, external clock (channel 2, 3) no operation is performed.
2 to 0	TPSC[2:0]	000	R/W	Timer Prescaler Select the TPU_TCNT count clock. The clock source can be selected independently for each channel. Table 21.5 shows the clock sources that can be set for each channel. For more information on count clock selection, see Table 21.6.

Table 21.5 TPU Clock Sources

	Internal Clock				External Clock	
Channel	**Bϕ/1**	**Bϕ/4**	**Bϕ/16**	**Bϕ/64**	**TPUTI2**	**TPUTI3**
0	O	O	O	O		
1	O	O	O	O		
2	O	O	O	O	O	
3	O	O	O	O		O

[Legend]
O: Setting

Table 21.6 Count Clock Selection by the TPSC[2:0] Bits

Channel	TPSC[2]	TPSC[1]	TPSC[0]	Description
0, 1	0	0	0	Internal clock: counts on Bϕ/1
			1	Internal clock: counts on Bϕ/4
		1	0	Internal clock: counts on Bϕ/16
			1	Internal clock: counts on Bϕ/64
	1	*	*	Setting prohibited
2	0	0	0	Internal clock: counts on Bϕ/1
			1	Internal clock: counts on Bϕ/4
		1	0	Internal clock: counts on Bϕ/16
			1	Internal clock: counts on Bϕ/64
	1	0	0	External clock: counts on TPUTI2 input
			1	Setting prohibited
		1	*	
3	0	0	0	Internal clock: counts on Bϕ/1
			1	Internal clock: counts on Bϕ/4
		1	0	Internal clock: counts on Bϕ/16
			1	Internal clock: counts on Bϕ/64
	1	0	0	External clock: counts on TPUTI3 input
			1	Setting prohibited
		1	*	

[Legend]
*: Don't care.

21.4.2 Timer Mode Register (TPU_TMDR)

TPU_TMDR specifies the operation mode for each channel. The TPU has one TPU_TMDR register for each channel. TPU_TMDR is initialized to H'0000 at a reset.

TPU_TMDR register settings should be made only while TPU_TCNT operation is stopped.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	BFWT	BFB	BFA	—	MD[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
6	BFWT	0	R/W	Buffer Write Timing Specifies TPU_TGRA and TPU_TGRB update timing when TPU_TGRC and TPU_TGRD are used as a compare match buffer. When TPU_TGRC and TPU_TGRD are not used as a compare match buffer register, this bit does not function. 0: TPU_TGRA and TPU_TGRB are rewritten at compare match of each register. 1: TPU_TGRA and TPU_TGRB are rewritten at counter clearing.
5	BFB	0	R/W	Buffer Operation B Specifies whether TPU_TGRB is used in normal operation, or TPU_TGRB and TPU_TGRD are used in combination for buffer operation. 0: TPU_TGRB normal operation 1: TPU_TGRB and TPU_TGRD used for buffer operation
4	BFA	0	R/W	Buffer Operation A Specifies whether TPU_TGRA is used in normal operation, or TPU_TGRA and TPU_TGRC are used in combination for buffer operation. 0: TPU_TGRA normal operation 1: TPU_TGRA and TPU_TGRC used for buffer operation
3	—	0	R	Reserved This bit is always read as 0 and cannot be modified.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	MD[2:0]	000	R/W	Timer Operating Mode Set the timer-operating mode. 000: Normal operating 001: Setting prohibited 010: PWM mode 011: Setting prohibited 100: Setting prohibited 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited

21.4.3 Timer I/O Control Register (TPU_TIOR)

TPU_TIOR controls the TPUTO0 through TPUTO3 pins. Each channel has one TPU_TIOR register. TPU_TIOR is initialized to H'0000 at a reset.

TPU_TIOR register setting should be made only while TPU_TCNT operation is stopped. Note that the setting of TPU_TMDR may affect TPU_TIOR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	IOA[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
2 to 0	IOA[2:0]	000	R/W	I/O Control Bits IOA2 to IOA0 specify the functions of TPU_TGRA and the TPUTO pin. For details, see Table 21.7.

Table 21.7 Settings for Bits IOA[2:0], Initial States of Pin TPUTO, and Results of Matching with TPU_TGRA

Channel	IOA[2]	IOA[1]	IOA[0]	Description	
0 to 3	0	0	0	Always output 0 (initial value)	
			1	Initial output of the TPUTO pin is 0	Output 0 on compare match with TPU_TGRA*
		1	0		Output 1 on compare match with TPU_TGRA
			1		Toggle output on compare match with TPU_TGRA*
	1	0	0	Always output 1	
			1	Initial output of the TPUTO pin is 1	Output 0 on compare match with TPU_TGRA*
		1	0		Output 1 on compare match with TPU_TGRA
			1		Toggle output on compare match with TPU_TGRA*

Note: * Do not use this setting in PWM mode.

21.4.4 Timer Interrupt Enable Register (TPU_TIER)

TPU_TIER is used to enable and disable interrupt requests for each channel. The TPU has one TPU_TIER register for each channel. TPU_TIER is initialized to H'0000 by a reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TC1EV	TG1ED	TG1EC	TG1EB	TG1EA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.

Bit	Bit Name	Initial Value	R/W	Description
4	TC1EV	0	R/W	Overflow Interrupt Enable When the TCFV flag in TPU_TSR is set to 1 (a TCNT overflow has occurred), this bit enables or disables interrupt requests corresponding to the state of the TCFV flag. 0: Interrupt requests by TCFV flag disabled 1: Interrupt requests by TCFV flag enabled
3	TG1ED	0	R/W	TPU_TGR Interrupt Enable D When the TGFD bit in TPU_TSR is set to 1 (a compare match between TPU_TCNT and TPU_TGRD has occurred), this bit enables or disables interrupt requests corresponding to the state of the TGFD flag. 0: Interrupt requests by TGFD flag disabled 1: Interrupt requests by TGFD flag enabled
2	TG1EC	0	R/W	TPU_TGR Interrupt Enable C When the TGFC bit in TPU_TSR is set to 1 (a compare match between TPU_TCNT and TPU_TGRC has occurred), this bit enables or disables interrupt requests corresponding to the state of the TGFC flag. 0: Interrupt requests by TGFC flag disabled 1: Interrupt requests by TGFC flag enabled
1	TG1EB	0	R/W	TPU_TGR Interrupt Enable B When the TGFB bit in TPU_TSR is set to 1 (a compare match between TPU_TCNT and TPU_TGRB has occurred), this bit enables or disables interrupt requests corresponding to the state of the TGFB flag. 0: Interrupt requests by TGFB flag disabled 1: Interrupt requests by TGFB flag enabled
0	TG1EA	0	R/W	TPU_TGR Interrupt Enable A When the TGFA bit in TPU_TSR is set to 1 (a compare match between TPU_TCNT and TPU_TGRA has occurred), this bit enables or disables interrupt requests corresponding to the state of the TGFA flag. 0: Interrupt requests by TGFA flag disabled 1: Interrupt requests by TGFA flag enabled

21.4.5 Timer Status Registers (TPU_TSR)

TPU_TSR displays information on the state of each channel. The TPU has one TPU_TSR register for each channel. TPU_TSR is initialized to H'0000 by a reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCFV	TGFD	TGFC	TGFB	TGFA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
4	TCFV	0	R/(W)*	Overflow Flag Status flag indicating overflow of TPU_TCNT [Clearing condition] Writing 0 to the TCFV bit after reading the bit when TCFV = 1 [Setting condition] Overflow of the value in TPU_TCNT (i.e. the value changing from H'FFFF to H'0000)
3	TGFD	0	R/(W)*	Compare Flag D Status flag indicating a match with TPU_TGRD [Clearing condition] Writing 0 to the TGFD bit after reading the bit when TGFD = 1 [Setting condition] A match between the values in TPU_TCNT and TPU_TGRD
2	TGFC	0	R/(W)*	Compare Flag C Status flag indicating a match with TPU_TGRC [Clearing condition] Writing 0 to the TGFC bit after reading the bit when TGFC = 1 [Setting condition] A match between the values in TPU_TCNT and TPU_TGRC

Bit	Bit Name	Initial Value	R/W	Description
1	TGFB	0	R/(W)*	Compare Flag B Status flag indicating a match with TPU_TGRB [Clearing condition] Writing 0 to the TGFB bit after reading the bit when TGFB = 1 [Setting condition] A match between the values in TPU_TCNT and TPU_TGRB
0	TGFA	0	R/(W)*	Compare Flag A Status flag indicating a match with TPU_TGRA [Clearing condition] Writing 0 to the TGFA bit after reading the bit when TGFA = 1 [Setting condition] A match between the values in TPU_TCNT and TPU_TGRA

Note: * Writing a 0 is the only way to clear this flag.

21.4.6 Timer Counter (TPU_TCNT)

TPU_TCNT indicates a 16-bit counter. The TPU has one TPU_TCNT per channel.

TPU_TCNT is initialized to H'0000 by a reset.

21.4.7 Timer General Register (TPU_TGR)

TPU_TGR indicates a 16-bit general register. Four general registers (TPU_TGRA, TPU_TGRB, TPU_TGRC, and TPU_TGRD) are provided for each channel. TPU_TGRC and TPU_TGRD can be designated for operation as buffer registers*. TPU_TGR is initialized to H'FFFF by a reset.

Note: * The combination of TPU_TGR and buffer register are TPU_TGRA—TPU_TGRC and TPU_TGRB—TPU_TGRD.

21.4.8 Timer Start Register (TPU_TSTR)

TPU_TSTR starts and stops TCNT operation for channels 0 to 3.

TPU_TSTR is initialized to H'0000 by a reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CST3	CST2	CST1	CST0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
3	CST3	0	R/W	Counter Start
2	CST2	0	R/W	These bits select either start or stop of TPU_TCNT
1	CST1	0	R/W	0: Stops TPU_TCNTm count operation
0	CST0	0	R/W	1: TPU_TCNTm count operation [Legend] m = 3 to 0

21.5 Operation

21.5.1 Overview

Operation overview for each mode is as follows.

(1) Ordinary Operation

Each channel is provided with TPU_TCNT and TPU_TGR registers. TPU_TCNT performs up-counting, and is also capable of free-running operation and periodic counting.

(2) Buffer Operation

When a compare match occurs, the buffer register value in the corresponding channel is transferred to TPU_TGR. Updating timing to rewrite from buffer registers can be selected either when a compare match occurs or when the counter is cleared.

(3) PWM Mode

In PWM mode, PWM waveform is output. The output level can be set by TPU_TIOR. PWM waveform, whose duty is in the range of 0 to 100%, can be output by the settings of TPU_TGRA and TPU_TGRB.

21.5.2 Basic Functions

(1) Counter Operation

When the bits CST[3:0] in TPU_TSTR are set to 1, the TPU_TCNT for the corresponding channel starts counting. TPU_TCNT can operate as a free-running counter, periodic counter, and so on.

(a) Example of count operation setting procedure

Figure 21.2 shows an example of the count operation setting procedure.

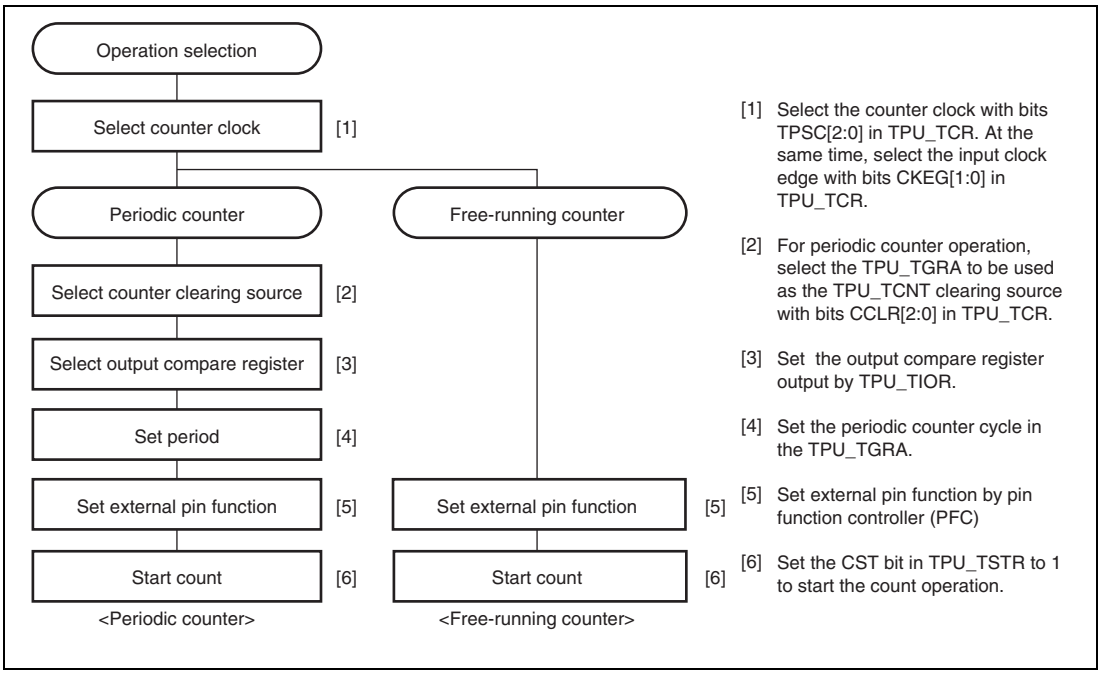


Figure 21.2 Example of Counter Operation Setting Procedure

(b) Free-running count operation and periodic count operation

Immediately after a reset, the TPU_TCNT counters are all designated as free-running counters. When the relevant bit in TPU_TSTR is set to 1, the corresponding TPU_TCNT counter starts up-count operation as a free-running counter. When TPU_TCNT has overflowed (changes from H'FFFF to H'0000), the TCFV bit in TPU_TSR is set to 1. TPU_TCNT starts counting up again from H'0000 after an overflow.

Figure 21.3 illustrates free-running counter operation.

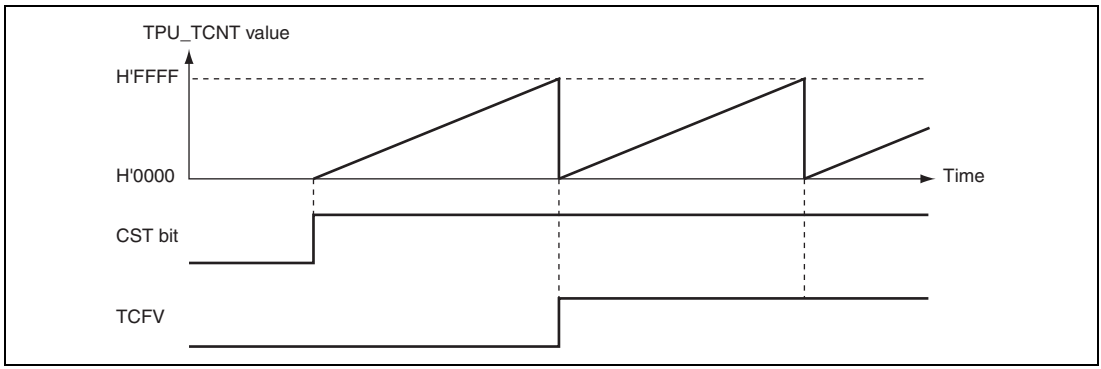


Figure 21.3 Free-Running Counter Operation

When a compare match is selected as the TPU_TCNT clearing source, the TPU_TCNT counter for the relevant channel performs periodic count operation. The TPU_TGR register for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR[2:0] in TPU_TCR. After the settings have been made, TPU_TCNT starts count-up operation as a periodic counter when the corresponding bit in TPU_TSTR is set to 1. When the count value matches the value in TPU_TGR, the TGF bit in TPU_TSR is set to 1 and TPU_TCNT is cleared to H'0000.

After a compare match, TPU_TCNT starts counting up again from H'0000.

Figure 21.4 illustrates periodic counter operation.

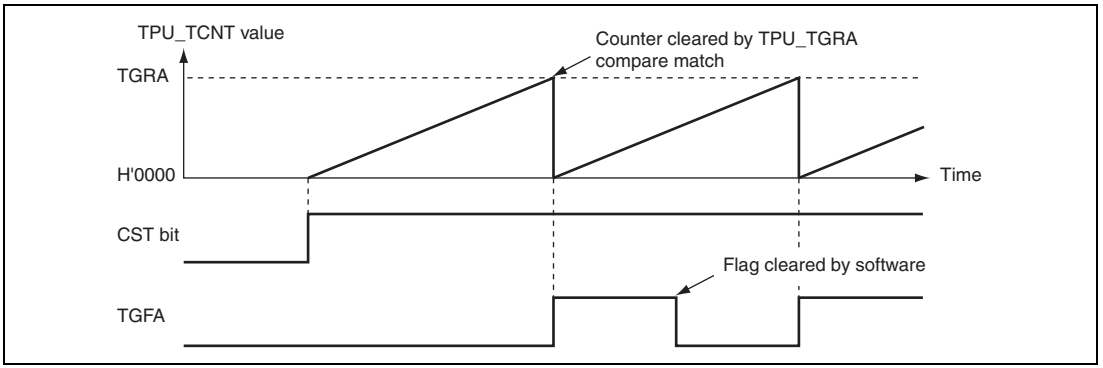


Figure 21.4 Periodic Counter Operation

(2) Waveform Output by Compare Match

The TPU can perform 0-, 1-, or toggle-output from the output pin (TPUTO pin) using a TPU_TGRA compare match.

(a) Example of setting procedure for waveform output by compare match

Figure 21.5 shows an example of the setting procedure for waveform output by a compare match.

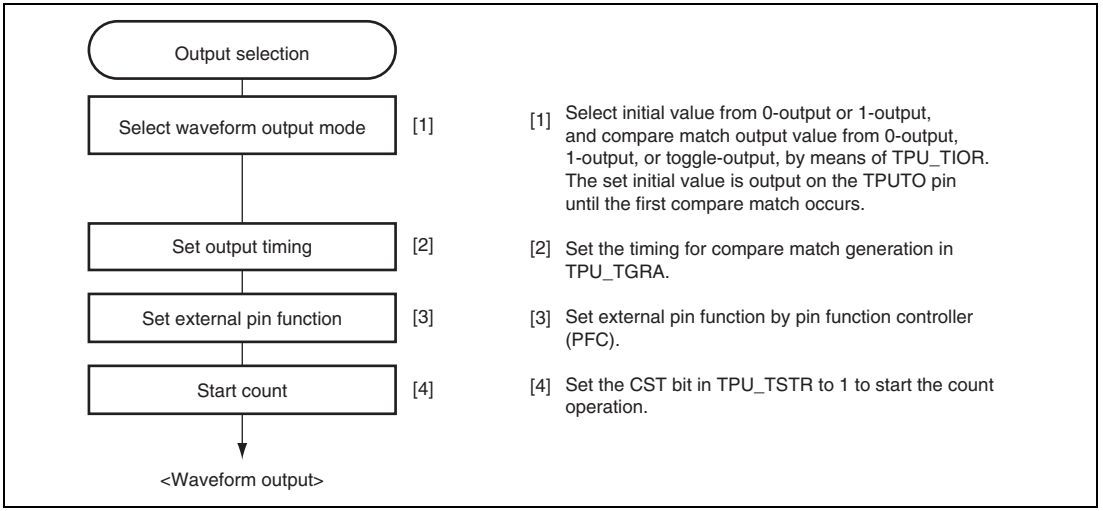


Figure 21.5 Example of Setting Procedure for Waveform Output by Compare Match

(b) Examples of waveform output operation

Figure 21.6 shows an example of 0-output and 1-output.

In this example, TPU_TCNT has been designated as a free-running counter, and settings have been made so that 1 or 0 is output by compare match A. When the set level and the pin level match, the pin level does not change.

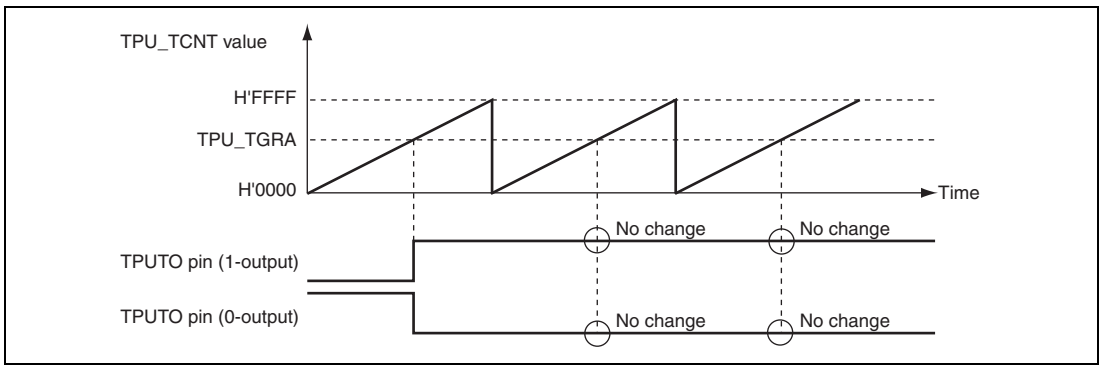


Figure 21.6 Example of 0-Output/1-Output Operation

Figure 21.7 shows an example of toggle output.

In this example, TPU_TCNT has been designated as a periodic counter (with counter clearing performed by compare match B), and settings have been made so that output is toggled by compare match A.

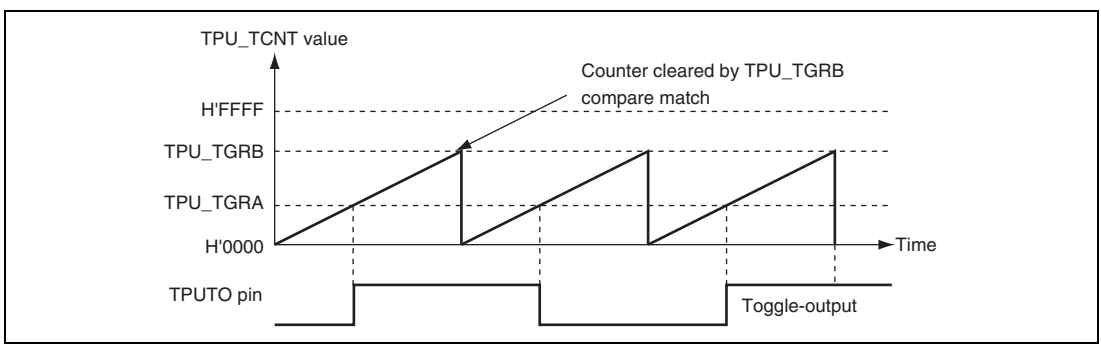


Figure 21.7 Example of Toggle Output Operation

21.5.3 Buffer Operation

TPU_TGRC and TPU_TGRD can be used as buffer registers.

Table 21.8 shows the register combinations used in buffer operation.

Table 21.8 Register Combinations in Buffer Operation

Timer General Register	Buffer Register
TPU_TGRA	TPU_TGRC
TPU_TGRB	TPU_TGRD

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register. Updating timing to rewrite from buffer registers can be selected either when compare match occurs or when the counter is cleared.

This operation is illustrated in Figure 21.8.

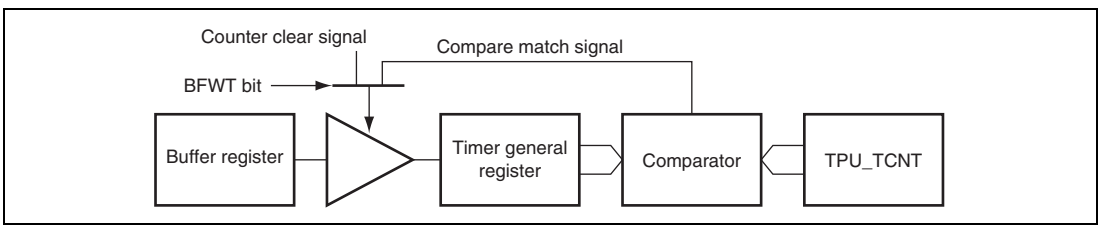


Figure 21.8 Compare Match Buffer Operation

(1) Example of Buffer Operation Setting Procedure

Figure 21.9 shows an example of the buffer operation setting procedure.

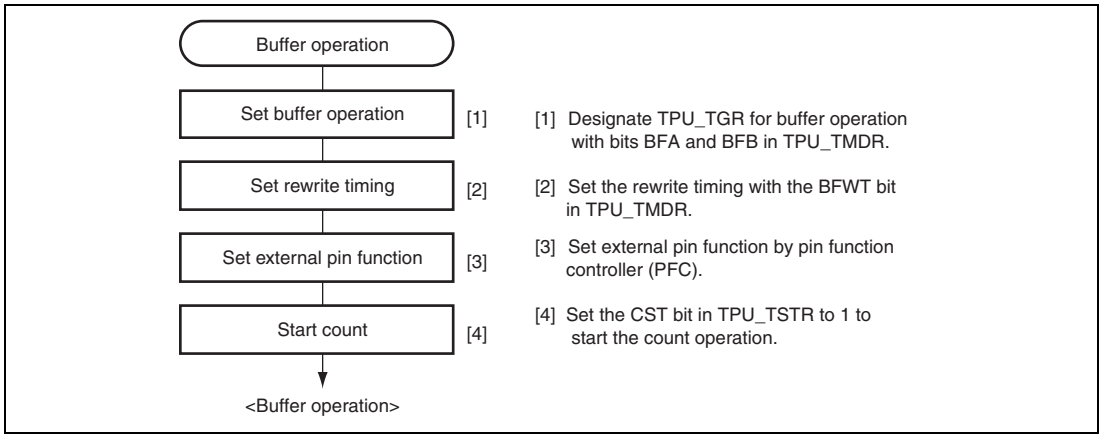


Figure 21.9 Example of Buffer Operation Setting Procedure

(2) Examples of buffer operation

Figure 21.10 shows an operation example in which PWM mode has been designated for channel 0, and buffer operation has been designated for TPU_TGRA and TPU_TGRC. The settings used in this example are TPU_TCNT cleared by compare match B, 1-output (TPUTO pin) at compare match A, initial value 0 output by counter clearing, and the rewrite timing from buffer register at counter clearing.

When compare match A occurs, the output changes. When a counter clear is generated by TPU_TGRB, the output changes and the value in buffer register TPU_TGRC is simultaneously transferred to the timer general register TPU_TGRA. This operation is repeated every time compare match A occurs.

For details on PWM modes, see section 21.5.4, PWM Modes.

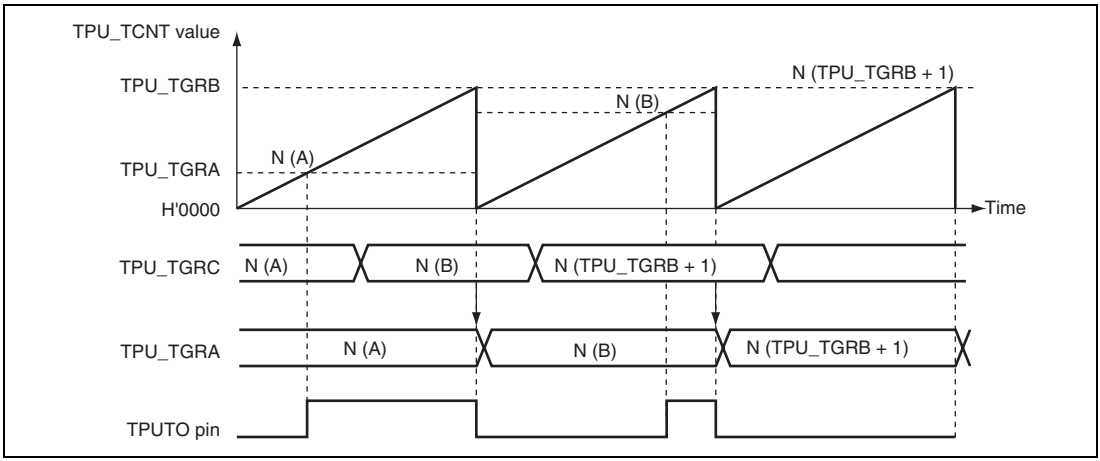


Figure 21.10 Example of Buffer Operation

21.5.4 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. 0-, or 1-output can be selected as the output level in response to compare match of each TPU_TGRA.

Designating TPU_TGRB compare match as the counter clearing source enables the cycle to be set in that register. All channels can be designated for PWM mode independently.

PWM output is generated from the TPUTO pin by using TPU_TGRA and TPU_TGRB as duty register and periodic register respectively. The initial output specified by TPU_TIOR is output to TPUTO pin by counter clearing due to periodic register compare match. Be sure to set TPU_TIOR so that the initial output level is different from the compare match output. Selecting the same level or toggle output activates no operation.

Conditions on 0% and 100% duties are shown below.

- 0% duty: When duty register (TPU_TGRA) is set to the value equal to periodic register (TPU_TGRB) + 1
- 100% duty: When duty register (TPU_TGRA) is set to 0

In PWM mode, up to four types of PMW outputs are available.

(1) Example of PWM Mode Setting Procedure

Figure 21.11 shows an example of the PWM mode setting procedure.

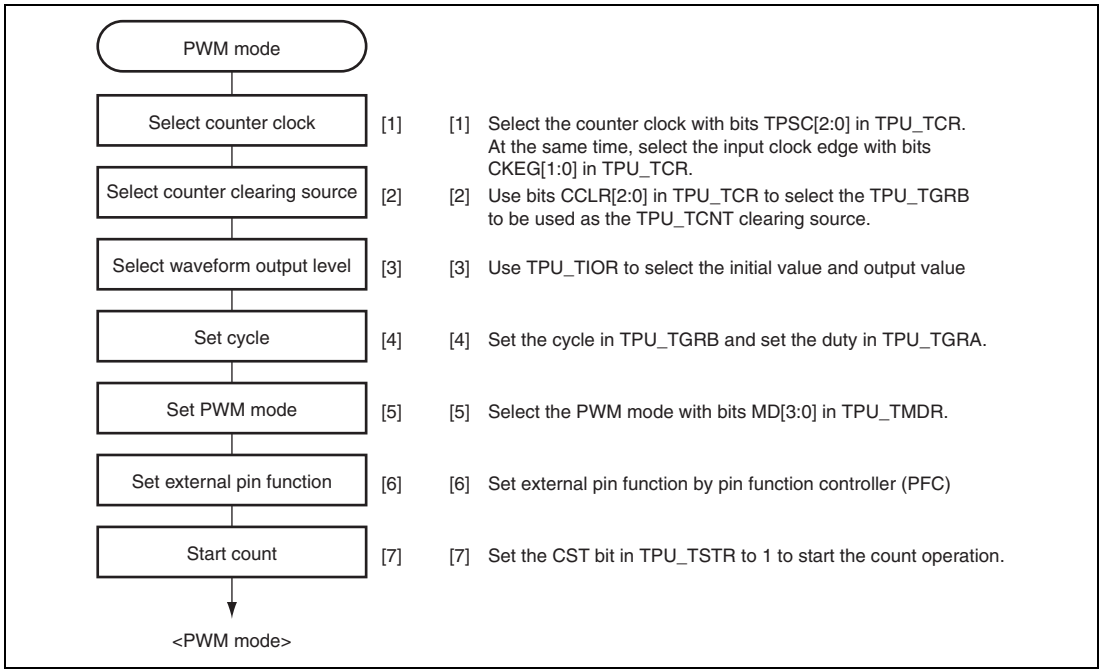


Figure 21.11 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 21.12 shows an example of PWM mode operation.

In this example, TPU_TGRB compare match is set as the TPU_TCNT clearing source, 0 is set for the TPU_TGRA initial output value, and 1 is set as the output value.

In this case, the value set in TPU_TGRB is used as the cycle, and the value set in TPU_TGRA as the duty cycle.

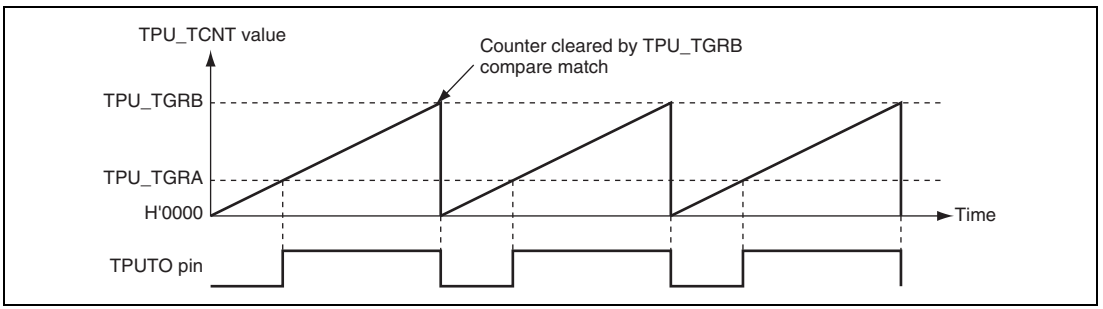


Figure 21.12 Example of PWM Mode Operation (1)

Figure 21.13 shows an example of PWM waveform output with 0% and 100% duties in PWM mode.

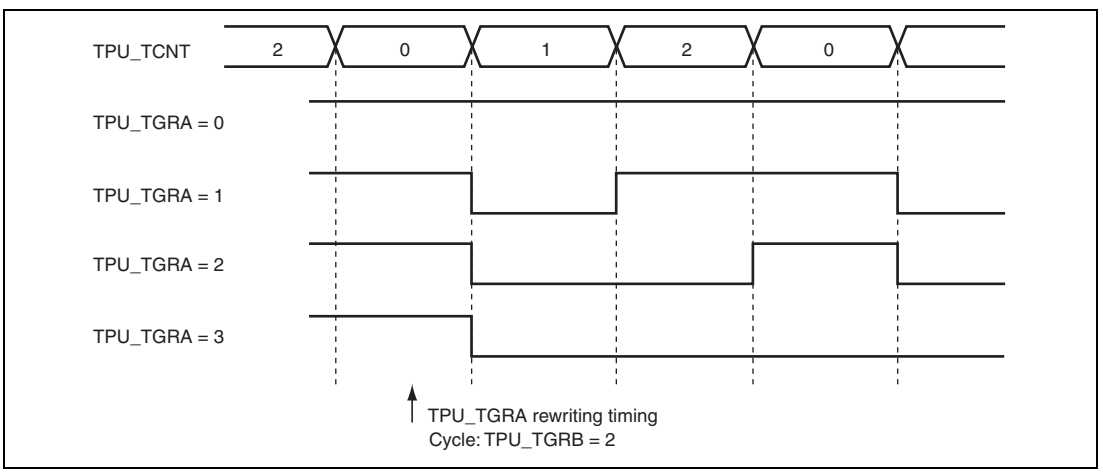


Figure 21.13 Example of PWM Mode Operation (2)

Section 22 Compare Match Timer (CMT)

This LSI includes a 32-bit compare match timer (CMT) of one channel.

22.1 Features

- 16 bits/32 bits can be selected.
- Provided with an auto-reload up counter.
- Provided with 32-bit constant registers and 32-bit up counters that can be written or read at any time.
- The CMT of this LSI can operate the counting even in R/U-standby mode.
- Allows selection among 3 counter input clocks:
  — External clock (RCLK) input: 1/8, 1/32, and 1/128
- One-shot operation and free-running operation are selectable.
- Allows selection of compare match or overflow for the interrupt source.
- Supports canceling of the standby state in R/U-standby mode.
- Module standby mode can be set.

Figure 22.1 shows a block diagram of the CMT.

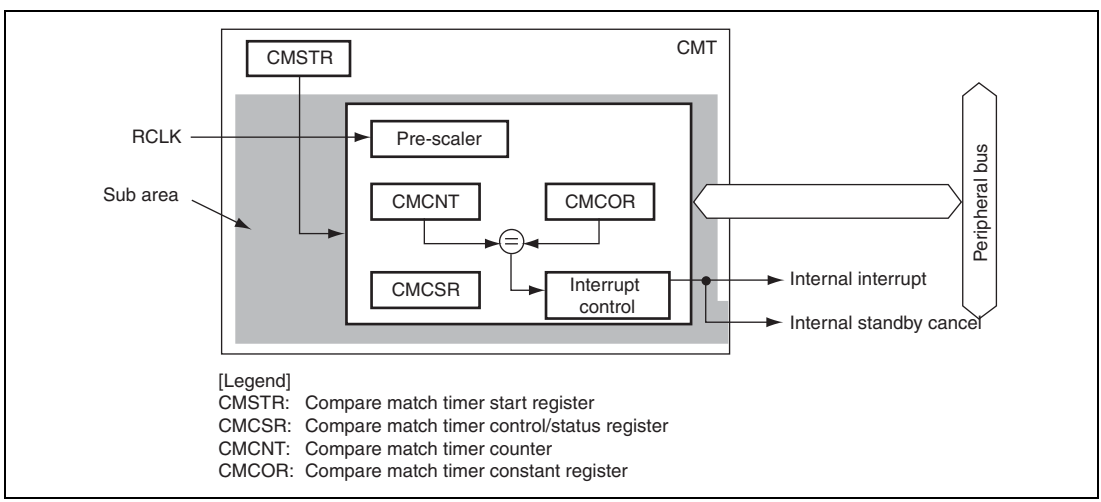


Figure 22.1 Block Diagram of CMT

22.2 Register Descriptions

Table 22.1 shows the CMT register configuration. Table 22.2 shows the register states in each operating mode.

Table 22.1 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Compare match timer start register	CMSTR	R/W	H'A44A 0000	16
Compare match timer control/status register	CMCSR	R/W	H'A44A 0060	16
Compare match timer counter	CMCNT	R/W	H'A44A 0064	32
Compare match timer constant register	CMCOR	R/W	H'A44A 0068	32

Table 22.2 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
CMSTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
CMCSR	Initialized	Initialized*	Retained	Retained	Initialized*	Retained
CMCNT	Initialized	Initialized	Retained	Retained	Initialized	Retained
CMCOR	Initialized	Initialized	Retained	Retained	Initialized	Retained

Note: * The CMF and OVF bits retain the value before a reset.

22.2.1 Compare Match Timer Start Register (CMSTR)

CMSTR is a 16-bit register that selects whether the compare match timer counter (CMCNT) is operated or halted.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	STR5	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	STR5	0	R/W	Count Start Selects whether to operate or halt the compare match timer counter (CMCNT). 0: Halts CMCNT count 1: Operates CMCNT count
4 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

22.2.2 Compare Match Timer Control/Status Register (CMCSR)

CMCSR is a 16-bit register that indicates the occurrence of compare matches, enables interrupts, and sets the counter input clocks.

Do not change bits other than the CMF and OVF bits during the compare match timer counter (CMCNT) operation. The CMF and OVF bits are not initialized. Be sure to write 0 to the CMF and OVF bits before they are used.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMF	OVF	WR FLG	—	—	—	CMS	CMM	CMTOUT_IE	—	CMR[1:0]		—	CKS[2:0]		
Initial value:	—	—	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R/(W)*	R/(W)*	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	CMF	Undefined	R/(W)*	Compare Match Flag This flag indicates whether or not values of the compare match timer counter (CMCNT) and compare match timer constant register (CMCOR) have matched. Software cannot write 1 to the bit. When one-shot is selected for the counter operation, counting resumes by clearing this bit. 0: CMCNT and CMCOR values have not matched [Clearing condition] • Write 0 to CMF 1: CMCNT and CMCOR values have matched
14	OVF	Undefined	R/(W)*	Overflow Flag This flag indicates whether or not the compare match timer counter (CMCNT) has overflowed and been cleared to 0. Software cannot write 1 to this bit. 0: CMCNT has not overflowed [Clearing condition] • Write 0 to OVF 1: CMCNT has overflowed

Bit	Bit Name	Initial Value	R/W	Description
13	WRFLG	0	R	Write State Flag CMCNT cannot be written to when this bit is 1. This bit indicates the period that the writing to CMCNT is masked for synchronization after the writing to CMCNT. Confirm that this flag is 0 when CMCNT is written to continuously.
12 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	CMS	0	R/W	Compare Match Timer Counter Size Selects whether the compare match timer counter (CMCNT) is used as a 16-bit counter or a 32-bit counter. This setting becomes the valid size for the compare match timer constant register (CMCOR). 0: Operates as a 32-bit counter 1: Operates as a 16-bit counter
8	CMM	0	R/W	Compare Match Mode Selects one-shot operation or free-running operation of the counter. 0: One-shot operation 1: Free-running operation
7	CMTOUT_IE	0	R/W	Compare Match Standby Mode Cancel Enable Enables or disables the cancel of various standby modes. 0: The cancel of various standby modes by this CMT is disabled. 1: The cancel of various standby modes by this CMT is enabled.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	CMR[1:0]	00	R/W	Compare Match Request These bits enable or disable an internal interrupt request in a compare match. 00: Disables an internal interrupt request 01: Setting prohibited 10: Enables an internal interrupt request 11: Setting prohibited Note: When various standby modes are canceled by this CMT, these bits should be set to B'10.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	CKS[2:0]	100	R/W	Clock Select These bits select the clock input to CMCNT. When the count start bit (STR5) in CMSTR is set to 1, CMCNT begins incrementing with the clock selected by these bits. 000: Setting prohibited 001: Setting prohibited 010: Setting prohibited 011: Setting prohibited 100: RCLK/8 101: RCLK/32 110: RCLK/128 111: Setting prohibited

Note: * Only 0 can be written to clear the flag.

22.2.3 Compare Match Timer Counter (CMCNT)

CMCNT is a 32-bit register that is used as an up-counter.

A counter operation is set by the compare match timer control/status register (CMCSR). Therefore, set CMCSR first, before starting a channel operation corresponding to the compare match timer start register (CMSTR). When the 16-bit counter operation is selected by the CMS bit, bits 15 to 0 of this register become valid. When the register should be written to, write the data that is added H'0000 to the upper half in a 32-bit operation. The contents of this register are initialized to H'00000000.

Counter operation is enabled when the peripheral clock is stopped for CMT. Counter operation is also enabled in a state that the core-area power supply is turned off in R/U-standby mode.

When CMCNT is going to be read during the counter operation, the value that is read out may be wrong because of an asynchronous clock. Read several times with software and compare the read-out values.

22.2.4 Compare Match Timer Constant Register (CMCOR)

CMCOR is a 32-bit register that sets the compare match period with CMCNT.

When the 16-bit counter operation is selected by the CMS bit in CMCSR, bits 15 to 0 of this register become valid. When the register should be written to, write the data that is added H'0000 to the upper half in a 32-bit operation.

An overflow is detected when CMCNT is cleared to 0 and this register is H'FFFFFFFF. The contents of this register are initialized to H'FFFFFFFF.

22.3 Operation

22.3.1 Counter Operation

The CMT starts the operation of the counter by writing B'1 to the STR5 bit in CMSTR after each register has been set. Complete all of the settings before starting the operation. Do not change the register settings other than by clearing flag bits.

The counter operates in one of two ways.

- One-Shot Operation

  One-shot operation is selected by setting the CMM bit in CMCSR to B'0. When the value in CMCNT matches the value in CMCOR, the value in CMCNT is cleared to H'00000000 and the CMF bit in CMCSR is set to B'1. Counting by CMCNT stops after it has been cleared.

  To detect an overflow interrupt, set the value in CMCOR to H'FFFFFFFF. When the value in CMCNT matches the value in CMCOR, CMCNT is cleared to H'00000000 and the CMF and OVF bits in CMCSR are set to B'1.

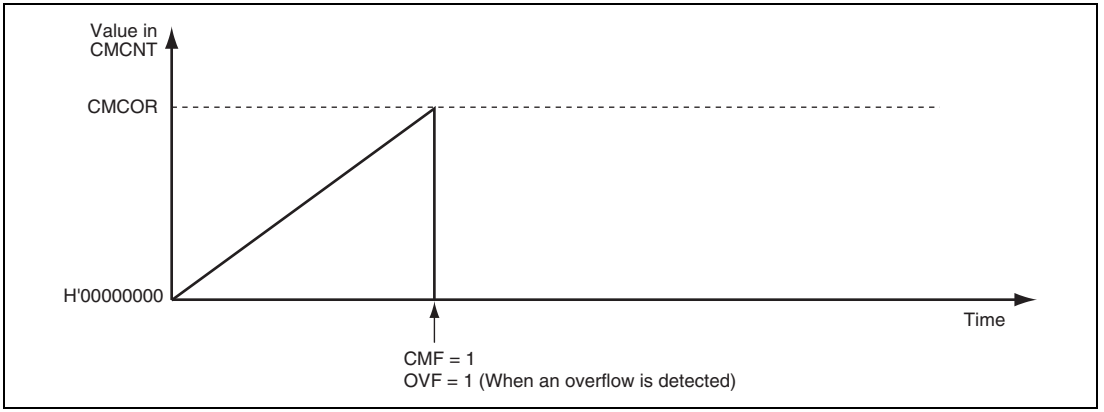


Figure 22.2 Counter Operation (One-Shot Operation)

- Free-Running Operation

  Free-running operation is selected by setting the CMM bit in CMCSR to B'1. When the value in CMCNT matches the value in CMCOR, CMCNT is cleared to H'00000000 and the CMF bit in CMCSR is set to B'1. CMCNT resumes counting-up after it has been cleared.

  To detect an overflow interrupt, set CMCOR to H'FFFFFFFF. When the values in CMCNT and CMCOR match, CMCNT is cleared to H'00000000 and the CMF and OVF bits in CMCSR are set to B'1.

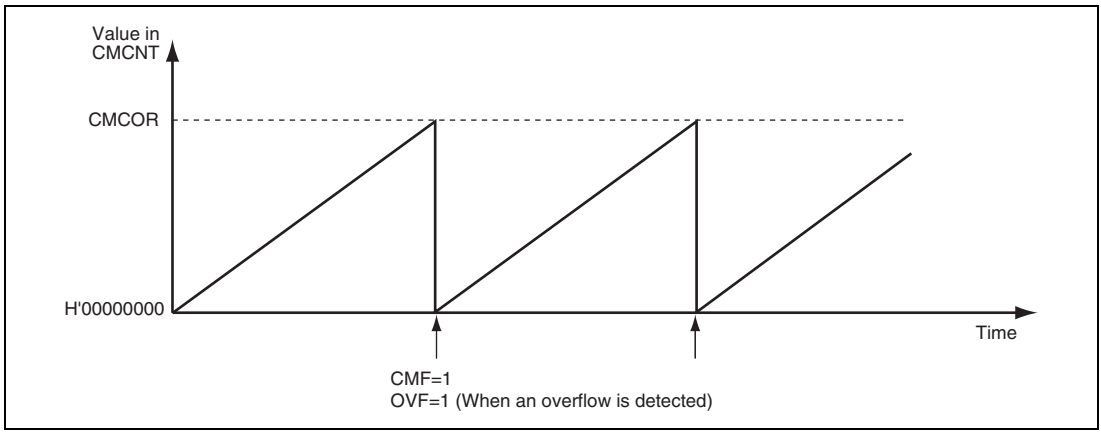


Figure 22.3 Counter Operation (Free-Running Operation)

22.3.2 Counter Size

In this module, the size of the counter is selectable as either 16 or 32 bits. This is selected by the CMS bit in CMCSR.

When the 16-bit size is selected, use a 32-bit value which has H'0000 as its upper half to set CMCOR.

To detect an overflow interrupt, the value must be set to H'0000FFFF.

22.3.3 Timing for Counting by CMCNT

In this module, the clock for the counter can be selected from among the following:

- External clock (RCLK): 1/8, 1/32, and 1/128

The clock for the counter is selected by the CKS bits in CMCSR. CMCNT is incremented at the rising edge of the selected clock.

22.3.4 Internal Interrupt Request to CPU

The setting of the CMR bits in CMCSR selects the sending of an internal interrupt request to the CPU at a compare match.

To clear the internal interrupt request to the CPU, the CMF bit should be set to B'0. Set the CMF bit to 0 in the handling routine for the CMT interrupt.

22.3.5 CMT Operation

As CMCNT can be operated only by the RCLK, it continues the counter operation even when this LSI makes the transition to R/U-standby mode and the core-area power supply is turned off after the necessary register settings have been made and B'1 has been written to the STR5 bit in CMSTR.

- Register Access

  The counter for the CMT is located in the sub area, because the counter has to be able to count-up even when the LSI is in R/U-standby mode. The registers that are needed for the counter's operation are set from the core area. The following restrictions thus apply to the modification of the registers.

After the following registers have been modified, do not make the transition to R/U-standby mode until you are sure that, according to the restrictions given below, the result of the modification has been reflected in the LSI's actual operation. Otherwise, the modification may not be reflected in an actual result.

1. CMCSR: Bits CKS, CMM, CMS, and CMTOUT_IE
   CMCOR: Bits 31 to 0
   CMSTR: Bit STR5

Two RCLK cycles are necessary before this register is either read or any modification of the value it holds is reflected in the LSI's actual operation.

2. CMCNT: Bits 31 to 0

Two RCLK cycles are necessary before this register is either read or any modification of the value it holds is reflected in the LSI's actual operation. Writing to this register, once started, is protected from further write operations until the writing operation has been completed.

- Canceling of Various Standby Modes

  When an interrupt occurs while the system is in standby or R/U-standby mode, various standby modes can be canceled if the CMTOUT_IE bit in CMCSR is set to B'1 and the CMR bits to B'10.

  Set the CMF bit in CMCSR to B'0 after various standby modes are canceled.

22.3.6 Compare Match Flag Set Timing

The CMF bit in CMCSR is set to 1 by the compare match signal generated when CMCOR and CMCNT match. The compare match signal is generated upon the final state of the match (timing at which the CMCNT value is updated to H'0000). Consequently, after CMCOR and CMCNT match, a compare match signal will not be generated until a CMCNT counter clock is input. Figure 22.4 shows the set timing of the CMF bit.

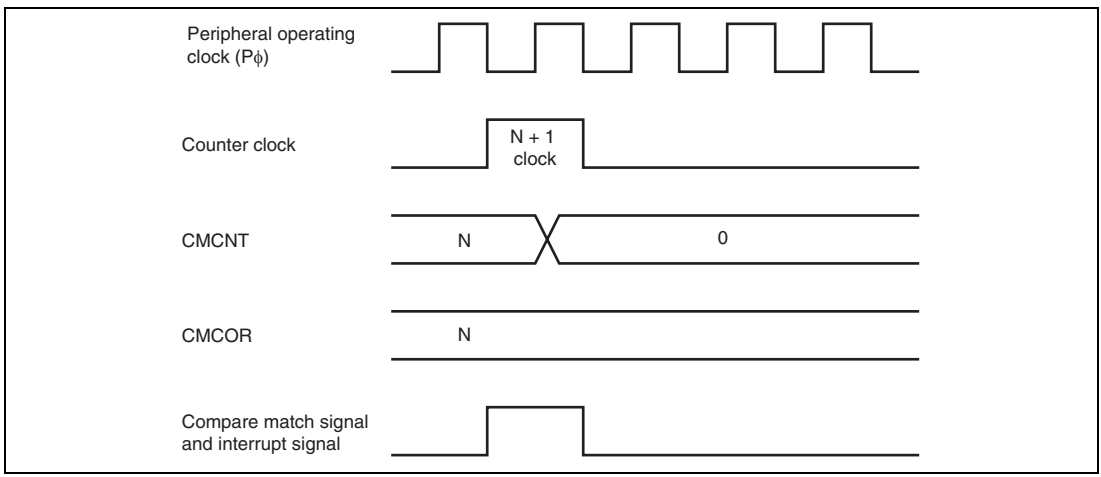


Figure 22.4 CMF Set Timing

Section 23 Multi Media Card Interface (MMCIF)

This module is a host interface that allows connection with Multi Media Card (referred to as MMC hereafter), which is a flash memory card, or HDD that conforms to the Consumer Electronics AT Attachment (CE-ATA) standard, which is an applied version of MMC standard.

23.1 Features

(1) MMC Interface

- Conforms to the MultiMediaCard System Specification
- Supports 1/4/8-bit MMC
- MMC clock frequency ≤ 1/2 × bus clock frequency (Bϕ).
- Error checking functions (CRC7, CRC16)
- Two types of interrupt requests: normal operation and error/timeout.
- DMA transfer requests: buffer write and buffer read
- Supports MMC mode (does not support SPI mode)

(2) CE-ATA Interface

- Conforms to the CE-ATA Digital Protocol
- Supports 1/4/8-bit CE-ATA devices
- MMC clock frequency ≤ 1/2 × bus clock frequency (Bϕ).
- Error checking functions (CRC7, CRC16)
- Two types of interrupt requests: normal operation interrupts and error/timeout interrupts
- DMA transfer requests: buffer write and buffer read
- Supports Command Completion Signal (CCS) and Command Completion Signal Disable (CCSD)

Figure 23.1 shows a block diagram of this module.

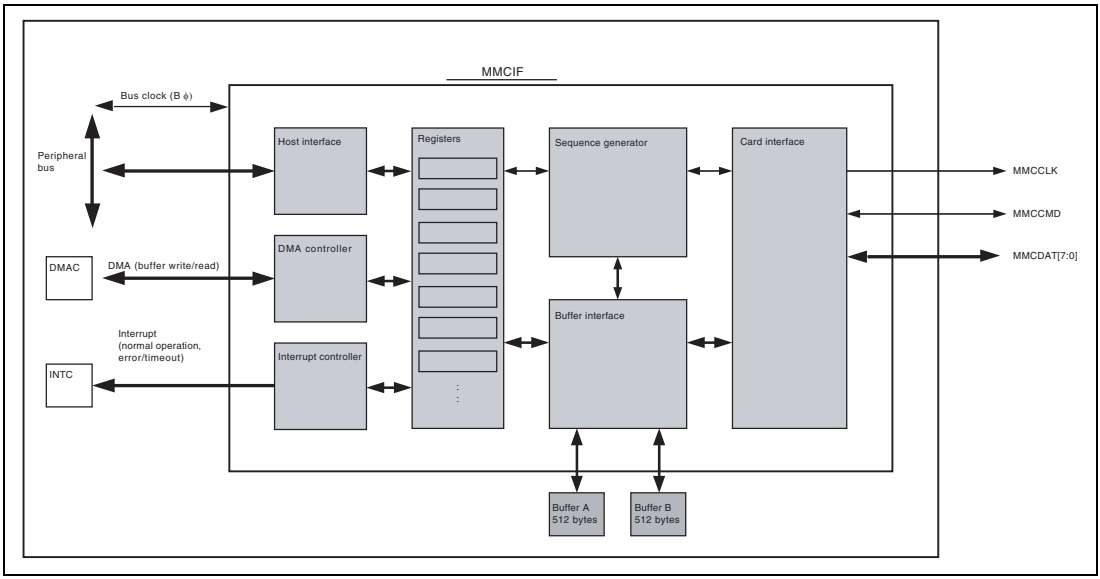


Figure 23.1 Block Diagram of MMCIF

23.2 Input/Output Pins

Table 23.1 shows the pin configuration of the MMCIF.

Table 23.1 Pin Configuration

Pin Name	I/O	Function
MMCCLK	Output	MMC clock
MMCCMD	Input/output	Command/response
MMCDAT[7:0]	Input/output	Transmit/receive data

23.3 Register Descriptions

Table 23.2 shows the register configuration of the MMCIF and Table 23.3 shows the register state in each processing mode.

Table 23.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Command setting register	CE_CMD_SET	R/W	H'A4CA0000	16, 32
Argument register	CE_ARG	R/W	H'A4CA0008	16, 32
Argument register for automatically-issued CMD12	CE_ARG_CMD12	R/W	H'A4CA000C	16, 32
Command control register	CE_CMD_CTRL	R/W	H'A4CA0010	16, 32
Transfer block setting register	CE_BLOCK_SET	R/W	H'A4CA0014	16, 32
Clock control register	CE_CLK_CTRL	R/W	H'A4CA0018	16, 32
Buffer access configuration register	CE_BUF_ACC	R/W	H'A4CA001c	16, 32
Response register 3	CE_RESP3	R/W	H'A4CA0020	16, 32
Response register 2	CE_RESP2	R/W	H'A4CA0024	16, 32
Response register 1	CE_RESP1	R/W	H'A4CA0028	16, 32
Response register 0	CE_RESP0	R/W	H'A4CA002C	16, 32
Response register for automatically-issued CMD12	CE_RESP_CMD12	R/W	H'A4CA0030	16, 32
Data register	CE_DATA	R/W	H'A4CA0034	16*, 32
Interrupt flag register	CE_INT	R/W	H'A4CA0040	16, 32
Interrupt mask register	CE_INT_MASK	R/W	H'A4CA0044	16, 32
Status register 1	CE_HOST_STS1	R/W	H'A4CA0048	16, 32
Status register 2	CE_HOST_STS2	R/W	H'A4CA004C	16, 32
Version register	CE_VERSION	R/W	H'A4CA007C	16, 32

Note: * For 16-bit access, H'A4CA0034 is the only address for access.

Note: Do not access registers other than those shown above.

Table 23.3 Register state in each Processing mode

Abbreviation	Power on reset	Manual reset	Software standby	Module standby	R-standby	U-standby	Sleep
CE_CMD_SET	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_ARG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_ARG_CMD12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_CMD_CTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_BLOCK_SET	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_CLK_CTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_BUF_ACC	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_RESP3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_RESP2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_RESP1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_RESP0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_RESP_CMD12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_DATA	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_INT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_INT_MASK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_HOST_STS1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_HOST_STS2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CE_VERSION	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

23.3.1 Command Setting Register (CE_CMD_SET)

CE_CMD_SET sets a command sequence. The command sequence starts when a command index has been set along with other required settings. If this register is accessed in 16-bit units, the command sequence starts when the settings have been made in bits 31 to 16. Note that writing to CE_CMD_SET is disabled while a command sequence is proceeding (i.e., the value of CMDSEQ in CE_HOST_STS1 is 1). For the setting values of CE_CMD_SET, see section 23.7.14, Setting Values of CE_CMD_SET.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMD[5:0]						RTYP[1:0]		RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIDXC[1:0]		RCRC7C[1:0]		—	CRC16C	—	CRCSTE	TBIT	OPDM	CCSH	—	—	—	DATW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 24	CMD[5:0]	All 0	R/W	Command Index Note: Setting a command index in these bits initiates the command sequence.
23, 22	RTYP[1:0]	00	R/W	Response Type 00: No response 01: 6-byte response (R1, R1b, R3, R4, R5) 10: 17-byte response (R2) 11: Setting prohibited
21	RBSY	0	R/W	Response Busy Select Selects whether “busy” is involved in response reception. 0: No response busy 1: Response busy involved (R1b)

Bit	Bit Name	Initial Value	R/W	Description
20	CCSEN	0	R/W	CCS Acceptance 0: Acceptance of CCS disabled 1: Acceptance of CCS enabled
19	WDAT	0	R/W	Presence/Absence of Data 0: No data 1: With data
18	DWEN	0	R/W	Read/Write 0: Read from the card 1: Write to the card
17	CMLTE	0	R/W	Single/Multi Block Transfer Select 0: Single-block transfer 1: Multi-block transfer
16	CMD12EN	0	R/W	Automatic CMD12 Issuance (valid when multi-block transfer is selected) 0: Disables automatic CMD12 issuance 1: Enables automatic CMD12 issuance Note: Set the transfer block size to 512 bytes
15, 14	RIDXC[1:0]	00	R/W	Response Index Check 00: Checks the response index 01: Checks the check bits 10: No checking 11: Setting prohibited
13, 12	RCRC7C [1:0]	00	R/W	Response CRC7 Check 00: Checks CRC7 (set the response type to 01) 01: Checks the check bits (set the response type to 01) 10: Checks internal CRC7 (R2 only) (set the response type to 10) 11: No checking
11	—	0	R	Reserved The write value should always be 0.
10	CRC16C	0	R/W	CRC16 Check in Reception 0: Checks CRC16 1: Does not check CRC16 (use when CMD14)

Bit	Bit Name	Initial Value	R/W	Description
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	CRCSTE	0	R/W	CRC Status Reception 0: Receives CRC status 1: Does not receive CRC status (use when CMD19)
7	TBIT	0	R/W	Transmission Bit Setting 0: Sets the transmission bit to high 1: Sets the transmission bit to low
6	OPDM	0	R/W	Open-Drain Output Mode 0: Normal output 1: Open-drain output Note: This setting is only applied to the MMCCMD line.
5	CCSH	0	R/W	Output High after CCS Reception 0: Outputs a high level at the third cycle after CCS is received 1: Does not output a high level after CCS is received
4 to 2	—	000	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	DATW[1:0]	00	R/W	Data Bus Width Setting 00: 1 bit 01: 4 bits 10: 8 bits 11: Setting prohibited

23.3.2 Argument Register (CE_ARG)

CE_ARG sets the argument for the command to be transmitted. This register must be set before CMD[5:0] in CE_CMD_SET is set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARG[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARG[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ARG[31:0]	H'00000000	R/W	Set bits 31 to 0 of the argument.

23.3.3 Argument Register for Automatically-Issued CMD12 (CE_ARG_CMD12)

CE_ARG_CMD12 is used to set the argument for the automatically-issued CMD12 in multi-block transfer. This register must be set before CMD[5:0] in CE_CMD_SET is set. For automatic issuance of CMD12, see section 23.6.2, Automatic CMD12 Issuance.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	C12ARG[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	C12ARG[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	C12ARG[31:0]	H'00000000	R/W	Set bits 31 to 0 of the argument.

23.3.4 Command Control Register (CE_CMD_CTRL)

CE_CMD_CTRL is used to terminate a command sequence forcibly. It also has the function to issue CCSD, which is effective when a CE-ATA device is connected. Do not issue CCSD when an MMC is connected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CCSD	BREAK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CCSD	0	R/W	CCSD Issuance Writing 1 to this bit while it is 0 issues a CCSD. After making sure that the CCSDE bit in CE_INT has become 1, write 0 to this bit to reset. Note: Before issuing a CCSD, make sure that the value of the CMDSEQ bit in CE_HOST_STS1 is 0.
0	BREAK	0	R/W	Forcible Termination of Command Sequence Writing 1 to this bit while it is 0 and then writing 0 to it discontinues the current command sequence. After this bit is set as described above, check if the value of the CMDSEQ bit in CE_HOST_STS1 has become 0, which indicates that the next processing can be performed. Note: Refer to section 23.8, Usage Notes.

23.3.5 Transfer Block Setting Register (CE_BLOCK_SET)

CE_BLOCK_SET specifies the size of the block and the number of blocks for the data to be transferred. This register must be set before CMD[5:0] in CE_CMD_SET is set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BLKCNT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BLKSIZ[15:0]															
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	BLKCNT [15:0]	H'0000	R/W	Number of Blocks for Transfer Note: This setting is valid for multi-block transfer.
15 to 0	BLKSIZ [15:0]	H'0200	R/W	Transfer Block Size Note: Transfer block size should be set as follows. • Single-block transfer: 1 to 512 bytes • Multi-block transfer: 512 bytes

23.3.6 Clock Control Register (CE_CLK_CTRL)

CE_CLK_CTRL controls the MMC clock and sets timeout values. Do not change the setting of this register while a command sequence is in progress.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CLKEN	—	—	—	—	CLKDIV[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SRSPTO[1:0]		SRBSYTO[3:0]				SRWDTO[3:0]				SCCSTO[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	CLKEN	0	R/W	MMC Clock Output Control 0: Does not output the MMC clock (tied to low level) 1: Outputs the MMC clock
23 to 20	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	CLKDIV[3:0]	0000	R/W	MMC Clock Frequency Setting 0000: Bus clock/2 0001: Bus clock/2^2 0111: Bus clock/2^8 1000: Bus clock/2^9 1001 to 1111: Setting prohibited
15,14	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13, 12	SRSPTO [1:0]	00	R/W	Response Timeout Setting Specifies the timeout period for the RSPTO bit of CE_INT. 00: 64 MMC clock cycles 01: 128 MMC clock cycles 10: 256 MMC clock cycles 11: Setting prohibited
11 to 8	SRBSYTO [3:0]	0000	R/W	Response Busy Timeout Setting Specifies the timeout period for the RBSYTO bit of CE_INT. 0000: 2^{14} MMC clock cycles 0001: 2^{15} MMC clock cycles : 1110: 2^{28} MMC clock cycles 1111: 2^{29} MMC clock cycles
7 to 4	SRWDTO [3:0]	0000	R/W	Write Data/Read Data Timeout Setting Specifies the timeout period for the WDATTO and RDATTO bits of CE_INT. 0000: 2^{14} MMC clock cycles 0001: 2^{15} MMC clock cycles : 1110: 2^{28} MMC clock cycles 1111: 2^{29} MMC clock cycles
3 to 0	SCCSTO [3:0]	0000	R/W	CCS Timeout Setting Specifies the timeout period for the CCSTO bit of CE_INT. 0000: 2^{14} MMC clock cycles 0001: 2^{15} MMC clock cycles : 1110: 2^{28} MMC clock cycles 1111: 2^{29} MMC clock cycles

23.3.7 Buffer Access Configuration Register (CE_BUF_ACC)

CE_BUF_ACC configures the method of accessing data registers and mode of DMA transfer. This register must be set before CMD[5:0] in CE_CMD_SET is set. For explanation of the buffers, see section 23.6.3, Buffer Structure.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DMAWEN	DMAREN	—	—	—	—	—	—	BUSW	ATYP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

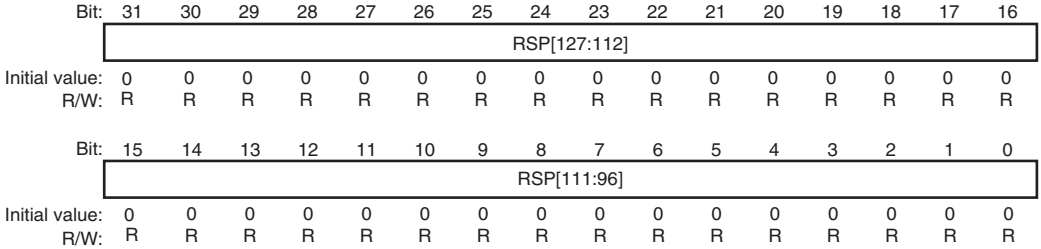
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	DMAWEN	0	R/W	Buffer Write DMA Transfer Request Enable 0: Disables DMA transfer request for buffer writing 1: Enables DMA transfer request for buffer writing
24	DMAREN	0	R/W	Buffer Read DMA Transfer Request Enable 0: Disables DMA transfer request for buffer reading 1: Enables DMA transfer request for buffer reading
23 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	BUSW	0	R/W	Data register access size selection 0: When access to CE_DATA in 32-bit. 1: When access to CE_DATA in 16-bit.
16	ATYP	0	R/W	0: When not swapped byte-wise. 1: When swapped byte-wise.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23.3.8 Response Registers 3 to 0 (CE_RESP3 to CE_RESP0)

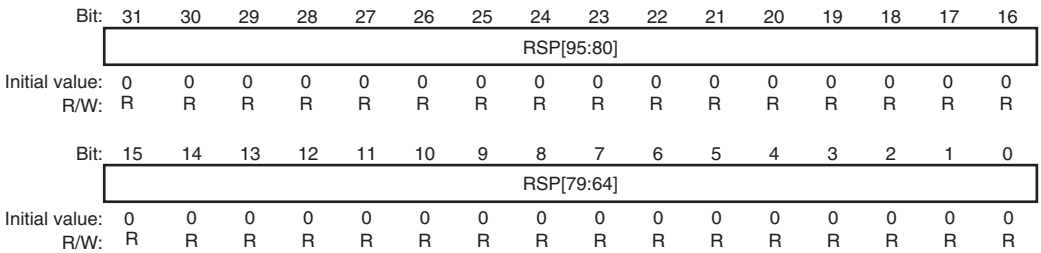
CE_RESP3 to CE_RESP0 are the registers for storing the response that has been received. For the formats of response values, see section 23.6.1, Command/Response Formats.

- CE_RESP3



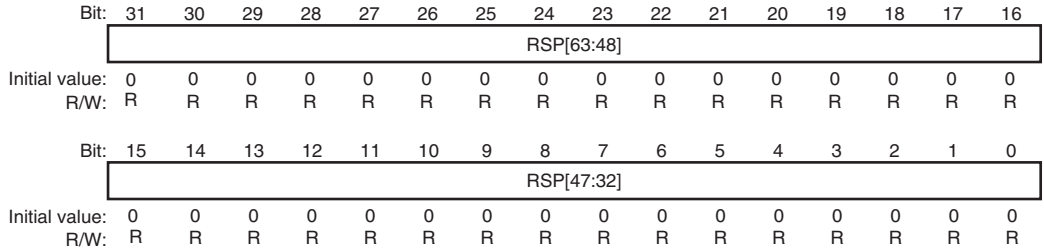
Bit	Bit Name	Initial Value	R/W	Description (CE_RESP3)
31 to 0	RSP[127:96]	H'00000000	R	R2 response [127:96]

- CE_RESP2



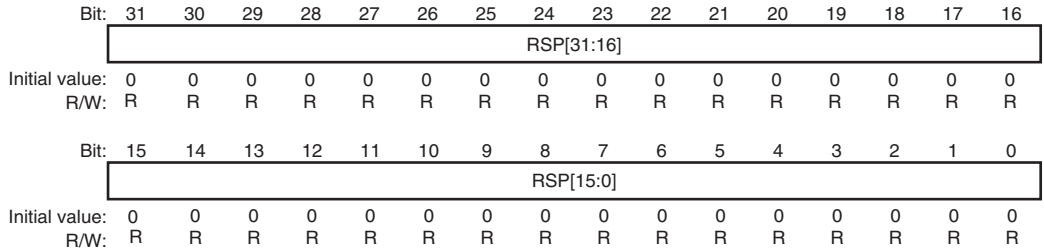
Bit	Bit Name	Initial Value	R/W	Description (CE_RESP2)
31 to 0	RSP[95:64]	H'00000000	R	R2 response [95:64]

- CE_RESP1



Bit	Bit Name	Initial Value	R/W	Description (CE_RESP1)
31 to 0	RSP[63:32]	H'00000000	R	R2 response [63:32]

- CE_RESP0



Bit	Bit Name	Initial Value	R/W	Description (CE_RESP0)
31 to 0	RSP[31:0]	H'00000000	R	Response [31:0] or R2 response [31:0]

23.3.9 Response Register for Automatically-Issued CMD12 (CE_RESP_CMD12)

CE_RESP_CMD12 is the register for storing the response to the automatically-issued CMD12.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP12[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP12[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP12[31:0]	H'00000000	R	CMD12 response [31:0]

23.3.10 Data Register (CE_DATA)

CE_DATA is used to access the buffers of this module. In 16-bit access, only DATA[31:16] is accessible. For the write/read data formats, see section 23.6.5, Data Formats.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	H'00000000	R/W	Buffer write/read data [31:0]

23.3.11 Interrupt Flag Register (CE_INT)

CE_INT indicates various statuses during execution of a command sequence. Each bit is set when its setting condition has been met. To clear flag(s), write 0 only to the bit(s) to be cleared and write 1 to the other bits.

For the operation in the case of an error or timeout, see section 23.6.6, Operation in the Case of Error/Timeout.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CCSDE	—	—	CMD12 DRE	CMD12 RBE	CMD12 CRE	DTRAN E	BUFR E	BUFW EN	BUFR EN	CCS RCV	—	RBSY E	CRSP E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD VIO	BUF VIO	—	—	WDAT ERR	RDAT ERR	RIDX ERR	RSP ERR	—	—	CCS TO	CRCS TO	WDAT TO	RDAT TO	RBSY TO	RSP TO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00	R	Reserved These bets are always read as 0. The writing value should always be 0.
29	CCSDE	0	R/W*	CCSD Issuance Complete [Setting conditions] CCSD has been issued [Clearing condition] Writing a 0 to this bit
28, 27	—	00	R	Reserved These bets are always read as 0. The write value should always be 0.
26	CMD12DRE	0	R/W*	Automatic CMD12 Issuance & Buffer Read Complete [Setting conditions] Response busy for automatically-issued CMD12 and buffer reading have been completed. [Clearing condition] Writing a 0 to this bit Note: When CMD12DRE has been set, CMD12RBE, CMD12CRE, and BUFRE have also been set. So, these bits should be cleared as well.

Bit	Bit Name	Initial Value	R/W	Description
25	CMD12RBE	0	R/W*	Automatic CMD12 Issuance Response Busy Complete [Setting conditions] Reception of the response and response busy for an automatically-issued CMD12 have been completed. [Clearing condition] Writing a 0 to this bit Note: When CMD12RBE has been set, CMD12CRE has also been set. So, this bit should be cleared as well.
24	CMD12CRE	0	R/W*	Automatic CMD12 Response Complete [Setting conditions] The response to an automatically-issued CMD12 has been received. [Clearing condition] Writing a 0 to this bit
23	DTRANE	0	R/W*	Data Transmission Complete [Setting conditions] Transmission of all blocks of data has been completed. • When configured to receive CRC status: Completion of busy (data busy) after reception of CRC status • When configured not to receive CRC status: Completion of data transmission [Clearing condition] Writing a 0 to this bit
22	BUFRE	0	R/W*	Buffer Read Complete [Setting conditions] All blocks of data have been received and the data have been read from the buffer [Clearing condition] Writing a 0 to this bit
21	BUFWEN	0	R/W*	Buffer Write Ready [Setting conditions] The buffer has become empty and ready for writing. [Clearing condition] Writing a 0 to this bit Note: This bit is not set when DMA transfer request for buffer writing is enabled.

Bit	Bit Name	Initial Value	R/W	Description
20	BUFREN	0	R/W*	Buffer Read Ready [Setting conditions] Transfer block size of data have been stored in the buffer and it has become ready for reading [Clearing condition] Writing a 0 to this bit Note: This bit is not set when DMA transfer request for buffer reading is enabled.
19	CCSRCV	0	R/W*	CCS Reception Complete [Setting conditions] CCS has been received. [Clearing condition] Writing a 0 to this bit
18	—	0	R	Reserved This bet is always read as 0.The write value should always be 0.
17	RBSYE	0	R/W*	Response Busy Complete [Setting conditions] Reception of a response and response busy have been completed. [Clearing condition] Writing a 0 to this bit Note: When RBSYE has been set, CRSPE has also been set. So, this bit should be cleared as well. Completion of reception of the response and response busy for automatically-issued CMD12 is reflected in CMD12RBE.
16	CRSPE	0	R/W*	Command/Response Complete [Setting conditions] A command has been transmitted or a response has been received • When configured not to receive response: A command has been transmitted • When configured to receive 6- or 17-byte response: A response has been received [Clearing condition] Writing a 0 to this bit Note: Completion of reception of the response to automatically-issued CMD12 is reflected in CMD12CRE.

Bit	Bit Name	Initial Value	R/W	Description
15	CMDVIO	0	R/W*	Command Issuance Error [Setting conditions] Illegal setting has been made in CE_CMD_SET or CE_BLOCK_SET. • During execution of a command sequence: Writing to CMD[5:0] in CE_CMD_SET • At the start of command sequence: Writing to CMD[5:0] in CE_CMD_SET when the registers have been set for one of the following combinations of selection — No response + response busy — No response + with data — No response + acceptance of CCS enabled — No data + automatic CMD12 issuance — With data + single-block transfer + automatic CMD12 issuance — With data + automatic CMD12 issuance + acceptance of CCS enabled — With data + transfer block size = 0 — With data + transfer block size ≥ 513 — With data + multi-block transfer + number of blocks for transfer = 0 [Clearing condition] Writing a 0 to this bit Note: When CMDVIO has been set, the command sequence is not stopped automatically.

Bit	Bit Name	Initial Value	R/W	Description
14	BUFVIO	0	R/W*	Buffer Access Error [Setting conditions] Illegal buffer access has been attempted. • CE_ DATA has been accessed exceeding the block size set in BLKSIZ[15:0] in CE_BLOCK_SET • While data is being read from the card: CE_DATA has been accessed with BUFREN not set (when DMA is used, with no DMA transfer request asserted for buffer reading) • While data is being written to the card: CE_DATA has been accessed with BUFWEN not set (when DMA is used, with no DMA transfer request asserted for buffer writing) [Clearing condition] Writing a 0 to this bit Note: When BUFVIO has been set, the command sequence is not stopped automatically.
13, 12	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
11	WDATERR	0	R/W*	Write Data Error [Setting conditions] Error is found in the data that has been written. • Error is in the status of the CRC status • Error is in the end bits of the CRC status [Clearing condition] Writing a 0 to this bit Note: When WDATERR has been set, the command sequence is stopped automatically.
10	RDATERR	0	R/W*	Read Data Error [Setting conditions] Error is found in the read data. • Error is in CRC16 of the read data • Error is in the end bits of the read data [Clearing condition] Writing a 0 to this bit Note: When RDATERR has been set, the command sequence is stopped automatically.

Bit	Bit Name	Initial Value	R/W	Description
9	RIDXERR	0	R/W*	Response Index Error [Setting conditions] Error has been found in the index value of the response. • When RIDXC[1:0] in CE_CMD_SET is 2'b00: The index value in the received response does not match that of the transmitted command • When RIDXC[1:0] in CE_CMD_SET is 2'b01: The check bits are not all 1 [Clearing condition] Writing a 0 to this bit Note: When RIDXERR has been set, the command sequence is stopped automatically.
8	RSPERR	0	R/W*	Response Error [Setting conditions] Error has been found in the response values of the response. • Transmission bit in the response is high • Error is in the end bits of the response • When RCRC7C[1:0] in CE_CMD_SET is 2'b00: Error is in CRC7 of the response • When RCRC7C [1:0] in CE_CMD_SET is 2'b01: The check bits are not all 0 • When RCRC7C[1:0] in CE_CMD_SET is 2'b10: Error is in internal CRC7 of the response [Clearing condition] Writing a 0 to this bit Note: When RSPERR has been set, the command sequence is stopped automatically.
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	CCSTO	0	R/W*	CCS Timeout [Setting conditions] CCS could not be received [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SCCSTO[3:0] of CE_CLK_CTRL. The command sequence is not stopped even if CCSTO is set.
4	CRCSTO	0	R/W*	CRC Status Timeout [Setting conditions] CRC status could not be received [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if CRCSTO is set.
3	WDATTO	0	R/W*	Write Data Timeout [Setting conditions] Data busy that follows CRC status does not end [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SRWDTO[3:0] of CE_CLK_CTRL. The command sequence is not stopped even if WDATTO is set.
2	RDATTO	0	R/W*	Read Data Timeout [Setting conditions] Read data could not be received [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SRWDTO[3:0] of CE_CLK_CTRL. The command sequence is not stopped even if RDATTO is set.

Bit	Bit Name	Initial Value	R/W	Description
1	RBSYTO	0	R/W*	Response Busy Timeout [Setting conditions] Response busy does not end [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SRBSYTO[3:0] of CE_CLK_CTRL. The command sequence is not stopped even if RBSYTO is set.
0	RSPTO	0	R/W*	Response Timeout [Setting conditions] Response could not be received [Clearing condition] Writing a 0 to this bit Note: The period for timeout detection is set in SRSPTO[1:0] of CE_CLK_CTRL. The command sequence is not stopped even if RSPTO is set.

Note: * A 0 is the only value that can be written to these bits. Writing a 1 is ignored.

23.3.12 Interrupt Mask Register (CE_INT_MASK)

CE_INT_MASK controls output of the CE_INT-related interrupt signals. If a flag in CE_INT is set to 1 while its corresponding bit in CE_INT_MASK is set to 1, an interrupt request is output. For details on interrupt requests, see section 23.4, Interrupt Requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MCCSDE	—	—	MCMD12DRE	MCMD12RBE	MCMD12CRE	MDTRANE	MBUFRE	MBUFWEN	MBUFREN	MCCSRCV	—	MRBSYE	MCRSPE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCMDVIO	MBUFVIO	—	—	MWDATERR	MRDATERR	MRIDXERR	MRSPERR	—	—	MCCSTO	MCRCSTO	MWDATTO	MRDATTO	MRBSYTO	MRSPTO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

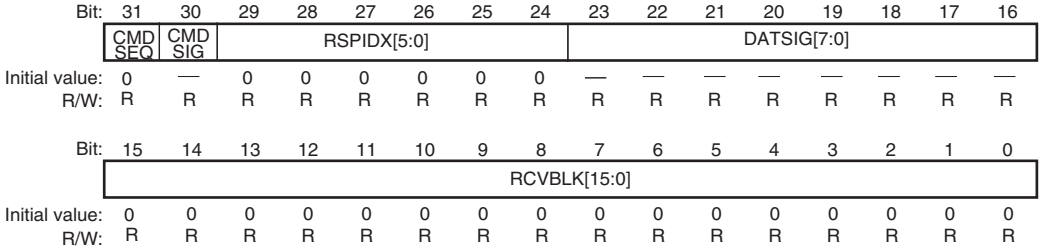
Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
29	MCCSDE	0	R/W	CCSDE Interrupt Mask 0: Disables interrupt output by the CCSDE flag 1: Enables interrupt output by the CCSDE flag
28, 27	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
26	MCMD12DRE	0	R/W	CMD12DRE Interrupt Mask 0: Disables interrupt output by the CMD12DRE flag 1: Enables interrupt output by the CMD12DRE flag
25	MCMD12RBE	0	R/W	CMD12RBE Interrupt Mask 0: Disables interrupt output by the CMD12RBE flag 1: Enables interrupt output by the CMD12RBE flag
24	MCMD12CRE	0	R/W	CMD12CRE Interrupt Mask 0: Disables interrupt output by the CMD12CRE flag 1: Enables interrupt output by the CMD12CRE flag
23	MDTRANE	0	R/W	DTRANE Interrupt Mask 0: Disables interrupt output by the DTRANE flag 1: Enables interrupt output by the DTRANE flag

Bit	Bit Name	Initial Value	R/W	Description
22	MBUFRE	0	R/W	BUFRE Interrupt Mask 0: Disables interrupt output by the BUFRE flag 1: Enables interrupt output by the BUFRE flag
21	MBUFWEN	0	R/W	BUFWEN Interrupt Mask 0: Disables interrupt output by the BUFWEN flag 1: Enables interrupt output by the BUFWEN flag
20	MBUFREN	0	R/W	BUFREN Interrupt Mask 0: Disables interrupt output by the BUFREN flag 1: Enables interrupt output by the BUFREN flag
19	MCCSRCV	0	R/W	CCSRCV Interrupt Mask 0: Disables interrupt output by the CCSRCV flag 1: Enables interrupt output by the CCSRCV flag
18	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	MRBSYE	0	R/W	RBSYE Interrupt Mask 0: Disables interrupt output by the RBSYE flag 1: Enables interrupt output by the RBSYE flag
16	MCRSPE	0	R/W	CRSPE Interrupt Mask 0: Disables interrupt output by the CRSPE flag 1: Enables interrupt output by the CRSPE flag
15	MCMDVIO	0	R/W	CMDVIO Interrupt Mask 0: Disables interrupt output by the CMDVIO flag 1: Enables interrupt output by the CMDVIO flag
14	MBUFVIO	0	R/W	BUFVIO Interrupt Mask 0: Disables interrupt output by the BUFVIO flag 1: Enables interrupt output by the BUFVIO flag
13, 12	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
11	MWDATERR	0	R/W	WDATERR Interrupt Mask 0: Disables interrupt output by the WDATERR flag 1: Enables interrupt output by the WDATERR flag

Bit	Bit Name	Initial Value	R/W	Description
10	MRDATERR	0	R/W	RDATERR Interrupt Mask 0: Disables interrupt output by the RDATERR flag 1: Enables interrupt output by the RDATERR flag
9	MRIDXERR	0	R/W	RIDXERR Interrupt Mask 0: Disables interrupt output by the RIDXERR flag 1: Enables interrupt output by the RIDXERR flag
8	MRSPERR	0	R/W	RSPERR Interrupt Mask 0: Disables interrupt output by the RSPERR flag 1: Enables interrupt output by the RSPERR flag
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
5	MCCSTO	0	R/W	CCSTO Interrupt Mask 0: Disables interrupt output by the CCSTO flag 1: Enables interrupt output by the CCSTO flag
4	MCRCSTO	0	R/W	CRCSTO Interrupt Mask 0: Disables interrupt output by the CRCSTO flag 1: Enables interrupt output by the CRCSTO flag
3	MWDATTO	0	R/W	WDATTO Interrupt Mask 0: Disables interrupt output by the WDATTO flag 1: Enables interrupt output by the WDATTO flag
2	MRDATTO	0	R/W	RDATTO Interrupt Mask 0: Disables interrupt output by the RDATTO flag 1: Enables interrupt output by the RDATTO flag
1	MRBSYTO	0	R/W	RBSYTO Interrupt Mask 0: Disables interrupt output by the RBSYTO flag 1: Enables interrupt output by the RBSYTO flag
0	MRSPTO	0	R/W	RSPTO Interrupt Mask 0: Disables interrupt output by the RSPTO flag 1: Enables interrupt output by the RSPTO flag

23.3.13 Status Register 1 (CE_HOST _STS1)

CE_HOST _STS1 indicates the number of blocks that have been transferred, the states of the CMD line and DAT lines, the index of the response that has been received, and whether a command sequence is in progress.



Bit	Bit Name	Initial Value	R/W	Description
31	CMDSEQ	0	R	Command Sequence in Progress 0: Command sequence is in the initial state 1: Command sequence is being executed
30	CMDSIG	Undefined	R	CMD Line Status Indicates the state on the command line.
29 to 24	RSPIDX [5:0]	H'00	R	Response Index Indicate the index of the response that has been received
23 to 16	DATSIG [7:0]	Undefined	R	DAT[7:0] Status Indicate the states on the DAT lines
15 to 0	RCVBLK [15:0]	H'0000	R	Number of Transferred Blocks Indicate the number blocks that have been transferred. • When the DWEN bit in CE_CMD_SET is 0 Number of blocks read from the card • When the DWEN bit in CE_CMD_SET is 1 Number of blocks written to the card

23.3.14 Status Register 2 (CE_HOST _STS2)

CE_HOST _STS2 indicates timeout and error statuses.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRC STE	CRC 16E	AC12 CRCE	RSP CRC7E	CRC STEBE	RDAT EBE	AC12R EBE	RSP EBE	AC12 IDXE	RSP IDXE	—	—	—	CRCST[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCS TO	RDAT TO	DATBS YTO	CRCST TO	AC12 BSYTO	RSPBS YTO	AC12 RSPTO	RSP TO	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CRCSTE	0	R	CRC Status Error Set to 1 when an error is found in the CRC status value.
30	CRC16E	0	R	Read Data CRC16 Error Set to 1 when an error is found in CRC16 in the read data.
29	AC12CRCE	0	R	Automatic CMD12 Response CRC7 Error Set to 1 when an error is found in CRC7 of the response to the automatically-issued CMD12.
28	RSPCRC7E	0	R	Command Response CRC7 Error Set to 1 when an error is found in CRC7 of the response.
27	CRCSTEBE	0	R	CRC Status End Bit Error Set to 1 when an error is found in the end bits in CRC status.
26	RDATEBE	0	R	Read Data End Bit Error Set to 1 when an error is found in the end bits in the read data.
25	AC12REBE	0	R	Automatic CMD12 Response End Bit Error Set to 1 when an error is found in the end bits of the response to the automatically-issued CMD12.
24	RSPEBE	0	R	Command Response End Bit Error Set to 1 when an error is found in the end bits of the response.
23	AC12IDXE	0	R	Automatic CMD12 Response Index Error Set to 1 when an error is found in the index of the response to the automatically-issued CMD12.

Bit	Bit Name	Initial Value	R/W	Description
22	RSPIDXE	0	R	Command Response Index Error Set to 1 when an error is found in the index of the response.
21 to 19	—	000	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	CRCST [2:0]	000	R	CRC Status Indicate the value of the CRC status that has been received.
15	CCSTO	0	R	CCS Timeout Set to 1 if CCS is not received within the period set by the SCCSTO bits in CE_CLK_CTRL.
14	RDATTO	0	R	Read Data Timeout Set to 1 if read data is not received within the period set by the SRWDTO bits in CE_CLK_CTRL.
13	DATBSYTO	0	R	Data Busy Timeout Set to 1 if data busy continues exceeding the period set by the SRWDTO bits in CE_CLK_CTRL.
12	CRCSTTO	0	R	CRC Status Timeout Set to 1 if CRC status could not be received.
11	AC12BSYTO	0	R	Automatic CMD12 Response Busy Timeout Set to 1 if the response busy of the automatically-issued CMD12 continues exceeding the period set by the SRBSYTO bits in CE_CLK_CTRL.
10	RSPBSYTO	0	R	Response Busy Timeout Set to 1 if the response busy continues exceeding the period set by the SRBSYTO bits in CE_CLK_CTRL.
9	AC12RSPTO	0	R	Automatic CMD12 Response Timeout Set to 1 if the response to the automatically-issued CMD12 is not received within the period set by the SRSPTO bits in CE_CLK_CTRL.
8	RSPTO	0	R	Response Timeout Set to 1 if the response is not received within the period set by the SRSPTO bits in CE_CLK_CTRL.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

23.3.15 Version Register (CE_VERSION)

CE_VERSION indicates the version number and controls software reset of this module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SW RST	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VERSION[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SWRST	0	R/W	Software Reset 0: Software reset cleared (normal operation) 1: Executes software reset
30 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	VERSION [15:0]	All 0	R	Version Information Indicates the version number of this module.

23.4 Interrupt Requests

This module generates these types of interrupt requests: normal operation and error/timeout. The interrupt flags are accommodated in the CE_INT. When a bit in the flag register is set to 1 and also the corresponding bit in the interrupt mask register is set to 1 (enabled), an interrupt request is generated.

Table 23.4 shows the specification of the interrupt requests.

Flag Register	Bit	Mask Register	Bit	Interrupt Request
CE_INT	CCSDE	CE_INT_MASK	MCCSDE	Normal operation interrupt
	CMD12DRE		MCMD12DRE	
	CMD12RBE		MCMD12RBE	
	CMD12CRE		MCMD12CRE	
	DTRANE		MDTRANE	
	BUFRE		MBUFRE	
	BUFWEN		MBUFWEN	
	BUFREN		MBUFREN	
	CCSRCV		MCCSRCV	
	RBSYE		MRBSYE	
	CRSPE		MCRSPE	
	CMDVIO		MCMDVIO	Error/timeout interrupt
	BUFVIO		MBUFVIO	
	WDATERR		MWDATERR	
	RDATERR		MRDATERR	
	RIDXERR		MRIDXERR	
	RSPERR		MRSPERR	
	CCSTO		MCCSTO	
	CRCSTO		MCRCSTO	
	WDATTO		MWDATTO	
	RDATTO		MRDATTO	
	RBSYTO		MRBSYTO	
	RSPTO		MRSPTO	

23.5 DMA Specifications

23.5.1 DMA for Buffer Writing

The DMA transfer request is asserted for buffer writing when the buffer has become empty while the DMAWEN bit in CE_BUF_ACC is set to 1.

The DMA transfer request stays asserted for the amount of data specified by BLKSIZ (the block size set in CE_BLOCK_SET) × BLKCNT (the number of blocks for transfer set in CE_BLOCK_SET), and negated after the last block has been transferred. Note that the BUFWEN bit in CE_INT will not be asserted during DMA transfer.

If an error has occurred during DMA transfer or DMA transfer is forcibly terminated, the command sequence is stopped automatically, which causes the DMA transfer request to be negated.

23.5.2 DMA for Buffer Reading

The DMA transfer request is asserted for buffer reading when the buffer stores data of the block size specified in CE_BLOCK_SET while the DMAREN bit in CE_BUF_ACC is set to 1.

The DMA transfer request stays asserted for the amount of data specified by BLKSIZ (the block size set in CE_BLOCK_SET) × BLKCNT (the number of blocks for transfer set in CE_BLOCK_SET), and negated after the last block has been transferred. Note that the BUFREN bit in CE_INT will not be asserted during DMA transfer.

If an error has occurred during DMA transfer or DMA transfer is forcibly terminated, the command sequence is stopped automatically, which causes the DMA transfer request to be negated.

23.6 Operation

This section describes the formats of the command and response, timing of automatic CMD12 issuance, buffer structures, buffer access select function, and the operation when an error has occurred.

23.6.1 Command/Response Formats

Figure 23.2 shows the format of the command to be transferred. The command index that is set in CMD[5:0] of CE_CMD_SET and the argument set in ARG[31:0] of CE_ARG are reflected in the command.

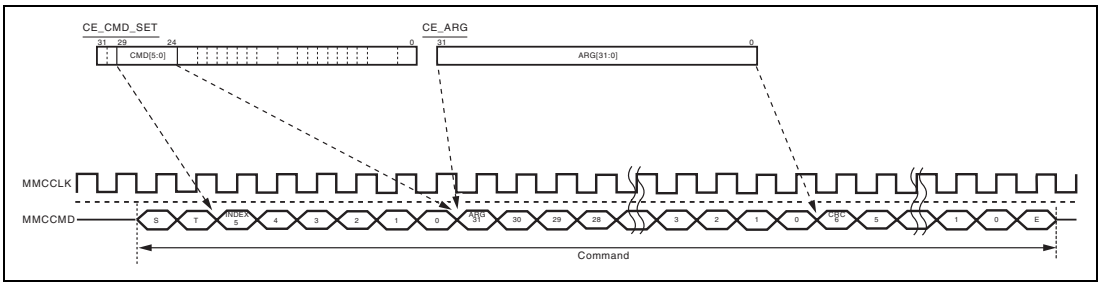


Figure 23.2 Command Format

Figures 23.3 and 23.4 show the formats when a 6-byte response and 17-byte response are received, respectively. The response index is stored in RSPIDX[5:0] of CE_HOST_STS1, and the status value of the response is stored to CE_RESP0 or CE_RESP3 to CE_RESP0.

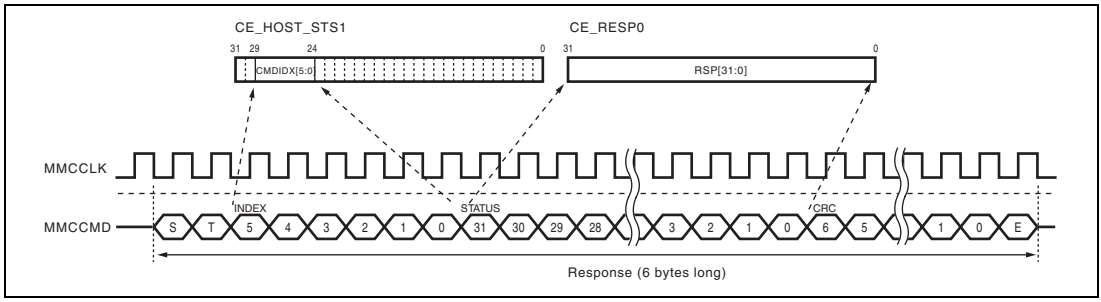


Figure 23.3 Format of 6-Byte Response

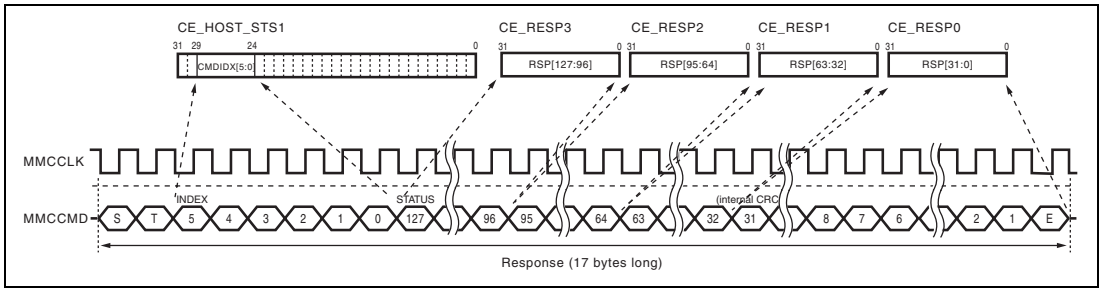


Figure 23.4 Format of 17-Byte Response (R2)

23.6.2 Automatic CMD12 Issuance

This module has the function that automatically issues CMD12 when multi-block transfer is performed with the CMD12EN in CE_CMD_SET set to 1. Timing of automatic CMD12 issuance is described for the case of multi-block reading and multi-block writing.

Figure 23.5 shows the timing of automatic CMD12 issuance in multi-block read. CMD12 is issued such that the end bit of the command is sent two bits before the end bit of the data during reception of the last block.

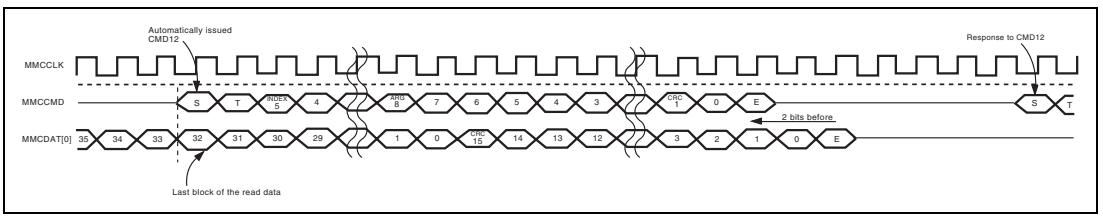


Figure 23.5 Timing of Automatically-Issued CMD12 in Multi-Block Read (1-Bit Mode)

Figure 23.6 shows the timing of automatic CMD12 issuance in multi-block write. CMD12 is issued after the data busy after transmission of the last block has ended.

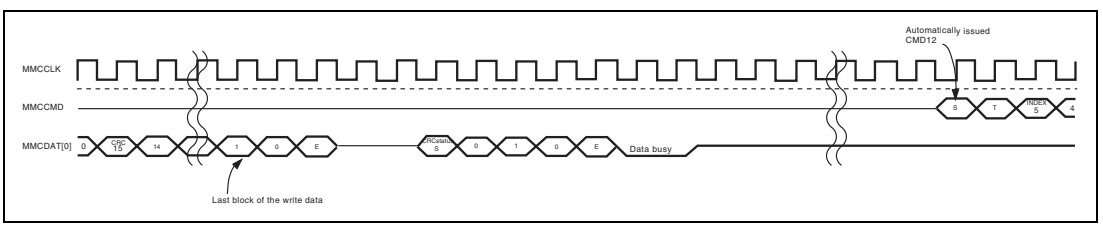


Figure 23.6 Timing of Automatically-Issued CMD12 in Multi-Block Write (1-Bit Mode)

23.6.3 Buffer Structure

This module has two 512-byte RAM units which are used for double buffering. If the transfer block size is set to 4 × n + 1 or 4 × n + 3, access should be made for 4 × n + 2 bytes or 4 × (n + 1) bytes in 16-bit access, and for 4 × (n + 1) bytes in 32-bit access. (n = 0, 1, 2, …, 127)

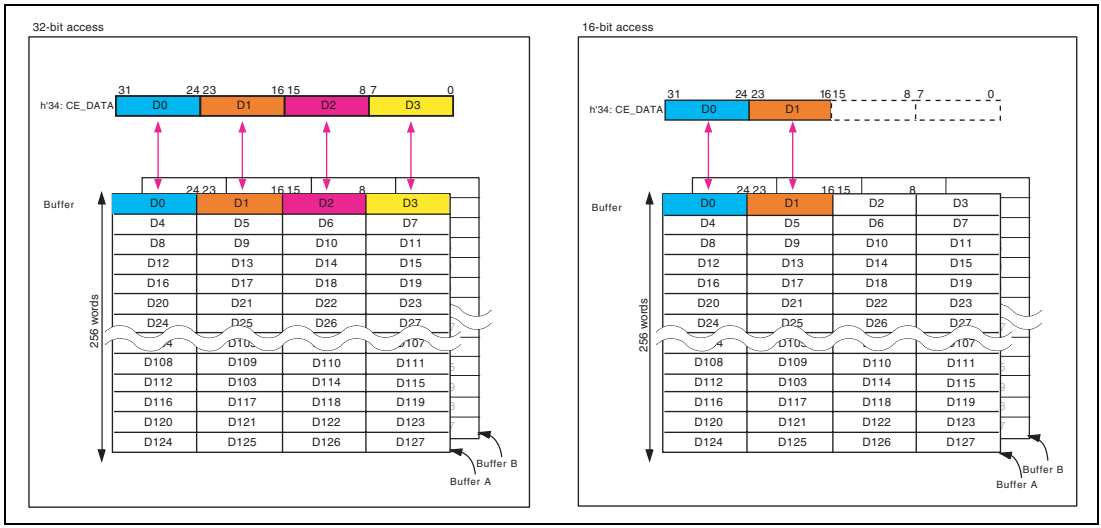


Figure 23.7 Double Buffer Structure

23.6.4 Buffer Access Select Function in Access to CE_DATA

This module has the buffer access select function that allows byte-wise swapping of data when the buffer is accessed by writing to or reading from CE_DATA. This function is enabled by the setting of CE_BUFF_ACC. Figure 23.8 shows the specification of 32-bit and 16-bit accesses.

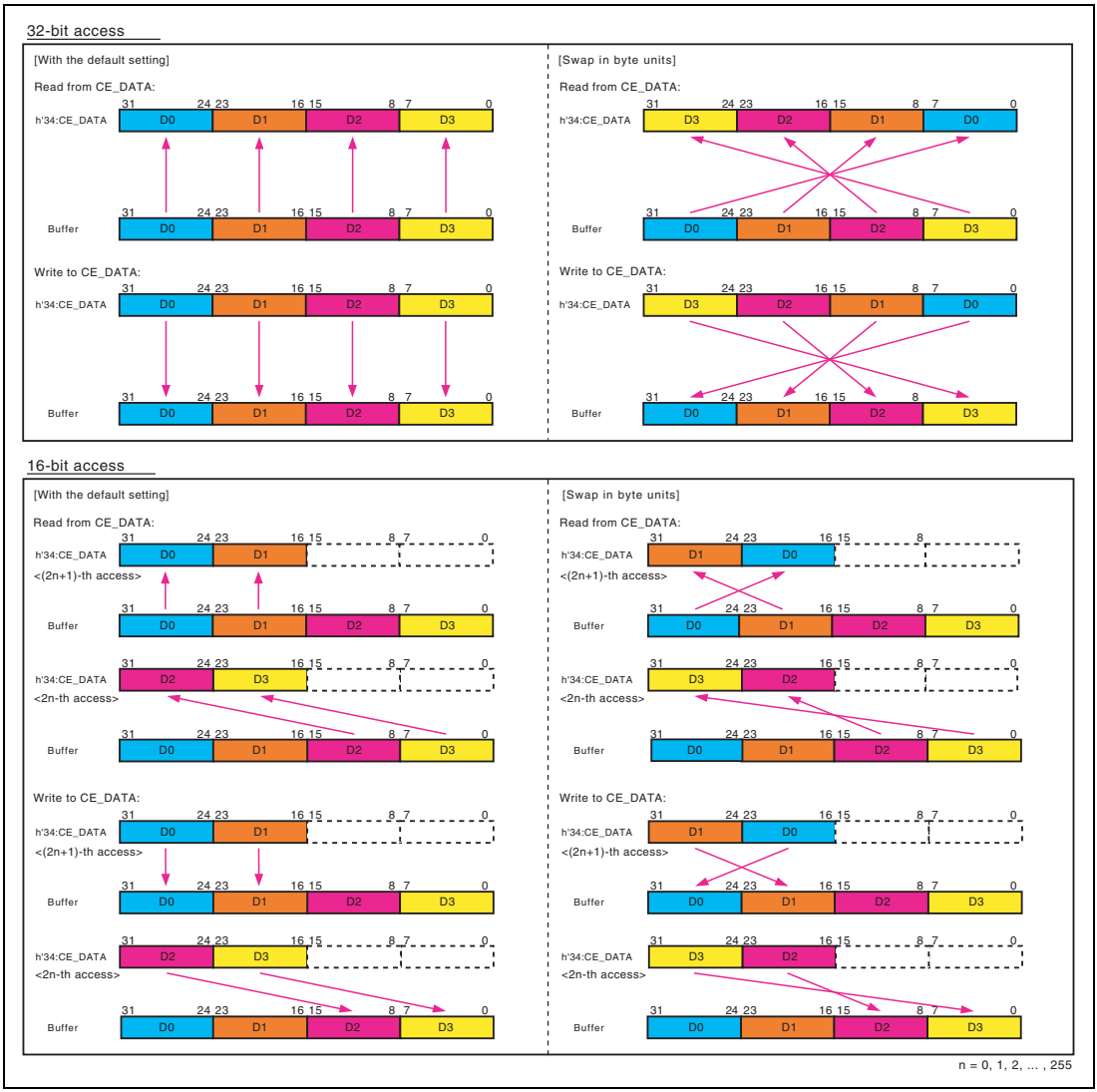


Figure 23.8 Specification of Byte-Swapping in 32/16-Bit Accesses

23.6.5 Data Formats

Figures 23.9 to 23.11 show the formats of data. In transmission, the value written to the buffer is reflected on the data lines, and in reception, the value of the received data is stored in the buffer.

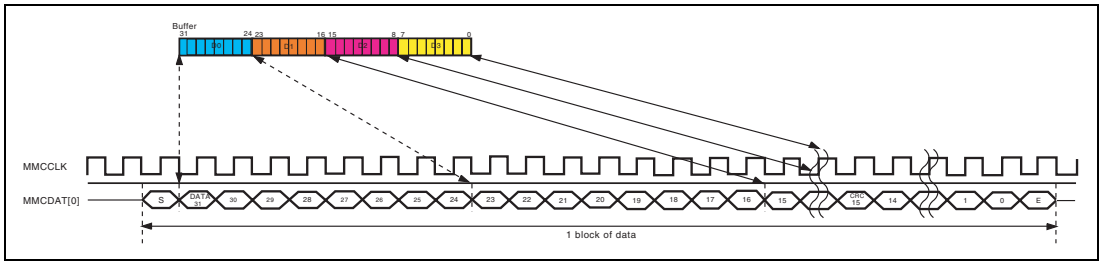


Figure 23.9 Data Format (1-Bit Mode)

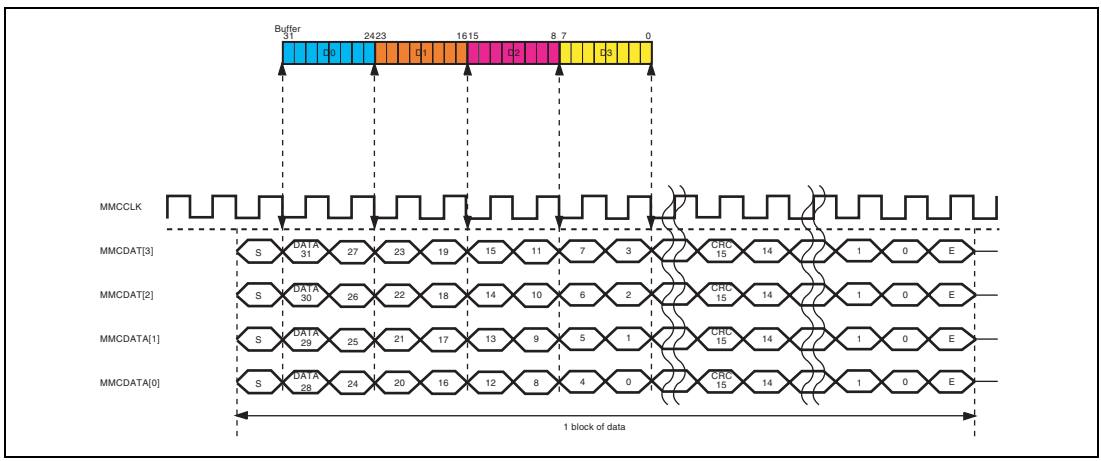


Figure 23.10 Data Format (4-Bit Mode)

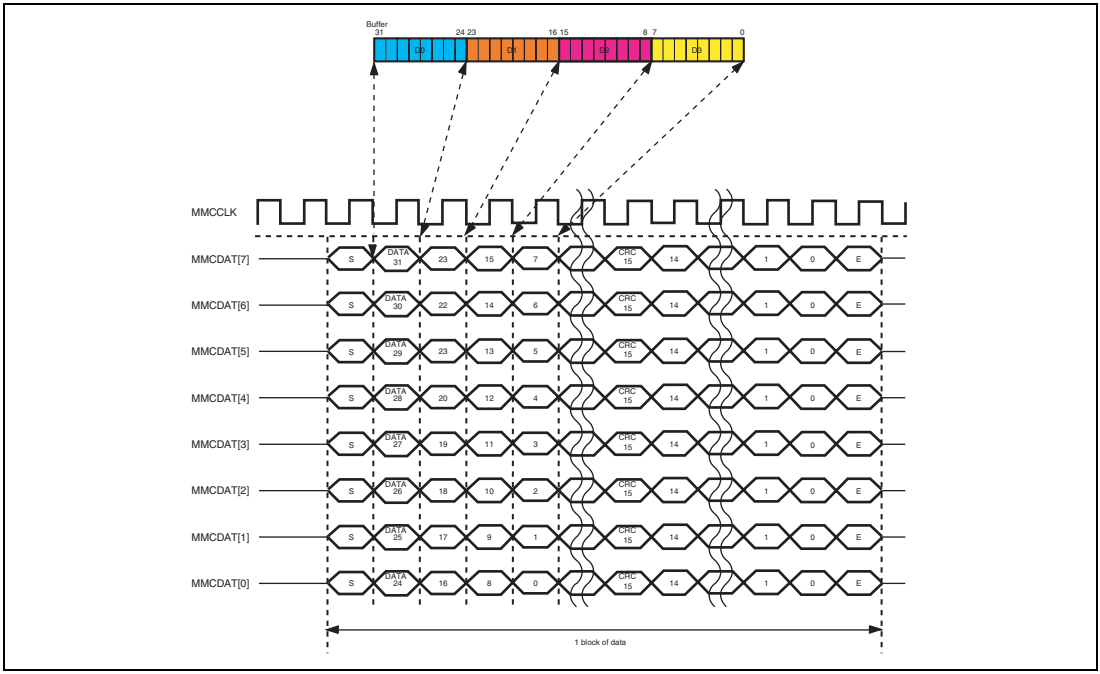


Figure 23.11 Data Format (8-Bit Mode)

23.6.6 Operation in the Case of Error/Timeout

The data for transmission or received data that had been stored in the buffers at the time of error occurrence are not guaranteed. After the error is recognized, check status register 1 and if the command sequence is still in progress, terminate it forcibly. Then, initialize the module and execute the command sequence again.

This module is not stopped when a timeout has occurred. If the command sequence does not end normally when the timeout has occurred and the flag in the status register 1 is still indicating that the command sequence is in progress, terminate the sequence forcibly and initialize the module.

For forcible termination, refer to section 23.8, Usage Notes.

23.7 Examples of Setting

This section shows the procedures for executing typical command sequences.

23.7.1 Legends

Figure 23.12 shows the legends for the symbols used in the figures in the following subsections.

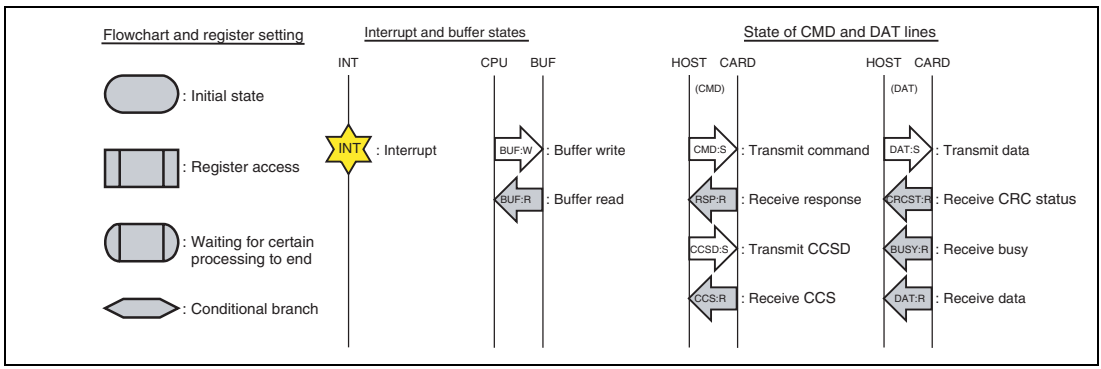


Figure 23.12 Legends for the Symbols Used in the Figures

23.7.2 Command Transmission

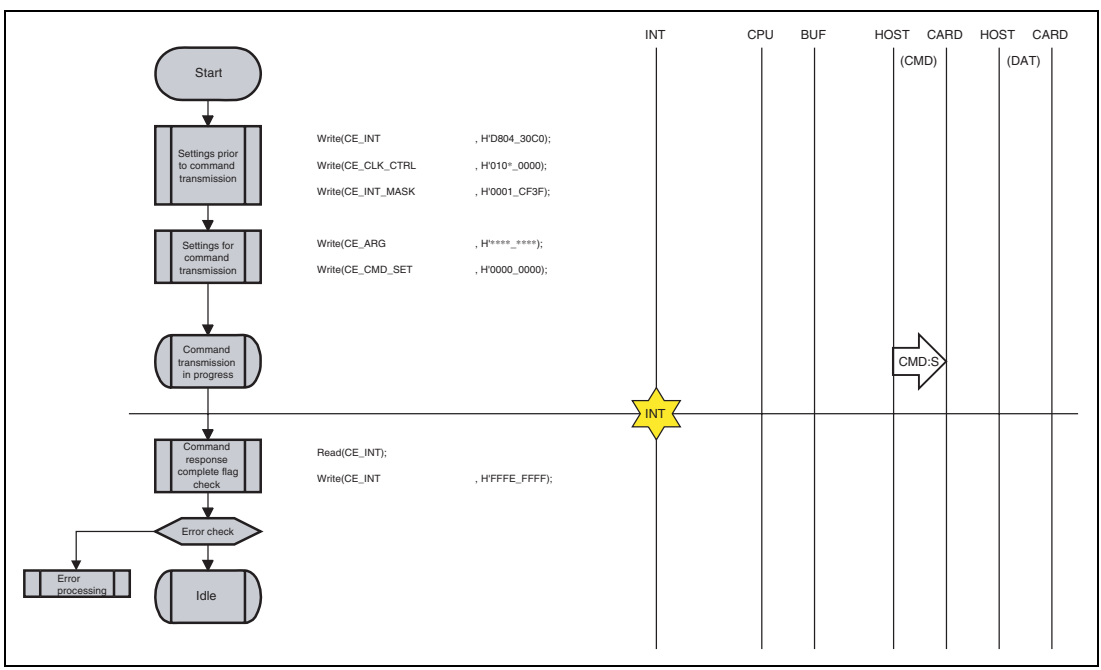


Figure 23.13 Command Transmission (CMD0)

23.7.3 Command Transmission → Response Reception

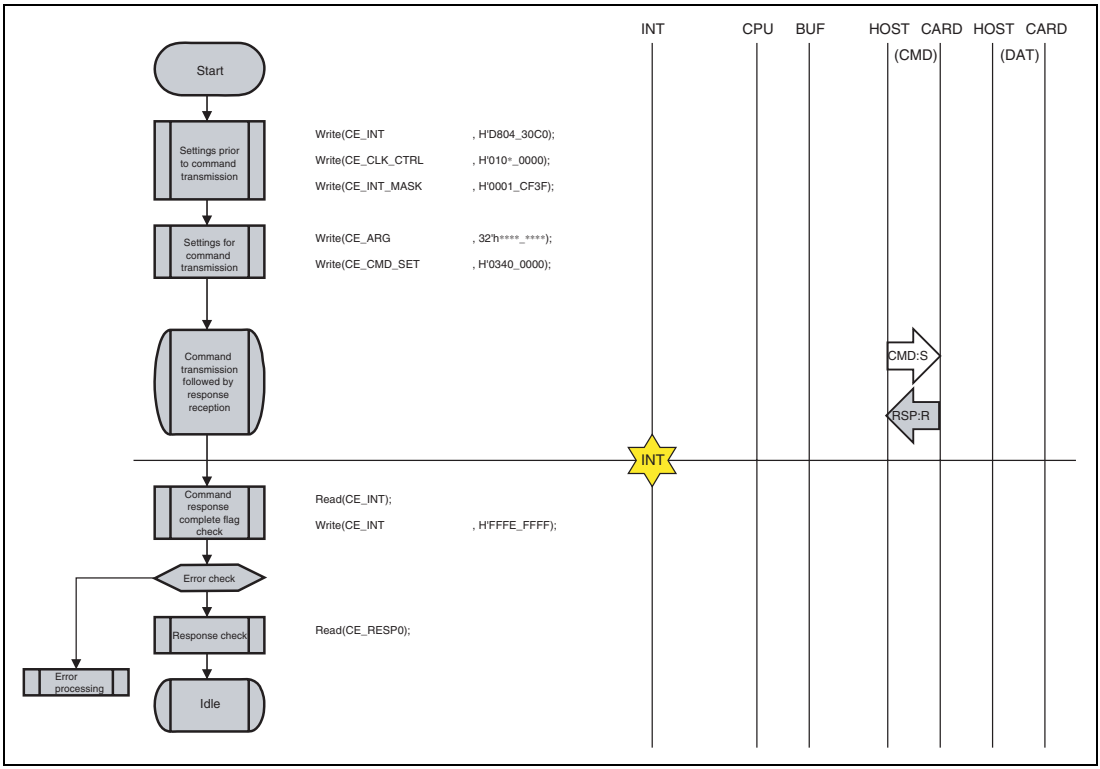


Figure 23.14 Command Transmission → Response Reception (CMD3)

23.7.4 Command Transmission → Response Reception (with Response Busy)

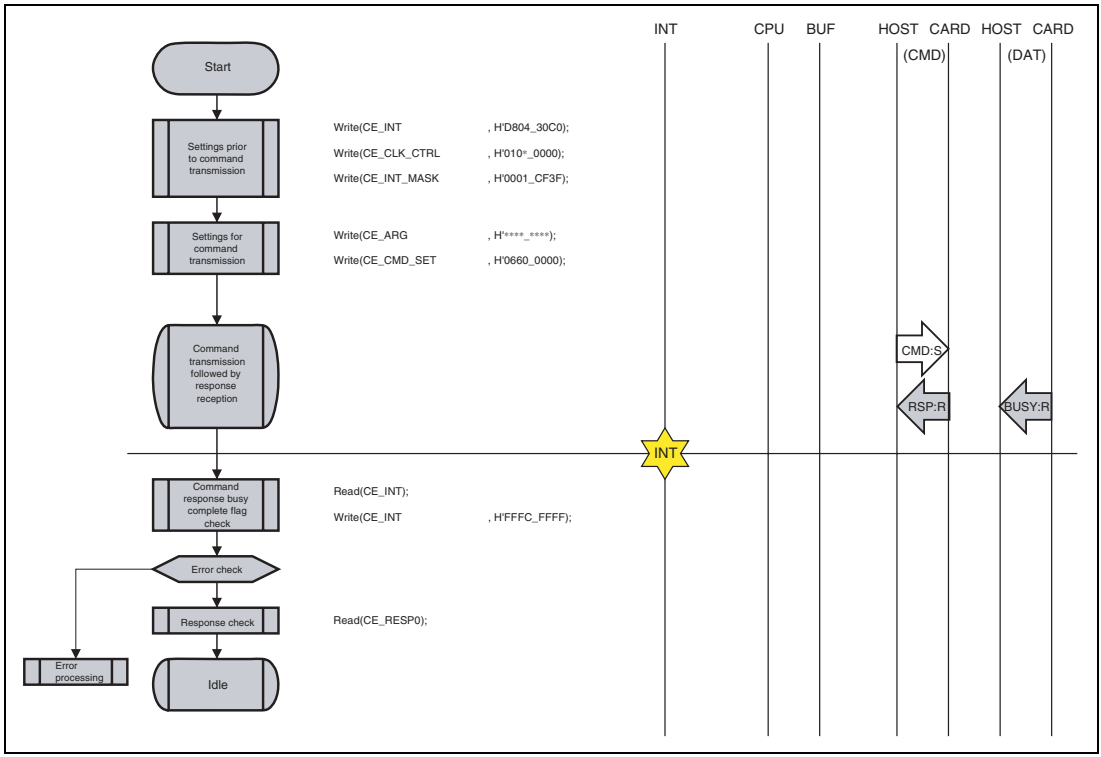


Figure 23.15 Command Transmission → Response Reception (with Response Busy) (CMD6)

23.7.5 Single-Block Read

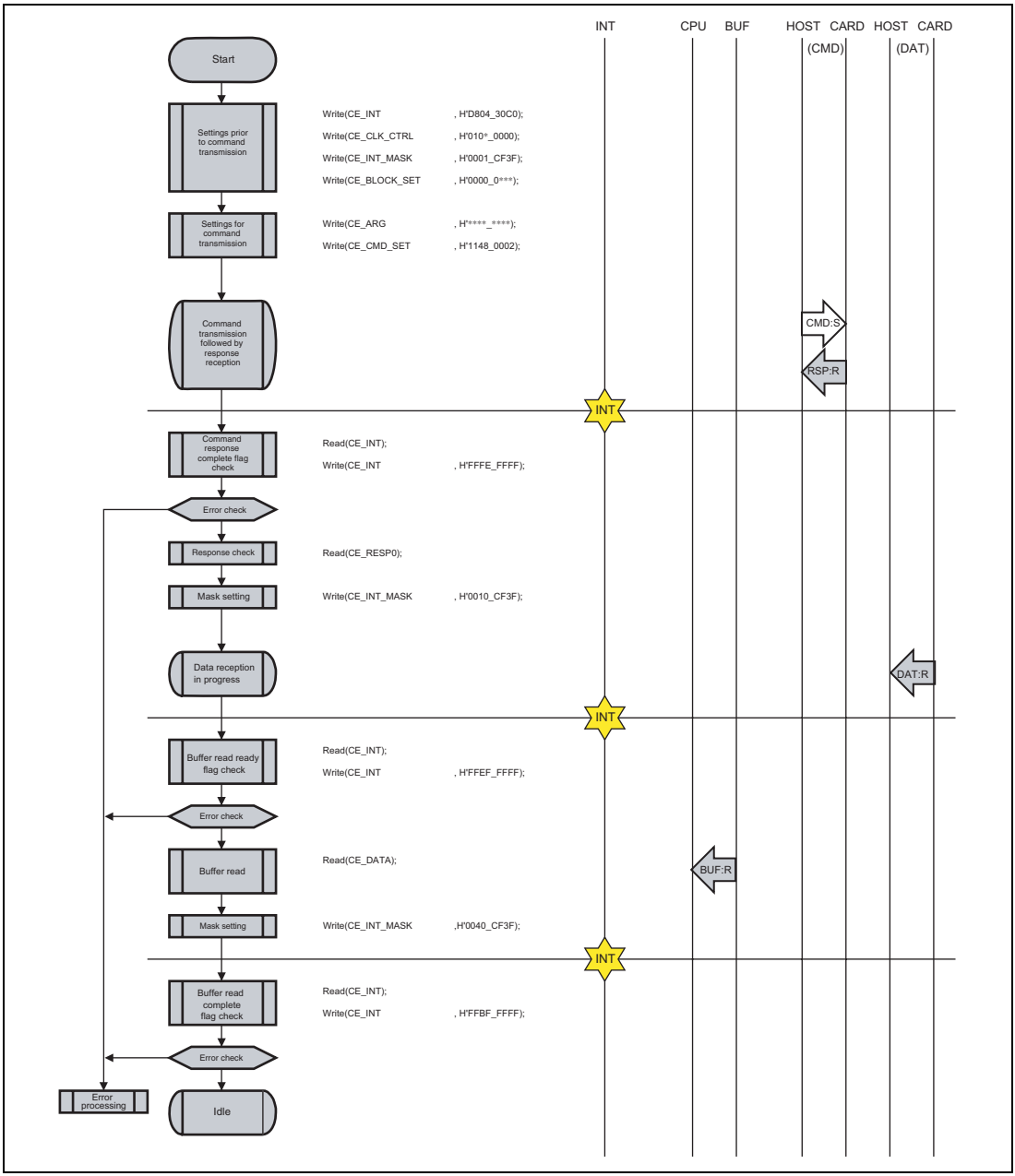


Figure 23.16 Single-Block Read (CMD17)

23.7.6 Multi-Block Read (with Automatic CMD12 Issuance)

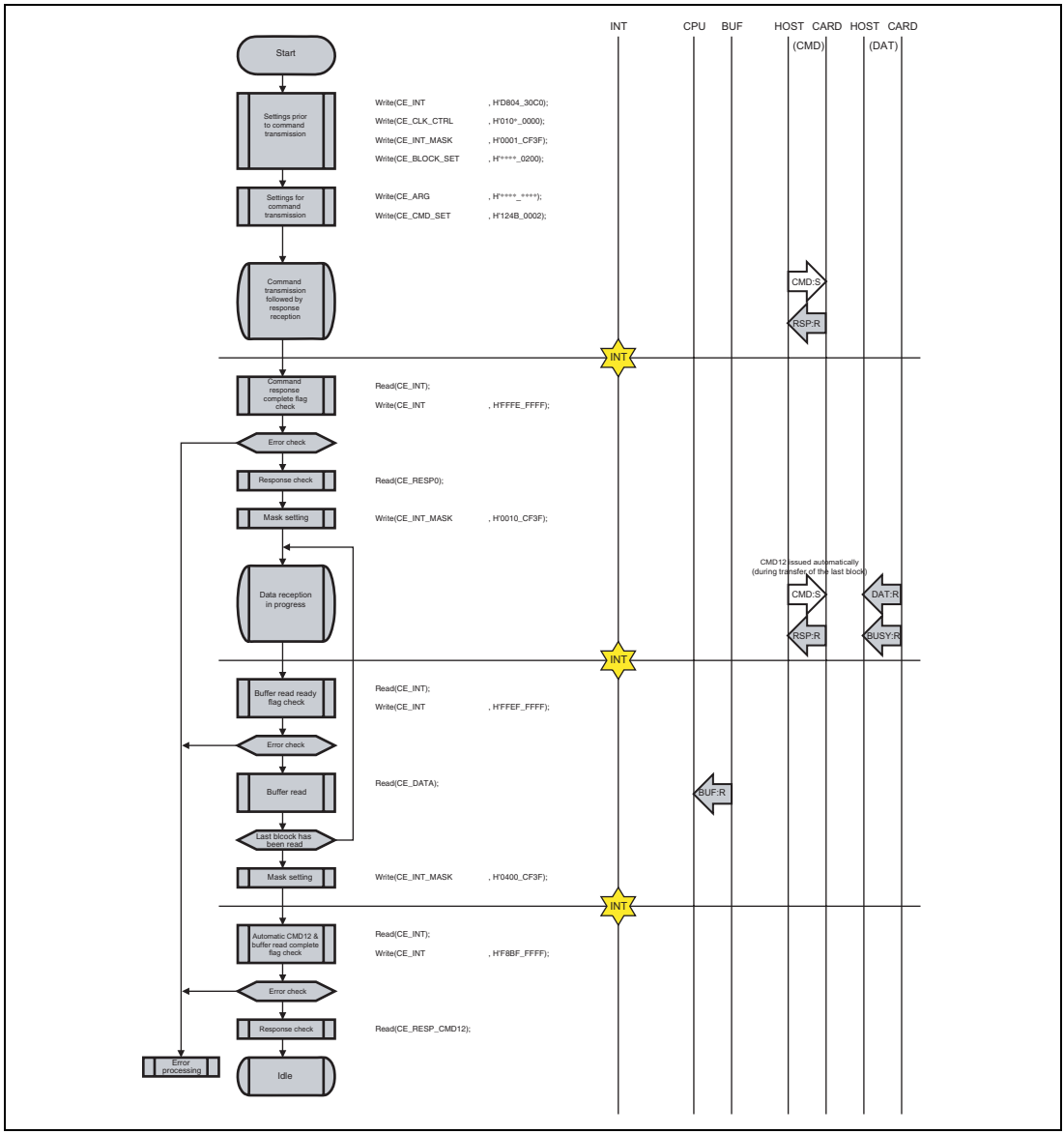


Figure 23.17 Multi-Block Read (with Automatic CMD12 Issuance) (CMD18)

23.7.7 Single-Block Write

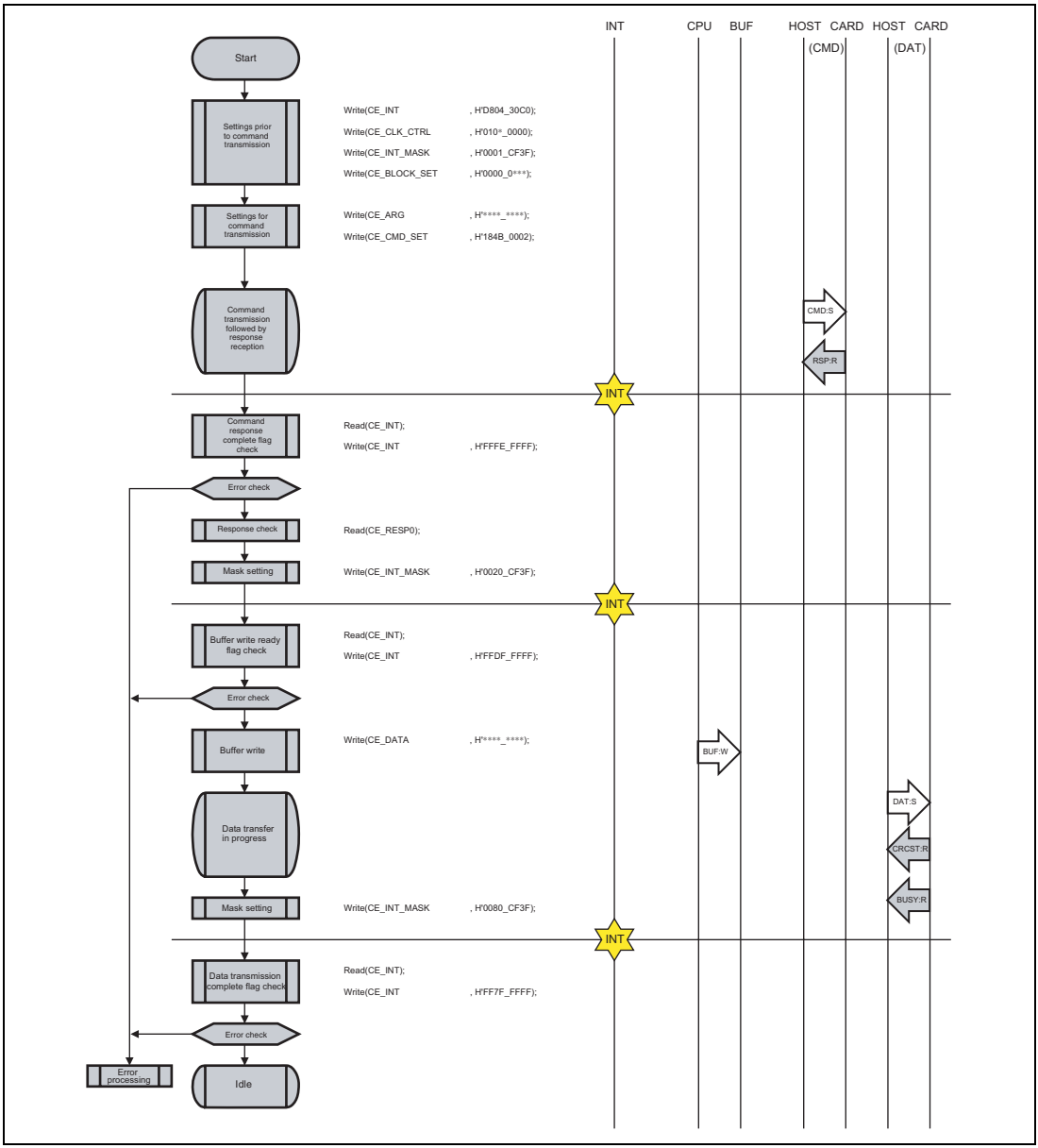


Figure 23.18 Single-Block Write (CMD24)

23.7.8 Multi-Block Write (with Automatic CMD12 Issuance)

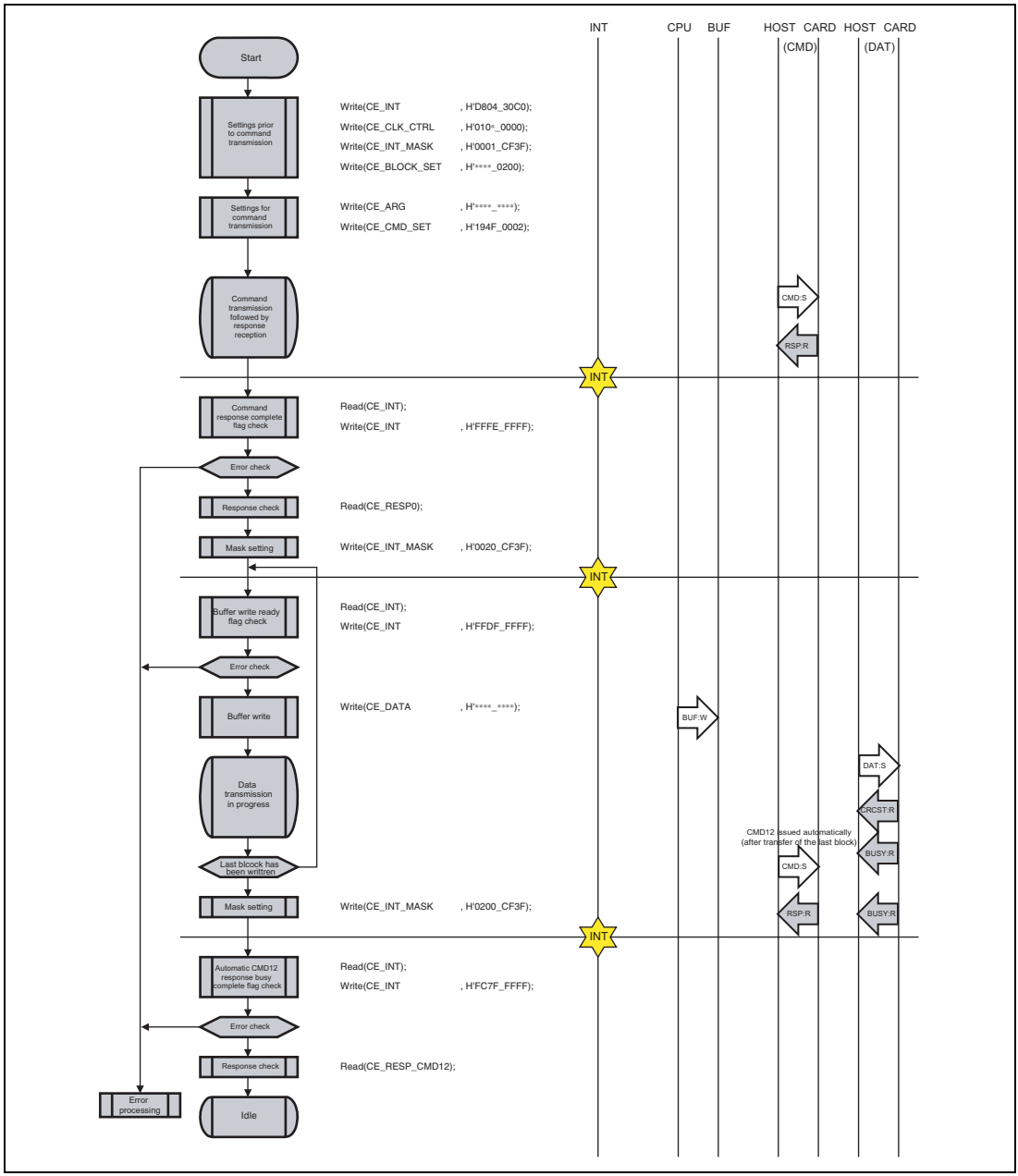


Figure 23.19 Multi-Block Write (with Automatic CMD12 Issuance) (CMD25)

23.7.9 Forcible Termination

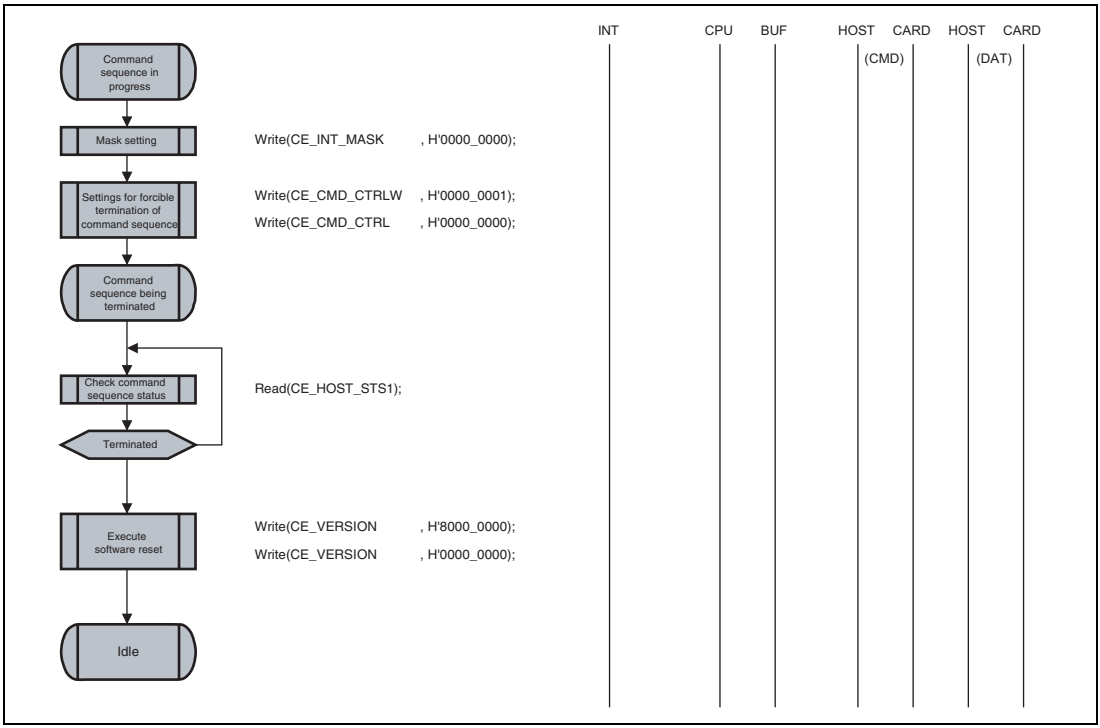


Figure 23.20 Forcible Termination

23.7.10 Command Transmission → Response Reception (with Response Busy and CCS)

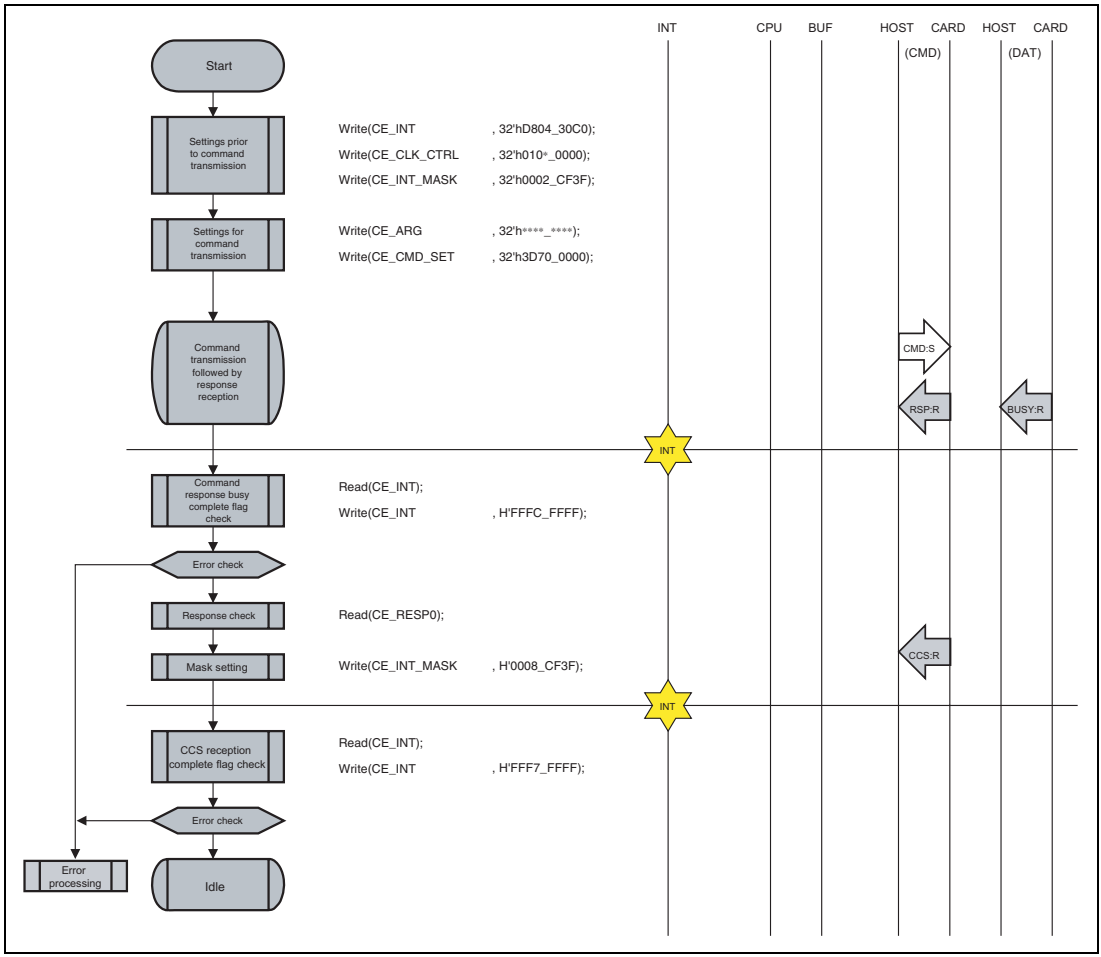


Figure 23.21 Command Transmission → Response Reception (with Response Busy and CCS) (CMD61)

23.7.11 Multi-Block Read (with CCS)

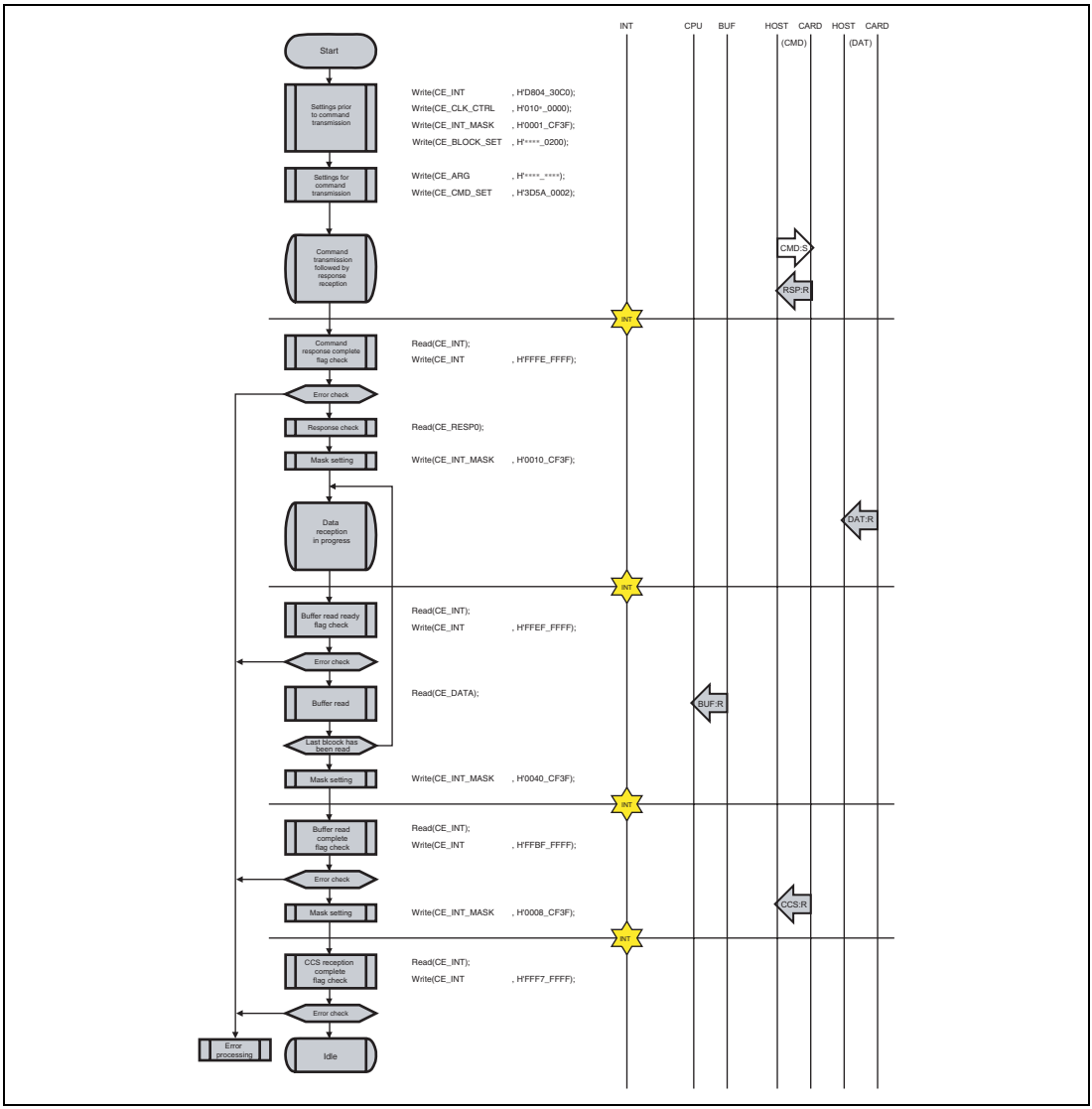


Figure 23.22 Multi-Block Read (with CCS) (CMD61)

23.7.12 Multi-Block Write (with Response Busy and CCS)

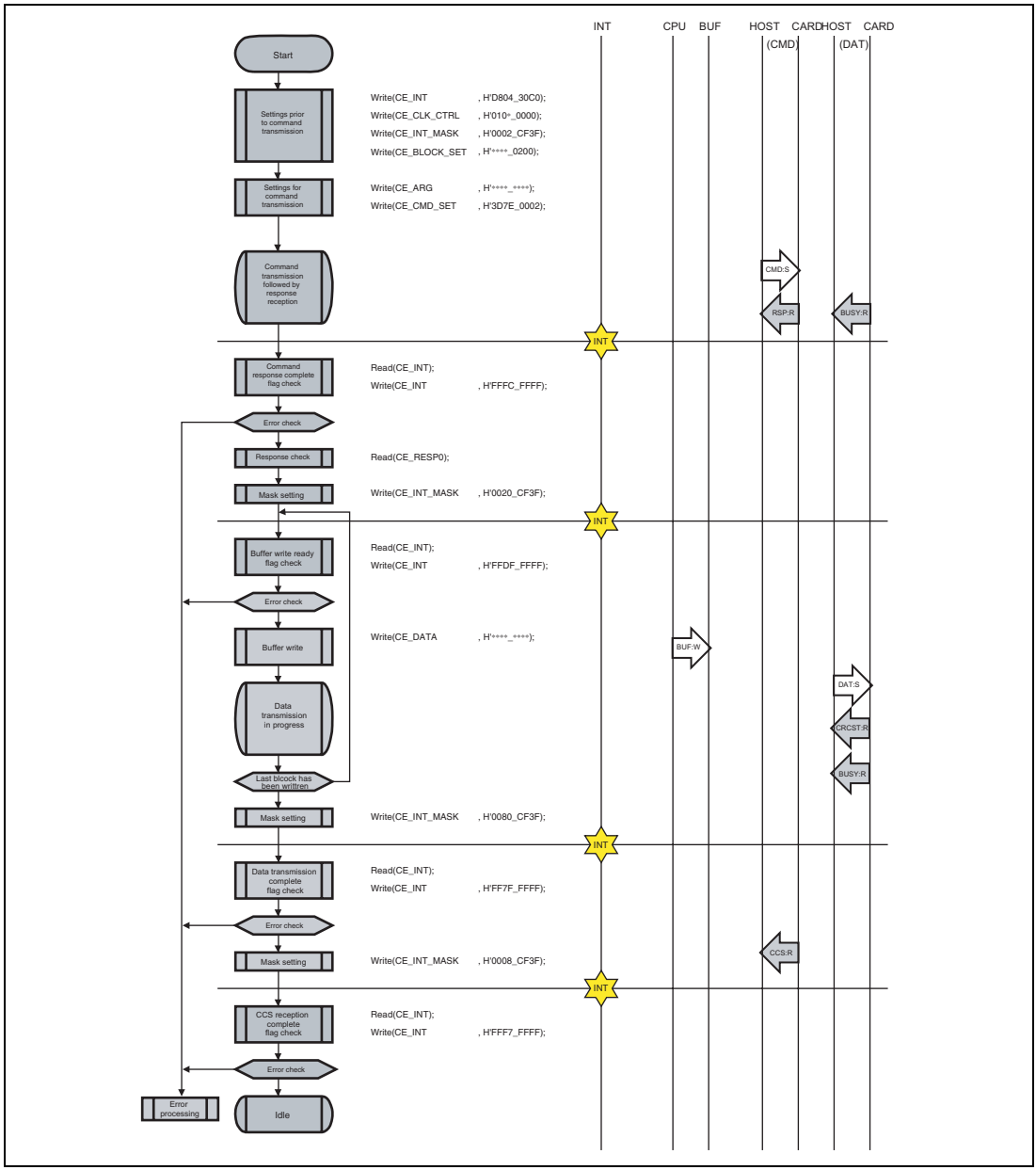


Figure 23.23 Multi-Block Write (with Response Busy and CCS) (CMD61)

23.7.13 Forcible Termination → Transmission of CCSD

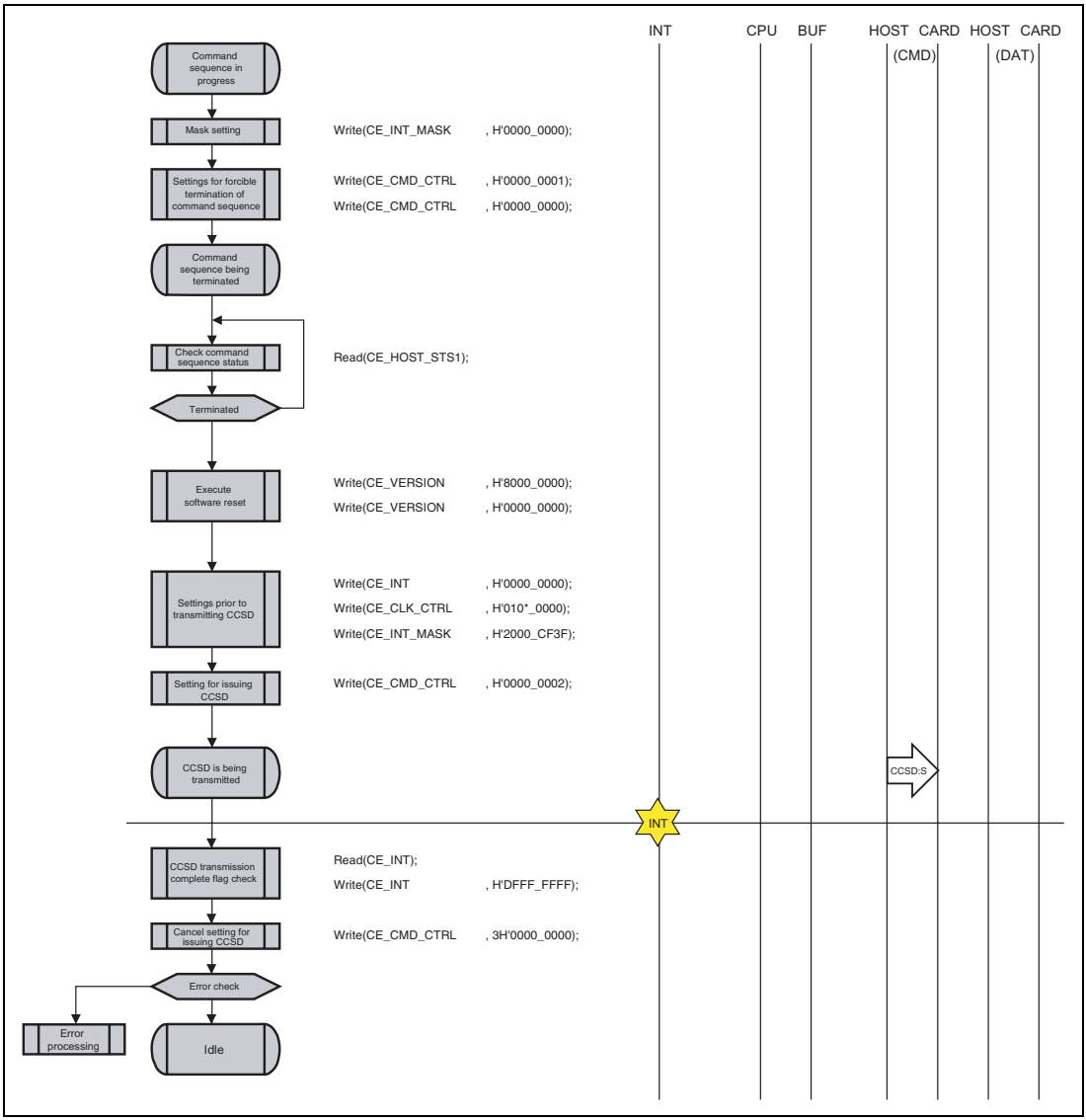


Figure 23.24 Forcible Termination → Transmission of CCSD

23.7.14 Setting Values of CE_CMD_SET

Table 23.5 lists the setting values required to issue commands.

Table 23.5 Setting Values of CE_CMD_SET

MMC/ CE-ATA	CMD	Abbreviation	RSP	CE_CMD_SET														
				CMD[5:0]	RTYP[1:0]	RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	CRC16C	CRCSTE	TBIT	OPDM	CCSH	DATW
MMC	CMD0	GO_IDLE_STATE	-	6'h00	2'b00	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD1	SEND_OP_COND	R3	6'h01	2'b01	0	0	0	0	0	0	2'b01	2'b01	0	0	0	0	0	2'b00
	CMD2	ALL_SEND_CID	R2	6'h02	2'b10	0	0	0	0	0	0	2'b01	2'b10	0	0	0	0	0	2'b00
	CMD3	SET_RELATIVE_ADDR	R1	6'h03	2'b00	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD4	SET_DSR	-	6'h04	2'b00	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD6	SWITCH	R1b	6'h06	2'b01	1	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD7	SELECT/DESETECT_CARD (dis->prg)	R1	6'h07	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
			R1b	6'h07	2'b01	1	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD8	SEND_EXT_CSD	R1	6'h08	2'b01	0	0	1	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD9	SEND_CSD	R2	6'h09	2'b10	0	0	0	0	0	0	2'b01	2'b10	0	0	0	0	0	2'b00
	CMD10	SEND_CID	R2	6'h0A	2'b10	0	0	0	0	0	0	2'b01	2'b10	0	0	0	0	0	2'b00
	CMD12	STOP_TRANSMISSION	R1b	6'h0C	2'b01	1	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD13	SEND_STATUS	R1	6'h0D	2'b01	0	0	0	0	0	0	2'b01	2'b00	0	0	0	0	0	2'b00
	CMD14	BUSTEST_R	R1	6'h0E	2'b01	0	0	1	0	0	0	2'b00	2'b00	1	0	0	0	0	2'b**
	CMD15	GO_INACTIVE_STATE	-	6'h0F	2'b00	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD16	SET_BLOCKLEN	R1	6'h10	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD17	READ_SINGLE_BLOCK	R1	6'h11	2'b01	0	0	1	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD18	READ_MULTIPLE_BLOCK (Open-ended)	R1	6'h12	2'b01	0	0	1	0	1	0	2'b00	2'b00	0	0	0	0	0	2'b**
			R1	6'h12	2'b01	0	0	1	0	1	1	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD19	BUSTEST_W	R1	6'h13	2'b01	0	0	1	1	0	0	2'b00	2'b00	0	1	0	0	0	2'b**
	CMD23	SET_BLOCK_COUNT	R1	6'h17	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD24	WRITE_BLOCK	R1	6'h18	2'b01	0	0	1	1	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD25	WRITE_MULTIPLE_BLOCK (Open-ended)	R1	6'h19	2'b01	0	0	1	1	1	0	2'b00	2'b00	0	0	0	0	0	2'b**
			R1	6'h19	2'b01	0	0	1	1	1	1	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD26	PROGRAM_CID	R1	6'h1A	2'b01	0	0	1	1	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD27	PROGRAM_CSD	R1	6'h1B	2'b01	0	0	1	1	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD28	SET_WRITE_PROT	R1b	6'h1C	2'b01	1	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD29	CLR_WRITE_PROT	R1b	6'h1D	2'b01	1	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD30	SEND_WRITE_PROT	R1	6'h1E	2'b01	0	0	1	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b**

MMC/ CE-ATA	CMD	Abbreviation	RSP	CE_CMD_SET														
				CMD[5:0]	RTYP[1:0]	RBSY	CCSEN	WDAT	DWEN	CMLTE	CMD12EN	RIDXC	RCRC7C	CRC16C	CRCSTE	TBIT	OPDM	CCSH	DATW
MMC	CMD35	ERASE_GROUP_START	R1	6'h23	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD36	ERASE_GROUP_END	R1	6'h24	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD38	ERASE	R1b	6'h26	2'b01	1	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD39	FAST_IO	R4	6'h27	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD40	GO_IRQ_STATE (send CMD)	R5	6'h28	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
		(send RSP)	R5	6'h28	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	1	1	0	2'b00
	CMD42	LOCK_UNLOCK	R1	6'h2A	2'b01	0	0	1	1	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD55	APP_CMD	R1	6'h37	2'b01	0	0	0	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b00
	CMD56	GEN_CMD (R)	R1b	Not supported															
		(W)	R1b	6'h38	2'b01	1	0	1	1	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
CE-ATA	CMD60	RW_MULTIPLE_REGISTER (R)	R1	6'h3C	2'b01	0	0	1	0	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
		(W)	R1b	6'h3C	2'b01	1	0	1	1	0	0	2'b00	2'b00	0	0	0	0	0	2'b**
	CMD61	RW_MULTIPLE_BLOCK(no dat)	R1b	6'h3D	2'b01	1	1	0	0	0	0	2'b00	2'b00	0	0	0	0	1	2'b**
		(R)	R1	6'h3D	2'b01	0	1	1	0	1	0	2'b00	2'b00	0	0	0	0	1	2'b**
		(W)	R1b	6'h3D	2'b01	1	1	1	1	1	0	2'b00	2'b00	0	0	0	0	1	2'b**

Note: This module does not support CMD11 and CMD20.

23.8 Usage Notes.

23.8.1 Forcible Termination

Forcible termination of a command sequence for the MMC/CE-ATA host interface by setting the BREAK bit in the CE_CMD_CTRL register may fail. This depends on the timing with which the bit is set. Therefore, when a command sequence is to be terminated forcibly, one of the two procedures below should be followed.

1. When a command sequence is to be forcibly terminated, issue a software reset instead of using the BREAK bit in the CE_CMD_CTRL register.
2. After the command for an R1b response has been issued and only when the command sequence is to be forcibly terminated in one of the states below, issue a software reset instead of using the BREAK bit in the CE_CMD_CTRL register.
   — Response busy timeout
   — Abnormal response value

Section 24 Flash Memory Boot ROM (FBR)

This LSI has the boot ROM, and it boots from the NAND flash memory that includes MMC controller by releasing a reset with the BOOT pin driven high.

24.1 Features

The boot ROM includes the boot program, which controls the MMCIF.

The boot program transfers the loader program to the ILRAM in this LSI, via the MMCIF module.

It jumps to the start address of the loader program after the transfer completion to the ILRAM by the boot program.

Figure 24.1 shows the overview of flash memory boot.

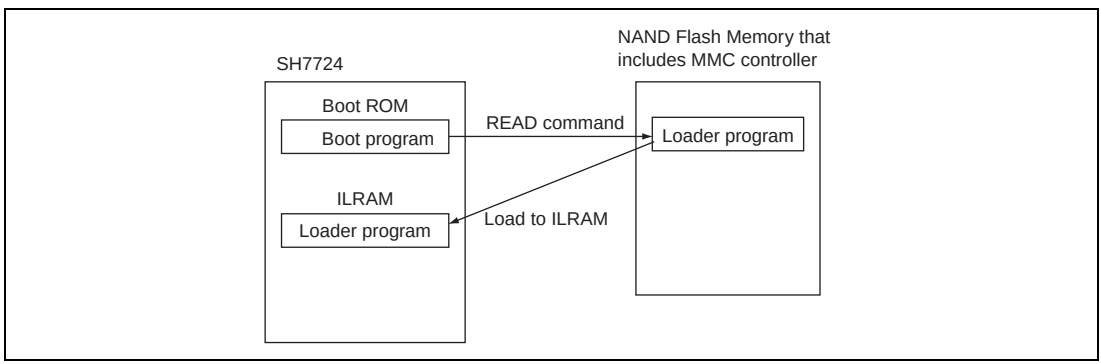


Figure 24.1 Overview of Flash Memory Boot

24.2 Input/Output Pins

Table 24.1 shows the pin configuration.

Table 24.1 Pin Configuration

Pin Name	Function	I/O	Description
BOOT	—	Input	Boots from the boot ROM. H: NAND flash memory boot via MMCIF L: Normal boot

Note: Refer to section 23, Multi Media Card Interface (MMCIF) for MMCIF pins.

24.3 Overview of Boot Program

Figure 24.2 shows the overview of the boot program.

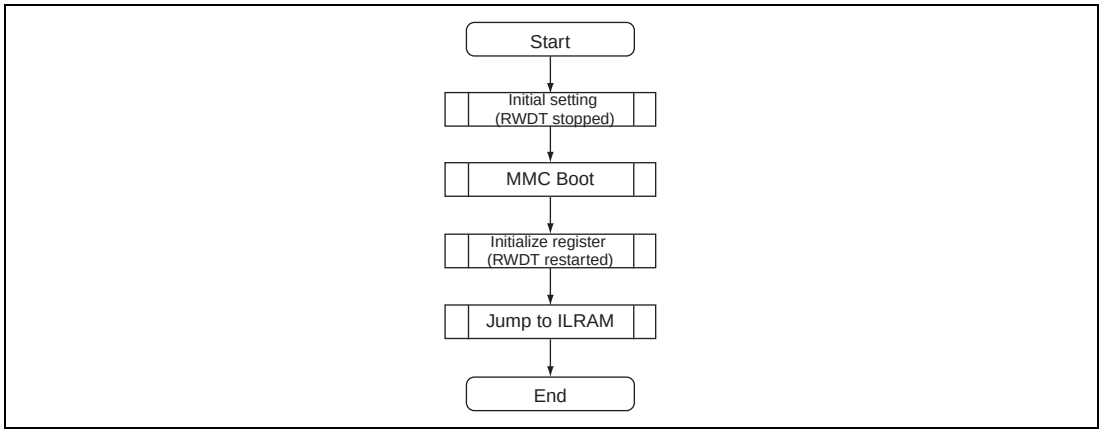


Figure 24.2 Overview of the Boot Program

Section 25 Clock-Synchronized Serial Interface with FIFO (MSIOF)

This LSI includes two-channel of clock-synchronized serial I/O module with FIFO (MSIOF0, MSIOF1).

25.1 Features

- FIFO capacity: 32 bits × 64 stages for transmission and 32 bits × 64 stages for reception
- MSB first or LSB first selectable for data transmission and reception
- Synchronization by frame synchronization pulse, level, or left/right channel switch
- Supports both master and slave modes
- Independent clock and synchronization signals for transmission and reception (common clock and synchronization signals are also selectable)
- Supports multiple-channel communication
  - Transfers multiple groups or words of data in one frame.
  - The word data size for each group can be selected from eight to 32 bits
  - Up to 256 words can be transferred in each group when one or two groups are used, or up to 16 words can be transferred in each group when three or four groups are used.
- Interrupts: One type in each channel
- Serial clock

  The internal clock (Bϕ) or external pin input (MSIOF0_MCK/MSIOF1_MCK) can be selected as the clock source.
- DMA transfer

  Supports DMA transfer by a transfer request for transmission and reception
- Serial format

  Supports serial format such as IIS, SPI (both master and slave modes), and μWIRE.

Figure 25.1 shows a block diagram of the MSIOF.

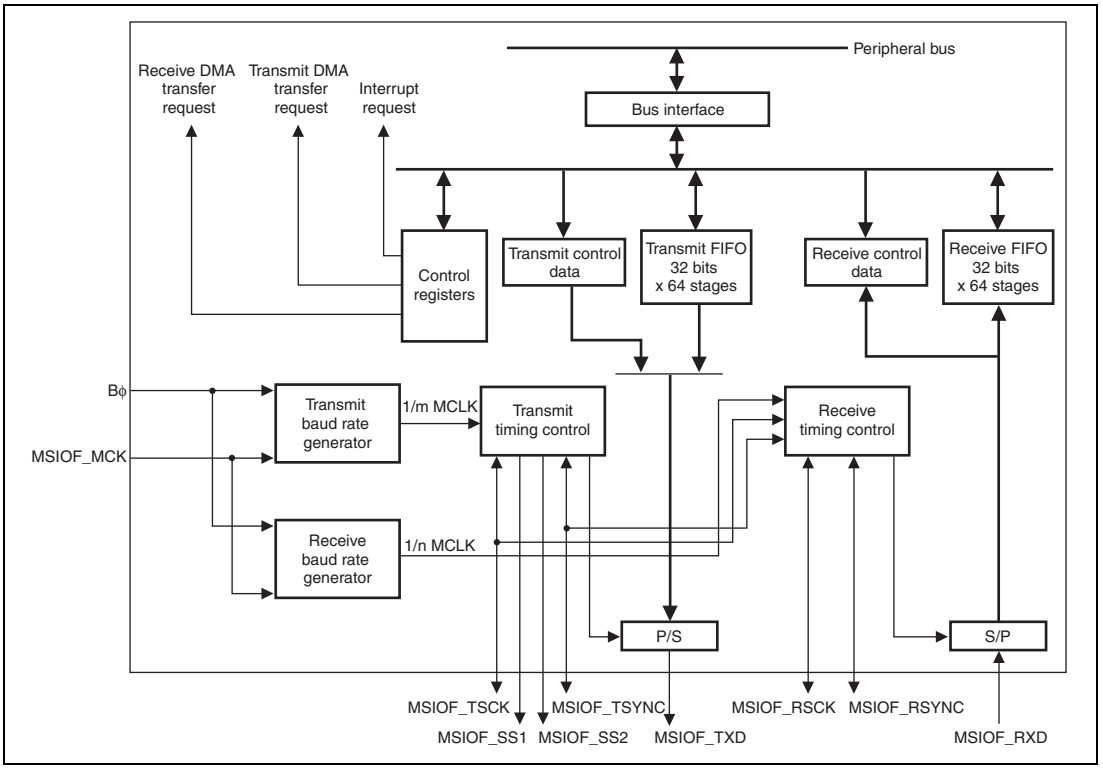


Figure 25.1 Block Diagram of MSIOF

25.2 Input/Output Pins

The pin configuration in this module is shown in Table 25.1.

Table 25.1 Pin Configuration

Pin Name	Abbreviation*	I/O	Function
MSIOF0_MCK MSIOF1_MCK	MSIOFMCK	Input	Master clock input
MSIOF0_TSCK MSIOF1_TSCK	MSIOFTSCK (SCK)	I/O	Serial clock for transmission Works as SCK when a common clock is used for transmission and reception.
MSIOF0_TSYNC MSIOF1_TSYNC	MSIOFTSYNC ($\overline{SS0}$)	I/O	Frame synchronization signal channel 0 for transmission Works as SYNC when a common synchronization signal is used for transmission and reception.
MSIOF0_SS1 MSIOF1_SS1	MSIOFSS1 ($\overline{SS1}$)	Output	Frame synchronization signal channel 1 for transmission Only the slave device can select this signal.
MSIOF0_SS2 MSIOF1_SS2	MSIOFSS2 ($\overline{SS2}$)	Output	Frame synchronization signal channel 2 for transmission Only the slave device can select this signal.
MSIOF0_RSCK MSIOF1_RSCK	MSIOFRSCK	I/O	Serial clock for reception
MSIOF0_RSYNC MSIOF1_RSYNC	MSIOFRSYNC	I/O	Frame synchronization signal for reception
MSIOF0_TXD MSIOF1_TXD	MSIOFTXD (MOSI/MISO)	Output	Transmit data
MSIOF0_RXD MSIOF1_RXD	MSIOFRXD (MISO/MOSI)	Input	Receive data

Note: * In SPI mode, the pins are called SCK, $\overline{SS0}$, $\overline{SS1}$, $\overline{SS2}$, MOSI, and MISO.

25.3 Register Descriptions

Table 25.2 shows the MSIOF register configuration and table 25.3 shows the register state in each processing mode.

Table 25.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
MSIOF0 Transmit mode register 1	MSIOF0_SITMDR1	R/W	H'A4C40000	32
MSIOF0 Transmit mode register 2	MSIOF0_SITMDR2	R/W	H'A4C40004	32
MSIOF0 Transmit mode register 3	MSIOF0_SITMDR3	R/W	H'A4C40008	32
MSIOF0 Receive mode register 1	MSIOF0_SIRMDR1	R/W	H'A4C40010	32
MSIOF0 Receive mode register 2	MSIOF0_SIRMDR2	R/W	H'A4C40014	32
MSIOF0 Receive mode register 3	MSIOF0_SIRMDR3	R/W	H'A4C40018	32
MSIOF0 Transmit clock select register	MSIOF0_SITSCR	R/W	H'A4C40020	16
MSIOF0 Receive clock select register	MSIOF0_SIRSCR	R/W	H'A4C40022	16
MSIOF0 Control register	MSIOF0_SICTR	R/W	H'A4C40028	32
MSIOF0 FIFO control register	MSIOF0_SIFCTR	R/W	H'A4C40030	32
MSIOF0 Status register	MSIOF0_SISTR	R/W	H'A4C40040	32
MSIOF0 Interrupt enable register	MSIOF0_SIIER	R/W	H'A4C40044	32
MSIOF0 Transmit control data register 1	MSIOF0_SITDR1	W	H'A4C40048	32
MSIOF0 Transmit control data register 2	MSIOF0_SITDR2	W	H'A4C4004C	32
MSIOF0 Transmit FIFO data register	MSIOF0_SITFDR	W	H'A4C40050	32
MSIOF0 Receive control data register 1	MSIOF0_SIRDR1	R	H'A4C40058	32
MSIOF0 Receive control data register 2	MSIOF0_SIRDR2	R	H'A4C4005C	32
MSIOF0 Receive FIFO data register	MSIOF0_SIRFDR	R	H'A4C40060	32
MSIOF1 Transmit mode register 1	MSIOF1_SITMDR1	R/W	H'A4C50000	32
MSIOF1 Transmit mode register 2	MSIOF1_SITMDR2	R/W	H'A4C50004	32
MSIOF1 Transmit mode register 3	MSIOF1_SITMDR3	R/W	H'A4C50008	32
MSIOF1 Receive mode register 1	MSIOF1_SIRMDR1	R/W	H'A4C50010	32
MSIOF1 Receive mode register 2	MSIOF1_SIRMDR2	R/W	H'A4C50014	32
MSIOF1 Receive mode register 3	MSIOF1_SIRMDR3	R/W	H'A4C50018	32
MSIOF1 Transmit clock select register	MSIOF1_SITSCR	R/W	H'A4C50020	16
MSIOF1 Receive clock select register	MSIOF1_SIRSCR	R/W	H'A4C50022	16
MSIOF1 Control register	MSIOF1_SICTR	R/W	H'A4C50028	32

Register Name	Abbreviation	R/W	Address	Access Size
MSIOF1 FIFO control register	MSIOF1_SIFCTR	R/W	H'A4C50030	32
MSIOF1 Status register	MSIOF1_SISTR	R/W	H'A4C50040	32
MSIOF1 Interrupt enable register	MSIOF1_SIIER	R/W	H'A4C50044	32
MSIOF1 Transmit control data register 1	MSIOF1_SITDR1	W	H'A4C50048	32
MSIOF1 Transmit control data register 2	MSIOF1_SITDR2	W	H'A4C5004C	32
MSIOF1 Transmit FIFO data register	MSIOF1_SITFDR	W	H'A4C50050	32
MSIOF1 Receive control data register 1	MSIOF1_SIRDR1	R	H'A4C50058	32
MSIOF1 Receive control data register 2	MSIOF1_SIRDR2	R	H'A4C5005C	32
MSIOF1 Receive FIFO data register	MSIOF1_SIRFDR	R	H'A4C50060	32

Table 25.3 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
MSIOF0_SITMDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SITMDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SITMDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRMDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRMDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRMDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SITSCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRSCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SICTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIFCTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SISTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIIER	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SITDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SITDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SITFDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF0_SIRFDR	Undefined	Undefined	Retained	Retained	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
MSIOF1_SITMDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SITMDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SITMDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRMDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRMDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRMDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SITSCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRSCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SICTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIFCTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SISTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIIER	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SITDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SITDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SITFDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRDR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
MSIOF1_SIRFDR	Undefined	Undefined	Retained	Retained	Initialized	Retained

25.3.1 MSIOF Transmit Mode Register 1 (MSIOF 0_SITMDR1, MSIOF 1_SITMDR1)

SITMDR1 is a 32-bit readable/writable register that specifies the MSIOF transmit mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TRMD	PCON	SYNCMD[1:0]		SYNCCH[1:0]		SYN CAC	BIT LSB	—	DTDL[2:0]			—	SYNCDL[2:0]		
Initial value:	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	WDP SE	PSFL[1:0]		DIV[3:0]				—	—	—	—	FLD[1:0]		CONT	TXS TP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
31	TRMD	1	R/W	Transfer Mode Selects the transfer mode. 0: Slave mode 1: Master mode
30	PCON	0	R/W	Transfer Signal Connection When PCON = 1, the receiver should be set to slave mode. 0: MSIOFTSCK and MSIOFTSYNC are used only for transmission. 1: MSIOFTSCK and MSIOFTSYNC are used as common signals for transmission and reception (MSIOFSCK and MSIOFSYNC). When the TXRST bit becomes valid while PCON = 1, the RXRST bit becomes valid at the same time.
29, 28	SYNCMD [1:0]	00	R/W	SYNC Mode These bits specify the mode for the MSIOFTSYNC signal. 00: Frame start synchronization pulse 01: 1-bit synchronization pulse for last data in slot 10: Level mode/SPI 11: L/R mode

Bit	Bit Name	Initial Value	R/W	Descriptions
27, 26	SYNCCH [1:0]	00	R/W	Synchronization Signal Channel Select These bits are valid only in master mode. 00: The frame synchronization signal is output from MSIOFTSYNC. 01: The frame synchronization signal is output from MSIOFSS1. 10: The frame synchronization signal is output from MSIOFSS2. 11: Setting prohibited
25	SYNCAC	0	R/W	MSIOFTSYNC Polarity 0: Active-high signal in synchronization pulse or level mode, or driven high then low in L/R mode 1: Active-low signal in synchronization pulse or level mode, or driven low then high in L/R mode
24	BITLSB	0	R/W	MSB/LSB First 0: MSB first 1: LSB first
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22 to 20	DTDL[2:0]	001	R/W	Data Pin Bit Delay for MSIOFTSYNC Pin 1xx is valid only when SYNCMD[1:0] = 10. When SYNCMD[1:0] = 10, these bits should be specified so that the sum of the delays specified through DTDL and SYNCDL becomes an integer value. 000: No bit delay 001: 1-clock-cycle delay 010: 2-clock-cycle delay 101: 0.5-clock-cycle delay 110: 1.5-clock-cycle delay Other than above: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Descriptions
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	SYNCDL [2:0]	000	R/W	Frame Synchronization Signal Timing Delay These bits extend the transmit frame synchronization signal. This setting is invalid when SYNCMD[1:0] = 01. 1xx is valid only when SYNCMD[1:0] = 10. When SYNCMD[1:0] = 10, these bits should be specified so that the sum of the delays specified through DTDL and SYNCDL becomes an integer value. 000: No bit delay 001: 1-clock-cycle delay 010: 2-clock-cycle delay 011: 3-clock-cycle delay 101: 0.5-clock-cycle delay 110: 1.5-clock-cycle delay Other than above: Setting prohibited
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	WDPSE	0	R/W	Word Location Setting Enable When this bit is set to 1, only the last word in each group is output as valid data. For the other words, the value specified with TXDIZ is output. 0: Makes the settings of SITMDR2 and SITMDR3 valid. 1: Enables transmission of only the last word in each group as valid data, and sets each BITLEN to 16 bits or less.

Bit	Bit Name	Initial Value	R/W	Descriptions
13, 12	PSFL[1:0]	00	R/W	Frame Size Specify When setting a value other than B'00, these bits should be specified so that the total size of the words specified in SITMDR2 and SITMDR3 is less than the frame size. 00: The frame size becomes the total of the words and bits in each group specified in SITMDR2 and SITMDR3. 01: Fixed at 8 words 10: Fixed at 16 words 11: Fixed at 32 words
11 to 8	DIV[3:0]	0000	R/W	Word Division One 32-bit word can be transmitted as two separate 16-bit words through two groups. The word count in each group should be set to 1. 0000: Not divided 0001: Upper word in group 1 and lower word in group 2 0010: Upper word in group 1 and lower word in group 3 0011: Upper word in group 1 and lower word in group 4 0100: Upper word in group 2 and lower word in group 1 0101: Upper word in group 2 and lower word in group 3 0110: Upper word in group 2 and lower word in group 4 0111: Upper word in group 3 and lower word in group 1 1000: Upper word in group 3 and lower word in group 2 1001: Upper word in group 3 and lower word in group 4 1010: Upper word in group 4 and lower word in group 1 1011: Upper word in group 4 and lower word in group 2 1100: Upper word in group 4 and lower word in group 3 Other than above: Setting prohibited Note: The groups that are not specified for word division should be used for control data.

Bit	Bit Name	Initial Value	R/W	Descriptions
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3, 2	FLD[1:0]	00	R/W	Frame Synchronization Signal Interval These bits specify the minimum idle time between frames in the number of serial clock cycles. This setting is valid only in master mode. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
1	CONT	0	R/W	Continuous Transmission Specifies continuous transmission of multiple frames of data within one frame. This setting is valid when SYNCMD[1:0] = 00 or 10. When the TFSE bit in SICTR is set to 1'b0 in master mode, transmission stops after one frame of data has been transmitted. 0: Does not perform continuous transmission 1: Performs continuous transmission
0	TXSTP	0	R/W	Transmission Stop Specifies the processing with no control data specified when the transmit FIFO becomes empty. This setting is valid only in master mode. When transmission is continued, the value specified with TXDIZ is output as transmit data. When TXSTP = 1, if data is stored in the transmit FIFO after transmission, data transmission is restarted. 0: Continues transmission when the transmit FIFO is empty. (The frame synchronization signal is output.) 1: Stops transmission when the transmit FIFO is empty. (The frame synchronization signal is not output.)

25.3.2 MSIOF Transmit Mode Register 2 (MSIOF0_ SITMDR2, MSIOF1_ SITMDR2)

SITMDR2 is a 32-bit readable/writable register that specifies the MSIOF transmit mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRP[1:0]		—	BITLEN1[4:0]					WDLEN1[7:0]							
Initial value:	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRPDL1[1:0]		GRPDL3[1:0]		CRDE4	CRDE3	CRDE2	CRDE1	CRDNUM4	CRDNUM3	CRDNUM2	CRDNUM1	GRPMASK4	GRPMASK3	GRPMASK2	GRPMASK1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
31, 30	GRP[1:0]	00	R/W	Group Count The number of groups is set to the value specified in these bits + 1. When three or more groups are specified, the word size (BITLEN) is set to 16 bits or less.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28 to 24	BITLEN1[4:0] (BITLEN3)	01111	R/W	Data Size (8 to 32 bits) The word size (bits) of groups 1 and 3 is set to the value specified in these bits + 1. (BITLEN1 = BITLEN3) Eight bits or a larger size should be specified. When three or four groups are used, the word size is set to 16 bits or less (the highest bit is ignored).
23 to 16	WDLEN1[7:0] (WDLEN1[3:0] WDLEN3[3:0])	H'00	R/W	Group Size (1 to 256 words) The word counts of groups 1 and 3 are set to WDLEN1 + 1 and WDLEN3 + 1 respectively. When one or two groups are used, the 8-bit value is set to WDLEN1. When three or four groups are used, the upper 4-bit value is set to WDLEN1, and the lower 4-bit value is set to WDLEN3.

Bit	Bit Name	Initial Value	R/W	Descriptions
15, 14	GRPDL1[1:0]	00	R/W	Group 1 Data Transmission Delay Specify the number of blank bits to be inserted before transmission of group 1. The value specified with TXDIZ in SICTR is output during transmission. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
13, 12	GRPDL3[1:0]	00	R/W	Group 3 Data Transmission Delay Specify the number of blank bits to be inserted before transmission of group 3. The value specified with TXDIZ in SICTR is output during transmission. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
11	CRDE4	0	R/W	Control Data Enable "x" in CRDEx represents a group. Each of these bits specifies whether the group transmits FIFO data or control data. 0: The specified group transmits FIFO data. 1: The specified group transmits control data.
10	CRDE3	0	R/W	
9	CRDE2	0	R/W	
8	CRDE1	0	R/W	
7	CRDNUM4	0	R/W	Control Data Transmit Count "x" in CRDNUMx represents a group. Each of these bits specifies the transmission count of control data when multiple words are set for a control data group. Each bit is valid only when the corresponding CRDEx bit is 1. When CRDNUMx = 1, control data is transmitted only for the first word, and the value specified with TXDIZ is output for remaining words. 0: Transmits control data continuously. 1: Transmits control data only once.
6	CRDNUM3	0	R/W	
5	CRDNUM2	0	R/W	
4	CRDNUM1	0	R/W	
3	GRPMASK4	0	R/W	Group Output Mask "x" in GRPMASKx represents a group. When GRPMASKx = 1, the group outputs the value specified with TXDIZ.
2	GRPMASK3	0	R/W	
1	GRPMASK2	0	R/W	
0	GRPMASK1	0	R/W	

25.3.3 MSIOF Transmit Mode Register 3 (MSIOF0_SITMDR3, MSIOF1_SITMDR3,)

SITMDR3 is a 32-bit readable/writable register that specifies the MSIOF transmit mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	BITLEN2[4:0]					WDLEN2[7:0]							
Initial value:	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRPDL2[1:0]		GRPDL4[1:0]		—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 24	BITLEN2[4:0] (BITLEN4)	01111	R/W	Word Size (8 to 32 bits) The word size is set to the value specified in these bits + 1. Eight bits or a larger size should be specified. When three or four groups are used, the word size is set to 16 bits or less (the highest bit is ignored).
23 to 16	WDLEN2[7:0] (WDLEN2[3:0] WDLEN4[3:0])	H'00	R/W	Group Size (1 to 256 words) The word counts of groups 2 and 4 are set to WDLEN2 + 1 and WDLEN4 + 1 respectively. When one or two groups are used, the 8-bit value is set to WDLEN2. When three or four groups are used, the upper 4-bit value is set to WDLEN2, and the lower 4-bit value is set to WDLEN4.
15, 14	GRPDL2[1:0]	00	R/W	Group 2 Data Transmission Delay These bits specify the number of blank bits to be inserted before transmission of group 2. The value specified with TXDIZ in SICTR is output during transmission. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay

Bit	Bit Name	Initial Value	R/W	Descriptions
13, 12	GRPDL4[1:0]	00	R/W	Group 4 Data Transmission Delay These bits specify the number of blank bits to be inserted before transmission of group 4. The value specified with TXDIZ in SICTR is output during transmission. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.3.4 MSIOF Receive Mode Register 1 (MSIOF0_SIRMDR1, MSIOF1_SIRMDR1)

SIRMDR1 is a 32-bit readable/writable register that specifies the MSIOF receive mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TRMD	—	SYNCMD[1:0]		—	—	SYN CAC	BIT LSB	—	DTDL[2:0]			—	SYNCDL[2:0]		
Initial value:	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R/W	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	WDP SE	PSFL[1:0]		MRG[3:0]				—	—	—	—	FLD[1:0]		CONT	RXS TP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
31	TRMD	1	R/W	Transfer Mode Selects the transfer mode. This bit should be set to 0 (slave mode) when setting the PCON bit in SITMDR1 to 1. 0: Slave mode 1: Master mode
30	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
29, 28	SYNCMD[1:0]	00	R/W	SYNC Mode These bits specify the mode for the MSIOFRSYNC signal. 00: Frame start synchronization pulse 01: 1-bit synchronization pulse for last data in slot 10: Level mode/SPI 11: L/R mode
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	SYNCAC	0	R/W	MSIOFRSYNC Polarity 0: Active-high signal in synchronization pulse or level mode, or driven high then low in L/R mode 1: Active-low signal in synchronization pulse or level mode, or driven low then high in L/R mode
24	BITLSB	0	R/W	MSB/LSB First 0: MSB first 1: LSB first
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22 to 20	DTDL[2:0]	001	R/W	Data Pin Bit Delay for MSIOFRSYNC Pin B'1xx is valid only in SPI mode. In SPI mode, these bits should be specified so that the sum of the delays specified through DTDL and SYNCDL becomes an integer value. 000: No bit delay 001: 1-clock-cycle delay 010: 2-clock-cycle delay 101: 0.5-clock-cycle delay 110: 1.5-clock-cycle delay Other than above: Setting prohibited
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
18 to 16	SYNCDL[2:0]	000	R/W	MSIOFRSYNC Timing Delay The synchronization signal is extended for the clock cycles specified in these bits. These bits are invalid when SYNCMD[1:0] = B'01. B'1xx is valid only in SPI mode. In SPI mode, these bits should be specified so that the sum of the delays specified through DTDL and SYNCDL becomes an integer value. 000: No bit delay 001: 1-clock-cycle delay 010: 2-clock-cycle delay 011: 3-clock-cycle delay 101: 0.5-clock-cycle delay 110: 1.5-clock-cycle delay Other than above: Setting prohibited
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	WDPSE	0	R/W	Word Location Setting Enable When this bit is set to 1, only the last word of each group is received as valid data. The other words are discarded as invalid data. 0: Makes the settings of SIRMDR2 and SIRMDR3 valid. 1: Enables reception of only the last word in each group as valid data, and sets each BITLEN to 16 bits or less.
13, 12	PSFL[1:0]	00	R/W	Frame Size Specify When setting a value other than B'00, these bits should be specified so that the total size of the words specified in SIRMDR2 and SIRMDR3 is less than the frame size. 00: Receives data with the frame size specified in SIRMDR2 and SIRMDR3. 01: Fixed at 8 words 10: Fixed at 16 words 11: Fixed at 32 words

Bit	Bit Name	Initial Value	R/W	Descriptions
11 to 8	MRG[3:0]	0000	R/W	Word Merge The data received as 16-bit words of two groups can be merged into one 32-bit word and the 32-bit word is stored in the receive FIFO. The word count of the two groups to be merged should be set to 1. In this case, data of groups other than word merging target groups should be handled as control data. 0000: Not merged 0001: Upper word in group 1 and lower word in group 2 0010: Upper word in group 1 and lower word in group 3 0011: Upper word in group 1 and lower word in group 4 0100: Upper word in group 2 and lower word in group 1 0101: Upper word in group 2 and lower word in group 3 0110: Upper word in group 2 and lower word in group 4 0111: Upper word in group 3 and lower word in group 1 1000: Upper word in group 3 and lower word in group 2 1001: Upper word in group 3 and lower word in group 4 1010: Upper word in group 4 and lower word in group 1 1011: Upper word in group 4 and lower word in group 2 1100: Upper word in group 4 and lower word in group 3 Other than above: Setting prohibited
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3, 2	FLD[1:0]	00	R/W	Frame Synchronization Signal Interval These bits specify the minimum idle time between frames in the number of serial clock cycles. This setting is valid only in master mode. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay

Bit	Bit Name	Initial Value	R/W	Descriptions
1	CONT	0	R/W	Continuous Reception Specifies continuous reception of multiple frames of data within one frame. This setting is valid when SYNCMD[1:0] =00 or 10. When the RFSE bit in SICTR is cleared to 1'b0 in master mode, reception stops after one frame of data has been received. 0: Does not perform continuous reception 1: Performs continuous reception
0	RXSTP	0	R/W	Reception Stop with Receive FIFO Full Specifies the processing when the receive FIFO becomes full. This setting is valid only in master mode. 0: Continues reception when the receive FIFO is full. (The receive frame synchronization signal is output.) 1: Terminates reception when the receive FIFO is full. (The receive frame synchronization signal is not output.)

25.3.5 MSIOF Receive Mode Register 2 (MSIOF0_SIRMDR2, MSIOF1_SIRMDR2)

SIRMDR2 is a 32-bit readable/writable register that specifies the MSIOF receive mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRP[1:0]		—	BITLEN1[4:0]					WDLEN1[7:0]							
Initial value:	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

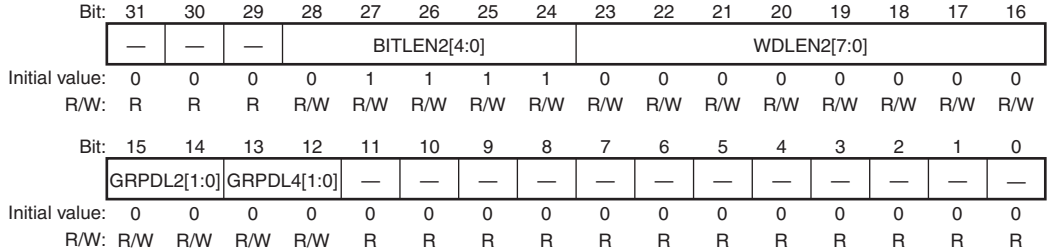
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRPDL1[1:0]		GRPDL3[1:0]		CRDE4	CRDE3	CRDE2	CRDE1	CRDNUM4	CRDNUM3	CRDNUM2	CRDNUM1	GRPMASK4	GRPMASK3	GRPMASK2	GRPMASK1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
31, 30	GRP[1:0]	00	R/W	Group Count The number of groups is set to the value specified in these bits + 1. When three or more groups are specified, the word size (BITLEN) is set to 16 bits or less.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28 to 24	BITLEN1[4:0] (BITLEN3)	01111	R/W	Word Size (8 to 32 bits) The word size (bits) of groups 1 and 3 is set to the value specified in these bits + 1. (BITLEN1 = BITLEN3) Eight bits or a larger size should be specified. When three or four groups are used, the word size is set to 16 bits or less (the highest bit is ignored).
23 to 16	WDLEN1[7:0] (WDLEN1[3:0] WDLEN3[3:0])	H'00	R/W	Group Size (1 to 256 words) The word counts of groups 1 and 3 are set to WDLEN1 + 1 and WDLEN3 + 1 respectively. When one or two groups are used, the 8-bit value is set to WDLEN1. When three or four groups are used, the upper 4-bit value is set to WDLEN1, and the lower 4-bit value is set to WDLEN3.

Bit	Bit Name	Initial Value	R/W	Descriptions
15, 14	GRPDL1[1:0]	00	R/W	Group 1 Data Reception Delay These bits specify the delay in data reception of group 1. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
13, 12	GRPDL3[1:0]	00	R/W	Group 3 Data Reception Delay These bits specify the delay in data reception of group 3. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
11	CRDE4	0	R/W	Control Data Enable
10	CRDE3	0	R/W	“x” in CRDEx represents a group. Each of these bits specifies whether the group receives FIFO data or control data. 0: The specified group receives FIFO data. 1: The specified group receives control data.
9	CRDE2	0	R/W	
8	CRDE1	0	R/W	
7	CRDNUM4	0	R/W	Control Data Reception Count
6	CRDNUM3	0	R/W	“x” in CRDNUMx represents a group. Each of these bits specifies the reception count of control data when multiple words are set for a control data group. Each bit is valid only when the corresponding CRDEx bit is 1. When CRDNUMx = 1, only the first word is received as control data, and the other words are discarded as invalid data. 0: Receives control data continuously. 1: Receives control data only once in a frame.
5	CRDNUM2	0	R/W	
4	CRDNUM1	0	R/W	
3	GRPMASK4	0	R/W	Group Input Mask
2	GRPMASK3	0	R/W	“x” in GRPMASKx represents a group. When GRPMASKx = 1, inputs of the group are masked and discarded as invalid data.
1	GRPMASK2	0	R/W	
0	GRPMASK1	0	R/W	

25.3.6 MSIOF Receive Mode Register 3 (MSIOF0_SIRMDR3, MSIOF1_SIRMDR3)

SIRMDR3 is a 32-bit readable/writable register that specifies the MSIOF receive mode.



Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 24	BITLEN2[4:0] (BITLEN4)	01111	R/W	Word Size (1 to 32 bits) The word size is set to the value specified in these bits + 1. Eight bits or a larger size should be specified. When three or four groups are used, the word size is set to 16 bits or less (the highest bit is ignored).
23 to 16	WDLEN2[7:0] (WDLEN4)	H'00	R/W	Group Size (1 to 256 words) The word counts of groups 2 and 4 are set to WDLEN2 + 1 and WDLEN4 + 1 respectively. When one or two groups are used, the 8-bit value is set to WDLEN2. When three or four groups are used, the upper 4-bit value is set to WDLEN2, and the lower 4-bit value is set to WDLEN4.
15, 14	GRPDL2[1:0]	00	R/W	Group 2 Data Reception Delay These bits specify the delay in data reception of group 2. 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay

Bit	Bit Name	Initial Value	R/W	Descriptions
13, 12	GRPDL4[1:0]	00	R/W	Group 4 Data Reception Delay These bits specify the delay in data reception of group 4 00: 0-clock-cycle delay 01: 1-clock-cycle delay 10: 2-clock-cycle delay 11: 3-clock-cycle delay
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.3.7 MSIOF Control Register (MSIOF0_SICTR, MSIOF1_SICTR)

SICTR is a 32-bit readable/writable register that specifies the MSIOF operating state.

The values written to the TXE, RXE, TFSE, RFSE, TSCKE, and RSCKE bits become valid (can be read from the bits) several cycles after writing. These bits should not be modified together; modify one bit at a time, check that the new value can be read from the bit, and then modify another bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSCKIZ[1:0]		RSCKIZ[1:0]		TEDG	REDG	—	—	TXDIZ[1:0]		—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSCKE	TFSE	RSCKE	RFSE	—	—	TXE	RXE	—	—	—	—	—	—	TXRST	RXRST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
31, 30	TSCKIZ[1:0]	00	R/W	Transmit Clock Input/Output When Transmission is Invalid (Clock Polarity Select in SPI Mode) When SPI mode is not used, these bits should always be set to 00. [Master mode] 00: Outputs MSIOFTSCK when transmission is invalid 01: Setting prohibited 10: Outputs 0 11: Outputs 1 [Slave mode] 00: Inputs MSIOFTSCK when transmission is invalid 01: Setting prohibited 10: Inputs 0 through MSIOFTSCK when transmission is invalid 11: Inputs 1 through MSIOFTSCK when transmission is invalid

Bit	Bit Name	Initial Value	R/W	Descriptions
29, 28	RSCKIZ[1:0]	00	R/W	Receive Clock Input/Output When Transmission is Invalid (Clock Polarity Select in SPI Mode) When SPI mode is not used, these bits should always be set to 00. [Master mode] 00: Outputs MSIOFRSCK when transmission is invalid 01: Setting prohibited 10: Outputs 0 11: Outputs 1 [Slave mode] 00: Inputs MSIOFRSCK when transmission is invalid 01: Setting prohibited 10: Inputs 0 through MSIOFRSCK when transmission is invalid 11: Inputs 1 through MSIOFRSCK when transmission is invalid
27	TEDG	0	R/W	Transmit Timing 0: Outputs data at the rising edge of the clock 1: Outputs data at the falling edge of the clock
26	REDG	0	R/W	Receive Timing 0: Samples data at the falling edge of the clock 1: Samples data at the rising edge of the clock
25, 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	TXDIZ[1:0]	00	R/W	Pin Output When Transmission is Invalid These bits specify the MSIOFTXD pin output state when transmission is disabled. 00: Outputs 0 01: Outputs 1 10: Output is in high-impedance state 11: Setting prohibited
21 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
15	TSCKE	0	R/W	Transmit Serial Clock Output Enable This bit is valid in master mode. When this bit is set to 1, the MSIOF initializes the baud rate generator and initiates the operation. At the same time, the MSIOF outputs the clock generated by the baud rate generator to the MSIOF_TSCK pin. After the clock is output, 1 can be read. When transmitting data, set this bit to 1 before setting the TFSE and TXE bits. After data transmission, clear the TFSE and TXE bits and then clear this bit to 0. [Write data] 0: Disables the MSIOF_TSCK output. (Outputs the value specified with TSCKIZ.) 1: Enables the MSIOF_TSCK output. [Read data] 0: Does not output MSIOF_TSCK. (Outputs the value specified with TSCKIZ.) 1: Outputs MSIOF_TSCK.
14	TFSE	0	R/W	Transmit Frame Synchronization Signal Output Enable This bit is valid in master mode. When this bit is set to 1, the MSIOF initializes the frame counter and initiates the operation. After the transmit frame synchronization signal is output, 1 can be read. When 0 is written to this bit, 0 is set after transmission of the frame. [Write data] 0: Disables the MSIOF_TSYNC output. (Outputs the value specified with SYNCAC.) 1: Enables the MSIOF_TSYNC output. [Read data] 0: Does not output MSIOF_TSYNC. (Outputs the value specified with SYNCAC.) 1: Outputs MSIOF_TSYNC.

Bit	Bit Name	Initial Value	R/W	Descriptions
13	RSCKE	0	R/W	Receive Serial Clock Output Enable This bit is valid in master mode. When this bit is set to 1, the MSIOF initializes the baud rate generator and initiates the operation. At the same time, the MSIOF outputs the clock generated by the baud rate generator to the MSIOF_RSCK pin. After the clock is output, 1 can be read. When receiving data, set this bit to 1 before setting the RFSE and RXE bits. After data reception, clear the RFSE and RXE bits and then clear this bit to 0. [Write data] 0: Disables the MSIOF_RSCK output. (Outputs the value specified with RSCKIZ.) 1: Enables the MSIOF_RSCK output. [Read data] 0: Does not output MSIOF_RSCK. (Outputs the value specified with RSCKIZ.) 1: Outputs MSIOF_RSCK.
12	RFSE	0	R/W	Receive Frame Synchronization Signal Output Enable This bit is valid in master mode. When this bit is set to 1, the MSIOF initializes the frame counter and initiates the operation. After the receive frame synchronization signal is output, 1 can be read. When 0 is written to this bit, 0 is set after reception of the frame. [Write data] 0: Disables the MSIOF_RSYNC output. (Outputs the value specified with SYNCAC.) 1: Enables the MSIOF_RSYNC output. [Read data] 0: Does not output MSIOF_RSYNC. (Outputs the value specified with SYNCAC.) 1: Outputs MSIOF_RSYNC.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
9	TXE	0	R/W	Transmit Enable When this bit is set to 1, the MSIOF starts data transmission from the beginning of the next frame (at the rising edge of the frame synchronization signal). After the valid data is output, 1 can be read. When value 1 set to this bit becomes valid, the MSIOF issues a transmit data transfer request according to the setting of the TFWM bits in SIFCTR. When transmit data is stored in the transmit FIFO, transmission of data from the MSIOF_TXD pin begins. When 0 is written to this bit, 0 is set after transmission of the frame. This bit is initialized upon a transmit reset. [Write data] 0: Disables the MSIOF_TXD output. (Outputs the value specified with TXDIZ.) 1: Enables the MSIOF_TXD output. [Read data] 0: Does not output MSIOF_TXD. (Outputs the value specified with TXDIZ.) 1: Outputs MSIOF_TXD.
8	RXE	0	R/W	Receive Enable When this bit is set to 1, the MSIOF starts data reception from the beginning of the next frame (at the rising edge of the frame synchronization signal). When value 1 set to this bit becomes valid, the MSIOF starts receiving data through the MSIOF_RXD pin. When receive data is stored in the receive FIFO, the MSIOF issues a receive data transfer request according to the setting of the RFWM bits in SIFCTR. This bit is initialized upon a receive reset. [Write data] 0: Disables data reception through MSIOF_RXD. 1: Enables data reception through MSIOF_RXD. [Read data] 0: Data is not received through MSIOF_RXD. 1: Data can be received through MSIOF_RXD.

Bit	Bit Name	Initial Value	R/W	Descriptions
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TXRST	0	R/W	Transmit Reset When value 1 set to this bit becomes valid, the MSIOF immediately sets transmit data through the MSIOF_TXD pin to 0, and initializes the transmit data registers and transmit-related status. When value 1 set to this bit becomes valid when PCON = 1, the RXRST bit is also becomes valid. The following registers and bits are initialized. SITDR1 and SITDR2 SITFDR Transmit FIFO write pointer TCRDY, TFEMP, and TDREQ bits in SISTR TXE bit This bit is read as 1 until the reset operation is completed for about SUBϕ20 cycles. No data should be written to SICTR or the transmit FIFO during this time period. [Write data] 0: Does not reset transmit operation. 1: Resets transmit operation. [Read data] 0: Transmit operation reset is completed. 1: Transmit operation is being reset.

Bit	Bit Name	Initial Value	R/W	Descriptions
0	RXRST	0	R/W	Receive Reset When value 1 set to this bit becomes valid, the MSIOF immediately disables reception through the MSIOF_RXD pin, and initializes the receive data registers and receive-related status. The following registers and bits are initialized. SIRDR1 and SIRDR2 SIRFDR Receive FIFO read pointer RCRDY, RFFUL, and RDREQ bits in SISTR RXE bit This bit is read as 1 until the reset operation is completed for about SUBϕ20 cycles. No data should be written to SICTR or read the receive FIFO during this time period. [Write data] 0: Does not reset receive operation. 1: Resets receive operation. [Read data] 0: Receive operation reset is completed. 1: Receive operation is being reset.

25.3.8 MSIOF Transmit Clock Select Register (MSIOF0_SITSCR, MSIOF1_SITSCR)

SITSCR is a 16-bit readable/writable register that specifies the conditions for generating the transmit serial clock in master mode. SITSCR can be specified when the TRMD bit in SITMDR1 is set to B'1.

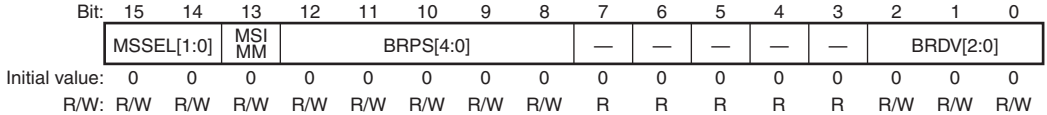
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSSEL[1:0]		MSIMM	BRPS[4:0]					—	—	—	—	—	BRDV[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
15, 14	MSSEL[1:0]	00	R/W	Master Clock Source Select The master clock is the clock input to the baud rate generator. 00: Selects Bϕ as the master clock 01: Selects the signal input from the MSIOFMCK pin as the master clock 10: Setting prohibited 11: Setting prohibited
13	MSIMM	0	R/W	Master Clock Direct Select When setting MSIMM = 1, the master clock source should be set to 33 MHz or lower. 0: Selects the clock output from the baud rate generator as the serial clock. 1: Selects the master clock itself as the serial clock.
12 to 8	BRPS[4:0]	00000	R/W	Prescalar Setting These bits specify the master clock division ratio in the count value of the prescalar in the baud rate generator. The specifiable value is from B'00000 (× 1/1) to B'11111 (× 1/32).
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
2 to 0	BRDV[2:0]	000	R/W	Baud Rate Generator's Division Ratio These bits specify the frequency division ratio for the output stage of the baud rate generator. The final frequency division ratio of the baud rate generator is determined as BRPS × BRDV (1/1024 max.). 000: Prescalar output × 1/2 001: Prescalar output × 1/4 010: Prescalar output × 1/8 011: Prescalar output × 1/16 100: Prescalar output × 1/32 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited

25.3.9 MSIOF Receive Clock Select Register (MSIOF0_SIRSCR, MSIOF1_SIRSCR)

SIRSCR is a 16-bit readable/writable register that specifies the conditions for generating the receive serial clock in master mode. SIRSCR can be specified when the TRMD bit in SIRMDR1 is set to B'1.



Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSSEL[1:0]		MSIMM	BRPS[4:0]					—	—	—	—	—	BRDV[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Descriptions
15, 14	MSSEL[1:0]	00	R/W	Master Clock Source Select The master clock is the clock input to the baud rate generator. 00: Selects Bϕ as the master clock 01: Selects the signal input from the MSIOFMCK pin as the master clock 10: Setting prohibited 11: Setting prohibited
13	MSIMM	0	R/W	Master Clock Direct Select When setting MSIMM = 1, the master clock source should be set to 33 MHz or lower. 0: Selects the clock output from the baud rate generator as the serial clock. 1: Selects the master clock itself as the serial clock.
12 to 8	BRPS[4:0]	00000	R/W	Prescalar Setting These bits specify the master clock division ratio in the count value of the prescalar in the baud rate generator. The specifiable value is from B'00000 (× 1/1) to B'11111 (× 1/32).
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
2 to 0	BRDV[2:0]	000	R/W	Baud Rate Generator's Division Ratio These bits specify the frequency division ratio for the output stage of the baud rate generator. The final frequency division ratio of the baud rate generator is determined as BRPS × BRDV (1/1024 max.). 000: Prescalar output × 1/2 001: Prescalar output × 1/4 010: Prescalar output × 1/8 011: Prescalar output × 1/16 100: Prescalar output × 1/32 101: Setting prohibited 110: Setting prohibited 111: Prescalar output × 1/1 Note: The B'111 value can be set when BRPS[4:0] bit value is only B'00000 or B'00001.

25.3.10 MSIOF Transmit Control Data Register 1 (MSIOF0_SITDR1, MSIOF1_STDR1)

SITDR1 is a 32-bit write-only register that specifies the transmit control data of the MSIOF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCRD1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCRD2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 16	TCRD1[15:0]	H'0000	W	• When one or two groups are used: These bits specify the transmit control data for the upper 16 bits of a word in group 1 • When three or four groups are used: These bits specify the transmit control data for a word in group 1.
15 to 0	TCRD2[15:0]	H'0000	W	• When one or two groups are used: These bits specify the transmit control data for the lower 16 bits of a word in group 1 when the word size is larger than 16 bits. • When three or four groups are used: These bits specify the transmit control data for a word in group 3.

25.3.11 MSIOF Transmit Control Data Register 2 (MSIOF0_SITDR2, MSIOF1_SITDR2)

SITDR2 is a 32-bit write-only register that specifies the transmit control data of the MSIOF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCRD3[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCRD4[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 16	TCRD3[15:0]	H'0000	W	• When one or two groups are used: These bits specify the transmit control data for the upper 16 bits of a word in group 2 • When three or four groups are used: These bits specify the transmit control data for a word in group 2.
15 to 0	TCRD4[15:0]	H'0000	W	• When one or two groups are used: These bits specify the transmit control data for the lower 16 bits of a word in group 2 when the word size is larger than 16 bits. • When three or four groups are used: These bits specify the transmit control data for a word in group 4.

25.3.12 MSIOF Transmit FIFO Data Register (MSIOF0_SITFDR, MSIOF1_SITFDR)

SITFDR is a 32-bit write-only register that specifies the transmit FIFO data of the MSIOF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SITFD1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SITFD2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 16	SITFD1[15:0]	H'0000	W	These bits specify the upper 16 bits of the FIFO data to be output through MSIOFTXD.
15 to 0	SITFD2[15:0]	H'0000	W	These bits specify the lower 16 bits of the FIFO data to be output through MSIOFTXD.

25.3.13 MSIOF Receive Control Data Register 1 (MSIOF0_SIRDR1, MSIOF1_SIRDR1)

SIRDR1 is a 32-bit read-only register that stores the receive control data of the MSIOF.

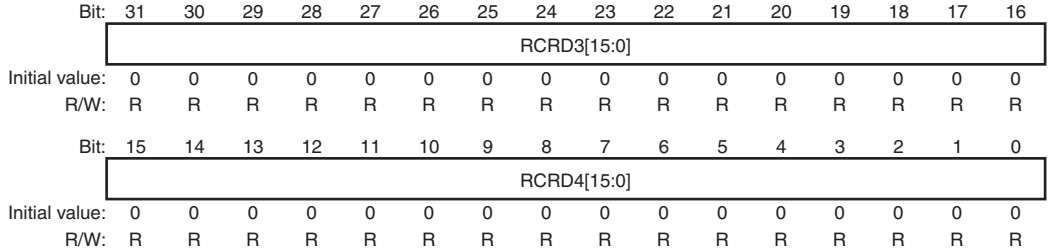
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RCRD1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCRD2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 16	RCRD1[15:0]	H'0000	R	• When one or two groups are used: Store the receive control data for the upper 16 bits of a word in group 1 • When three or four groups are used: Store the receive control data for a word in group 1.
15 to 0	RCRD2[15:0]	H'0000	R	• When one or two groups are used: Store the receive control data for the lower 16 bits of a word in group 1 when the word size is larger than 16 bits. • When three or four groups are used: Store the receive control data for a word in group 3.

25.3.14 MSIOF Receive Control Data Register 2 (MSIOF0_SIRDR2, MSIOF1_SIRDR2)

SIRDR2 is a 32-bit read-only register that stores the receive control data of the MSIOF.



Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 16	RCRD3[15:0]	H'0000	R	• When one or two groups are used: Store the receive control data for the upper 16 bits of a word in group 2. • When three or four groups are used: Store the receive control data for a word in group 2.
15 to 0	RCRD4[15:0]	H'0000	R	• When one or two groups are used: Store the receive control data for the lower 16 bits of a word in group 2 when the word size is larger than 16 bits. • When three or four groups are used: Store the receive control data for a word in group 4.

25.3.15 MSIOF Receive FIFO Data Register (MSIOF0_SIRFDR, MSIOF1_SIRFDR)

SIRFDR is a 32-bit read-only register that stores the receive FIFO data of the MSIOF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SIRFD1[15:0]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SIRFD2[15:0]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 16	SIRFD1[15:0]	Undefined	R	Store the upper 16 bits of the FIFO data received through MSIOFRXD.
15 to 0	SIRFD2[15:0]	Undefined	R	Store the lower 16 bits of the FIFO data received through MSIOFRXD.

25.3.16 MSIOF Status Register (MSIOF0_SISTR, MSIOF1_SISTR)

Each bit in SISTR becomes an MSIOF interrupt source when the corresponding bit in SIIER is set to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	TCRDY	TFEMP	TDREQ	—	—	—	—	TEOF	—	TFSERR	TFOVF	TFUDF	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RCRDY	RFFUL	RDREQ	—	—	—	—	RCRDY	—	RFSERR	RFUDF	RFOVF	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	TCRDY	0	R/W	Transmit Control Data Ready If SITDR1 or SITDR2 is written while this bit is cleared to 0, SICTR is overwritten and the previous contents are not output through the MSIOFTXD pin. This bit is valid when the TXE bit in SICTR is 1. Writing 1 to this bit clears the content. Writing 0 is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: SITDR1 and SITDR2 are not ready to be written 1: SITDR1 and SITDR2 are ready to be written
29	TFEMP	0	R/W	Transmit FIFO Empty This bit is valid when the TXE bit in SICTR is 1. Writing 1 to this bit clears the content. Writing 0 is invalid. This bit indicates a state; if SITFDR is written, the MSIOF clears this bit. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: Transmit FIFO is not empty 1: Transmit FIFO is empty

Bit	Bit Name	Initial Value	R/W	Descriptions
28	TDREQ	0	R	Transmit Data Transfer Request A transmit data transfer request is issued when the empty space in the transmit FIFO exceeds the size specified by the TFWM bits in SIFCTR. When transferring transmit data through the DMAC, this bit is always cleared after one DMAC access. After DMAC access, when the conditions for setting this bit are satisfied, the MSIOF again sets this bit to 1. This bit is valid when the TXE bit in SICTR is 1. This bit indicates a state; if the size of empty space in the transmit FIFO becomes less than the size specified by the TFWM bits in SIFCTR, the MSIOF clears this bit. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. However, when the TDMAE bit is 1, only a DMAC transfer request is issued. 0: The size of empty space in the transmit FIFO has not exceeded the size specified by the TFWM bits in SIFCTR. 1: The size of empty space in the transmit FIFO has exceeded the size specified by the TFWM bits in SIFCTR.
27 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	TEOF	0	R/W	Completion of Frame Transmission The completion of frame transmission is issued when transmission of one frame of data is completed. This bit becomes valid when the TXE bit in SICTR is 1. Writing 1 to this bit clears the content. Writing 0 is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: Completion of one frame transmission not detected 1: Completion of one frame transmission detected
22	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
21	TFSERR	0	R/W	Transmit Frame Synchronization Error A transmit frame synchronization error occurs when the next transmit frame synchronization timing arrives before the previous data or control data transmission has been completed. If a transmit frame synchronization error occurs, the MSIOF performs transmission only for the slots that can be transferred. This bit is valid when the TXE bit in SICTR is 1. When 1 is written to this bit, it is cleared. Writing 0 to this bit is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: No transmit frame synchronization error has occurred 1: A transmit frame synchronization error has occurred
20	TFOVF	0	R/W	Transmit FIFO Overflow A transmit FIFO overflow means that there has been an attempt to write to SITFDR when the transmit FIFO is full. If a transmit FIFO overflow occurs, the MSIOF ignores the write operation causing the overflow. This bit is valid when the TXE bit in SICTR is 1. When 1 is written to this bit, it is cleared. Writing 0 to this bit is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: No transmit FIFO overflow has occurred 1: A transmit FIFO overflow has occurred

Bit	Bit Name	Initial Value	R/W	Descriptions
19	TFUDF	0	R/W	Transmit FIFO Underflow A transmit FIFO underflow means that loading for transmission has occurred when the transmit FIFO is empty. If a transmit FIFO underflow occurs, the MSIOF repeatedly sends the previous transmit data. This bit is valid when the TXE bit in SICTR is 1. When 1 is written to this bit, it is cleared. Writing 0 to this bit is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: No transmit FIFO underflow has occurred 1: A transmit FIFO underflow has occurred
18 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	RCRDY	0	R	Receive Control Data Ready When data different from the previous one is stored during control data reception, a write of valid data is assumed and this bit is set to 1. In the first data reception, this bit is set to 1 when data other than 0 is stored. If valid data is written to SIRDR1 or SIRDR2 when this bit is set to 1, SIRDR1 or SIRDR2 is overwritten with the latest data. This bit is valid when the RXE bit in SICTR is 1. This bit indicates a state; if SIRDR1 or SIRDR2 is read, the MSIOF clears this bit. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: SIRDR1 and SIRDR2 store no valid data. 1: SIRDR1 and SIRDR2 store valid data.
13	RFFUL	0	R/W	Receive FIFO Full This bit is valid when the RXE bit in SICTR is 1. Writing 1 to this bit clears the content. Writing 0 is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: Receive FIFO is not full 1: Receive FIFO is full

Bit	Bit Name	Initial Value	R/W	Descriptions
12	RDREQ	0	R	Receive Data Transfer Request A receive data transfer request is issued when the valid data space in the receive FIFO exceeds the size specified by the RFWM bits in SIFCTR. When transferring receive data through the DMAC, this bit is always cleared by one DMAC access. After DMAC access, when the conditions for setting this bit are satisfied, the MSIOF again sets this bit to 1. This bit is valid when the RXE bit in SICTR is 1. This bit indicates a state; if the size of valid data space in the receive FIFO becomes less than the size specified by the RFWM bits in SIFCTR, the MSIOF clears this bit. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. However, when the RDMAE bit is 1, only a DMAC transfer request is issued. 0: The size of valid data space in the receive FIFO has not exceeded the size specified by the RFWM bits in SIFCTR. 1: The size of valid data space in the receive FIFO has exceeded the size specified by the RFWM bits in SIFCTR.
11 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	REOF	0	R/W	Completion of Frame Reception The completion of frame reception is issued when reception of one frame of data is completed. This bit becomes valid when the RXE bit in SICTR is 1. Writing 1 to this bit clears the content. Writing 0 is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: Completion of one frame reception not detected 1: Completion of one frame reception detected
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
5	RFSERR	0	R/W	Receive Frame Synchronization Error A receive frame synchronization error occurs when the next receive frame synchronization timing arrives before the previous data or control data reception has been completed. If a receive frame synchronization error occurs, the MSIOF performs reception only for the slots that can be transferred. This bit is valid when the RXE bit in SICTR is 1. When 1 is written to this bit, it is cleared. Writing 0 to this bit is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: No receive frame synchronization error has occurred 1: A receive frame synchronization error has occurred
4	RFUDF	0	R/W	Receive FIFO Underflow A receive FIFO underflow means that reading of SIRFDR has occurred when the receive FIFO is empty. If a receive FIFO underflow occurs, the value read from SIRFDR is not guaranteed. This bit is valid when the RXE bit in SICTR is 1. When 1 is written to this bit, it is cleared. Writing 0 to this bit is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: No receive FIFO underflow has occurred 1: A receive FIFO underflow has occurred
3	RFOVF	0	R/W	Receive FIFO Overflow A receive FIFO overflow means that writing has been caused by receiving operation when the receive FIFO is full. If a receive FIFO overflow occurs, the receive data causing the overflow is lost. This bit is valid when the RXE bit in SICTR is 1. When 1 is written to this bit, it is cleared. Writing 0 to this bit is invalid. If the interrupt request from this bit is enabled, an MSIOF interrupt is issued. 0: No receive FIFO overflow has occurred 1: A receive FIFO overflow has occurred

Bit	Bit Name	Initial Value	R/W	Descriptions
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.3.17 MSIOF Interrupt Enable Register (MSIOF0_SIIER, MSIOF1_SIIER)

SIIER is a 32-bit readable/writable register that enables the issue of MSIOF interrupts. When each bit in this register is set to 1 and the corresponding bit in SISTR is set to 1, the MSIOF issues an interrupt.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TDMAE	TCRDYE	TFEMPE	TDREQE	—	—	—	—	TEOFE	—	TFSERRE	TFOVFE	TFUDFE	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDMAE	RCRDYE	RFFULE	RDREQE	—	—	—	—	REOFE	—	RFSERRE	RFUDFE	RFOVFE	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31	TDMAE	0	R/W	Transmit Data DMA Transfer Request Enable Specifies whether to send an interrupt as an interrupt request to the CPU or a transfer request to the DMAC. The TDREQE bit can be set as the request source. 0: Sends an interrupt request to the CPU 1: Sends a DMA transfer request to the DMAC
30	TCRDYE	0	R/W	Transmit Control Data Ready Enable 0: Disables interrupts due to transmit control data ready 1: Enables interrupts due to transmit control data ready
29	TFEMPE	0	R/W	Transmit FIFO Empty Enable 0: Disables interrupts due to transmit FIFO empty 1: Enables interrupts due to transmit FIFO empty
28	TDREQE	0	R/W	Transmit Data Transfer Request Enable 0: Disables interrupts due to transmit data transfer requests 1: Enables interrupts due to transmit data transfer requests
27 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
23	TEOFE	0	R/W	Frame Transmission Completion Enable 0: Disables interrupts due to completion of frame transmission 1: Enables interrupts due to completion of frame transmission
22	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
21	TFSERRE	0	R/W	Transmit Frame Synchronization Error Enable 0: Disables interrupts due to transmit frame synchronization errors 1: Enables interrupts due to transmit frame synchronization errors
20	TFOVFE	0	R/W	Transmit FIFO Overflow Enable 0: Disables interrupts due to transmit FIFO overflow 1: Enables interrupts due to transmit FIFO overflow
19	TFUDFE	0	R/W	Transmit FIFO Underflow Enable 0: Disables interrupts due to transmit FIFO underflow 1: Enables interrupts due to transmit FIFO underflow
18 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	RDMAE	0	R/W	Receive Data DMA Transfer Request Enable Specifies whether to send an interrupt as an interrupt request to the CPU or a transfer request to the DMAC. The RDREQE bit can be set as the request source. 0: Sends an interrupt request to the CPU 1: Sends a DMA transfer request to the DMAC
14	RCRDYE	0	R/W	Receive Control Data Ready Enable 0: Disables interrupts due to receive control data ready 1: Enables interrupts due to receive control data ready
13	RFFULE	0	R/W	Receive FIFO Full Enable 0: Disables interrupts due to receive FIFO full 1: Enables interrupts due to receive FIFO full

Bit	Bit Name	Initial Value	R/W	Descriptions
12	RDREQE	0	R/W	Receive Data Transfer Request Enable 0: Disables interrupts due to receive data transfer requests 1: Enables interrupts due to receive data transfer requests
11 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	REOFE	0	R/W	Frame Reception Completion Enable 0: Disables interrupts due to completion of frame reception 1: Enables interrupts due to completion of frame reception
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5	RFSERRE	0	R/W	Receive Frame Synchronization Error Enable 0: Disables interrupts due to receive frame synchronization errors 1: Enables interrupts due to receive frame synchronization errors
4	RFUDFE	0	R/W	Receive FIFO Underflow Enable 0: Disables interrupts due to receive FIFO underflow 1: Enables interrupts due to receive FIFO underflow
3	RFOVFE	0	R/W	Receive FIFO Overflow Enable 0: Disables interrupts due to receive FIFO overflow 1: Enables interrupts due to receive FIFO overflow
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.3.18 MSIOF FIFO Control Register (MSIOF0_SIFCTR, MSIOF1_SIFCTR)

SIFCTR is a 32-bit readable/writable register that indicates the area available for the transmit/receive FIFO transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TFWM[2:0]			—	—	TFUA[6:0]							—	—	—	—
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFWM[2:0]			—	—	RFUA[6:0]							—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Descriptions
31 to 29	TFWM[2:0]	000	R/W	Transmit FIFO Watermark A transfer request of the transmit FIFO is issued by the TDREQE bit in SIIER. The transmit FIFO always operates as a 64-stage FIFO regardless of these bit settings. 000: Issues a transfer request when 64 stages of the transmit FIFO are empty. 001: Issues a transfer request when 32 or more stages of the transmit FIFO are empty. 010: Issues a transfer request when 24 or more stages of the transmit FIFO are empty. 011: Issues a transfer request when 16 or more stages of the transmit FIFO are empty. 100: Issues a transfer request when 12 or more stages of the transmit FIFO are empty. 101: Issues a transfer request when 8 or more stages of the transmit FIFO are empty. 110: Issues a transfer request when 4 or more stages of the transmit FIFO are empty. 111: Issues a transfer request when 1 or more stages of transmit FIFO are empty.
28, 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Descriptions
26 to 20	TFUA[6:0]	H'40	R	Transmit FIFO Usable Area Indicate the number of words that can be transferred by the CPU or DMAC as B'000000 (full) to B'100000 (empty).
19 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 13	RFWM[2:0]	000	R/W	Receive FIFO Watermark A transfer request of the receive FIFO is issued by the RDREQE bit in SIIER. The receive FIFO always operates as a 32-stage FIFO regardless of these bit settings. 000: Issues a transfer request when 1 stage or more of the receive FIFO are valid. 001: Issues a transfer request when 4 or more stages of the receive FIFO are valid. 010: Issues a transfer request when 8 or more stages of the receive FIFO are valid. 011: Issues a transfer request when 16 or more stages of the receive FIFO are valid. 100: Issues a transfer request when 24 or more stages of the receive FIFO are valid. 101: Issues a transfer request when 32 or more stages of the receive FIFO are valid. 110: Issues a transfer request when 48 or more stages of the receive FIFO are valid. 111: Issues a transfer request when 64 stages of the receive FIFO are valid.
12, 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 4	RFUA[6:0]	H'00	R	Receive FIFO Usable Area Indicate the number of words that can be transferred by the CPU or DMAC as B'0000000 (empty) to B'1000000 (full).
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

25.4 Operation

25.4.1 Operating Mode

(1) Independent Transmit/Receive Mode

Dedicated clock and frame synchronization signals can be specified each for transmission and reception.

Master mode: MSIOFTSCK, MSIOFTSYNC, MSIOFRSCK, and MSIOFRSYNC are output.

Slave mode: MSIOFTSCK, MSIOFTSYNC, MSIOFRSCK, and MSIOFRSYNC are input.

(2) Common Transmit/Receive Mode

Common clock and frame synchronization signals are used for transmission and reception.

Master mode: MSIOFTSCK (MSIOFSCK) and MSIOFTSYNC (MSIOFSYNC) are output.

Slave mode: MSIOFTSCK (MSIOFSCK) and MSIOFTSYNC (MSIOFSYNC) are input.

25.4.2 Serial Clocks

(1) Clock Output in Master Mode

In master mode, the baud rate generator is used to generate the serial clock. The division ratio is from 1/1 to 1/1024.

(2) Clock Input in Slave Mode

In slave mode, the clock input for transmission and reception is used as the serial clock.

(3) Multiple Channel Function

This module provides the multiple channel function as shown in Figure 25.2.

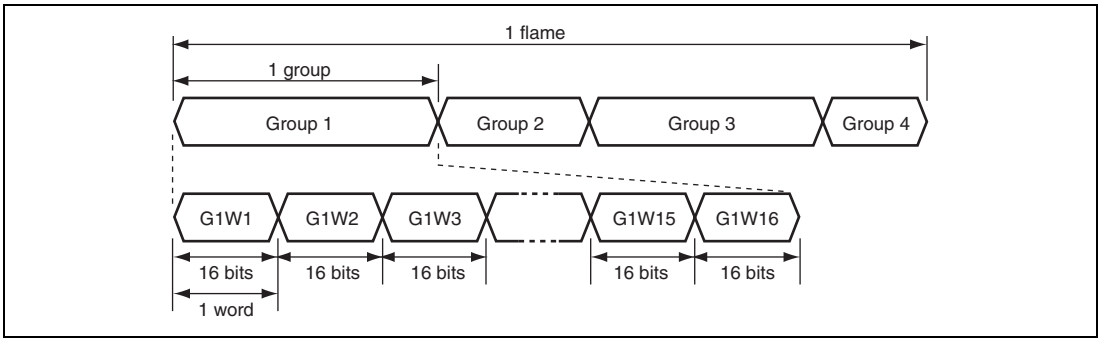


Figure 25.2 Multiple Channel Structure

The following conditions can be specified for the multiple channel function.

- Up to four groups in one frame
- Up to 256 words in one group
- Eight bits to 32 bits in one word
- Up to three blank bits can be inserted between groups

Note the following limitations by the number of groups.

Table 25.4 Maximum Word Size and Group Size Limited by Number of Groups

Number of Groups	Maximum Word Size	Maximum Group Size
One or two	32 bits	256 words
Three or four	16 bits	16 words

25.4.3 Serial Timing

(1) MSIOFTSYNC and MSIOFRSYNC

The MSIOFTSYNC and MSIOFRSYNC are frame synchronization signals. The following four modes are available.

- Frame start synchronization pulse: 1-bit-width pulse indicating the start of a frame
- Last bit synchronization pulse: 1-bit-width pulse indicating the last bit in a word
- Level/SPI: Level signal asserted during frame transmission
- L/R: 1/2-frame-width pulse indicating the first-half groups in a high level and the last-half groups in a low level

Figures 25.3 to 25.6 show the MSIOFTSYNC and MSIOFRSYNC synchronization timing in these modes.

(a) Frame Start Synchronization Pulse

The rising edge of the synchronization pulse indicates the start of a frame.

The delay between the rising edge of the synchronization signal and the start of data transmission or reception can be specified with the DTDL bits. The width of the synchronization pulse can be extended through the SYNCDL bit setting.

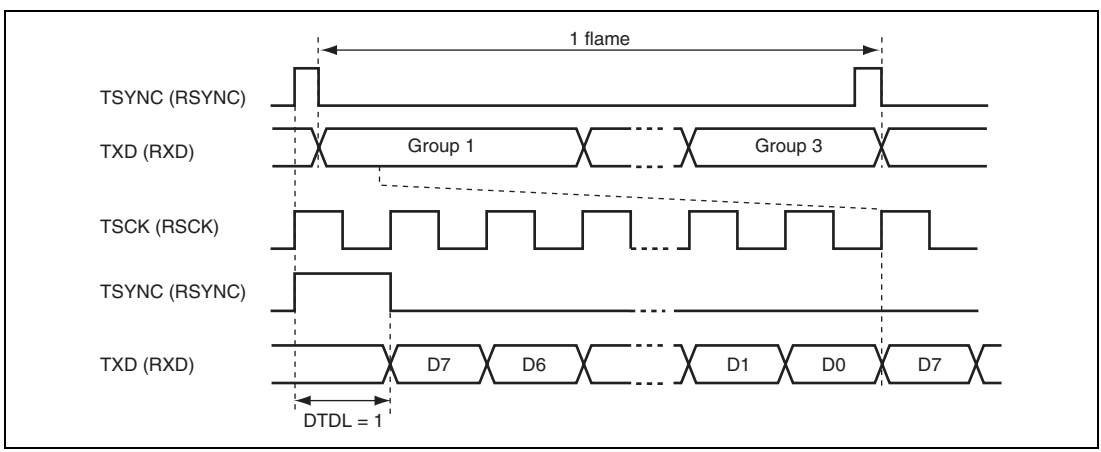


Figure 25.3 Synchronization Timing in Start Frame Synchronization Pulse Mode

(b) Last Bit Synchronization Pulse

The rising edge of the synchronization pulse indicates the last bit of each word.

The delay between the last bit of a word and the synchronization pulse can be specified with the DTDL bits.

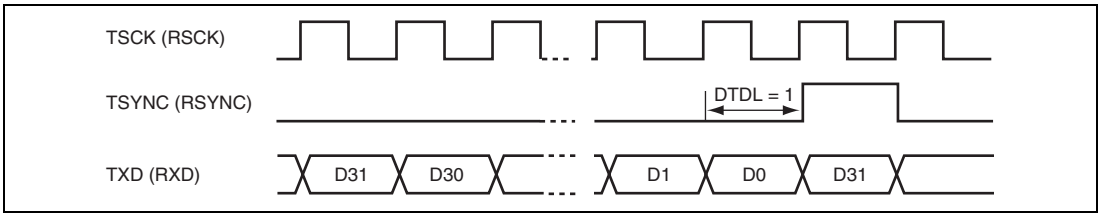


Figure 25.4 Synchronization Timing in Last Bit Synchronization Pulse Mode

(c) Level Synchronization

The synchronization signal is driven high for the frame length.

The delay between the rising edge of the synchronization signal and the start of transmission or reception can be specified with the DTDL bits and the delay between the end of transmission or reception and the falling edge of the synchronization signal can be specified with the SYNCDL bits (DTDL = 0 to 2 and SYNCDL = 0 to 3). This mode is available for the SPI master or slave operation.

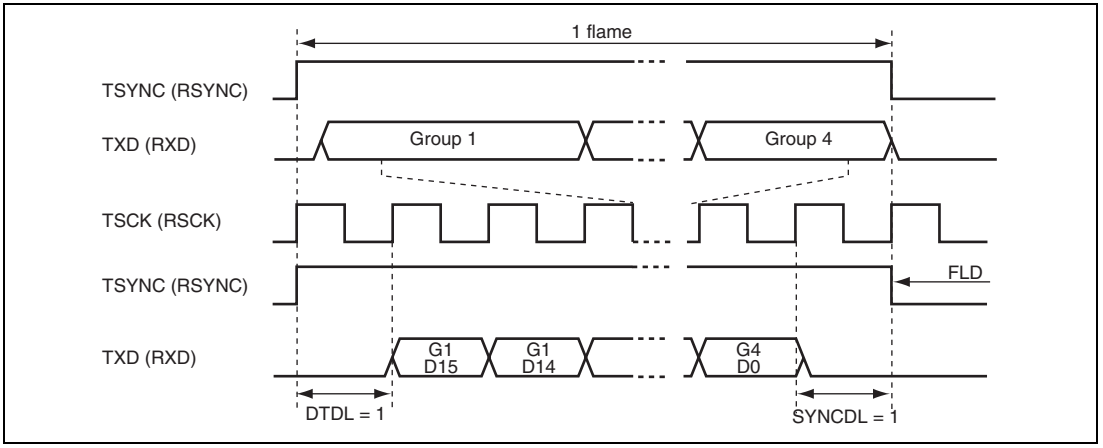


Figure 25.5 Synchronization Timing in Level Synchronization Mode

(d) L/R Synchronization

A high level in the synchronization signal indicates the first-half groups and a low level indicates the last-half groups.

The delay between the rising or falling edge of the synchronization signal and the start of transmission or reception can be specified with the DTDL bits and the delay between the end of transmission or reception and the falling or rising edge of the synchronization signal can be specified with the SYNCDL bits (DTDL = 0 to 2 and SYNCDL = 0 to 3).

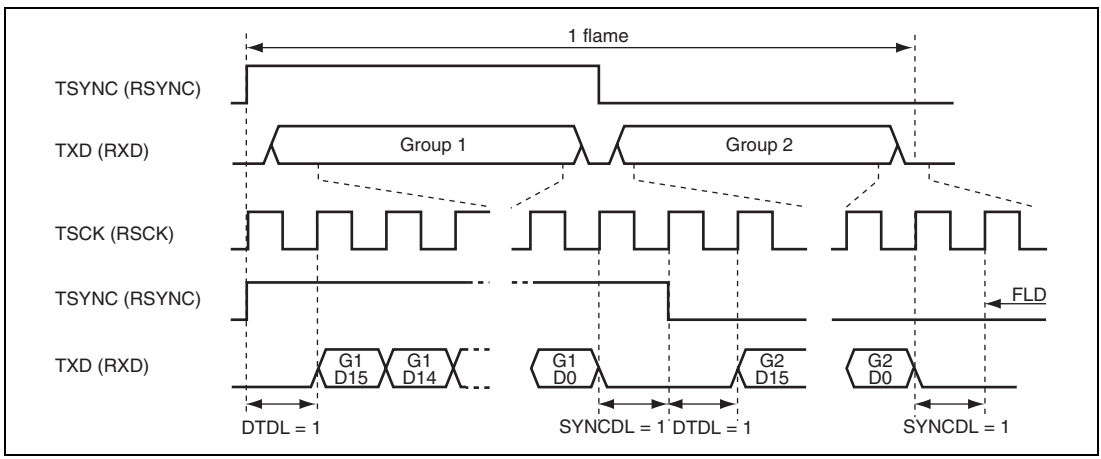


Figure 25.6 Synchronization Timing in L/R Synchronization Mode

(2) Transmit/Receive Timing

The timing of MSIOFTXD transmission relative to MSIOFTSCK and of MSIOFRXD reception relative to MSIOFRSCK can be specified as sampling on either edge, as listed below. The respective settings are made in TEDG and REDG bits of SICTR.

- Falling-edge sampling
- Rising-edge sampling

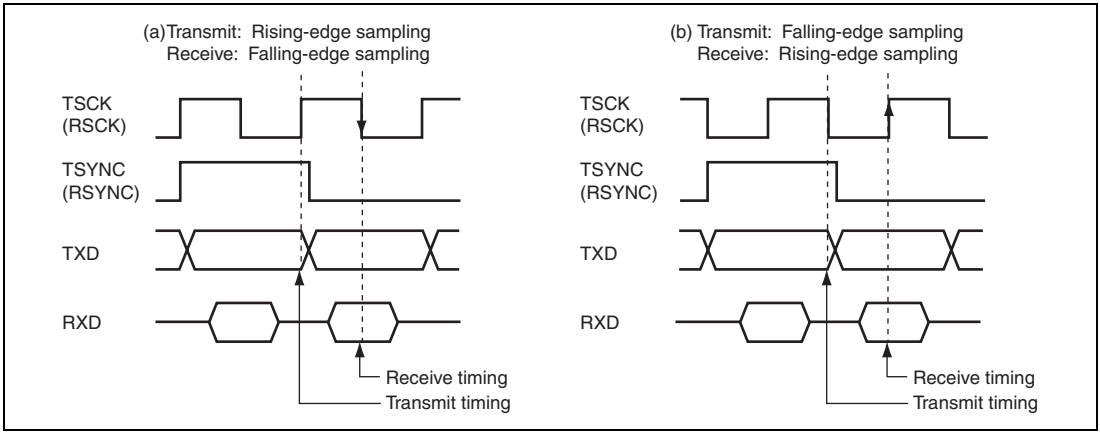


Figure 25.7 Transmit/Receive Timing.

25.4.4 Control Data Interface

For the groups specified with the CRDE bits in SITMDR2 and SIRMDR2, data is not stored in the FIFOs but handled as control data to be stored in SITDR1/SITDR2 and SIRDR1/SIRDR2.

Control data is transferred in the following two ways, which can be selected through the CRDNUM bits in SITMDR2 and SIRMDR2, as shown in Figure 25.8.

- Only the first word in the group is transferred as valid data.
- All words in the group is transferred as valid data.

For the invalid data period during transmission, the value specified in the TXDIZ bit is output. The invalid receive data is not stored in SIRDR1 or SIRDR2.

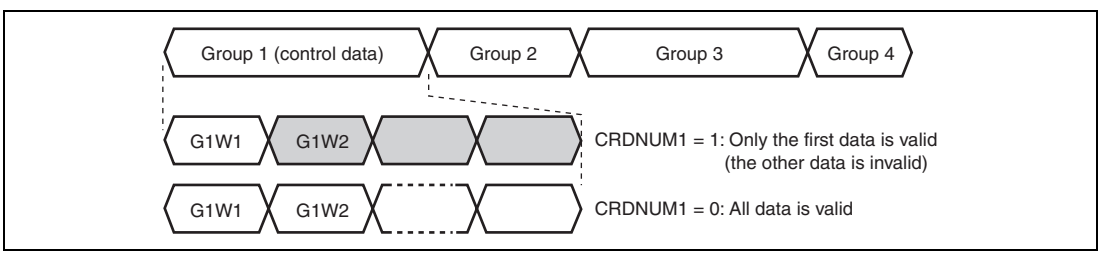


Figure 25.8 Control Data Allocation

25.4.5 Transfer Data Allocation to Registers

(1) Transmit/Receive Data

Transmit/receive data (FIFO data and control data) should be written to or read from the following registers.

- Transmit FIFO data writing: SITFDR (32-bit/16-bit access)
- Control data writing: SITDR1 and SITDR2 (32-bit/16-bit access)
- Receive FIFO data reading: SIRFDR (32-bit/16-bit access)
- Control data reading: SIRDR1 and SIRDR2 (32-bit/16-bit access)

For 16-bit accesses to FIFO data, only the upper 16 bits can be accessed. For control data, both the upper 16 bits and lower 16 bits can be accessed.

If the word bit length is shorter than the transmit FIFO data access width, the upper bits are transmitted as valid data. For example, when data is written to the transmit FIFO with 32-bit accesses and 16-bit words are transmitted, the upper 16 bits will be transmitted and the lower 16 bits will be discarded as invalid data.

Similarly, if the word bit length is longer than the receive FIFO data access width, the upper bits are received as valid data. For example, when 32-bit length data is written to the receive FIFO and it is accessed in 16 bits, the lower 16 bits are discarded as invalid data.

Figure 25.9 shows the bit alignment of these registers.

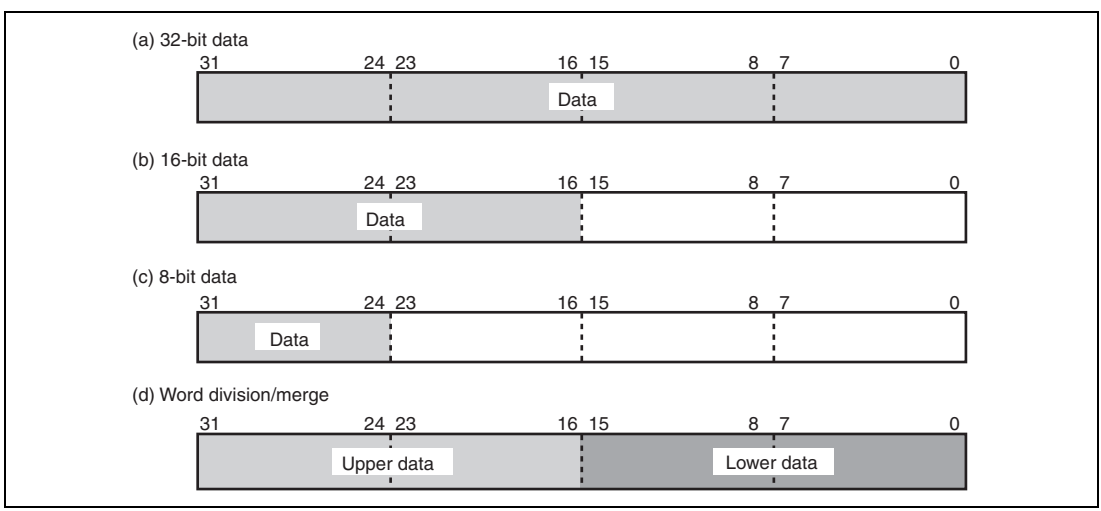


Figure 25.9 Transmit/Receive Data Bit Alignment

25.4.6 FIFO

(1) Overview

The transmit and receive FIFOs of the MSIOF have the following features.

- 32 bits × 64 stages for transmission and 32 bits × 64 stages for reception
- The FIFO pointer is updated in one read or write cycle regardless of the access size of the CPU or DMAC. (One access cannot be divided into multiple accesses.)

(2) Transfer Request

A request for FIFO data transfer can be issued to the CPU or DMAC as the following interrupt sources.

- FIFO transmit request: TDREQ (interrupt source for transmission)
- FIFO receive request: RDREQ (interrupt source for reception)

The conditions for requesting FIFO data transfer can be specified separately for transmission and reception. The conditions for the transmit FIFO and receive FIFO are specified in the TFWM[2:0] bits and RFWM[2:0] bits in SIFCTR, respectively.

Table 25.5 Condition for Issuing Transmit Request

TFWM[2:0]	Number of Requested Stages	Condition for Transmit Request	Areas Used
000	1	64 stages of empty area	Smallest
001	32	32 or more stages of empty area	
010	40	24 or more stages of empty area	
011	48	16 or more stages of empty area	
100	52	12 or more stages of empty area	
101	56	8 or more stages of empty area	
110	60	4 or more stages of empty area	
111	64	1 or more stages of empty area	Largest

Table 25.6 Condition for Issuing Receive Request

RFWM[2:0]	Number of Requested Stages	Condition for Receive Request	Areas Used
000	1	1 or more stages of valid data	Smallest
001	4	4 or more stages of valid data	
010	8	8 or more stages of valid data	
011	16	16 or more stages of valid data	
100	24	24 or more stages of valid data	
101	32	32 or more stages of valid data	
110	48	48 or more stages of valid data	
111	64	64 stages of valid data	Largest

The maximum FIFO stages are always available even if the data area or empty area exceeds the size specified for the transfer condition. Accordingly, an overflow error or an underflow error occurs if the data area or empty area exceeds the maximum number of FIFO stages.

The FIFO transfer request is canceled when the specified condition is not satisfied even if the FIFO does not become empty or full.

(3) Number of FIFOs

The number of transmit and receive FIFO stages used are indicated by the following registers.

- Transmit FIFO: The number of empty FIFO stages is indicated by the TFUA[6:0] bits in SIFCTR.
- Receive FIFO: The number of valid data stages is indicated by the RFUA[6:0] bits in SIFCTR.

These registers show the number of data stages that can be transferred by the CPU or DMAC.

25.4.7 Transmit and Receive Procedures

(1) Transmission in Master Mode

Figure 25.10 shows an example of settings and operation for transmission in master mode.

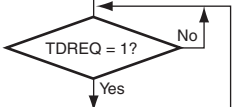
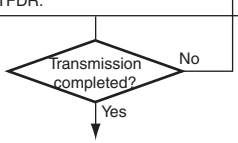
No.	Flow chart	MSIOF settings	MSIOF operation
1	Set SITMDR, SITSCR, SIFCTR, and SIIER.	Set the operating mode, serial clock, FIFO threshold value, and enable interrupts.	
2	Set the TSCKE bit in SICTR to 1.	Set output sampling and start the baud rate generator.	Outputs the serial clock.
3	Read 1 from the TSCKE bit.		
4	Set SITFDR and SITDR1/2.	Set transmit data.	
5	Set the TXE bit in SICTR to 1	Enable transmission.	
6	Read 1 from the TXE bit.		
7	Set the TFSE bit in SICTR to 1.	Start outputting frame synchronization signal.	Outputs the frame synchronization signal.
8	Read 1 from the TFSE bit.		
9	Transmit the SITFDR and SITDR1/2 contents through TXD in synchronization with MSIOFTSYNC.		Transmits data.
10			Issues a transmit data transfer request according to the FIFO threshold value.
11	Set SITFDR.	Set transmit data.	
12	 Clear the TXE and TFSE bits to 0.	Disable transmission and output of frame synchronization signal.	Terminates transmission.

Figure 25.10 Example of Transmit Procedure in Master Mode

(2) Reception in Master Mode

Figure 25.11 shows an example of settings and operation for reception in master mode.

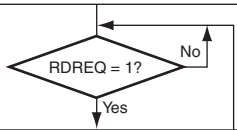
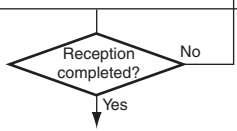
No.	Flow chart	MSIOF settings	MSIOF operation
1	Set SIRMDR1/2/3, SIRSCR, SIFCTR, and SIIER.	Set the operating mode, serial clock, and FIFO threshold value, and enable interrupts.	
2	Set the RSCKE bit in SICTR to 1.	Set output sampling and start the baud rate generator.	Outputs the serial clock.
3	Read 1 from the RSCKE bit.		
4	Set the RXE bit in SICTR to 1	Enable transmission.	
5	Read 1 from the RXE bit.		
6	Set the RFSE bit in SICTR to 1.	Start outputting frame synchronization signal.	Outputs the frame synchronization signal.
7	Read 1 from the RFSE bit.		
8	Receive data through RXD in synchronization with MSIOFRSYNC and store it in SIRFDR and SIRDR1/2.		Receives data.
9			Issues a receive data transfer request according to the FIFO threshold value.
10	Read SIRFDR.	Read receive data.	
11	 Clear the RXE and RFSE bits to 0.	Disable reception and output of frame synchronization signal.	Terminates reception.

Figure 25.11 Example of Receive Procedure in Master Mode

(3) Transmission in Slave Mode

Figure 25.12 shows an example of settings and operation for transmission in slave mode.

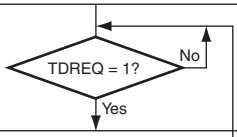
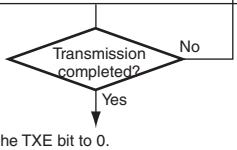
No.	Flow chart	MSIOF settings	MSIOF operation
1	Set SITMDR, SIFCTR, and SIIER.	Set the operating mode and FIFO threshold value, and enable interrupts.	
2	Set SITFDR and SITDR1/2.	Set transmit data.	
3	Set the TXE bit in SICTR to 1.	Enable transmission.	
4	Read 1 from the TXE bit.		
5	Transmit the SITFDR and SITDR1/2 contents through TXD in synchronization with MSIOFTSYNC.		Transmits data.
6			Issues a transmit data transfer request according to the FIFO threshold value.
7	Set SITFDR.	Set transmit data.	
8	 Clear the TXE bit to 0.	Disable transmission.	Terminates transmission.

Figure 25.12 Example of Transmit Procedure in Slave Mode

(4) Reception in Slave Mode

Figure 25.13 shows an example of settings and operation for reception in slave mode.

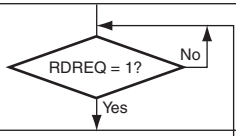
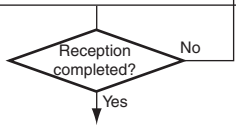
No.	Flow chart	MSIOF settings	MSIOF operation
1	Set SIRMDR1/2/3, SIFCTR, and SIIER.	Set the operating mode and FIFO threshold value, and enable interrupts.	
2	Set the RXE bit in SICTR to 1.	Set output sampling and enable reception.	
3	Receive data through RXD in synchronization with MSIOFRSYNC and store it in SIRFDR and SIRDR1/2.		Receives data.
4			Issues a receive data transfer request according to the FIFO threshold value.
5	Read SIRFDR.	Read receive data.	
6	 Clear the RXE and RFSE bits to 0.	Disable reception and output of frame synchronization signal.	Terminates reception.

Figure 25.13 Example of Receive Procedure in Slave Mode

(5) Reset

After a power-on reset is applied, module stop state is canceled, or a reset signal is asserted through the TXRST or RXRST setting in SICTR, it takes about 20 cycles by Bϕ to complete the internal reset of the module. During this period, do not modify the control register or access the FIFOs.

The MSIOF can separately reset the transmit and receive units by setting the following bits to 1.

- Transmit reset: TXRST bit in SICTR
- Receive reset: RXRST bit in SICTR

Table 25.7 Transmit and Receive Reset

Type	Objects Initialized
Transmit reset	SITDR1, SITDR2, and SITFDR registers
	Transmit FIFO write pointer
	TCRDY, TFEMP, and TDREQ bits in SISTR
	TXE bit in SICTR
Receive reset	SIRDR1, SIRDR2, and SIRFDR registers
	Receive FIFO read pointer
	RCRDY, RFFUL, and RDREQ bits in SISTR
	RXE bit in SICTR

(6) Initial Operating Mode

After a power-on reset, both the transmit and receive units are initialized to master mode (0 is output through CLK and SYNC). When using slave mode, keep the connected device outputting 0 until the operating mode setting is completed.

25.4.8 Interrupts

The MSIOF has one type of interrupt.

(1) Interrupt Sources

Interrupts can be generated by several sources. Each source is shown as an MSIOF status in SISTR. Table 25.8 lists the MSIOF interrupt sources.

Table 25.8 MSIOF Interrupt Sources

No.	Classification	Bit Name	Function Name	Description
1	Transmission	TDREQ	Transmit FIFO transfer request	The transmit FIFO has empty space of specified size or more.
2		TFEMP	Transmit FIFO empty	The transmit FIFO is empty.
3	Reception	RDREQ	Receive FIFO transfer request	The receive FIFO stores data of specified size or more.
4		RFFUL	Receive FIFO full	The receive FIFO is full.
5	Control	TCRDY	Transmit control data ready	The transmit control data register is ready to be written.
6		RCRDY	Receive control data ready	The receive control data register stores valid data.
7	Error	TFUDF	Transmit FIFO underflow	Serial data transmit timing has arrived while the transmit FIFO is empty.
8		TFOVF	Transmit FIFO overflow	The transmit FIFO is written to while it is full.
9		RFOVF	Receive FIFO overflow	Serial data is received while the receive FIFO is full.
10		RFUDF	Receive FIFO underflow	The receive FIFO is read while the receive FIFO is empty.
11		TFSERR	Transmit FS error	A transmit frame synchronization signal is input before the specified number of bits has been reached (in slave mode).
12		RFSERR	Receive FS error	A receive frame synchronization signal is input before the specified number of bits has been reached (in slave mode).

Whether an interrupt is issued from an interrupt source is determined by the SIIER settings. If an interrupt source bit is set to 1 and the corresponding bit in SIIER is set to 1, an MSIOF interrupt is issued.

(2) Regarding Transmit and Receive Classification

The transmit and receive interrupt sources are signals indicating the state; after being set, if the state changes, they are automatically cleared by the MSIOF.

When the DMA transfer is used, a DMA transfer request signal is pulled low for one cycle at the end of DMA transfer.

(3) Processing when Errors Occur

- Transmit FIFO underflow (TFUDF)
  The value specified with TXDIZ in SICTR is output.
- Transmit FIFO overflow (TFOVF)
  The contents of the transmit FIFO are protected, and the write operation causing the overflow is ignored.
- Receive FIFO overflow (RFOVF)
  Data causing the overflow is discarded and lost.
- Receive FIFO underflow (RFUDF)
  An undefined value is output on the bus.
- Transmit FS error (TFSERR)
  The internal counter is reset according to the synchronization signal in which an error occurs.
- Receive FS error (RFSERR)
  The internal counter is reset according to the synchronization signal in which an error occurs.

25.4.9 Transmit and Receive Timing

Examples of the MSIOF serial transmission and reception are shown in figures 25.14 and 25.15.

(1) 16-Bit Synchronization Pulse

Synchronization pulse method, one group, one word, 16 bits in a word, no bit delay

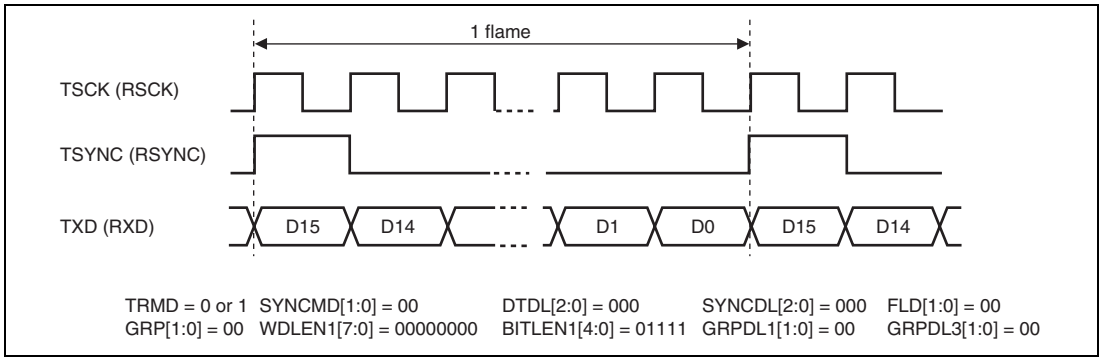


Figure 25.14 Transmit and Receive Timing (16 Bits)

(2) 32-Bit Synchronization Pulse

Synchronization pulse method, one group, 32 words, 32 bits in a word, 1-bit delay

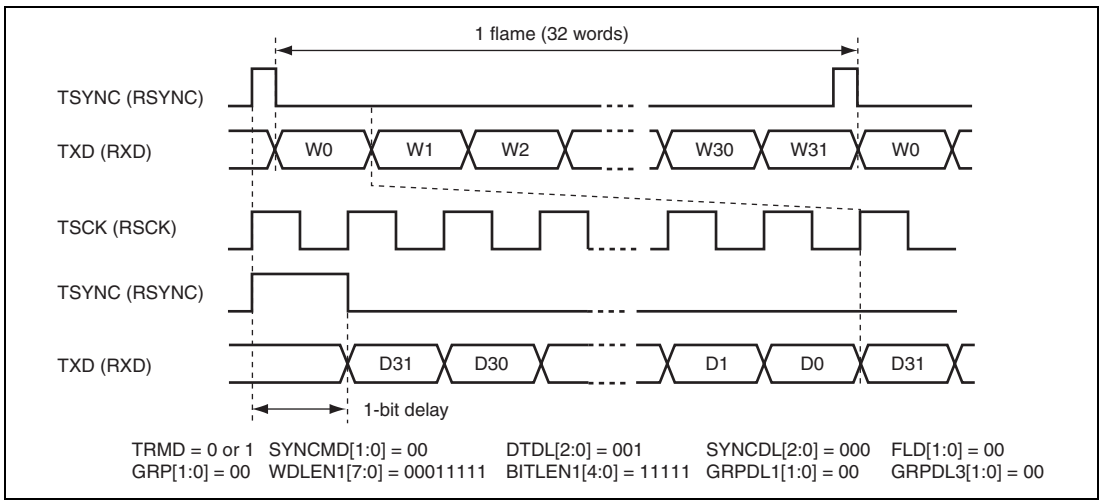


Figure 25.15 Transmit and Receive Timing (32 Bits)

(3) 24-Bit IIS Transmit

L/R method, two groups, one for each word, 24 bits in a word, 1-bit delay.

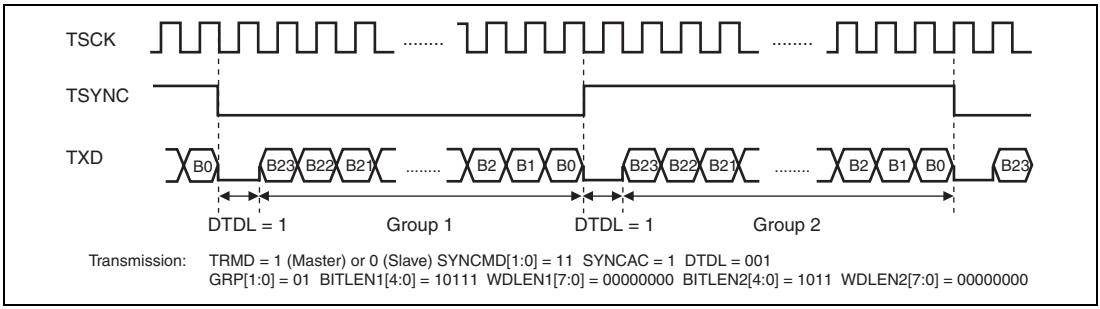


Figure 25.16 Transmit Timing (24 Bits IIS)

(4) µ-WIRE (Transmit Only)

Level method, two groups, one for each word, 8/16 bits in a word, 1-bit delay

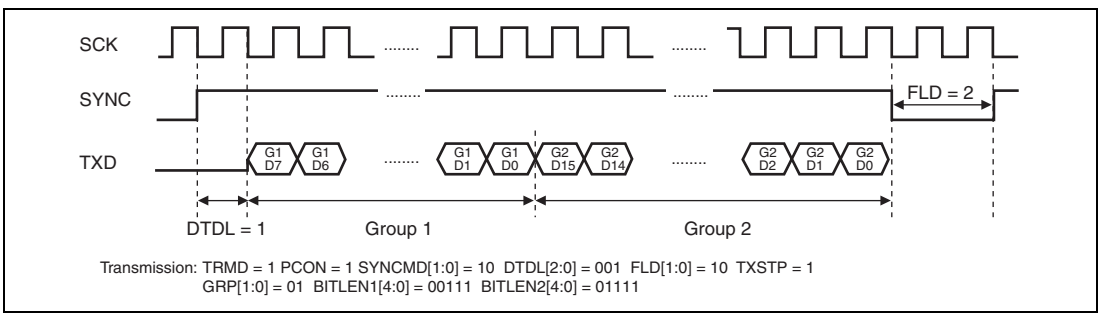


Figure 25.17 Transmit Timing (µ-WIRE transmit)

(5) µ-WIRE (Switches for Transmission and Reception)

Level method, two groups, one for each word, 16/32 bits in a word, 1-bit delay.

One output mask for transmission groups, two input masks for reception group.

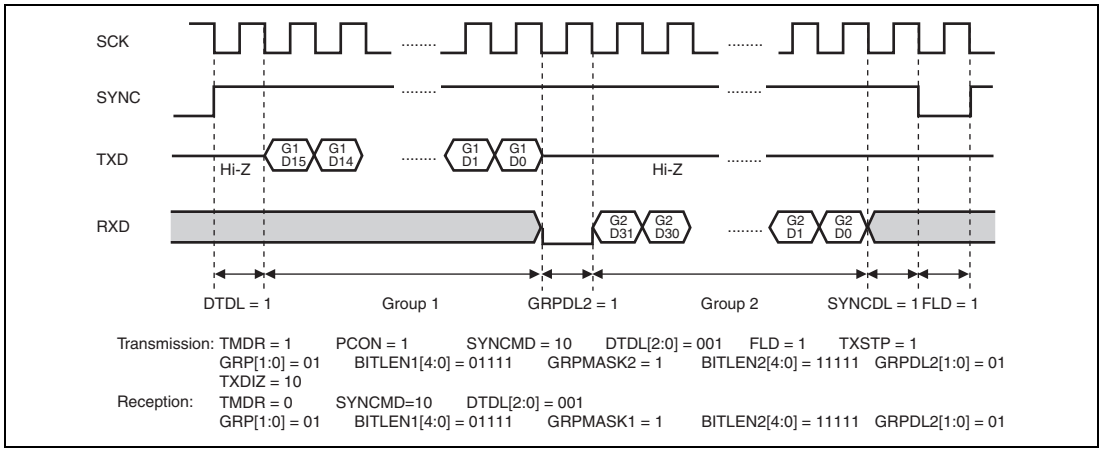


Figure 25.18 Transmit and Receive Timing (µ-WIRE)

25.4.10 SPI

The MSIOF can be used as an SPI device by setting the registers appropriately.

(1) Example of SPI Device Connection

Figure 25.19 shows an example of connection with an SPI device.

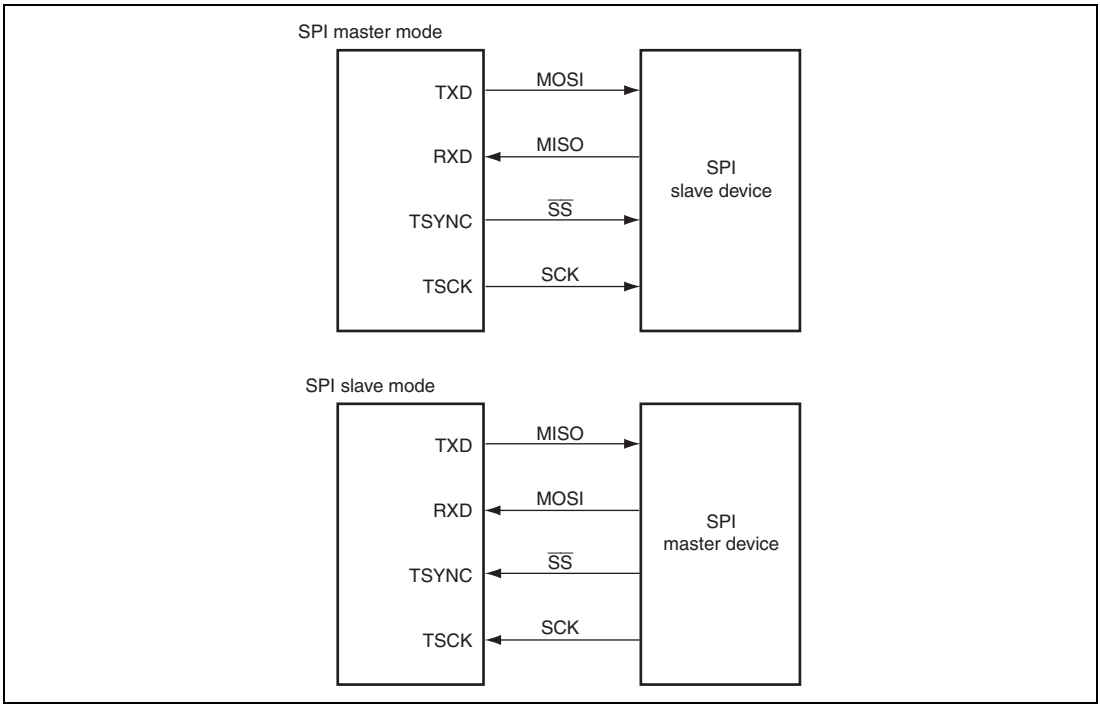
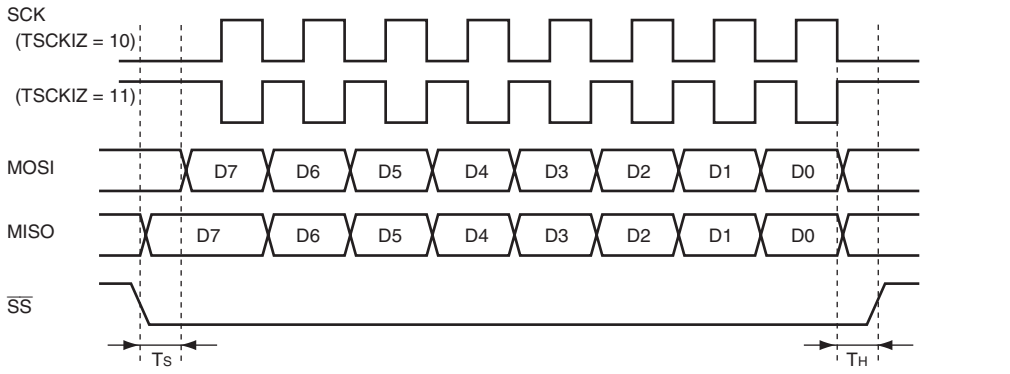


Figure 25.19 Example of SPI Device Connection

(2) SPI Serial Clock Timing

Figures 25.20 and 25.21 show the data and clock timing in SPI mode. As shown in the figures, four types of serial transfer formats can be used.



Transmit: TRMD = 1 (master) or 0 (slave) PCON = 1 SYNCMD[1:0] = 10 SYNCAC = 1
GRP[1:0] = 00 BITLEN1[4:0] = 00111 WDLEN1[7:0] = 00000000 GRPDL1[1:0] = 00 GRPDL3[1:0] = 00

Receive: TRMD = 0 SYNCMD[1:0] = 10 SYNCAC = 1
GRP[1:0] = 00 BITLEN1[4:0] = 00111 WDLEN1[7:0] = 00000000 GRPDL1[1:0] = 00 GRPDL3[1:0] = 00

Set DTDL to satisfy T_S (0 to 2 can be specified in 0.5-clock-cycle units)
Set SYNCDL to satisfy T_H (0 to 2 can be specified in 0.5-clock-cycle units)
The sum of DTDL and SYNCDL settings should be an integer.

SICTR: Upper: TSCKIZ[1:0] = 10, RSCKIZ[1:0] = 10, TEDG = 1, REDG = 1
Lower: TSCKIZ[1:0] = 11, RSCKIZ[1:0] = 11, TEDG = 0, REDG = 0

Figure 25.20 SPI Clock and Data Timing 1

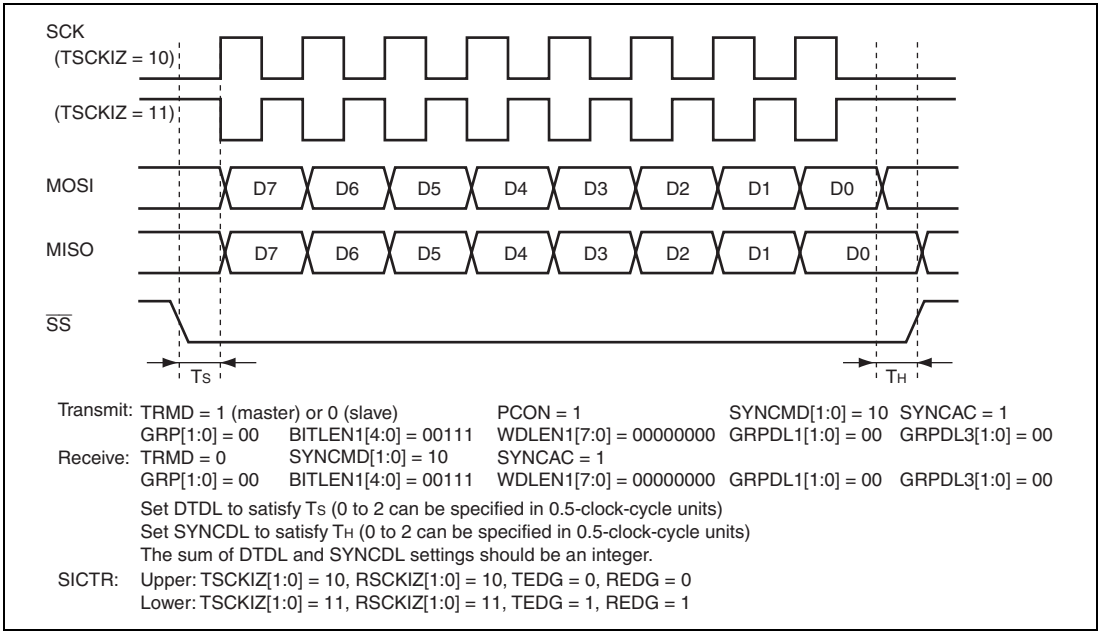


Figure 25.21 SPI Clock and Data Timing 2

25.4.11 Usage Notes for SPI Mode

1. In SPI master mode, the FLD bits in SITMDR1 should be set so that four or more Bϕ cycles are secured.
2. In SPI slave mode, intervals between frame synchronization signals should be set so that four or more Bϕ cycles are secured.
3. In SPI mode, SICTR should be set again after setting the bits related to input (RSCKIZ, REDG) in SICTR to 0 respectively when canceling module stop or resetting output/input.

Section 26 Serial Communication Interface with FIFO (SCIF)

This LSI has a three-channel serial communication interface with FIFO (SCIF) that supports both asynchronous and clock synchronous serial communication. It also has 16-stage FIFO registers for both transmission and reception independently for each channel that enable this LSI to perform efficient high-speed continuous communication.

26.1 Features

- Asynchronous serial communication:
  - Serial data communication is performed by start-stop in character units. The SCIF can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are eight selectable serial data communication formats.
  - Data length: 7 or 8 bits
  - Stop bit length: 1 or 2 bits
  - Parity: Even, odd, or none-
  - Receive error detection: Parity, framing, and overrun errors
  - Break detection: Break is detected when a framing error is followed by at least one frame at the space 0 level (low level).
- Clock synchronous serial communication:
  - Serial data communication is synchronized with a clock signal. The SCIF can communicate with other chips having a clock synchronous communication function. There is one serial data communication format.
  - Data length: 8 bits
  - Receive error detection: Overrun errors
- Full duplex communication: The transmitting and receiving sections are independent, so the SCIF can transmit and receive simultaneously. Both sections use 16-stage FIFO buffering, so high-speed continuous data transfer is possible in both the transmit and receive directions.
- On-chip baud rate generator with selectable bit rates

- Internal or external transmit/receive clock source: From either baud rate generator (internal) or SCK pin (external)
- Four types of interrupts: Transmit-FIFO-data-empty interrupt, break interrupt, receive-FIFO-data-full interrupt, and receive-error interrupts are requested independently on each channel.
- When the transmit FIFO is empty or the receive FIFO contains any received data, the DMA controller (DMAC) can be activated to perform data transfer by generating a DMA transfer request.
- When the SCIF is not in use, it can be stopped by halting the clock supplied to it, saving power.
- The quantity of data in the transmit and receive FIFO data registers and the number of receive errors of the received data in the receive FIFO data register can be ascertained.
- A time-out error (DR) can be detected when receiving in asynchronous mode.

Figure 26.1 shows a block diagram of the SCIF.

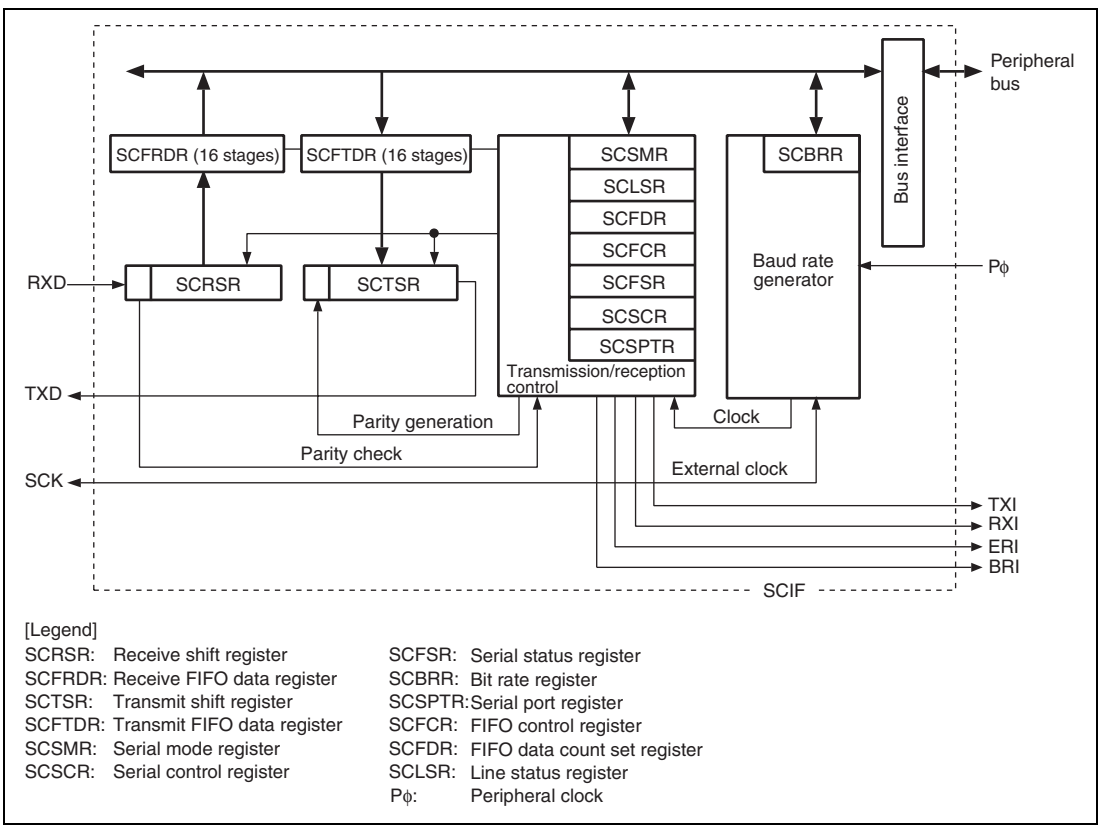


Figure 26.1 Block Diagram of SCIF

26.2 Input/Output Pins

Table 26.1 shows the pin configuration of the SCIF.

Table 26.1 Pin Configuration

Channel	Pin Name	Function	I/O	Description
0	SCIF0_TXD	Transmit data	Output	Transmit data pin
	SCIF0_RXD	Received data	Input	Received data pin
	SCIF0_SCK	Serial clock	I/O	Clock I/O pin
1	SCIF1_TXD	Transmit data	Output	Transmit data pin
	SCIF1_RXD	Received data	Input	Received data pin
	SCIF1_SCK	Serial clock	I/O	Clock I/O pin
2	SCIF2_TXD	Transmit data	Output	Transmit data pin
	SCIF2_RXD	Received data	Input	Received data pin
	SCIF2_SCK	Serial clock	I/O	Clock I/O pin

Note: In the following descriptions, channel numbers in pin names and signal names are omitted and TXD, RXD, and SCK are used as generic terms.

26.3 Register Descriptions

The SCIF has the following registers.

Table 26.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Serial mode register 0	SCSMR0	R/W	H'FFE0 0000	16
Bit rate register 0	SCBRR0	R/W	H'FFE0 0004	8
Serial control register 0	SCSCR0	R/W	H'FFE0 0008	16
Transmit FIFO data register 0	SCFTDR0	W	H'FFE0 000C	8
Serial status register 0	SCFSR0	R/W*[1]	H'FFE0 0010	16
Receive FIFO data register 0	SCFRDR0	R	H'FFE0 0014	8
FIFO control register 0	SCFCR0	R/W	H'FFE0 0018	16
FIFO data count register 0	SCFDR0	R	H'FFE0 001C	16
Line status register 0	SCLSR0	R/W*[2]	H'FFE0 0024	16
Serial mode register 1	SCSMR1	R/W	H'FFE1 0000	16
Bit rate register 1	SCBRR1	R/W	H'FFE1 0004	8
Serial control register 1	SCSCR1	R/W	H'FFE1 0008	16
Transmit FIFO data register 1	SCFTDR1	W	H'FFE1 000C	8
Serial status register 1	SCFSR1	R/W*[1]	H'FFE1 0010	16
Receive FIFO data register 1	SCFRDR1	R	H'FFE1 0014	8
FIFO control register 1	SCFCR1	R/W	H'FFE1 0018	16
FIFO data count register 1	SCFDR1	R	H'FFE1 001C	16
Line status register 1	SCLSR1	R/W*[2]	H'FFE1 0024	16
Serial mode register 2	SCSMR2	R/W	H'FFE2 0000	16
Bit rate register 2	SCBRR2	R/W	H'FFE2 0004	8
Serial control register 2	SCSCR2	R/W	H'FFE2 0008	16
Transmit FIFO data register 2	SCFTDR2	W	H'FFE2 000C	8
Serial status register 2	SCFSR2	R/W*[1]	H'FFE2 0010	16
Receive FIFO data register 2	SCFRDR2	R	H'FFE2 0014	8
FIFO control register 2	SCFCR2	R/W	H'FFE2 0018	16
FIFO data count register 2	SCFDR2	R	H'FFE2 001C	16
Line status register 2	SCLSR2	R/W*[2]	H'FFE2 0024	16

Table 26.3 Register States in Each Operating Mode

Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-standby	U-standby	Sleep
SCSMR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCBRR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCSCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFTDR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFSR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFRDR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFDR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCLSR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCSMR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCBRR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCSCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFTDR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFSR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFRDR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFDR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCLSR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCSMR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCBRR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCSCR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFTDR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFSR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFRDR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFCR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCFDR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SCLSR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

26.3.1 Receive Shift Register (SCRSR)

SCRSR receives serial data. Data input at the RXD pin is loaded into SCRSR in the order received, LSB (bit 0) first, converting the data to parallel form. When one byte has been received, it is automatically transferred to SCFRDR.

The CPU cannot read or write to SCRSR directly.

26.3.2 Receive FIFO Data Register (SCFRDR)

SCFRDR is an 8-bit length16-stage FIFO register that stores serial received data. The SCIF completes the reception of one byte of serial data by moving the received data from SCRSR into SCFRDR for storage. Thereafter, SCRSR becomes ready for next reception and continuous reception is possible until 16 bytes are stored, which makes SCFRDR full. The CPU can read but not write to SCFRDR. If data is read when there is no received data in the SCFRDR, the value is undefined.

When SCFRDR is full of received data, subsequent serial data is lost.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SCFRD[7:0]	Undefined	R	FIFO for serial received data

26.3.3 Transmit Shift Register (SCTSR)

SCTSR is used to transmit serial data. The SCIF loads transmit data from SCFTDR into SCTSR, then transmits the data serially from the TXD pin, LSB (bit 0) first. After transmitting one data byte, the SCIF automatically loads the next transmit data from SCFTDR into SCTSR and starts transmitting again.

The CPU cannot read from or write to SCTSR directly.

26.3.4 Transmit FIFO Data Register (SCFTDR)

SCFTDR is an 8-bit length16-stage FIFO register that stores data for serial transmission. When data for transmission is written to SCFTDR while the transmit shift register (SCTSR) is empty, the SCIF transfers the data written to SCFTDR to SCTSR and starts serial transmission. Continuous serial transmission can be performed until there is no transmit data left in SCFTDR.

SCFTDR is write-only and cannot be read by the CPU. When SCFTDR is full (16 bytes), no more data can be written. If writing of new data is attempted, the data is ignored.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SCFTD[7:0]	Undefined	W	FIFO for serial transmit data

26.3.5 Serial Mode Register (SCSMR)

SCSMR is a 16-bit register that specifies the serial communication format of the SCIF and selects the clock source for the baud rate generator.

The CPU can always read from and write to SCSMR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CA	CHR	PE	OE	STOP	—	CKS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	CA	0	R/W	Communication Mode Selects whether the SCIF operates in asynchronous or clock synchronous mode. 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	0	R/W	Character Length Selects 7-bit or 8-bit data length in asynchronous mode. When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted. In clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting. 0: 8-bit data 1: 7-bit data
5	PE	0	R/W	Parity Enable Selects whether to add a parity bit to transmit data and to check the parity of received data, in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the PE setting. 0: Parity bit not added or checked 1: Parity bit added and checked* Note: * When PE is set to 1, an even or odd parity bit is added to transmit data, depending on the parity mode (OE) setting. Received data parity is checked according to the even/odd (OE) mode setting.

Bit	Bit Name	Initial Value	R/W	Description
4	OE	0	R/W	Parity Mode Selects even or odd parity when parity bits are added and checked. The OE setting is used only in asynchronous mode and only when the parity enable bit (PE) is set to 1 to enable parity addition and checking. The OE setting is ignored in clock synchronous mode, or in asynchronous mode when parity addition and checking is disabled. 0: Even parity*[1] 1: Odd parity*[2] Notes: 1. If even parity is selected, the parity bit is added to transmit data to make an even number of 1s in the transmitted character and parity bit combined. Received data is checked to see if it has an even number of 1s in the received character and parity bit combined. 2. If odd parity is selected, the parity bit is added to transmit data to make an odd number of 1s in the transmitted character and parity bit combined. Received data is checked to see if it has an odd number of 1s in the received character and parity bit combined.
3	STOP	0	R/W	Stop Bit Length Selects one or two bits as the stop bit length in asynchronous mode. When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character. The setting of this bit is only valid in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added. 0: One stop bit*[1] 1: Two stop bits*[2] Notes: 1. When transmitting, a single 1-valued bit is added at the end of each character for transmission. 2. When transmitting, two 1-valued bits are added at the end of each character for transmission.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CKS[1:0]	00	R/W	Clock Select Select the internal clock source of the on-chip baud rate generator. 00: Pϕ 01: Pϕ/4 10: Pϕ/16 11: Pϕ/64 Note: Pϕ: Peripheral clock

26.3.6 Serial Control Register (SCSCR)

SCSCR operates the SCIF transmitter/receiver, enables/disables interrupt requests, and selects the transmit/receive clock source. The CPU can always read and write to SCSCR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TIE	RIE	TE	RE	REIE	—	CKE[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit-FIFO-data-empty interrupt (TXI) requests when the serial transmit data is transferred from the transmit FIFO data register (SCFTDR) to the transmit shift register (SCTSR), when the quantity of data in SCFTDR becomes less than the specified number of transmission triggers, and when the TDFE flag in the serial status register (SCFSR) is set to1. The TXI interrupt request can be cleared either by writing to SCFTDR a greater quantity of transmit data than the specified transmission trigger number after reading 1 from the TDFE flag and then clearing TDFE to 0, or by clearing TIE to 0. 0: Transmit-FIFO-data-empty interrupt request (TXI) is disabled 1: Transmit-FIFO-data-empty interrupt request (TXI) is enabled

Bit	Bit Name	Initial Value	R/W	Description
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables the receive FIFO data full (RXI) interrupts requested when the RDF flag or DR flag in serial status register (SCFSR) is set to1, receive-error (ERI) interrupts requested when the ER flag in SCFSR is set to1, and break (BRI) interrupts requested when the BRK flag in SCFSR or the ORER flag in line status register (SCLSR) is set to1. RXI interrupt requests can be cleared by reading the DR or RDF flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE to 0. ERI or BRI interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0. 0: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled 1: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled
5	TE	0	R/W	Transmit Enable Enables or disables serial transmission by the SCIF. With TE set to 1, serial transmission starts when transmit data is written to SCFTDR. 0: Transmission disabled 1: Transmission enabled* Note: * Select the transmit format in SCSMR and SCFCR and reset the transmit FIFO before setting TE to 1.

Bit	Bit Name	Initial Value	R/W	Description
4	RE	0	R/W	Receive Enable Enables or disables serial reception by the SCIF. With RE set to 1, serial reception starts when a start bit is detected in asynchronous mode, or synchronous clock is detected in clock synchronous mode. 0: Reception disabled*[1] 1: Reception enabled*[2] Notes:1. Clearing RE to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, PER, and ORER). These flags retain their previous values. 2. Select the receive format in SCSMR and SCFCR and reset the receive FIFO before setting RE to 1.
3	REIE	0	R/W	Receive Error Interrupt Enable Enables or disables the receive-error (ERI) interrupts and break (BRI) interrupts. The setting of REIE bit is valid only when RIE bit is set to 0. ERI or BRI interrupt requests can be cleared by reading the ER, BR or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0. Even if RIE is set to 0, when REIE is set to 1, ERI or BRI interrupt requests are enabled. Set so If SCIF wants to inform INTC of ERI or BRI interrupt requests during DMA transfer. 0: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled 1: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled*
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	CKE[1:0]	00	R/W	Clock Enable Select the SCIF clock source and enable or disable clock output from the SCK pin. Depending on CKE[1:0], the SCK pin can be used for serial clock output or serial clock input. CKE[1:0] must be set before selecting the operating mode of the SCIF by the SCSMR register. • Asynchronous mode 00: Internal clock; SCK pin is used as an input pin (input signal is ignored) 01: Setting prohibited 10: External clock; SCK pin used for clock input*[1] 11: Setting prohibited • Clock synchronous mode 00: Setting prohibited 01: Internal clock; SCK pin used for serial clock output*[2] 10: External clock; SCK pin used for serial clock input 11: Setting prohibited Notes: 1. The input clock frequency is 16 times the bit rate. 2. The output clock frequency is the same as the bit rate.

26.3.7 Serial Status Register (SCFSR)

SCFSR is a 16-bit register. The upper 8 bits indicate the number of receive errors in the receive FIFO data register (SCFRDR), and the lower 8 bits indicate the states of SCIF operation.

The CPU can always read the upper 8 bits of SCFSR and always read and write from/to the lower 8 bits. However, it cannot write 1 to the flags ER, TEND, TDFE, BRK, RDF, and DR. These flags can be cleared to 0 only after 1 has been read from them. The PER flag and FER flag are read-only and cannot be written to.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PERC[3:0]				FERC[3:0]				ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R	R	R/W*	R/W*

Note: * Only 0 can be written to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	PERC[3:0]	0000	R	Number of Parity Errors Indicate the number of data bytes including a parity error in the received data stored in the receive FIFO data register (SCFRDR). After the ER bit in SCFSR is set, the value in PERC[3:0] indicates the number of parity errors in SCFRDR. When parity errors have been found in all bytes in the 16 bytes of received data in SCFRDR, PERC[3:0] shows 0000.
11 to 8	FERC[3:0]	0000	R	Number of Framing Errors Indicate the number of data bytes including a framing error in the received data stored in SCFRDR. After the ER bit in SCFSR is set, the value in FERC[3:0] indicates the number of framing errors in SCFRDR. When framing errors have been found in all bytes in the 16 bytes of received data in SCFRDR, FERC[3:0] shows 0000.

Bit	Bit Name	Initial Value	R/W	Description
7	ER	0	R/W*	Receive Error Indicates the occurrence of a framing error or a parity error during reception. Clearing the RE bit to 0 in SCSCR does not affect the ER bit, which retains its value. Even if a receive error has occurred, the received data is transferred to SCFRDR and the receive operation is continued. Whether or not the data read from SCFRDR includes any error is shown in the FER and PER bits in SCFSR. 0: A framing error or parity error has not occurred during reception. [Clearing conditions] • Power-on reset or manual reset • A 0 is written to ER after 1 is read from it 1: A framing error or parity error has occurred during reception. [Setting conditions] • At the end of reception, the stop bit of the last byte of received data is checked and it is found to be 0. In two stop-bit mode, only the first stop bit is checked and the second one is not checked. • The total number of 1s in the received data plus parity bit does not match the even/odd parity specified by the OE bit in SCSMR
6	TEND	1	R/W*	Transmit End Indicates that when the last bit of a serial character was transmitted, SCFTDR did not contain valid data, so transmission has ended. 0: Transmission is in progress [Clearing condition] • When 0 is written to TEND after 1 is read from it after transmit data is written to SCFTDR • Data is written to SCFTDR by the DMAC 1: Transmission has ended [Setting conditions] • Power-on reset or manual reset • TE in the serial control register (SCSCR) is clear • SCFTDR does not contain any transmit data when the last bit of a one-byte serial character is transmitted

Bit	Bit Name	Initial Value	R/W	Description
5	TDFE	1	R/W*	Transmit FIFO Data Empty Indicates that writing of transmit data to SCFTDR has been enabled after data is transferred from SCFTDR to SCTSR and the quantity of data in SCFTDR has become equal to or less than the transmission trigger number specified by the TTRG bits in SCFCR. 0: The quantity of transmit data written to SCFTDR is greater than the specified transmission trigger number [Clearing conditions] • After 1 is read from TDFE, data of the quantity exceeding the specified transmission trigger number is written to SCFTDR and then 0 is written to TDFE • Data of the quantity exceeding the specified transmission trigger number is written to SCFTDR by the DMAC 1: The quantity of transmit data in SCFTDR is equal to or less than the specified transmission trigger number [Setting conditions] • Power-on reset or manual reset • The quantity of transmit data in SCFTDR becomes equal to or less than the specified transmission trigger number as a result of transmission Note: Since SCFTDR is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is "16 minus the specified transmission trigger number". If an attempt is made to write more data, the excess data is ignored. The quantity of data in SCFTDR is indicated by the upper 8 bits of SCFDR.

Bit	Bit Name	Initial Value	R/W	Description
4	BRK	0	R/W*	Break Detection Indicates that a break signal has been detected in received data. 0: No break signal has been received [Clearing conditions] • Power-on reset or manual reset • When 0 is written to BRK after 1 is read from it 1: Break signal has been received After a break is detected, transfer of the received data (H'00) to SCFRDR stops. When the break ends and the receive signal becomes mark (=1), the transfer of received data resumes. [Setting condition] • Data causing a framing error is received, and space (=0, ie, low level) lasts for one or more frame length in the subsequent reception
3	FER	0	R	Framing Error Indication In asynchronous mode, indicates a framing error in the received data that is to be read from SCFRDR next. 0: No framing error has occurred in the received data to be read from SCFRDR next. [Clearing conditions] • Power-on reset or manual reset • No framing error is found in the data to be read from SCFRDR next 1: A framing error has occurred in the next received data to be read from SCFRDR next. [Setting condition] • A framing error is found in the data to be read from SCFRDR next

Bit	Bit Name	Initial Value	R/W	Description
2	PER	0	R	Parity Error Indication In asynchronous mode, indicates a parity error in the received data that is to be read from SCFRDR next. 0: No parity error has been found in the received data to be read from SCFRDR next. [Clearing conditions] • Power-on reset or manual reset • No parity error is found in the data to be read from SCFRDR next 1: A parity error has occurred in the next received data to be read from SCFRDR next. [Setting condition] • A parity error is found in the data to be read from SCFRDR next
1	RDF	0	R/W*	Receive FIFO Data Full Indicates that received data in SCRSR has been transferred to SCFRDR and the quantity of data in SCFRDR has become equal to or more than the receive trigger number specified by the RTRG bits in SCFCR. 0: The quantity of received data in SCFRDR is less than the specified receive trigger number [Clearing conditions] • Power-on reset or manual reset • After 1 is read from RDF, SCFRDR is read until the quantity of received data in SCFRDR becomes less than the specified receive trigger number and then 0 is written to RDF • SCFRDR is read by the DMAC until the quantity of received data in SCFRDR becomes less than the specified receive trigger number 1: The quantity of received data in SCFRDR is equal to or more than the specified receive trigger number [Setting condition] • Received data of the quantity equal to or more than the specified receive trigger number is stored in SCFRDR Note: SCFRDR is a 16-byte FIFO register. At least, data of the specified receive trigger number can be read when RDF is 1. If an attempt is made to read after all the data in SCFRDR has been read, undefined data will be read. The quantity of received data in SCFRDR is indicated by the lower 8 bits of SCFDR.

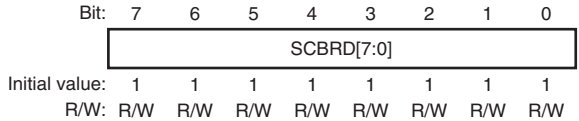
Bit	Bit Name	Initial Value	R/W	Description
0	DR	0	R/W*	Received data Ready Indicates that the quantity of data in SCFRDR is less than the specified receive trigger number, and that the next data has not yet been received after the elapse of 15 ETU from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1. 0: Reception is in progress, or no received data remains in SCFRDR after reception ended normally. [Clearing conditions] • Power-on reset or manual reset • After 1 is read from DR, all received data are read and then 0 is written to DR. • All received data are read by the DMAC 1: Next data has not been received [Setting condition] • SCFRDR contains less data than the specified receive trigger number, and the next data has not yet been received after the elapse of 15 ETU from the last stop bit. Note: 15 ETU is equivalent to 1.5 frames with the 8-bit, 1-stop-bit format. (ETU: elementary time unit)

Note: * Only 0 can be written to clear the flag.

26.3.8 Bit Rate Register (SCBRR)

SCBRR is an 8-bit register that is used to set the bit rate of serial transmission/reception in relation to the operating clock of the baud rate generator selected by the CKS1[1:] bits in SCSMR.

The CPU can always read and write to SCBRR.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SCBRD[7:0]	H'FF	R/W	Bit rate setting

The SCBRR setting is calculated as follows:

- Asynchronous mode:

$$N = \{P\phi / (64 \times 2^{2n-1} \times B)\} \times 10^6 - 1$$

- Clock synchronous mode:

$$N = \{P\phi / (8 \times 2^{2n-1} \times B)\} \times 10^6 - 1$$

B: Bit rate (bits/s)
N: SCBRR setting for baud rate generator ($0 \leq N \leq 255$)
(The setting must satisfy the electrical characteristics.)
$P\phi$: Operating frequency for peripheral modules (MHz)
n: Baud rate generator clock source (n = 0, 1, 2, 3) (for the clock sources and values of n, see Table 26.4.)

Table 26.4 SCSMR Settings

		SCSMR Setting
n	**Clock Source**	**CKS[1:0]**
0	Pϕ	00
1	Pϕ/4	01
2	Pϕ/16	10
3	Pϕ/64	11

Note: The bit rate error in asynchronous mode is given by the following formula:

$$\text{Error (\%)} = \{\{P\phi / ((N + 1) \times 64 \times 2^{2n-1} \times B)\} \times 10^{6} - 1\} \times 100$$

26.3.9 FIFO Control Register (SCFCR)

SCFCR resets the quantity of data in the transmit and receive FIFO data registers, sets the trigger data quantity, and contains an enable bit for loop-back testing. SCFCR can always be read and written to by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RTRG[1:0]		TTRG[1:0]		—	TFRST	RFRST	LOOP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7, 6	RTRG[1:0]	00	R/W	Receive FIFO Data Trigger Set the quantity of received data at which the received data full (RDF) flag in SCFSR is set. The RDF flag is set to 1 when the quantity of received data stored in SCFRDR has become equal to or more than the set trigger number shown below as the reception proceeds.

• Asynchronous mode	• Clock synchronous mode
00: 1	00: 1
01: 4	01: 2
10: 8	10: 8
11: 14	11: 14

Bit	Bit Name	Initial Value	R/W	Description
5, 4	TTRG[1:0]	00	R/W	Transmit FIFO Data Trigger Set the quantity of remaining transmit data at which the transmit FIFO data register empty (TDFE) flag in SCFSR is set. The TDFE flag is set to 1 when the quantity of transmit data in SCFTDR has become equal to or less than the set trigger number shown below as the transmission proceeds. 00: 8 (8)* 01: 4 (12)* 10: 2 (14)* 11: 0 (16)* Note: * Values in parentheses mean the number of empty bytes in SCFTDR when the TDFE flag is set to 1.
3	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TFRST	0	R/W	Transmit FIFO Data Register Reset Invalidates the transmit data in SCFTDR to reset SCFTDR in an empty state. 0: Resetting disabled* 1: Resetting enabled Note: * Resetting is performed by a power-on reset or manual reset, or when a standby state is entered.
1	RFRST	0	R/W	Receive FIFO Data Register Reset Invalidates the received data in SCFRDR to reset SCFRDR in an empty state. 0: Resetting disabled* 1: Resetting enabled Note: * Resetting is performed by a power-on reset or manual reset, or when a standby state is entered.
0	LOOP	0	R/W	Loopback Test Internally connects the transmit output pin (TXD) and receive input pin (RXD), enabling loopback testing. 0: Loopback test disabled 1: Loopback test enabled

26.3.10 FIFO Data Count Set Register (SCFDR)

SCFDR is a 16-bit register that indicates the quantity of data stored in SCFTDR and SCFRDR.

It indicates the quantity of transmit data in SCFTDR with the upper 8 bits, and the quantity of received data in SCFRDR with the lower 8 bits. SCFDR can always be read by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TFDC[4:0]					—	—	—	RFDC[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	TFDC[4:0]	00000	R	Number of Data Bytes in Transmit FIFO Indicate the quantity of non-transmitted data stored in SCFTDR. H'00 means no transmit data, and H'10 means that SCFTDR is full of transmit data (16 bytes).
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	RFDC[4:0]	00000	R	Number of Data Bytes in Receive FIFO Indicate the quantity of received data stored in SCFRDR. H'00 means no received data, and H'10 means that SCFRDR full of received data (16 bytes).

26.3.11 Line Status Register (SCLSR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ORER
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*

Note: * Only 0 can be written to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ORER	0	R/W*	Overrun Error Indicates the occurrence of an overrun error. 0: Receiving is in progress or has ended normally*[1] [Clearing conditions] • Power-on reset or manual reset • When 0 is written to ORER after 1 is read from it 1: An overrun error has occurred*[2] [Setting condition] • Next serial reception is finished while SCFRDR is full of 16 bytes of received data. Notes: 1. Clearing the RE bit to 0 in SCSCR does not affect the ORER bit, which retains the value before RE is cleared. 2. SCFRDR retains the data before an overrun error has occurred, and the next received data is discarded. When the ORER bit is set to 1, the SCIF cannot continue the next serial reception.

Note: * Only 0 can be written to clear the flag.

26.4 Operation

26.4.1 Overview

For serial communication, the SCIF has an asynchronous mode in which characters are synchronized individually, and a clock synchronous mode in which communication is synchronized with clock pulses.

The SCIF has a 16-stage FIFO buffer for both transmission and receptions, reducing the overhead of the CPU, and enabling continuous high-speed communication.

The transmission/reception format is selected by SCSMR as shown in Table 26.5. The SCK pin function is determined by the combination of the CA bit in SCSMR and the CKE[1:0] bits in SCSCR.

(1) Asynchronous Mode

- Data length is selectable: 7 or 8 bits
- Parity bit is selectable. So is the stop bit length (1 or 2 bits). The combination of the preceding selections constitutes the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, received data ready, and breaks.
- The number of stored data bytes is indicated for both the transmit and receive FIFO registers.
- An internal or external clock can be selected as the SCIF clock source.
  - When an internal clock is selected, the SCIF operates using the clock of on-chip baud rate generator.
  - When an external clock is selected, the external clock input must have a frequency 16 times the bit rate. (The on-chip baud rate generator is not used.)

(2) Clock Synchronous Mode

- Data length is 8 bits only.
- In receiving, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the SCIF clock source.
  - When an internal clock is selected, the SCIF operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  - When an external clock is selected, the SCIF is driven by the synchronization clock that is input from the SCK pin.

Table 26.5 SCSMR Settings and SCIF Communication Formats

SCSMR Settings					SCIF Communication Format		
Bit 7 CA	Bit 6 CHR	Bit 5 PE	Bit 3 STOP	Mode	Data Length	Parity Bit	Stop Bit Length
0	0	0	0	Asynchronous	8 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
	1	0	0		7 bits	Not set	1 bit
			1				2 bits
		1	0			Set	1 bit
			1				2 bits
1	x	x	x	Clock synchronous	8 bits	Not set	None

[Legend]

x: Don't care

26.4.2 Operation in Asynchronous Mode

In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections of the SCIF are independent, so full duplex communication is possible. The transmitter and receiver are 16-byte FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmitting and receiving.

Figure 26.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. The SCIF monitors the line and starts serial communication when the line goes to the space (low) state, indicating a start bit. One serial character consists of a start bit (low), data (LSB first), parity bit (high or low), and stop bit (high), in that order.

When receiving in asynchronous mode, the SCIF synchronizes at the falling edge of the start bit. The SCIF samples each data bit on the eighth pulse of a clock with a frequency 16 times the bit rate. Received data is latched at the center of each bit.

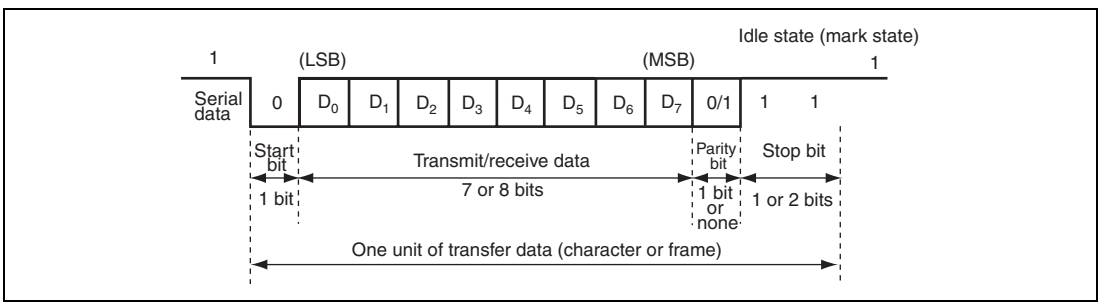


Figure 26.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits)

(1) Transmit/Receive Formats

Table 26.6 lists the eight communication formats that can be selected in asynchronous mode. The format is selected by settings in SCSMR.

Table 26.6 Serial Communication Formats (Asynchronous Mode)

SCSMR Bits			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-bit data								STOP		
0	0	1	START	8-bit data								STOP	STOP	
0	1	0	START	8-bit data								P	STOP	
0	1	1	START	8-bit data								P	STOP	STOP
1	0	0	START	7-bit data							STOP			
1	0	1	START	7-bit data							STOP	STOP		
1	1	0	START	7-bit data							P	STOP		
1	1	1	START	7-bit data							P	STOP	STOP	

[Legend]
START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIF transmit/receive clock.

(3) SCIF Initialization (Asynchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in SCSCR, then initialize the SCIF as follows. When changing the operation mode or the communication format, always clear the TE and RE bits to 0 before following the procedure given below.

1 Clearing TE to 0 initializes SCTSR. Clearing TE and RE to 0, however, does not initialize SCFSR, SCFTDR, or SCFRDR, which retain their contents.

2. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the SCFSR is set. The TE bit can be cleared to 0 during transmission, but the transmit data goes to the Mark state after the bit is cleared to 0. Set the TFRST bit in SCFCR to 1 and reset SCFTDR before TE is set again to start transmission.

Figure 26.3 shows a sample flowchart for initializing the SCIF.

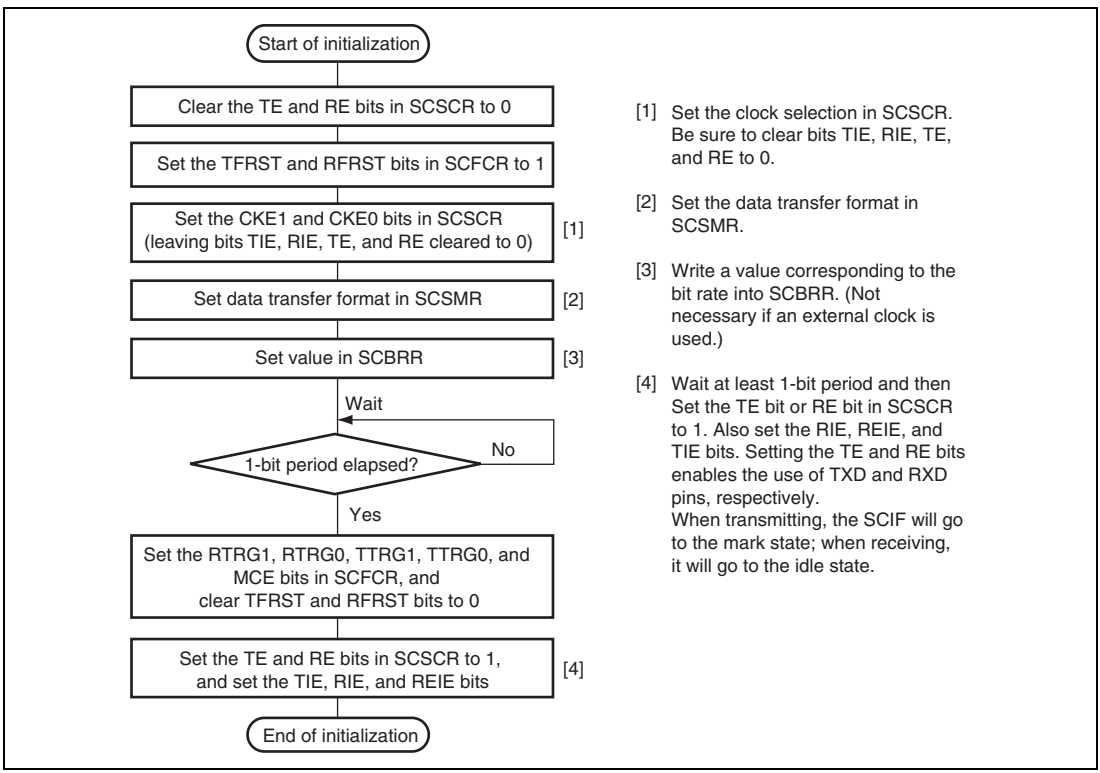


Figure 26.3 Sample Flowchart for SCIF Initialization

(4) Transmitting Serial Data (Asynchronous Mode)

Figure 26.4 shows a sample flowchart for serial transmission.

Use the following procedure for serial data transmission after enabling the SCIF for transmission.

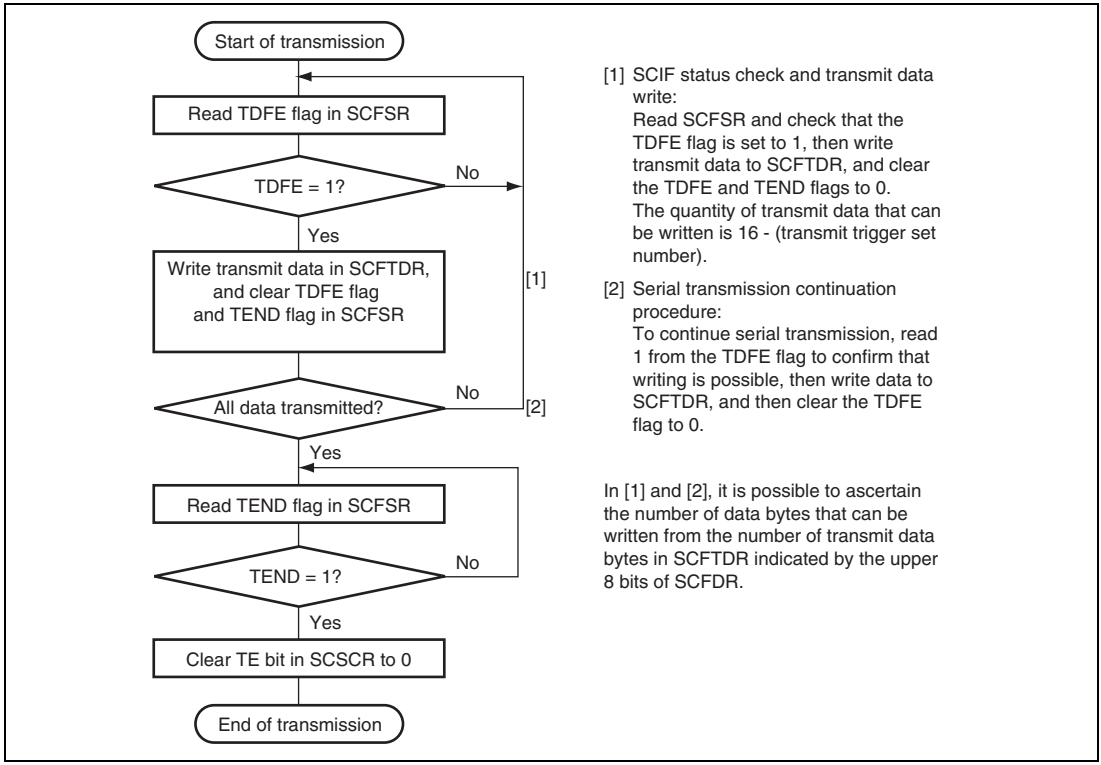


Figure 26.4 Sample Flowchart for Transmitting Serial Data

In serial transmission, the SCIF operates as described below.

1. When data is written into SCFTDR, the SCIF transfers the data from SCFTDR to SCTSR and starts transmission. Confirm that the TDFE flag in SCFSR is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is at least (16 – transmit trigger number setting).
2. When data is transferred from SCFTDR to SCTSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR falls below the transmit trigger number set in SCFCR, the TDFE flag is set. If the TIE bit in SCSR is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.

   The serial transmit data is sent from the TXD pin in the following order.

   A. Start bit: One-bit 0 is output.
   B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
   C. Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   D. Stop bit(s): One or two 1 bits (stop bits) are output.
   E. Mark state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The SCIF checks the SCFTDR transmit data at the timing for sending the stop bit. If data is present, the data is transferred from SCFTDR to SCTSR, the stop bit is sent, and then serial transmission of the next frame is started.

   When there is no transmit data after the stop bit is sent, the TEND flag in SCFSR is set to 1 and the SCIF enters a mark state, in which 1 is output from the TXD pin.

Figure 26.5 shows an example of the operation for transmission.

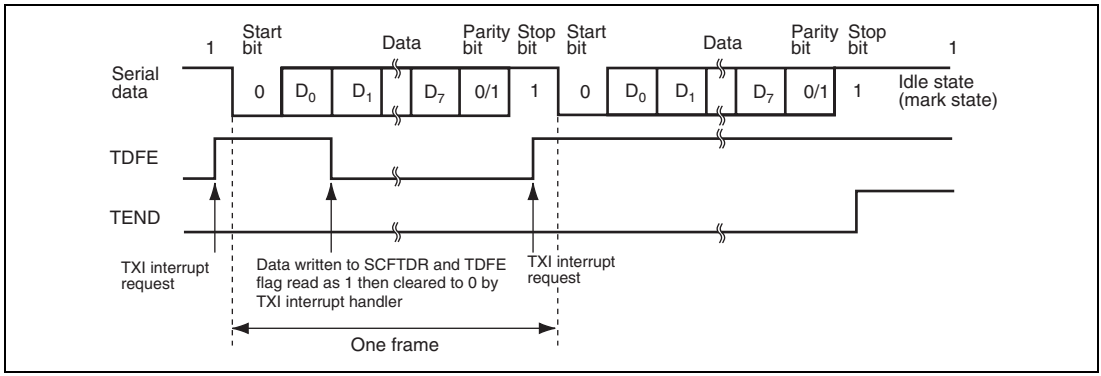
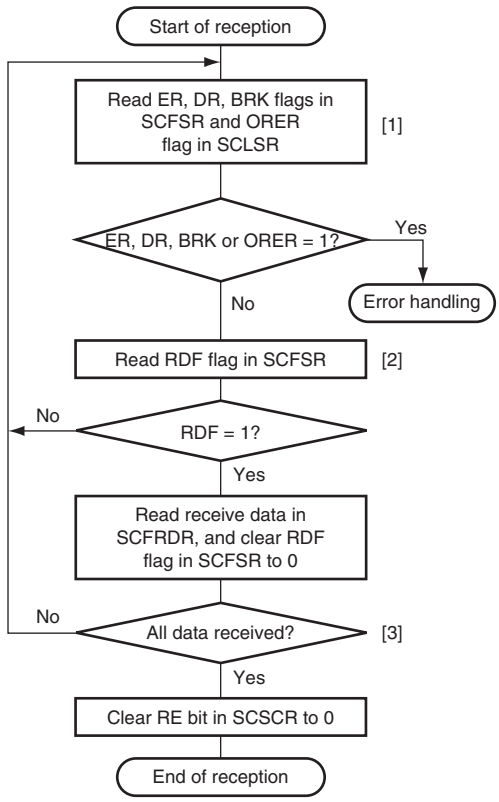


Figure 26.5 Example of Transmit Operation (8-Bit Data, Parity, 1 Stop Bit)

(5) Receiving Serial Data (Asynchronous Mode)

Figures 26.6 and 26.7 show sample flowcharts for serial reception.

Use the following procedure for serial data reception after enabling the SCIF for reception.



[1] Receive error handling and break detection:

Read the DR, ER, and BRK flags in SCFSR, and the ORER flag in SCLSR, to identify any error, perform the appropriate error handling, then clear the DR, ER, BRK, and ORER flags to 0.

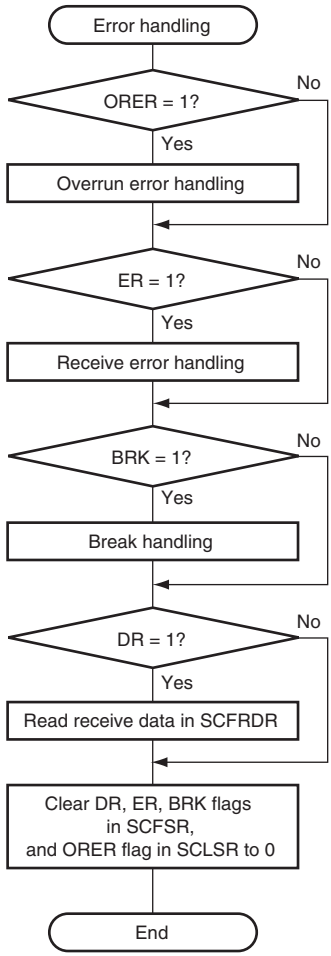
[2] SCIF status check and receive data read:

Read SCFSR and check that RDF flag = 1, then read the receive data in SCFRDR, read 1 from the RDF flag, and then clear the RDF flag to 0. The transition of the RDF flag from 0 to 1 can also be identified by a receive FIFO data full interrupt (RXI).

[3] Serial reception continuation procedure:

To continue serial reception, read at least the receive trigger set number of receive data bytes from SCFRDR, read 1 from the RDF flag, then clear the RDF flag to 0. The number of receive data bytes in SCFRDR can be ascertained by reading from SCFDR.

Figure 26.6 Sample Flowchart for Receiving Serial Data



- Whether a framing error or parity error has occurred in the receive data that is to be read from SCFRDR can be ascertained from the FER and PER bits in SCFSR.
- When a break signal is received, receive data is not transferred to SCFRDR while the BRK flag is set. However, note that the last data in SCFRDR is H'00, and the break data in which a framing error occurred is stored.

Figure 26.7 Sample Flowchart for Receiving Serial Data (cont)

In serial reception, the SCIF operates as described below.

1. The SCIF monitors the transmission line, and if a 0 start bit is detected, performs internal synchronization and starts reception.
2. The received data is stored in SCRSR in LSB-to-MSB order.
3. The parity bit and stop bit are received.

   After receiving these bits, the SCIF carries out the following checks.

   A. Stop bit check: The SCIF checks whether the stop bit is 1. If there are two stop bits, only the first is checked.

   B. The SCIF checks whether received data can be transferred from the receive shift register (SCRSR) to SCFRDR.

   C. Overrun check: The SCIF checks that the ORER flag is 0, indicating that the overrun error has not occurred.

   D. Break check: The SCIF checks that the BRK flag is 0, indicating that the break state is not set.*

   If all the above checks are passed, the received data is stored in SCFRDR.

Note: * Even when a parity error or a framing error occurs, reception will not be suspended.

4. If the RIE bit in SCSCR is set to 1 when the RDF or DR flag changes to 1, a receive-FIFO-data-full interrupt (RXI) request is generated. If the RIE bit or the REIE bit in SCSCR is set to 1 when the ER flag changes to 1, a receive-error interrupt (ERI) request is generated. If the RIE bit or the REIE bit in SCSCR is set to 1 when the BRK or ORER flag changes to 1, a break reception interrupt (BRI) request is generated.

Figure 26.8 shows an example of the operation for reception.

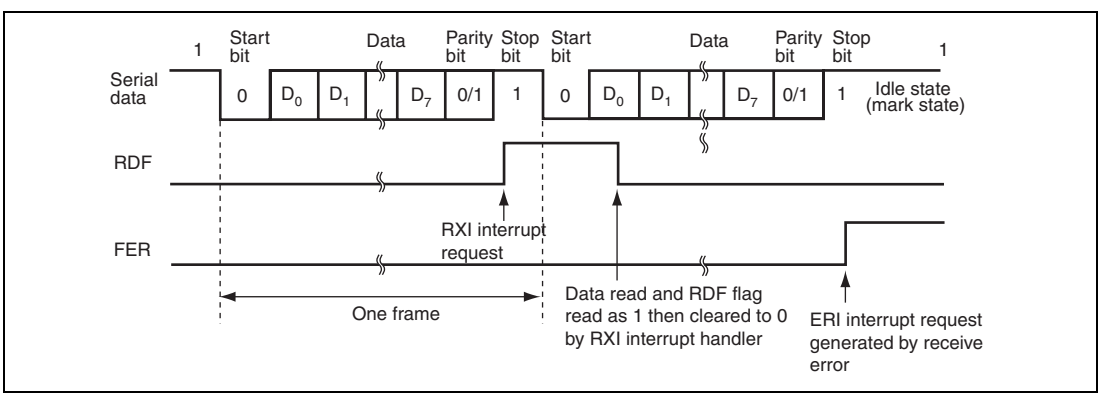


Figure 26.8 Example of SCIF Receive Operation (8-Bit Data, Parity, 1 Stop Bit)

26.4.3 Operation in Clock Synchronous Mode

In clock synchronous mode, the SCIF transmits and receives data in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The SCIF transmitter and receiver are independent, so full-duplex communication is possible while sharing the same clock. The transmitter and receiver are also 16-byte FIFO buffered, so continuous transmitting or receiving is possible by reading or writing data while transmitting or receiving is in progress.

Figure 26.9 shows the general format in clock synchronous serial communication.

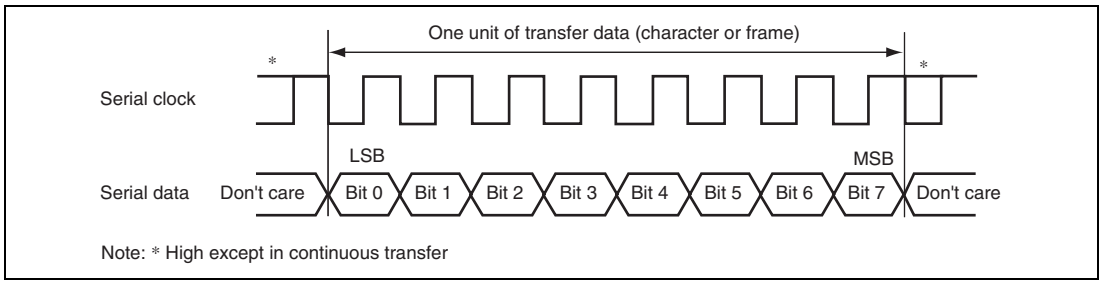


Figure 26.9 Data Format in Clock Synchronous Communication

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial clock.

In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the last data, the communication line remains in the state of the last data.

In clock synchronous mode, the SCIF receives data by synchronizing with the rising edge of the serial clock.

(1) Transmit/Receive Formats

The data length is fixed at eight bits. No parity bit can be added.

(2) Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the SCIF transmit/receive clock.

(3) SCIF Initialization (Clock Synchronous Mode)

Before transmitting, receiving, or changing the mode or communication format, the software must clear the TE and RE bits to 0 in SCSCR, then initialize the SCIF. Clearing TE to 0 initializes SCTSR. Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and SCRDR, which retain their contents.

Figure 26.10 shows a sample flowchart for initializing the SCIF.

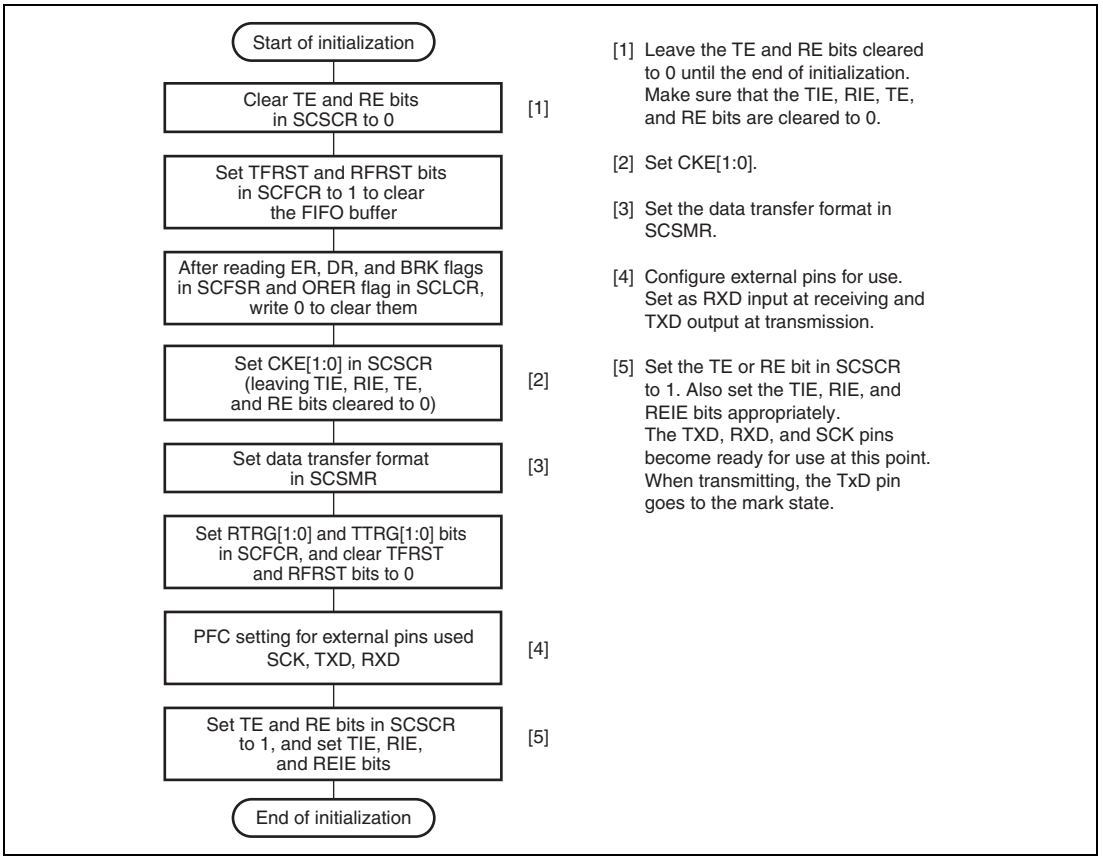


Figure 26.10 Sample Flowchart for SCIF Initialization

(4) Transmitting Serial Data (Clock Synchronous Mode)

Figure 26.11 shows a sample flowchart for transmitting serial data.

Use the following procedure for serial data transmission after enabling the SCIF for transmission.

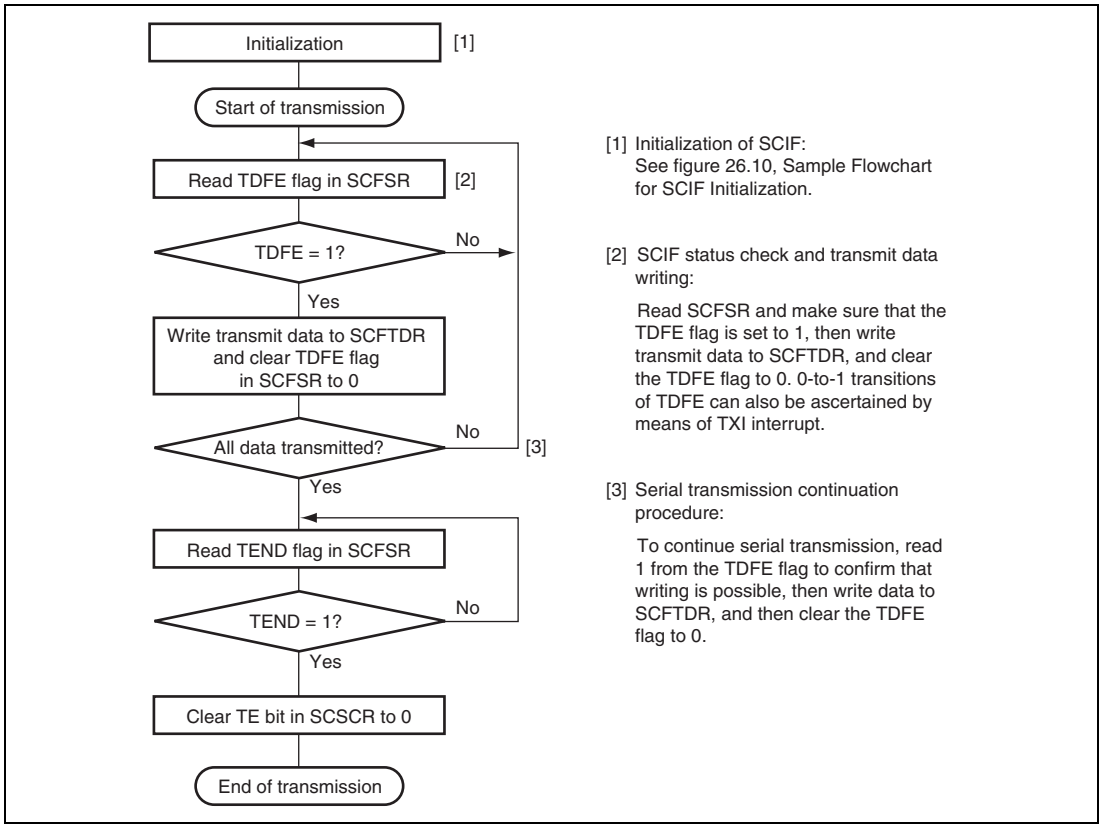


Figure 26.11 Sample Flowchart for Transmitting Serial Data

In serial transmission, the SCIF operates as described below.

1. When data is written into SCFTDR, the SCIF transfers the data in SCFTDR to SCTSR and starts transmission. Confirm that the TDFE flag in SCFSR is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is at least (16 – transmit trigger setting).
2. When data is transferred from SCFTDR to SCTSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR falls below the transmit trigger number set in SCFCR, the TDFE flag is set. If the TIE bit in SCSCR is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.

   The SCIF outputs data in synchronization with the input clock. Data is output from the TXD pin in order from the LSB (bit 0) to the MSB (bit 7).
3. The SCIF checks the SCFTDR transmit data at the timing for sending the MSB (bit 7). If data is present, the data is transferred from SCFTDR to SCTSR, and then serial transmission of the next frame is started. If there is no transmit data, the TEND flag in SCFSR is set to 1 after the MSB (bit 7) is sent and the TXD pin holds its state.

Figure 26.12 shows an example of SCIF transmit operation.

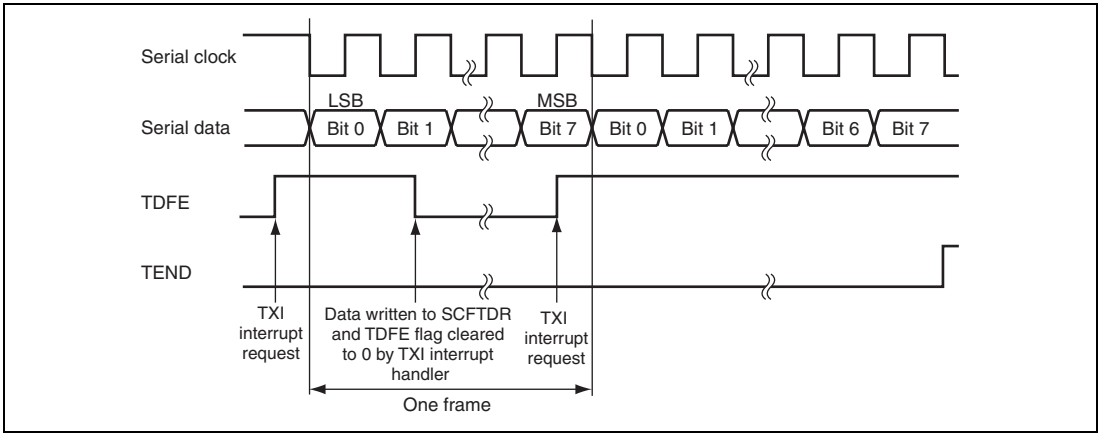


Figure 26.12 Example of SCIF Transmit Operation

(5) Receiving Serial Data (Clock Synchronous Mode)

Figures 26.13 and 26.14 show sample flowcharts for receiving serial data. When switching from asynchronous mode to clock synchronous mode without SCIF initialization, make sure that the ORER bit in SCLSR and the PERC and FERC bits in SCFCR are cleared to 0.

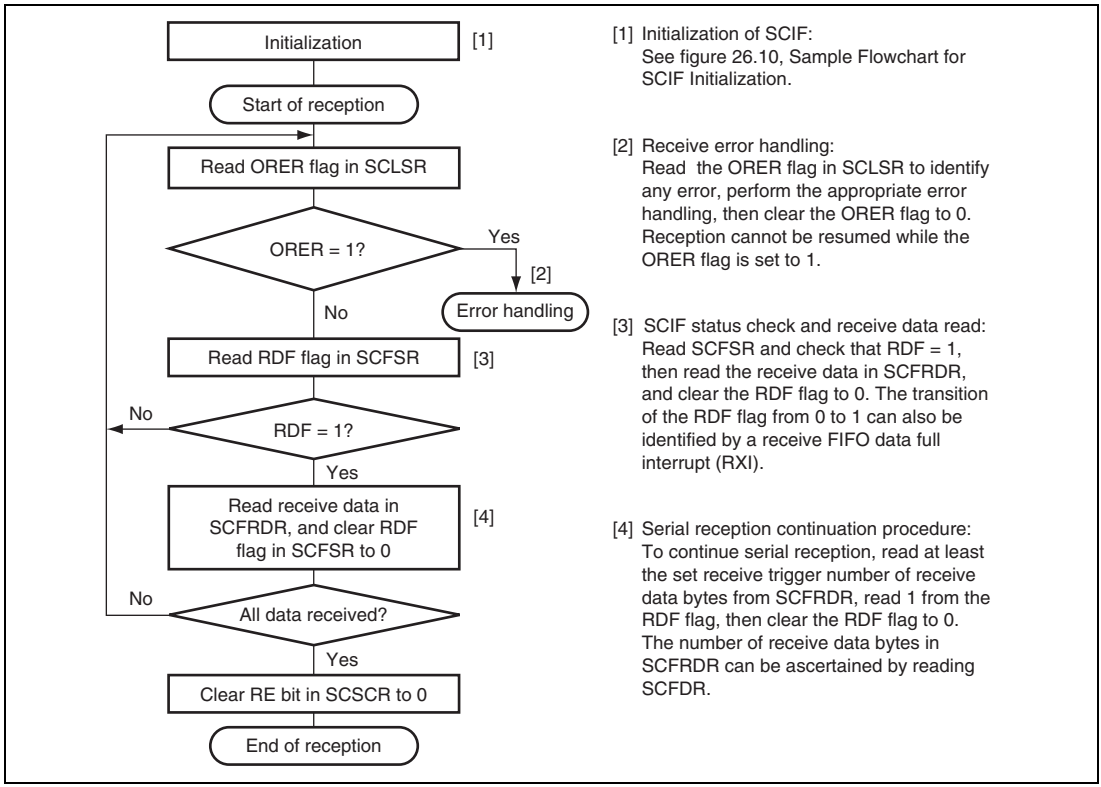


Figure 26.13 Sample Flowchart for Receiving Serial Data (1)

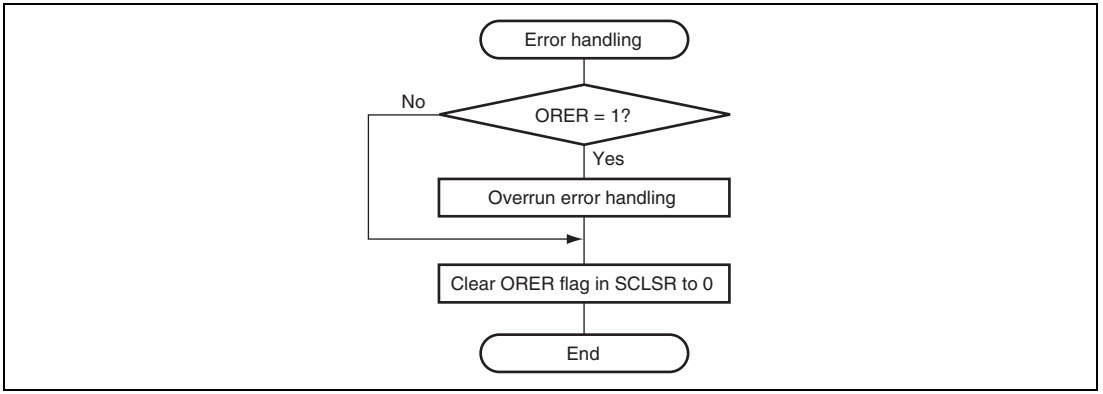


Figure 26.14 Sample Flowchart for Receiving Serial Data (2)

In serial reception, the SCIF operates as described below.

1. The SCIF starts the reception in synchronization with the serial clock output.
2. Received data is shifted into SCRSR in order from the LSB to the MSB. After receiving the data, the SCIF checks the received data can be loaded from SCRSR into SCFRDR or not. If this check is passed, the SCIF stores the received data in SCFRDR. If the check is not passed (overrun error is detected), further reception is prevented.
3. After setting RDF to 1, if the receive FIFO data full interrupt enable bit (RIE) is set to 1 in SCSCR, the SCIF requests a receive-data-full interrupt (RXI). If the ORER bit is set to 1 and the receive-data-full interrupt enable bit (RIE) or the receive error interrupt enable bit (REIE) in SCSCR is also set to 1, the SCIF requests a break interrupt (BRI).

Figure 26.15 shows an example of SCIF receive operation.

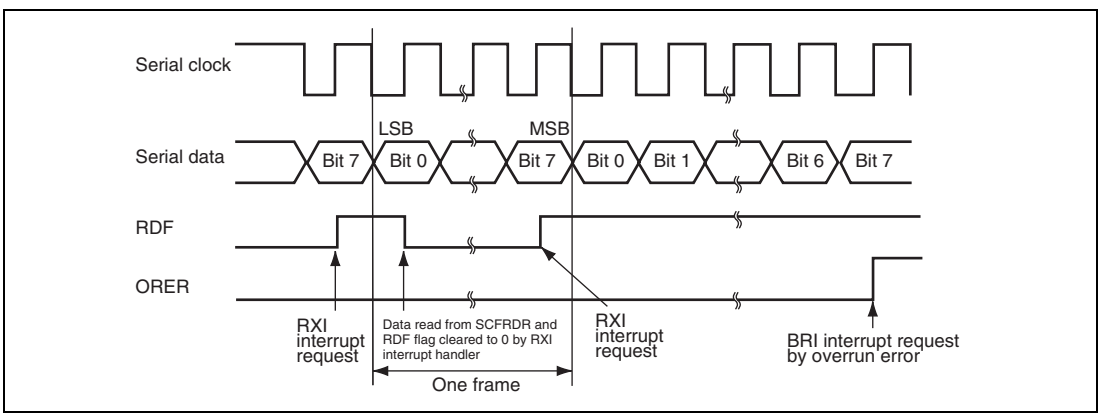
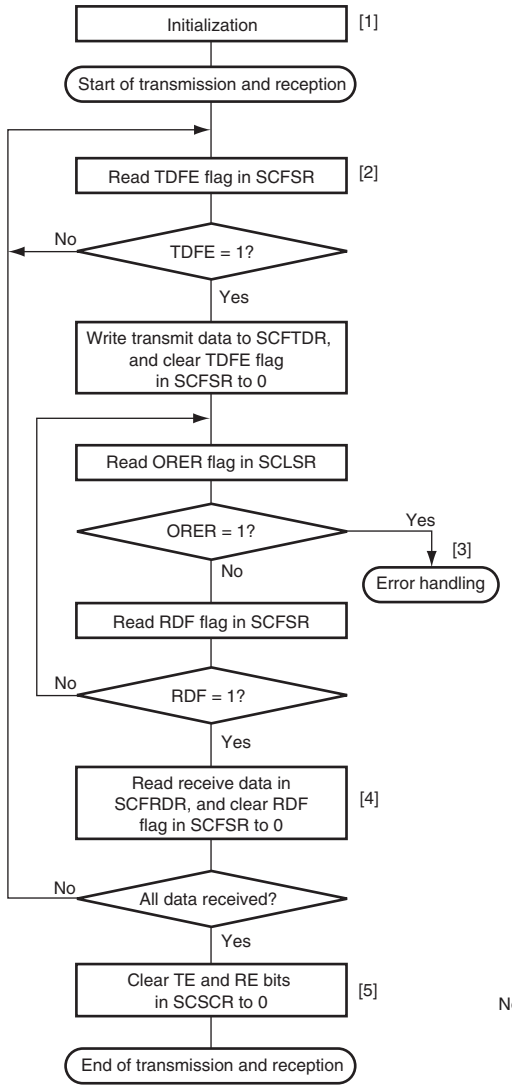


Figure 26.15 Example of SCIF Receive Operation

(6) Transmitting and Receiving Serial Data Simultaneously (Clock Synchronous Mode)

Figure 26.16 shows a sample flowchart for transmitting and receiving serial data simultaneously.

Use the following procedure for the simultaneous transmission/reception of serial data, after enabling the SCIF for transmission/reception.



[1] Initialization of SCIF:

See figure 26.10, Sample Flowchart for SCIF Initialization.

[2] SCIF status check and transmit data write:

Read SCFSR and check that the TDFE flag is set to 1, then write transmit data to SCFTDR, and clear the TDFE flag to 0. The transition of the TDFE flag from 0 to 1 can also be identified by a transmit FIFO data empty interrupt (TXI).

[3] Receive error handling:

Read the ORER flag in SCLSR to identify any error, perform the appropriate error handling, then clear the ORER flag to 0. Reception cannot be resumed while the ORER flag is set to 1.

[4] SCIF status check and receive data read:

Read SCFSR and check that RDF flag is set to 1, then read the receive data in SCFRDR, and clear the RDF flag to 0. The transition of the RDF flag from 0 to 1 can also be identified by a receive FIFO data full interrupt (RXI).

[5] Serial transmission and reception continuation procedure:

To continue serial transmission and reception, read 1 from the RDF flag and the receive data in SCFRDR, and clear the RDF flag to 0 before receiving the MSB in the current frame. Similarly, read 1 from the TDFE flag to confirm that writing is possible before transmitting the MSB in the current frame. Then write data to SCFTDR and clear the TDFE flag to 0.

Note: When switching from a transmit operation or receive operation to simultaneous transmission and reception operations, clear the TE and RE bits to 0, and then set them simultaneously to 1.

Figure 26.16 Sample Flowchart for Transmitting/Receiving Serial Data

26.5 SCIF Interrupt Sources and DMAC

The SCIF has four types of interrupt sources for each channel: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive-FIFO-data-full (RXI), and break (BRI). However, only a single INTEVT code is assigned per channel, so interrupt sources must be identified by software. The interrupt sources are enabled or disabled separately for each channel by means of the TIE, RIE, and REIE bits in SCSCR.

When the TXI request is enabled by the TIE bit and the TDFE flag in SCFSR is set to 1, a TXI interrupt request and transmit-FIFO-data-empty DMA transfer request are generated. When the TXI request is disabled by the TIE bit and the TDFE flag in SCFSR is set to 1, only a transmit-FIFO-data-empty DMA transfer request is generated. This transmit-FIFO-data-empty DMA transfer request can activate the DMAC to perform data transfer.

When the RXI request is enabled by the RIE bit and the RDF flag or the DR flag in SCFSR is set to 1, an RXI interrupt request and receive-FIFO-data-full DMA transfer request are generated. When the RXI request is disabled by the RIE bit and the RDF flag or the DR flag in SCFSR is set to 1, only a receive-FIFO-data-full DMA transfer request is generated. This receive-FIFO-data-full DMA transfer request can activate the DMAC to perform data transfer. Note that the RXI interrupt request or receive-FIFO-data-full DMA transfer request resulting from the DR flag is only generated in asynchronous mode.

When the BRK flag in SCFSR or the ORER flag in SCLSR is set to 1, a BRI interrupt request is generated.

To perform transmission/reception using the DMAC, configure and enable the DMAC first and configure the SCIF next. The SCIF should be configured such that the RXI and TXI interrupt requests are not sent to the interrupt controller. If not configured as such, the interrupt requests sent to the interrupt controller will be cleared by the DMAC regardless of the interrupt handling program.

Clearing the RIE bit to 0 and setting the REIE bit to 1 in SCSCR generates only an ERI interrupt request without generating an RXI interrupt request.

26.6 Usage Notes

Note the following when using the SCIF.

26.6.1 SCFTDR Writing and TDFE Flag

The TDFE flag in SCFSR is set when the number of transmit data bytes written in SCFTDR has fallen below the transmit trigger number set by bits TTRG[1:0] in SCFCR. After the TDFE flag is set, transmit data up to the number of empty bytes in SCFTDR can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in SCFTDR is equal to or less than the transmit trigger number, the TDFE flag will be set to 1 again after being read as 1 and cleared to 0. TDFE flag clearing should therefore be carried out when SCFTDR contains more than the transmit trigger number of transmit data bytes.

The number of transmit data bytes in SCFTDR can be ascertained through SCFDR.

26.6.2 SCFRDR Reading and RDF Flag

The RDF flag in SCFSR is set when the number of received data bytes in SCFRDR has become equal to or greater than the receive trigger number set by bits RTRG[1:0] in SCFCR. After RDF flag is set, received data equivalent to the trigger number can be read from SCFRDR, allowing efficient continuous reception.

However, if the number of data bytes in SCFRDR exceeds the trigger number, the RDF flag will be set to 1 again if it is cleared to 0. The RDF flag should therefore be cleared to 0 after being read as 1 after reading the number of the received data in SCFRDR which is less than the trigger number.

The number of received data bytes in SCFRDR can be ascertained through SCFDR.

26.6.3 Received data Sampling Timing and Receive Margin (Asynchronous Mode)

The SCIF operates on a base clock with a frequency 16 times the bit rate. In reception, the SCIF synchronizes internally with the fall of the start bit, which it samples on the base clock. Received data is latched at the rising edge of the eighth base clock pulse. This is shown in Figure 26.17.

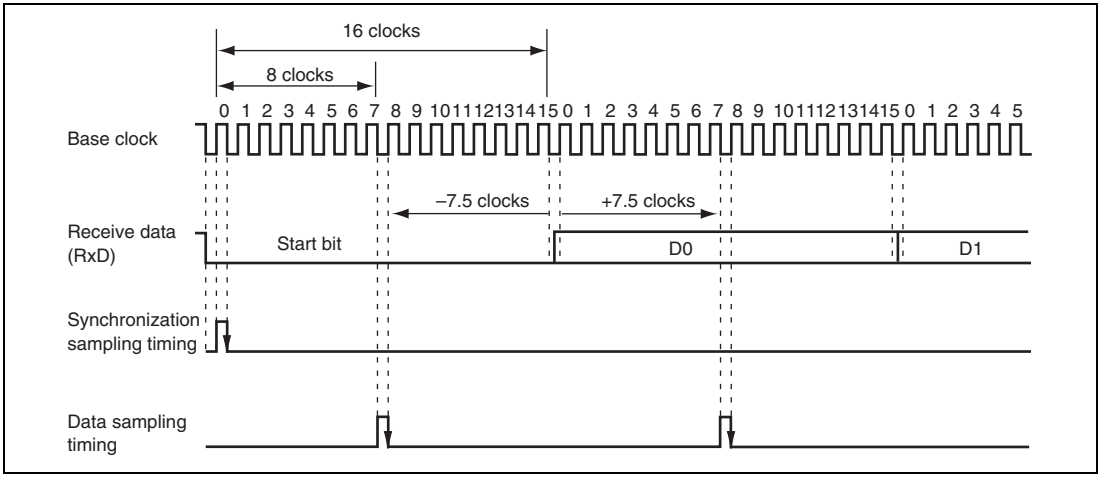


Figure 26.17 Received data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed by equation 1 .

Equation 1:

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)\,F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\ \%$$

Where: M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16 or 8)
D: Clock duty (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation 1, if F = 0, D = 0.5 and N = 16, the receive margin is 46.875%, as given by equation 2.

Equation 2:

When D = 0.5 and F = 0:

$M = (0.5 - 1/(2 \times 16)) \times 100\%$

$= 46.875\%$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

26.6.4 Using the DMAC

When performing transmission/reception using the DMAC, the SCIF should be configured such that the RXI and TXI interrupt requests are not sent to the interrupt controller. If not configured as such, however, the interrupt requests sent to the interrupt controller will be cleared by the DMAC regardless of the interrupt handling program.

26.6.5 Interrupts

Although the SCIF has four types of interrupt sources for each channel, only a single INTEVT code is assigned per channel. Therefore, interrupt sources must be identified by software.

Section 27 Serial Communication Interface with FIFO A (SCIFA)

This LSI has three channels (channel 3 to channel 5) of serial communication interface (SCIFA) that includes FIFO buffers. The SCIFA can perform asynchronous and synchronous serial communications. It has 64-stage FIFO registers for both transmission and reception, which allow efficient high-speed continuous communication.

27.1 Features

- Asynchronous or synchronous mode can be selected for serial communication mode.
- On-chip baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source: From either baud rate generator (internal) or SCK pin (external)
- Six types of interrupts (asynchronous mode):

  Transmit-data-stop, transmit-FIFO-data-empty, receive-FIFO-data-full, receive-error (framing error/parity error), break-receive, and receive-data-ready interrupts. A common interrupt vector is assigned to each interrupt source.
- Two types of interrupts (synchronous mode)
- The direct memory access controller (DMAC) can be activated to transfer data in the event of transmit-FIFO-data-empty, transmit-data-stop, or receive-FIFO-data-full. Note that the transfer request to the DMAC is common to transmit-FIFO-data-empty and transmit-data-stop.
- On-chip modem control functions ($\overline{\text{CTS}}$ and $\overline{\text{RTS}}$)
- Transmit data stop function is available
- While the SCIFA is not used, it can be stopped by stopping the clock for it to reduce power consumption.
- The number of data bytes in the transmit and receive FIFO registers and the number of receive errors of the received data in the receive FIFO register can be known.
- Full-duplex communication capability

  The transmitter and receiver are independent units, enabling transmission and reception to be performed simultaneously.

  The transmitter and receiver both have a 64-stage FIFO buffer structure, enabling fast and continuous serial data transmission and reception.

- Asynchronous mode:

  Serial data communications are performed by start-stop in character units. The SCI can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are eight selectable serial data communication formats.

  — Data length: Seven or eight bits
  — Stop bit length: One or two bits
  — Parity: Even, odd, or none
  — LSB first
  — Receive error detection: Parity, framing, and overrun errors
  — Break detection: Break is detected when the received data next the generated framing error is the space 0 level and has the framing error.

- Synchronous mode:

  Serial data communication is synchronized with a clock. Serial data communication can be carried out with other chips that have a synchronous communication function.

  — Data length: 8 bits
  — LSB-first transfer

Figure 27.1 shows the block diagram of SCIFA.

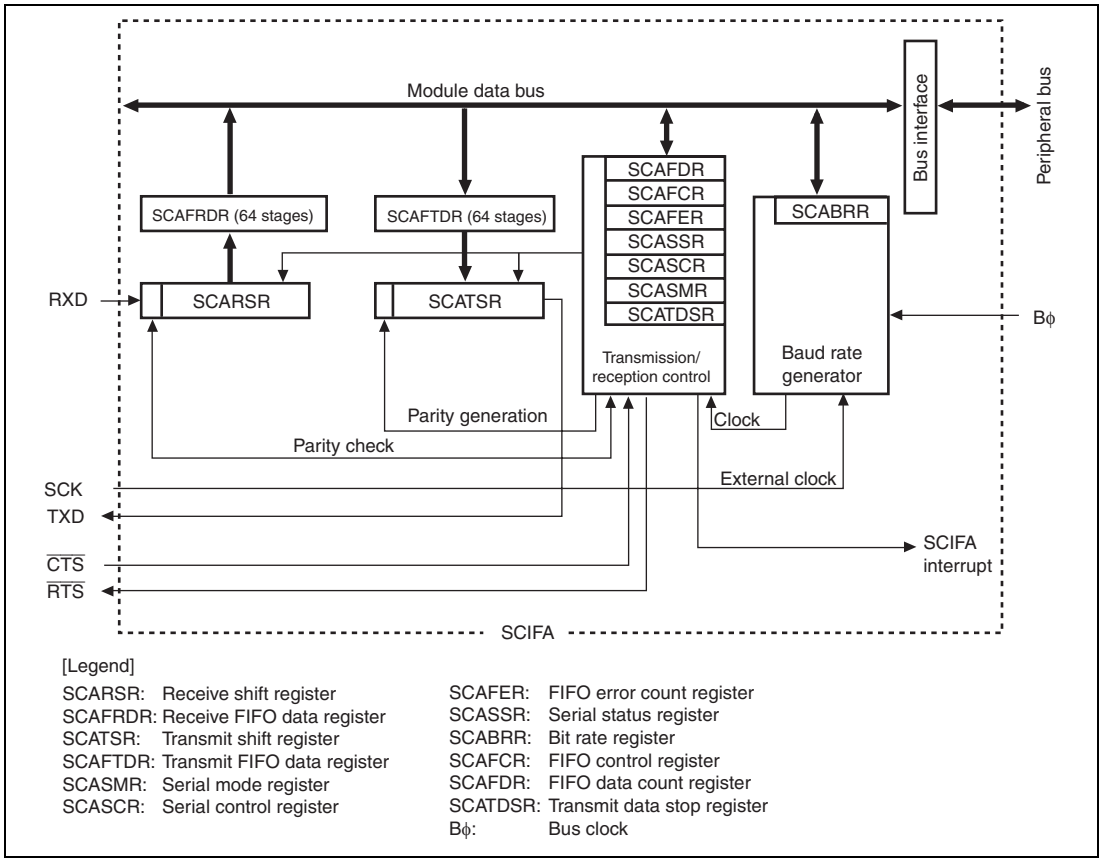


Figure 27.1 Block Diagram of SCIFA

27.2 Input/Output Pins

Table 27.1 shows the pin configuration of SCIFA.

Table 27.1 Pin configuration

Channel	Pin Name	I/O	Function
3	SCIF3_SCK	Input/output	Clock input/output
	SCIF3_RXD	Input	Received data input
	SCIF3_TXD	Output	Transmit data output
	$\overline{\text{SCIF3\_CTS}}$	Input	Clear to send
	$\overline{\text{SCIF3\_RTS}}$	Output	Request to send
4	SCIF4_SCK	Input/output	Clock input/output
	SCIF4_RXD	Input	Received data input
	SCIF4_TXD	Output	Transmit data output
5	SCIF5_SCK	Input/output	Clock input/output
	SCIF5_RXD	Input	Received data input
	SCIF5_TXD	Output	Transmit data output

Note: In the following description, channel numbers in pin names are omitted and SCK, RXD, TXD, $\overline{\text{CTS}}$, and $\overline{\text{RTS}}$ are used as the generic abbreviations.

27.3 Register Descriptions

The register configuration of the SCIFA is shown in Table 27.2, and the register states in each processing mode are shown in Table 27.3.

Table 27.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Serial mode register A3	SCASMR3	R/W	H'A4E3 0000	16
Bit rate register A3	SCABRR3	R/W	H'A4E3 0004	8
Serial control register A3	SCASCR3	R/W	H'A4E3 0008	16
Transmit data stop register A3	SCATDSR3	R/W	H'A4E3 000C	8
FIFO error count register A3	SCAFER3	R	H'A4E3 0010	16
Serial status register A3	SCASSR3	R/W*	H'A4E3 0014	16
FIFO control register A3	SCAFCR3	R/W	H'A4E3 0018	16
FIFO data count register A3	SCAFDR3	R	H'A4E3 001C	16
Transmit FIFO data register A3	SCAFTDR3	W	H'A4E3 0020	8
Receive FIFO data register A3	SCAFRDR3	R	H'A4E3 0024	8
Serial mode register A4	SCASMR4	R/W	H'A4E4 0000	16
Bit rate register A4	SCABRR4	R/W	H'A4E4 0004	8
Serial control register A4	SCASCR4	R/W	H'A4E4 0008	16
Transmit data stop register A4	SCATDSR4	R/W	H'A4E4 000C	8
FIFO error count register A4	SCAFER4	R	H'A4E4 0010	16
Serial status register A4	SCASSR4	R/W*	H'A4E4 0014	16
FIFO control register A4	SCAFCR4	R/W	H'A4E4 0018	16
FIFO data count register A4	SCAFDR4	R	H'A4E4 001C	16
Transmit FIFO data register A4	SCAFTDR4	W	H'A4E4 0020	8
Receive FIFO data register A4	SCAFRDR4	R	H'A4E4 0024	8
Serial mode register A5	SCASMR5	R/W	H'A4E5 0000	16
Bit rate register A5	SCABRR5	R/W	H'A4E5 0004	8
Serial control register A5	SCASCR5	R/W	H'A4E5 0008	16
Transmit data stop register A5	SCATDSR5	R/W	H'A4E5 000C	8
FIFO error count register A5	SCAFER5	R	H'A4E5 0010	16
Serial status register A5	SCASSR5	R/W*	H'A4E5 0014	16
FIFO control register A5	SCAFCR5	R/W	H'A4E5 0018	16

Register Name	Abbreviation	R/W	Address	Access Size
FIFO data count register A5	SCAFDR5	R	H'A4E5 001C	16
Transmit FIFO data register A5	SCAFTDR5	W	H'A4E5 0020	8
Receive FIFO data register A5	SCAFRDR5	R	H'A4E5 0024	8

Note: * To bits 9 to 7, 5, 4, 1, and 0, only 0 can be written to clear the flag.

Table 27.3 Register States in Each Operating Mode

Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
SCASMR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCABRR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASCR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCATDSR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFER3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASSR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFCR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFTDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFRDR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASMR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCABRR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASCR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCATDSR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFER4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASSR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFCR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFDR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFTDR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFRDR4	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASMR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCABRR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASCR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCATDSR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFER5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCASSR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFCR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFDR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFTDR5	Initialized	Initialized	Retained	Retained	Initialized	Retained
SCAFRDR5	Initialized	Initialized	Retained	Retained	Initialized	Retained

27.3.1 Receive Shift Register (SCARSR)

SCARSR receives serial data. Data input at the RXD pin is loaded into the SCARSR in the order received, LSB (bit 0) first, converting the data to parallel form. When one byte has been received, it is automatically transferred to SCAFRDR, which is a receive FIFO data register. The CPU cannot read from or write to the SCARSR directly.

27.3.2 Receive FIFO Data Register (SCAFRDR)

The 64-byte receive FIFO data register (SCAFRDR) stores serial received data. The SCIFA completes the reception of one byte of serial data by moving the received data from SCARSR into SCAFRDR for storage. Continuous receive can be performed until 64 bytes are stored, which makes SCAFRDR full.

The CPU can read but cannot write to SCAFRDR. When data is read without received data in SCAFRDR, the value is undefined. When the received data in this register becomes full, the subsequent serial data is lost.

Bit:	7	6	5	4	3	2	1	0
	SCFRD[7:0]							
Initial value:	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial value	R/W	Description
7 to 0	SCFRD[7:0]	Undefined	R	FIFO Data Registers for Serial Received data

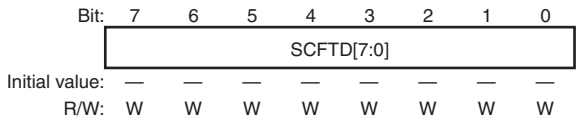
27.3.3 Transmit Shift Register (SCATSR)

SCATSR transmits serial data. The SCIFA loads transmit data from SCAFTDR into SCATSR, then transmits the data serially from the TXD pin, LSB (bit 0) first. After transmitting one data byte, the SCI automatically loads the next transmit data from SCAFTDR into SCATSR and starts transmitting again. The CPU cannot read or write SCATSR directly.

27.3.4 Transmit FIFO Data Register (SCAFTDR)

SCAFTDR is a 64-byte 8-bit-length FIFO register that stores data for serial transmission. When the SCIFA detects that SCATSR is empty, it moves transmit data written in SCAFTDR into SCATSR and starts serial transmission. Continuous serial transmission is performed until SCAFTDR becomes empty. As SCAFTDR is write-only, it cannot be read by the CPU.

When the transmit data in SCAFTDR is full (64 bytes), next data cannot be written. If attempted to write, the data is ignored.



Bit	Bit Name	Initial value	R/W	Description
7 to 0	SCFTD[7:0]	Undefined	R	FIFO Data Registers for Serial Transmit Data

27.3.5 Serial Mode Register (SCASMR)

SCASMR is a 16-bit readable/writable register that specifies the SCIFA serial communication format and selects the clock source for the baud rate generator and the sampling rate.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SRC[2:0]			CA	CHR	PE	OE	STOP	—	CKS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read 0. The write value should always be 0.
10 to 8	SRC[2:0]	000	R/W	Sampling Control Select sampling rate. 000: Sampling rate 1/16 001: Sampling rate 1/5 010: Sampling rate 1/7 011: Sampling rate 1/11 100: Sampling rate 1/13 101: Sampling rate 1/17 110: Sampling rate 1/19 111: Sampling rate 1/27
7	CA	0	R/W	Communication Mode Selects whether the SCIFA operates in asynchronous or synchronous mode. 0: Asynchronous mode 1: Synchronous mode
6	CHR	0	R/W	Character Length Selects seven-bit or eight-bit data. This bit is only valid in asynchronous mode. In synchronous mode, the data length is always eight bits, regardless of the CHR setting. 0: Eight-bit data 1: Seven-bit data* Note: * When seven-bit data is selected, the MSB (bit 7) in SCAFTDR is not transmitted.

Bit	Bit Name	Initial Value	R/W	Description
5	PE	0	R/W	Parity Enable Selects whether to add a parity bit to transmit data and to check the parity of received data. This setting is only valid in asynchronous mode. In synchronous mode, parity bit addition and checking is not performed, regardless of the PE setting. 0: Parity bit not added or checked 1: Parity bit added and checked Note: * When PE is set to 1, an even or odd parity bit is added to transmit data, depending on the parity mode (OE) setting. Received data parity is checked according to the even/odd (OE) mode setting.
4	OE	0	R/W	Parity Mode Selects even or odd parity when parity bits are added and checked. The OE setting is used only when the PE is set to 1 to enable parity addition and check. The OE setting is ignored when parity addition and check is disabled. 0: Even parity*[1] 1: Odd parity*[2] Notes: 1. If even parity is selected, the parity bit is added to transmit data to make an even number of 1s in the transmitted character and parity bit combined. Received data is checked to see if it has an even number of 1s in the received character and parity bit combined. 2. If odd parity is selected, the parity bit is added to transmit data to make an odd number of 1s in the transmitted character and parity bit combined. Received data is checked to see if it has an odd number of 1s in the received character and parity bit combined.

Bit	Bit Name	Initial Value	R/W	Description
3	STOP	0	R/W	Stop Bit Length Selects one or two bits as the stop bit length. In receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character. This setting is only valid in asynchronous mode. In synchronous mode, this setting is invalid since stop bits are not added. 0: One stop bit*[1] 1: Two stop bits*[2] Notes: 1. In transmitting, a single bit of 1 is added at the end of each transmitted character. 2. In transmitting, two bits of 1 are added at the end of each transmitted character.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	CKS[1:0]	00	R/W	Clock Select These bits select the internal clock source of the on-chip baud rate generator. 00: Bϕ 01: Bϕ/4 10: Bϕ/16 11: Bϕ/64 Note: Bϕ is the bus clock.

Note: In synchronous mode, bits other than CKS[1:0] are fixed to 0.

27.3.6 Serial Control Register (SCASCR)

SCASCR is a 16-bit readable/writable register that operates the SCI transmitter/receiver, enables/disables interrupt requests, and selects the transmit/receive clock source.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDRQE	RDRQE	—	—	TSIE	ERIE	BRIE	DRIE	TIE	RIE	TE	RE	—	—	CKE[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	TDRQE	0	R/W	Transmit Data Transfer Request Enable Selects whether to issue the transmit-FIFO-data-empty interrupt request or DMA transfer request when TIE = 1 and transmit FIFO empty interrupt is generated at the transmission. 0: Interrupt request is issued to CPU 1: Transmit data transfer request is issued to DMAC
14	RDRQE	0	R/W	Received data Transfer Request Enable Selects whether to issue the receive-FIFO-data-full interrupt or DMA transfer request when RIE = 1 and receive FIFO data full interrupt is generated at the reception. 0: Interrupt request is issued to CPU 1: Received data transfer request is issued to DMAC
13,12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	TSIE	0	R/W	Transmit Data Stop Interrupt Enable Enables or disables the generation of the transmit-data-stop interrupt requested when the TSE bit in SCAFCR is enabled and the TSF flag in SCASSR is set to 1. 0: The transmit-data-stop-interrupt disabled* 1: The transmit-data-stop-interrupt enabled Note: * The transmit data stop interrupt request is cleared by reading the TSF flag after it has been set to 1, then clearing the flag to 0, or clearing the TSIE bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
10	ERIE	0	R/W	Receive Error Interrupt Enable Enables or disables the generation of a receive-error (framing error/parity error) interrupt requested when the ER flag in SCASSR is set to 1. 0: The receive-error interrupt disabled* 1: The receive-error interrupt enabled Note: * The receive-error interrupt request is cleared by reading the ER flag after it has been set to 1, then clearing the flag to 0, or clearing the ERIE bit to 0.
9	BRIE	0	R/W	Break Interrupt Enable Enables or disables the generation of break-receive interrupt requested when the BRK flag in SCASSR is set to 1. 0: The break-receive interrupt disabled* 1: The break receive interrupt enabled Note: * The break-receive interrupt request is cleared by reading the BRK flag after it has been set to 1, then clearing the flag to 0, or clearing the BRIE bit to 0.
8	DRIE	0	R/W	Received data Ready Interrupt Enable Disables or enables the generation of receive-data-ready interrupt when the DR flag in SCASSR is set to 1. 0: The receive-data-ready interrupt disabled 1: The receive-data-ready interrupt enabled Note: * The receive-data-ready interrupt request is cleared by reading the DR flag after it has been set to 1, then clearing the flag to 0, or clearing the DRIE bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables the transmit-FIFO-data-empty interrupt requested when the TDFE flag of SCASSR is set to 1. 0: Transmit-FIFO-data-empty interrupt request disabled* 1: Transmit-FIFO-data-empty interrupt request enabled Note: * The transmit-FIFO-data empty interrupt request can be cleared by writing the greater number of transmit data bytes than the specified transmission trigger number to SCAFTDR and by clearing TDFE to 0 after reading 1 from TDFE, or can be cleared by clearing TIE to 0.
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables the receive-FIFO-data-full interrupt requested when the RDF flag of SCASSR is set to1. 0: Receive-FIFO-data-full interrupt request disabled* 1: Receive-FIFO-data-full interrupt request enabled Note: * The receive-FIFO-data -full interrupt request can be cleared by reading the RDF flag after it has been set to 1, then clearing the flag to 0, or by clearing the RIE bit to 0.
5	TE	0	R/W	Transmit Enable Enables or disables the SCIFA serial transmitter. 0: Transmitter disabled 1: Transmitter enabled* Note: * SCASMR and SCAFCR should be set to select the transmit format and reset the transmit FIFO before setting the TE bit to 1.
4	RE	0	R/W	Receive Enable Enables or disables the SCIFA serial receiver. 0: Receiver disabled*[1] 1: Receiver enabled*[2] Notes: 1. Clearing RE to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, PER, and ORER). These flags retain their previous values. 2. SCASMR and SCAFCR should be set to select the receive format and reset the receive FIFO before setting the RE bit to 1.

Bit	Bit Name	Initial Value	R/W	Description
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	CKE[1:0]	00	R/W	Clock Enable These bits select the SCIFA clock source and should be set before selecting the SCIFA operating mode by SCASMR. • Asynchronous mode 00: Internal clock, SCK pin used as input pin (input signal is ignored)*1 01: Setting prohibited 10: External clock, SCK pin used for clock input*3 11: Setting prohibited • Synchronous mode 00: Setting prohibited 01: Internal clock, SCK pin used for synchronous clock output*2 10: External clock, SCK pin used for clock input 11: Setting prohibited Notes: 1. When the data sampling is done using on-chip baud rate generator, CKE[1:0] should be set to 00. 2. The output clock frequency is the same as the bit rate. 3. Input the clock which is appropriate for the sampling rate. For example, when the sampling rate is 1/16, input the clock frequency 8 times the bit rate. When the external clock is not input, CKE[1:0] should be set to 00.

27.3.7 FIFO Error Count Register (SCAFER)

SCAFER is a 16-bit read-only register that indicates the number of received data errors (framing error/parity error).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PER[5:0]						—	—	FER[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15,14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	PER[5:0]	000000	R	Parity Error Indicate the number of data bytes which contain parity errors in received data stored in SCAFRDR in asynchronous mode. PER[5:0] indicate the number of data bytes with parity errors after the ER bit in SCASSR is set. If all 64-byte received data in SCAFRDR have parity errors, bits PER[5:0] are all 0s.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	FER[5:0]	000000	R	Framing Error Indicate the number of data bytes which contain framing errors in received data stored in SCAFRDR in asynchronous mode. FER[5:0] indicate the number of data bytes with framing errors after the ER bit in SCASSR is set. If all 64-byte received data in SCAFRDR have framing errors, bits FER[5:0] are all 0s.

27.3.8 Serial Status Register (SCASSR)

SCASSR is a 16-bit readable/writable register that indicates SCIFA states. The ORER, TSF, ER, TDFE, BRK, RDF, or DR flag cannot be set to 1. These flags can be cleared to 0 only if they have first been read (after being set to 1). The flags TEND, FER, and PER are read-only bits and cannot be modified.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ORER	TSF	ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R	R/(W)*	R/(W)*	R	R	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	ORER	0	R/(W)*	Overrun Error Flag Indicates that the overrun error occurred during reception. This bit is valid only in asynchronous mode. 0: Indicates during reception, or reception has been completed without any error*[1] [Clearing conditions] Power-on reset, manual reset Writing 0 after reading ORER = 1 1: Indicates that the overrun error is generated during reception*[2] [Setting condition] When receive FIFO is full and the next serial data reception is completed Notes: 1. When the RE bit in SCASCR is cleared to 0, the ORER flag is not affected and retains its previous state. 2. SCAFRDR holds the data received before the overrun error, and newly received data is lost. When ORER is set to 1, subsequent serial data reception cannot be carried out.

Bit	Bit Name	Initial Value	R/W	Description
8	TSF	0	R/(W)*	Transmit Data Stop Flag Indicates that the number of transmit data bytes matches the value set in SCATDSR. 0: Transmit data number does not match the value set in SCATDSR [Clearing conditions] • Power-on reset, manual reset • Writing 0 after reading TSF = 1 1: Transmit data number matches the value set in SCATDSR
7	ER	0	R/(W)*	Receive Error Indicates that a framing error or parity error occurred during reception in asynchronous mode.*[1] 0: Receive is normally completed without any framing or parity error [Clearing conditions] • Power-on reset, manual reset • ER is read as 1, then written to with 0. 1: A framing error or a parity error has occurred during receiving [Setting conditions] • The stop bit is 0 after checking whether or not the last stop bit of the received data is 1 at the end of one-data receive.*[2] • The total number of 1s in the received data and in the parity bit does not match the even/odd parity specification specified by the OE bit in SCASMR. Notes: 1. Indicates clearing the RE bit to 0 in SCASCR does not affect the ER bit, which retains its previous value. Even if a receive error occurs, the received data is transferred to SCAFRDR and the receive operation is continued. Whether or not the data read from SCAFRDR includes a receive error can be detected by the FER and PER bits in SCASSR. 2. In the stop mode, only the first stop bit is checked; the second stop bit is not checked.

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	1	R	Transmit End Indicates that when the last bit of a serial character was transmitted, SCAFTDR did not contain valid data, so transmission has ended. 0: Transmission is in progress [Clearing condition] • Data is written to SCAFTDR. 1: End of transmission [Setting condition] • SCAFTDR contains no transmit data when the last bit of a one-byte serial character is transmitted.
5	TDFE	1	R/(W)*	Transmit FIFO Data Empty Indicates that data is transferred from SCAFTDR to SCATSR, the number of data bytes in SCAFTDR becomes less than the transmission trigger number specified by the TTRG[1:0] bits in SCAFCR, and writing the transmit data to SCAFTDR is enabled. 0: The number of transmit data bytes written to SCAFTDR is greater than the specified transmission trigger number [Clearing condition] • Data exceeding the specified transmission trigger number is written to SCAFTDR, software reads TDFE after it has been set to 1, then writes 0 to TDFE. 1: The number of transmit data bytes in SCAFTDR becomes less than the specified transmission trigger number [Setting conditions] • Power-on reset • The number of transmit data bytes in SCAFTDR becomes less than the specified transmission trigger number as a result of transmission* Note: * Since SCAFTDR is a 64-byte FIFO register, the maximum number of data bytes which can be written when TDFE is 1 is "64 minus the specified transmission trigger number". If attempted to write excess data, the data is ignored. The number of data bytes in SCAFTDR is indicated in SCAFDR.

Bit	Bit Name	Initial Value	R/W	Description
4	BRK	0	R/(W)*	Break Detection Indicates that a break signal is detected in received data in asynchronous mode. 0: No break signal is being received [Clearing conditions] • Power-on reset, manual reset • BRK is read as 1, then written to with 0 1: A break signal is received * [Setting conditions] Data including a framing error is received • Data causing a framing error is received, and space (=0, i.e., low level) lasts for one or more frame length in the subsequent reception Note: * When a break is detected, transfer of the received data (H'00) to SCAFRDR stops after detection. When the break ends and the receive signal becomes mark 1, the transfer of the received data resumes.
3	FER	0	R	Framing Error Indicates a framing error in the data read from SCAFRDR in asynchronous mode. 0: No framing error occurred in the data read from SCAFRDR [Clearing conditions] • Power-on reset • No framing error is present in the data read from SCAFRDR 1: A framing error occurred in the data read from SCAFRDR [Setting condition] • A framing error is present in the data read from SCAFRDR

Bit	Bit Name	Initial Value	R/W	Description
2	PER	0	R	Parity Error Indicates a parity error in the data read from SCAFRDR in asynchronous mode. 0: No parity error occurred in the data read from SCAFRDR [Clearing conditions] • Power-on reset • No parity error is present in the data read from SCAFRDR 1: A parity error occurred in the data read from SCAFRDR [Setting condition] • A parity error is present in the data read from SCAFRDR
1	RDF	0	R/(W)*	Receive FIFO Data Full Indicates that received data is transferred to SCAFRDR, the number of data bytes in SCAFRDR becomes more than the reception trigger number specified by the RTRG[1:0] bits in SCAFCR. 0: The number of transmit data bytes written to SCAFRDR is less than the specified reception trigger number [Clearing conditions] • Power-on reset • SCAFRDR is read until the number of received data bytes in SCAFRDR becomes less than the specified reception trigger number, and RDF is read as 1, then written to with 0. 1: The number of received data bytes in SCAFRDR is more than the specified reception trigger number [Setting condition] • Received data of the number of bytes greater than the specified reception trigger number is being stored to SCAFRDR.* Note: * Since SCAFRDR is a 64-byte FIFO register, the maximum number of data bytes which can be read when RDF is 1 is the specified reception trigger number. If attempted to read after all data in SCAFRDR have been read, the data is undefined. The number of received data bytes in SCAFRDR is indicated by the lower bits of SCAFDR.

Bit	Bit Name	Initial Value	R/W	Description
0	DR	0	R/(W)*	Received Data Ready Indicates that SCAFRDR stores data which is less than the specified reception trigger number, and that next data is not yet received after 15 etu has elapsed from the last stop bit in asynchronous mode. 0: Receive is in progress, or no received data remains in SCAFRDR after the receive ended normally. [Clearing conditions] (Initial value) • Power-on reset • All received data in SCAFRDR is read, and DR is read as 1, then written to with 0. 1: Next received data is not received [Setting condition] • SCAFRDR stores the data which is less than the specified reception trigger number, and that next data is not yet received after 15 etu has elapsed from the last stop bit.* Note: * This is equivalent to 1.5 frames with the 8-bit 1-stop-bit format. (etu: Element Time Unit)

Note: * The only value that can be written is 0 to clear the flag.

27.3.9 Bit Rate Register (SCABRR)

SCABRR is an eight-bit readable/writable register that, together with the baud rate generator clock source selected by the CKS[1:0] bits in SCASMR, determines the serial transmit/receive bit rate.

Bit:	7	6	5	4	3	2	1	0
	SCBRD[7:0]							
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SCBRD[7:0]	H'FF	R/W	Bit Rate Setting

The SCABRR setting is calculated as follows:

Asynchronous Mode:

1. When sampling rate is 1/16

$$N = \frac{B\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1$$

2. When sampling rate is 1/5

$$N = \frac{B\phi}{10 \times 2^{2n-1} \times B} \times 10^6 - 1$$

3. When sampling rate is 1/11

$$N = \frac{B\phi}{22 \times 2^{2n-1} \times B} \times 10^6 - 1$$

4. When sampling rate is 1/13

$$N = \frac{B\phi}{26 \times 2^{2n-1} \times B} \times 10^6 - 1$$

5. When sampling rate is 1/27

$$N = \frac{B\phi}{54 \times 2^{2n-1} \times B} \times 10^6 - 1$$

Synchronous Mode:

$$N = \frac{B\phi}{4 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bits/s)
N: SCABRR setting for baud rate generator
Asynchronous mode: $0 \leq N \leq 255$
Synchronous mode: $1 \leq N \leq 255$
Bϕ: Bus operating frequency (MHz)
n: Baud rate generator input clock (n = 0 to 3)
(See the table below for the relation between n and the clock.)

Table 27.4 SCASMR Setting

		SCASMR Setting
n	**Clock Source**	**CKS[1:0]**
0	Bϕ	00
1	Bϕ/4	01
2	Bϕ/16	10
3	Bϕ/64	11

Find the bit rate error in asynchronous mode by the following formula:

1. When sampling rate is 1/16

$$\text{Error (\%)} = \left(\frac{B\phi \times 10^6}{(1+N) \times B \times 32 \times 2^{2n-1}} - 1 \right) \times 100$$

2. When sampling rate is 1/5

$$\text{Error (\%)} = \left(\frac{B\phi \times 10^6}{(1+N) \times B \times 10 \times 2^{2n-1}} - 1 \right) \times 100$$

3. When sampling rate is 1/11

$$\text{Error (\%)} = \left(\frac{B\phi \times 10^6}{(1+N) \times B \times 22 \times 2^{2n-1}} - 1 \right) \times 100$$

4. When sampling rate is 1/13

$$\text{Error (\%)} = \left(\frac{B\phi \times 10^6}{(1+N) \times B \times 26 \times 2^{2n-1}} - 1 \right) \times 100$$

5. When sampling rate is 1/27

$$\text{Error (\%)} = \left(\frac{B\phi \times 10^6}{(1+N) \times B \times 54 \times 2^{2n-1}} - 1 \right) \times 100$$

27.3.10 FIFO Control Register (SCAFCR)

SCAFCR is a 16-bit readable/writable register that resets the number of data bytes in the transmit and receive FIFO registers, sets the number of trigger data, and contains an enable bit for the loop back test.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSE	TCRST	—	—	—	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	TSE	0	R/W	Transmit Data Stop Enable Enables or disables transmit data stop function. This function is enabled only in asynchronous mode. Since this function is not supported in synchronous mode, clear this bit to 0 in synchronous mode. 0: Transmit data stop function disabled 1: Transmit data stop function enabled
14	TCRST	0	R/W	Transmit Count Reset Clears the transmit count to 0. This bit is available while the transmit data stop function is enabled. 0: Transmit count reset disabled* 1: Transmit count reset enabled (cleared to 0) Note: * The transmit count is reset (cleared to 0) by a power-on reset.
13 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	RSTRG [2:0]	000	R/W	Trigger of the $\overline{\text{RTS}}$ Output Active The $\overline{\text{RTS}}$ signal goes to high, when the number of received data bytes stored in SCAFRDR has become equal to or more than the trigger number setting listed below. 000: 63 001: 1 010: 8 011: 16 100: 32 101: 48 110: 54 111: 60
7, 6	RTRG[1:0]	00	R/W	Receive FIFO Data Trigger Number Set the number of received data bytes at which the received data full (RDF) flag in SCASSR is set. The RDF flag is set when the number of received data bytes stored in SCAFRDR has become equal to or more than the trigger number setting listed below. 00: 1 01: 16 10: 32 11: 48
5, 4	TTRG[1:0]	00	R/W	Transmit FIFO Data Trigger Number Set the number of remaining transmit data bytes at which the transmit FIFO data register empty (TDFE) flag in SCASSR is set. The TDFE flag is set when the number of transmit data bytes in SCAFTDR has become equal to or less than the trigger number setting listed below. 00: 32 (32) 01: 16 (49) 10: 2 (62) 11: 0 (64) Note: * Values in brackets mean the number of empty bytes in SCAFTDR when the TDFE is set.

Bit	Bit Name	Initial Value	R/W	Description
3	MCE	0	R/W	Modem Control Enable Enables the modem control signals $\overline{\text{CTS}}$ and $\overline{\text{RTS}}$. 0: Disables the modem signal* 1: Enables the modem signal Note: * The $\overline{\text{CTS}}$ is fixed to active 0 regardless of the input value, and the $\overline{\text{RTS0}}$ is also fixed to 0.
2	TFRST	0	R/W	Transmit FIFO Data Register Reset Cancels the transmit data in the transmit FIFO data register and resets the data to the empty state. 0: Disables reset operation* 1: Enables reset operation Note: * The reset is executed in a power-on reset.
1	RFRST	0	R/W	Receive FIFO Data Register Reset Cancels the received data in the receive FIFO data register and resets the data to the empty state. 0: Disables reset operation* 1: Enables reset operation Note: * The reset is executed in a power-on reset.
0	LOOP	0	R/W	Loop Back Test Internally connects the transmit output pin (TXD) and receive input pin (RXD), $\overline{\text{RTS}}$ and $\overline{\text{CTS}}$ and enables the loop back test. 0: Disables the loop back test 1: Enables the loop back test

27.3.11 FIFO Data Count Register (SCAFDR)

SCAFDR is a 16-bit register which indicates the number of data bytes stored in SCAFTDR and SCAFRDR.

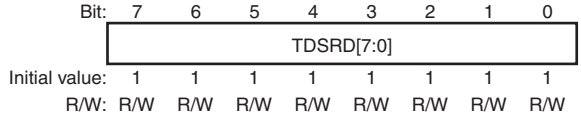
The bits 14 to 8 of this register indicate the number of transmit data bytes stored in SCAFTDR that have not yet been transmitted.

The bits 6 to 0 of this register indicate the number of received data bytes stored in SCAFRDR.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	T[6:0]	H'00	R	These bits indicate the number of non-transmitted data stored in SCAFTDR. The H'00 means no transmit data, and the H'40 means that SCAFTDR is full of transmit data.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 0	R[6:0]	H'00	R	These bits indicate the number of received data bytes stored in SCAFRDR. The H'00 means no received data, and the H'40 means that SCAFRDR is full of received data.

27.3.12 Transmit Data Stop Register (SCATDSR)

SCATDSR is an 8-bit readable/writable register that sets the number of data bytes to be transmitted. This register is available when the TSE bit in SCAFCR is enabled. The transmit operation stops after all data set by this register have been transmitted. Settable values are H'00 (1 byte) to H'FF (256 bytes). The initial value of this register is H'FF.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	TDSRD[7:0]	H'FF	R/W	Transmit Data Stop Setting

27.4 Operation

27.4.1 Overview

For serial communication, the SCIFA has asynchronous mode in which characters are synchronized individually and synchronous mode in which synchronization is achieved with clock pulses. The SCIFA has the 64-byte FIFO buffer for both transmission and reception, reduces an overhead of the CPU, and enables continuous high-speed communication.

27.4.2 Asynchronous Mode

Operation in asynchronous mode is described below.

The transmission and reception format is selected in SCASMR, as listed in Table 27.5.

- Data length is selectable from seven or eight bits.
- Parity bit is selectable. So is the stop bit length (one or two bits). The combination of the preceding selections constitutes the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, overrun errors, receive FIFO data full, received data ready, and breaks.
- The number of stored data for both the transmit and receive FIFO registers is displayed.
- Clock source: Internal clock/external clock
  - Internal clock: SCIFA operates using the on-chip baud rate generator
  - External clock: The clock appropriate for the sampling rate should be input. For example, when the sampling rate is 1/16, input the clock frequency 8 times the bit rate. (The internal baud rate generator should not be used.)

Table 27.5 SCASMR Setting and SCIFA Transmit/Receive Format

SCASMR Setting				SCIFA Transmit/Receive Format			
Bit 6	Bit 5	Bit 3		Data Length	Multi-processor Bit	Parity Bit	Stop Bit Length
CHR	PE	STOP	Mode				
0	0	0	Asynchronous mode	8-bit data	None	None	1 bit
		1					2 bits
	1	0				Yes	1 bit
		1					2 bits
1	0	0		7-bit data		None	1 bit
		1					2 bits
	1	0				Yes	1 bit
		1					2 bits

27.4.3 Serial Operation

(1) Transmit/Receive Formats

Table 27.6 lists eight communication formats that can be selected. The format is selected by settings in SCASMR.

Table 27.6 Serial Transmit/Receive Formats

SCASMR Setting			Serial Transmit/Receive Format and Frame Length											
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12
0	0	0	START	8-Bit data								STOP		
0	0	1	START	8-Bit data								STOP	STOP	
0	1	0	START	8-Bit data								P	STOP	
0	1	1	START	8-Bit data								P	STOP	STOP
1	0	0	START	7-Bit data							STOP			
1	0	1	START	7-Bit data							STOP	STOP		
1	1	0	START	7-Bit data							P	STOP		
1	1	1	START	7-Bit data							P	STOP	STOP	

Note: START: Start bit
STOP: Stop bit
P: Parity bit

(2) Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input at the SCK pin can be selected as the SCIFA's serial clock, according to the setting of the CKE bit in SCASCR.

When an external clock is input at the SCK pin, the clock appropriate for the sampling rate should be input. For example, when the sampling rate is 1/16, the clock frequency should be 8 times the bit rate used.

(3) Transmitting and Receiving Data

(a) SCIFA Initialization

Before transmitting or receiving, clear the TE and RE bits to 0 in SCASCR, then initialize the SCIFA as follows.

When changing the communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing TE to 0 initializes SCATSR. Clearing TE and RE to 0, however, does not initialize SCASSR, SCAFTDR, or SCAFRDR, which retain their previous contents.

Clear TE to 0 after all transmit data are transmitted and the TEND bit in the SCASSR is set. The transmitting data enters the high impedance state after clearing to 0 although the bit can be cleared to 0 in transmitting. Set the TFRST bit in the SCAFCR to 1 and reset SCAFTDR before TE is set again to start transmission.

Figure 27.2 is a sample flowchart for initializing the SCIFA.

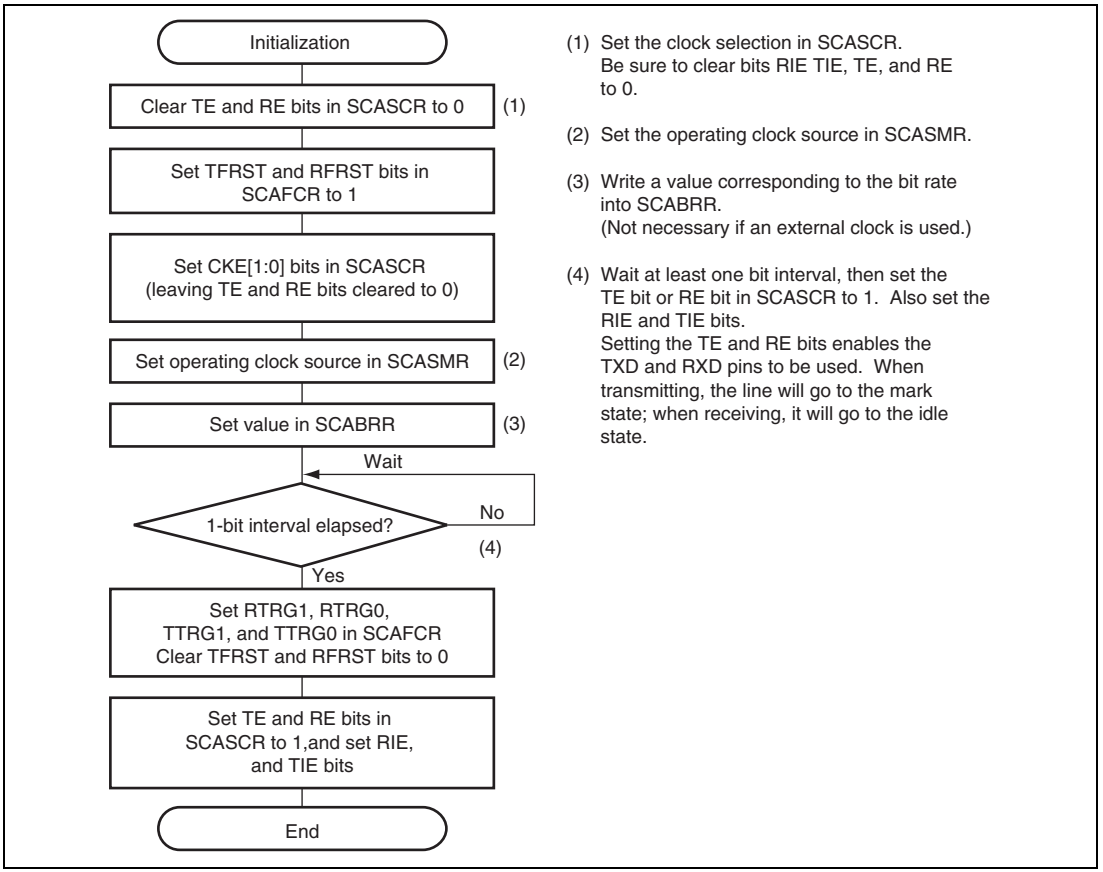


Figure 27.2 Sample SCIFA Initialization Flowchart

(b) Serial Data Transmission

Figure 27.3 shows a sample serial transmission flowchart. After SCIFA transmission is enabled, use the following procedure to perform serial data transmission.

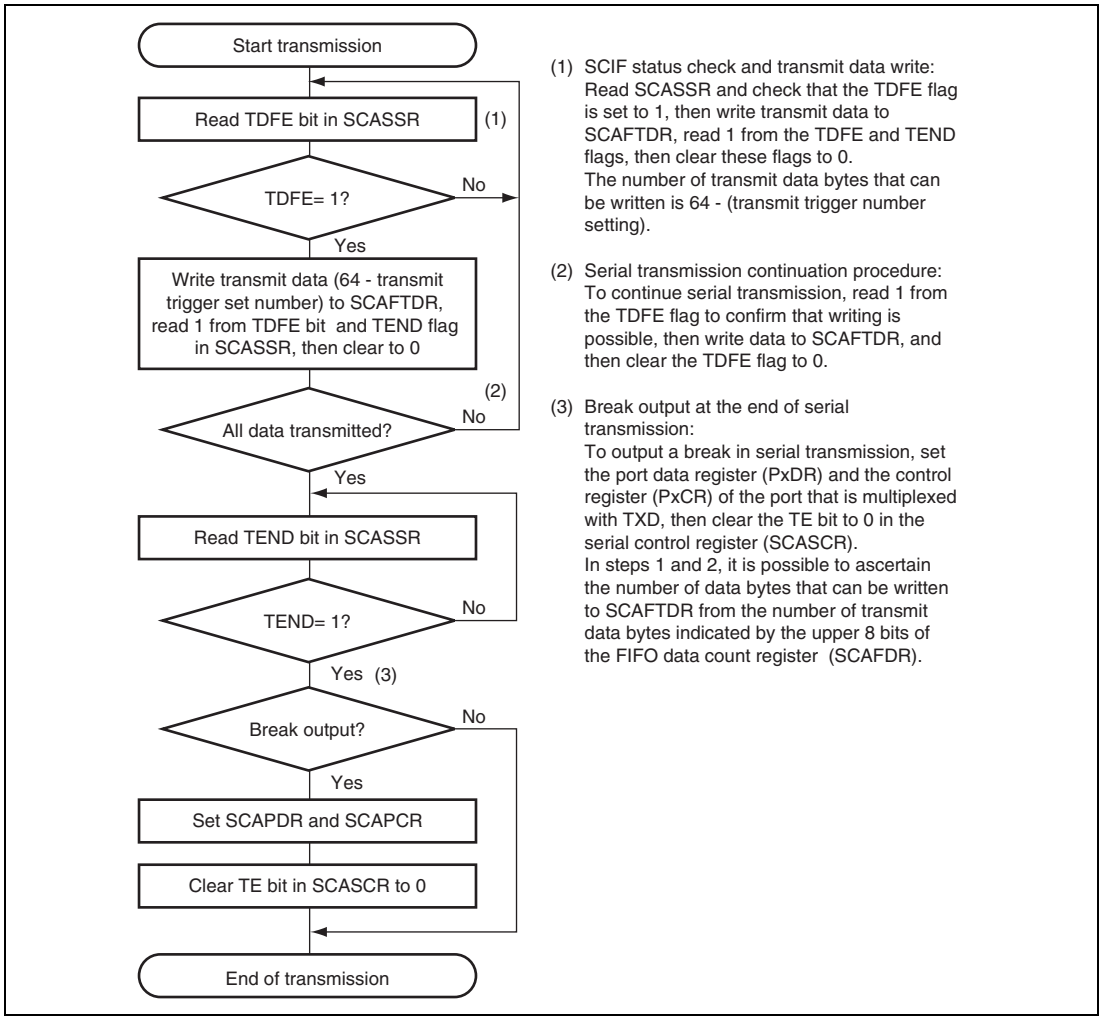


Figure 27.3 Sample Serial Transmission Flowchart

In serial transmission, the SCIFA operates as described below.

1. When data is written into SCAFTDR, the SCIFA transfers the data from SCAFTDR to SCATSR and starts transmitting. Confirm that the TDFE flag in SCASSR is set to 1 before writing transmit data to SCAFTDR. The number of data bytes that can be written is at least (64 – transmission trigger number setting).
2. When data is transferred from SCAFTDR to SCATSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCAFTDR. When the number of transmit data bytes in SCAFTDR falls below the transmission trigger number set in SCAFCR, the TDFE flag is set. If the TIE bit in SCASCR is set to 1 at this time, a transmit-FIFO-data-empty interrupt request is generated.
   When the number of transmit data bytes matches the data set in SCATDSR while the transmit data stop function is used, the transmit operation is stopped and the TSF flag in SCASSR is set. When the TSIE bit in SCASCR is set to 1, transmit data stop interrupt request is generated. A common interrupt vector is assigned to the transmit-FIFO-data-empty interrupt and the transmit-data-stop interrupt.

   The serial transmit data is sent from the TXD pin in the following order.

   A. Start bit: One-bit 0 is output.
   B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
   C. Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
   D. Stop bit(s): One- or two-bit 1s (stop bits) are output.
   E. Mark state: 1 is output continuously until the start bit that starts the next transmission is sent.

3. The SCIFA checks SCAFTDR transmit data at the timing for sending the stop bit. If data is present, the data is transferred from SCAFTDR to SCATSR, the stop bit is sent, and then serial transmission of the next frame is started.

   If there is no transmit data, the TEND flag in SCASSR is set to 1, the stop bit is sent, and then the line goes to the mark state in which 1 is output continuously.

   Figure 27.4 shows an example of the operation for transmission in asynchronous mode.

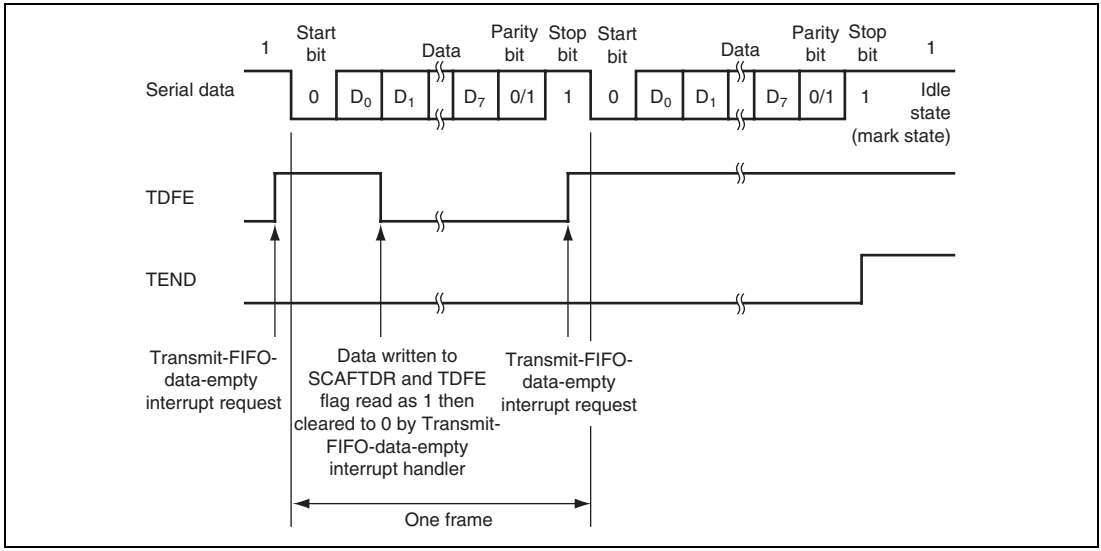


Figure 27.4 Example of Transmit Operation (Example with 8-Bit Data, Parity, One Stop Bit)

(c) Transmit Data Stop

When the value of the SCATDSR register and the number of transmit data bytes match, transmit operation stops. Setting the TSIE bit (interrupt enable bit) allows the generation of an interrupt and activation of DMAC.

Figure 27.5 shows an example of the operation for transmit data stop function.

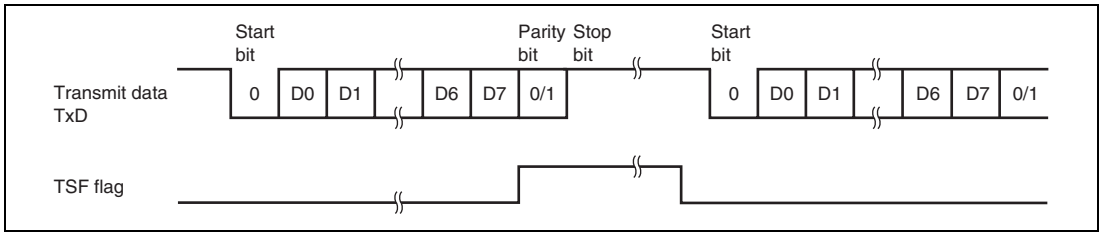


Figure 27.5 Example of Transmit Data Stop Function

Figure 27.6 shows the transmit data stop function flowchart.

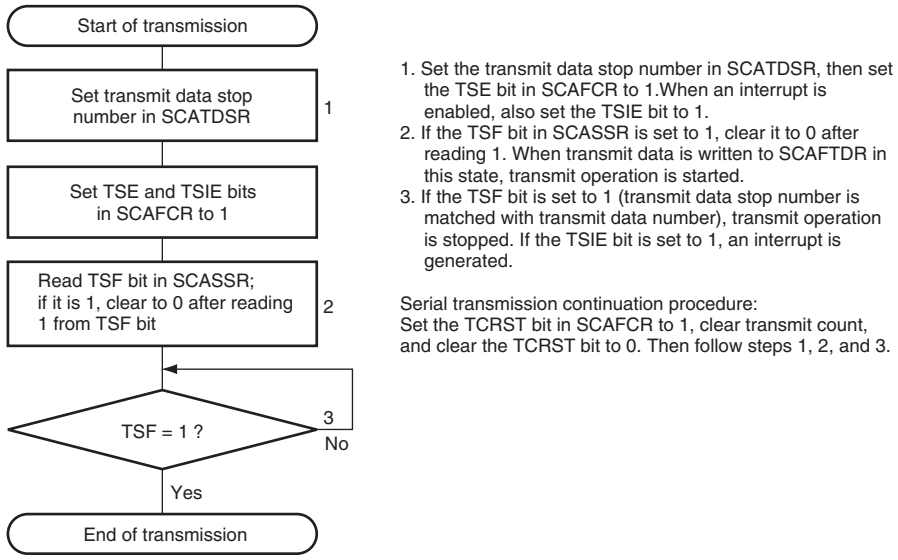


Figure 27.6 Transmit Data Stop Function Flowchart

(d) Serial Data Reception

Figures 27.7 and 27.8 show sample serial reception flowcharts. After SCIFA reception is enabled, use the following procedure to perform serial data reception.

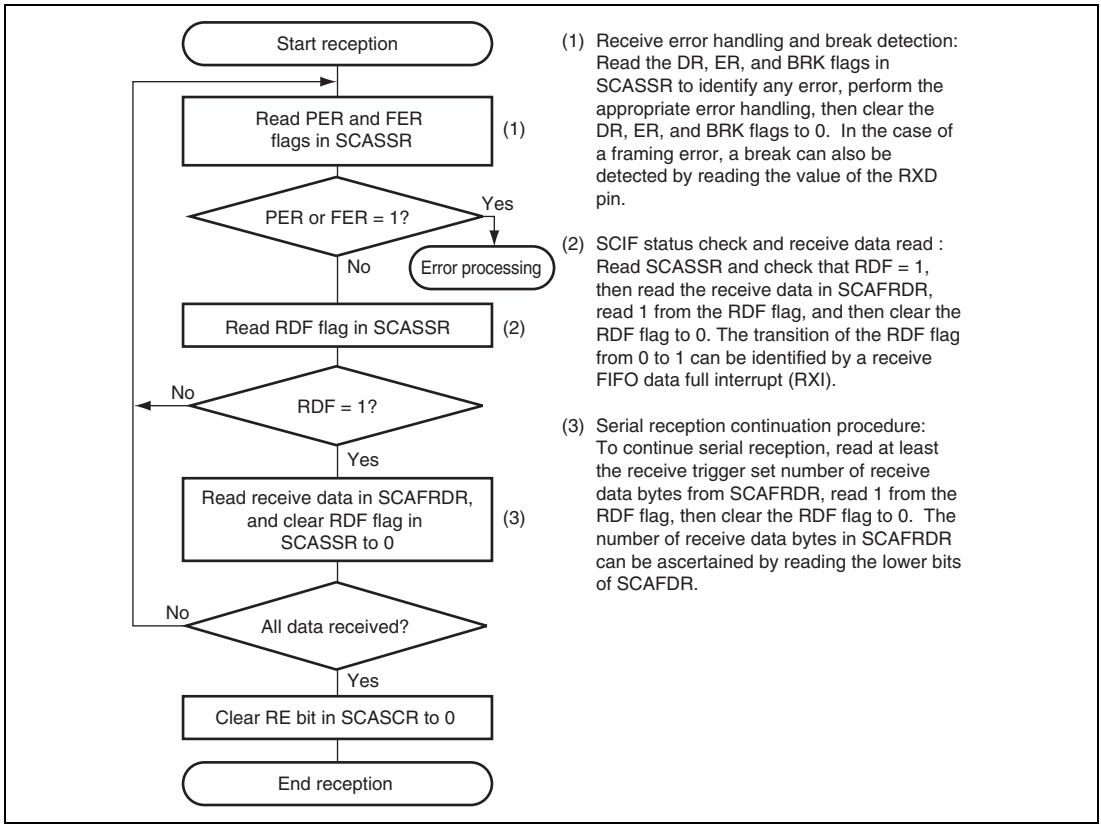
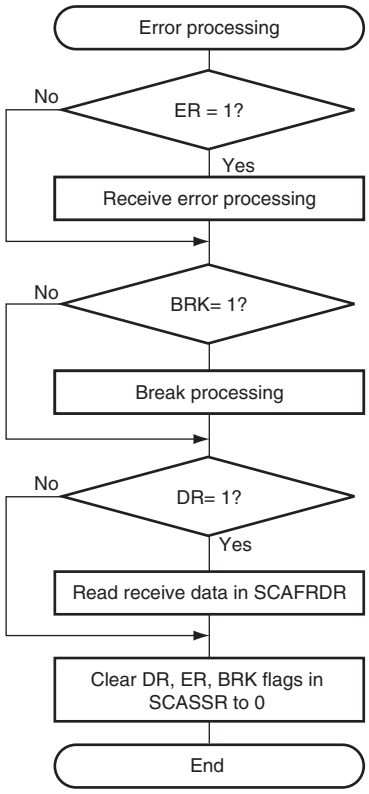


Figure 27.7 Sample Serial Reception Flowchart (1)



1. Whether a framing error or parity error has occurred in the receive data read from SCAFRDR can be ascertained from the FER and PER bits in SCASSR.
2. When a break signal is received, receive data is not transferred to SCAFRDR while the BRK flag is set. However, note that the last data in SCAFRDR is H'00 and the break data in which a framing error occurred is stored.

Figure 27.8 Sample Serial Reception Flowchart (2)

In serial reception, the SCIFA operates as described below.

1. The SCIFA monitors the transmission line, and if a 0 start bit is detected, performs internal synchronization and starts reception.
2. The received data is stored in SCARSR in LSB-to-MSB order.
3. The parity bit and stop bit are received.

   After receiving these bits, the SCIFA carries out the following checks.

   A. Stop bit check: The SCIFA checks whether the stop bit is 1. If there are two stop bits, only the first is checked.

   B. The SCIFA checks whether received data can be transferred from SCARSR to SCAFRDR.

   C. Break check: The SCIFA checks that the BRK flag is 0, indicating that the break state is not set.

   If all the above checks are passed, the received data is stored in SCAFRDR.

Note: Even when the receive error (framing error/parity error) is generated, receive operation is continued.

4. If the RIE bit in SCASCR is set to 1 when the RDF flag changes to 1, a receive-FIFO-data-full interrupt request is generated.
   If the ERIE bit in SCASCR is set to 1 when the ER flag changes to 1, a receive-error interrupt request is generated.
   If the BRIE bit in SCASCR is set to 1 when the BRK flag changes to 1, a break reception interrupt request is generated.
   If the DRIE bit in SCASCR is set to 1 when the DR flag changes to 1, a received data ready interrupt request is generated.
   Note that a common vector is assigned to each interrupt source.

Figure 27.9 shows an example of the operation for reception.

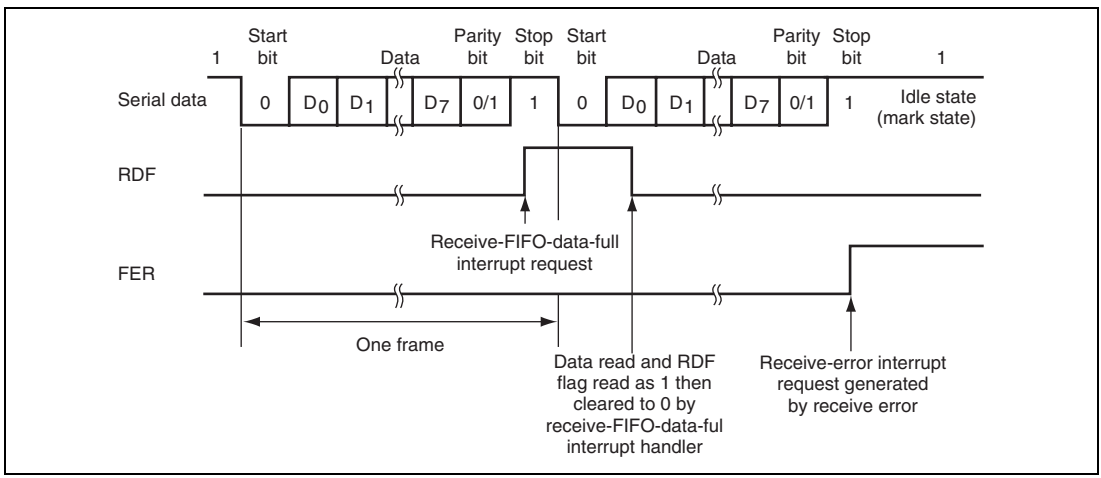


Figure 27.9 Example of SCIFA Receive Operation (Example with 8-Bit Data, Parity, One Stop Bit)

When modem control is enabled, transmission can be stopped and restarted in accordance with the $\overline{\text{CTS}}$ input value. When $\overline{\text{CTS}}$ is set to 1, if transmission is in progress, the line goes to the mark state after transmission of one frame. When $\overline{\text{CTS}}$ is set to 0, the next transmit data is output starting from the start bit.

Figure 27.10 shows an example of the operation when modem control is used.

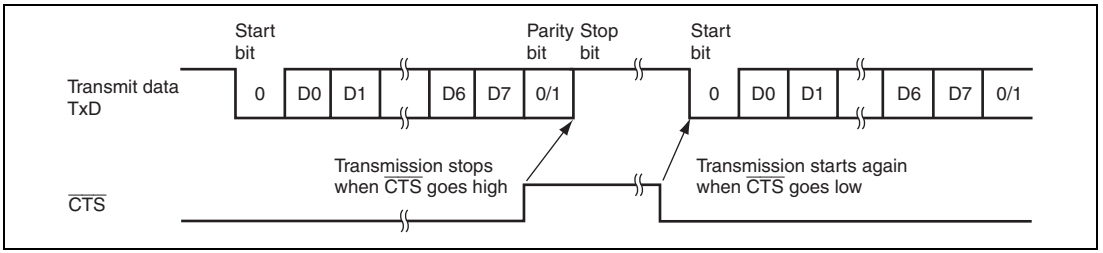


Figure 27.10 Example of $\overline{\text{CTS}}$ Control Operation

When modem control is enabled, the $\overline{\text{RTS}}$ signal goes high after the number of receive FIFO (SCAFRDR) has exceeded the number of $\overline{\text{RTS}}$ output triggers.

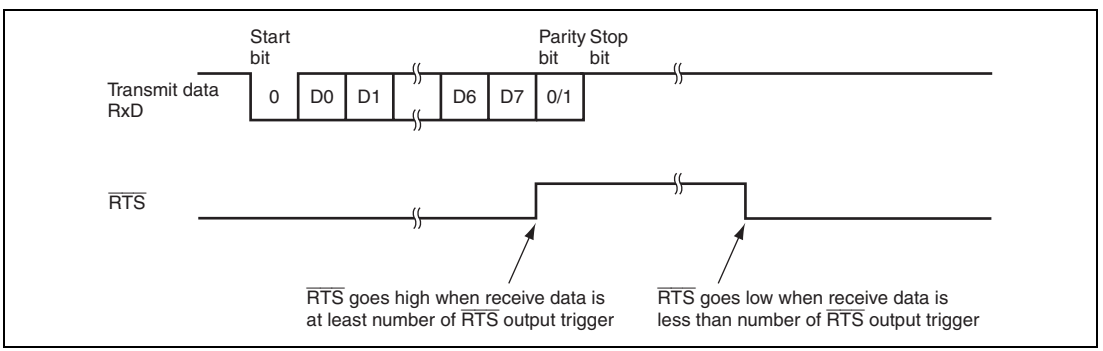


Figure 27.11 Example of $\overline{\text{RTS}}$ Control Operation

27.4.4 Synchronous Mode

Operation in synchronous mode is described below.

The SCIFA has 64-stage FIFO buffers for both transmission and reception, reducing the CPU overhead and enabling fast, continuous communication to be performed.

The operating clock source is selected using SCASMR. The SCIFA clock source is determined by the CKE[1:0] bits in SCASCR.

- Transmit/receive format: Fixed 8-bit data
- Indication of the number of data bytes stored in the transmit and receive FIFO registers
- Internal clock or external clock used as the SCIFA clock source

  When the internal clock is selected:
  The SCIFA operates on the baud rate generator clock and outputs a serial clock from SCK pin.

  When the external clock is selected:
  The SCIFA operates on the external clock input through the SCK pin.

27.4.5 Serial Operation in Synchronous Mode

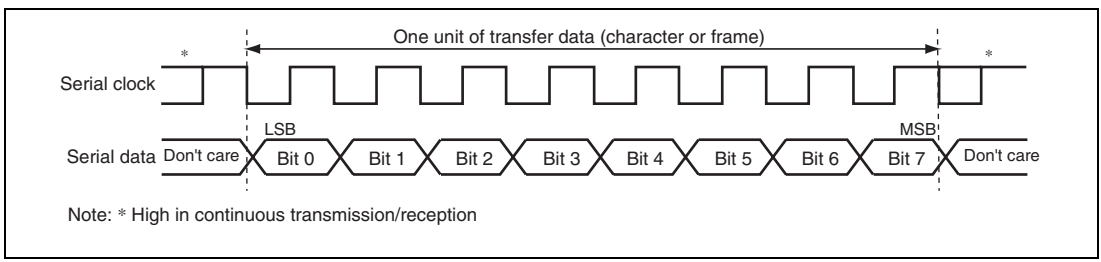


Figure 27.12 Data Format in Synchronous Communication

In synchronous serial communication, data on the communication line is output from a falling edge of the serial clock to the next falling edge. Data is guaranteed valid at the rising edge of the serial clock.

In serial communication, each character is output starting with the LSB and ending with the MSB. After the MSB is output, the communication line remains in the state of the MSB.

In synchronous mode, the SCIFA receives data in synchronization with the rising edge of the serial clock.

(1) Data Transfer Format

A fixed 8-bit data format is used. No parity or multiprocessor bit is added.

(2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input through the SCK pin can be selected as the serial clock for the SCIFA, according to the setting of the CKE[1:0] bits in SCASCR.

Eight serial clock pulses are output in the transfer of one character, and when no transmission/reception is performed, the clock is fixed high. However, when the operation mode is reception only, the synchronous clock output continues while the RE bit is set to 1. To fix the clock high every time one character is transferred, write to SCAFTDR the same number of dummy data bytes as the data bytes to be received and set the TE and RE bits to 1 at the same time to transmit the dummy data. When the specified number of data bytes are transmitted, the clock is fixed high.

(3) Data Transfer Operations

(a) SCIFA Initialization

Before transmitting and receiving data, it is necessary to clear the TE and RE bits in SCASCR to 0, then initialize the SCIFA as described below.

When the clock source, etc., is changed, the TE and RE bits must be cleared to 0 before making the change using the following procedure. When the TE bit is cleared to 0, SCATSR is initialized. Note that clearing the TE and RE bits to 0 does not change the contents of SCASSR, SCAFTDR, or SCAFRDR. The TE bit should be cleared to 0 after all transmit data has been sent and the TEND bit in SCASSR has been set to 1. The TE bit should not be cleared to 0 during transmission; if attempted, the TXD pin will go to the high-impedance state. Before setting TE to 1 again to start transmission, the TFRST bit in SCAFCR should first be set to 1 to reset SCAFTDR.

Figure 27.13 shows sample SCIFA initialization flowcharts.

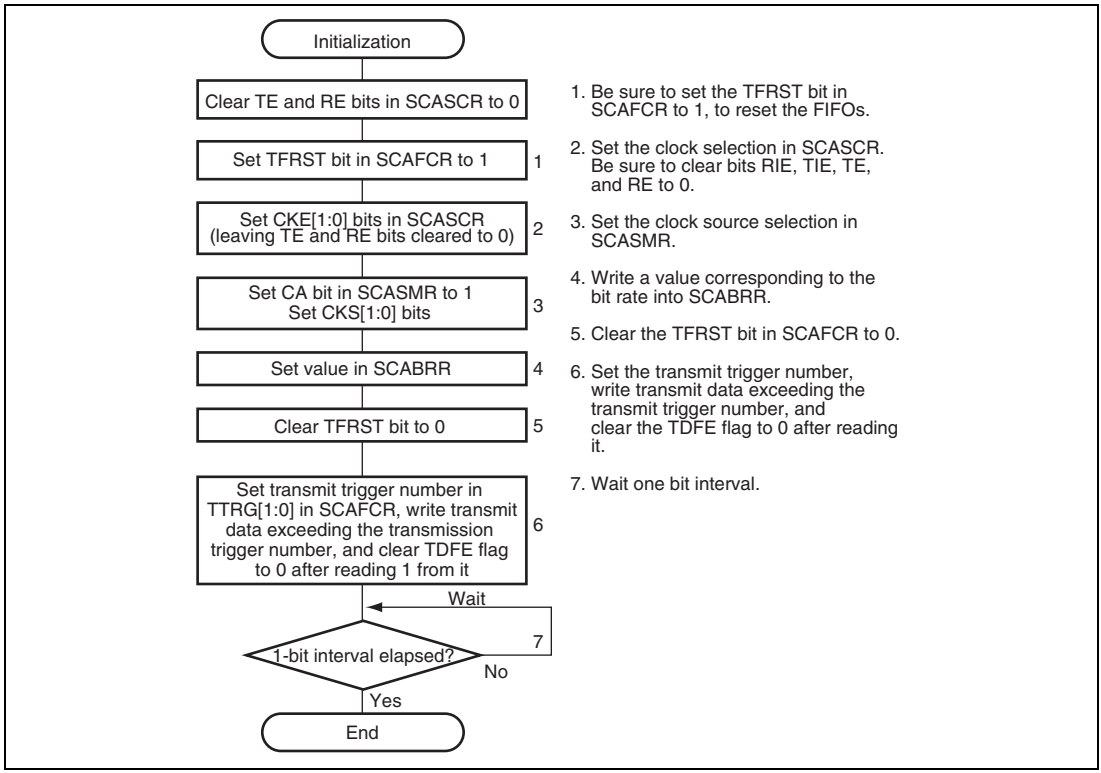


Figure 27.13 Sample SCIFA Initialization Flowchart (1) (Transmission)

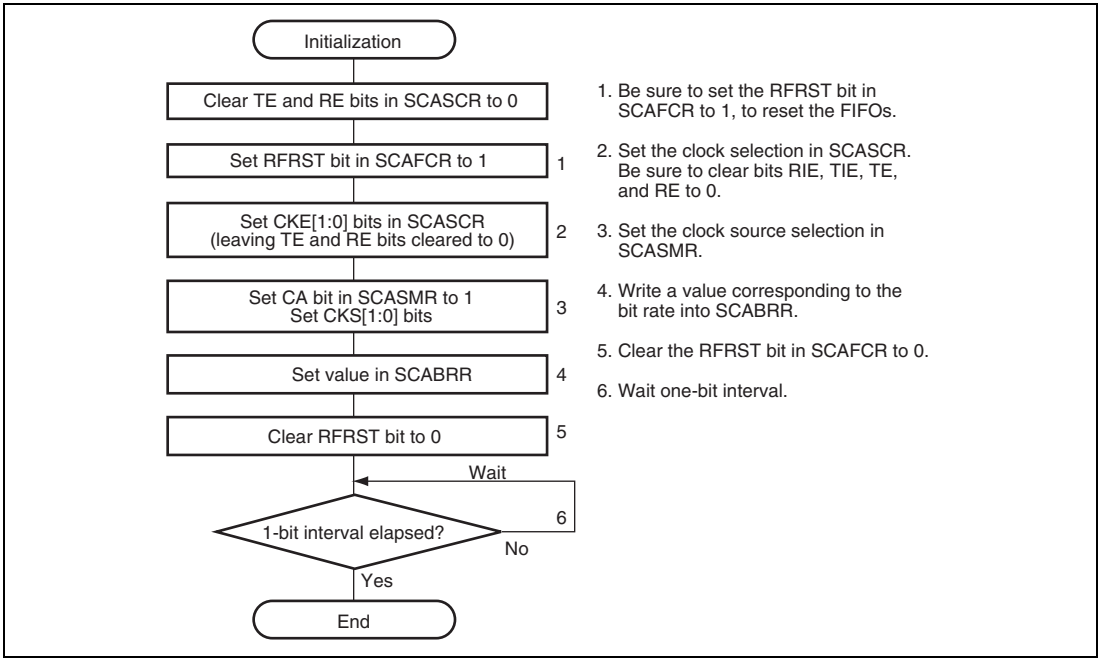


Figure 27.14 Sample SCIFA Initialization Flowchart (2) (Reception)

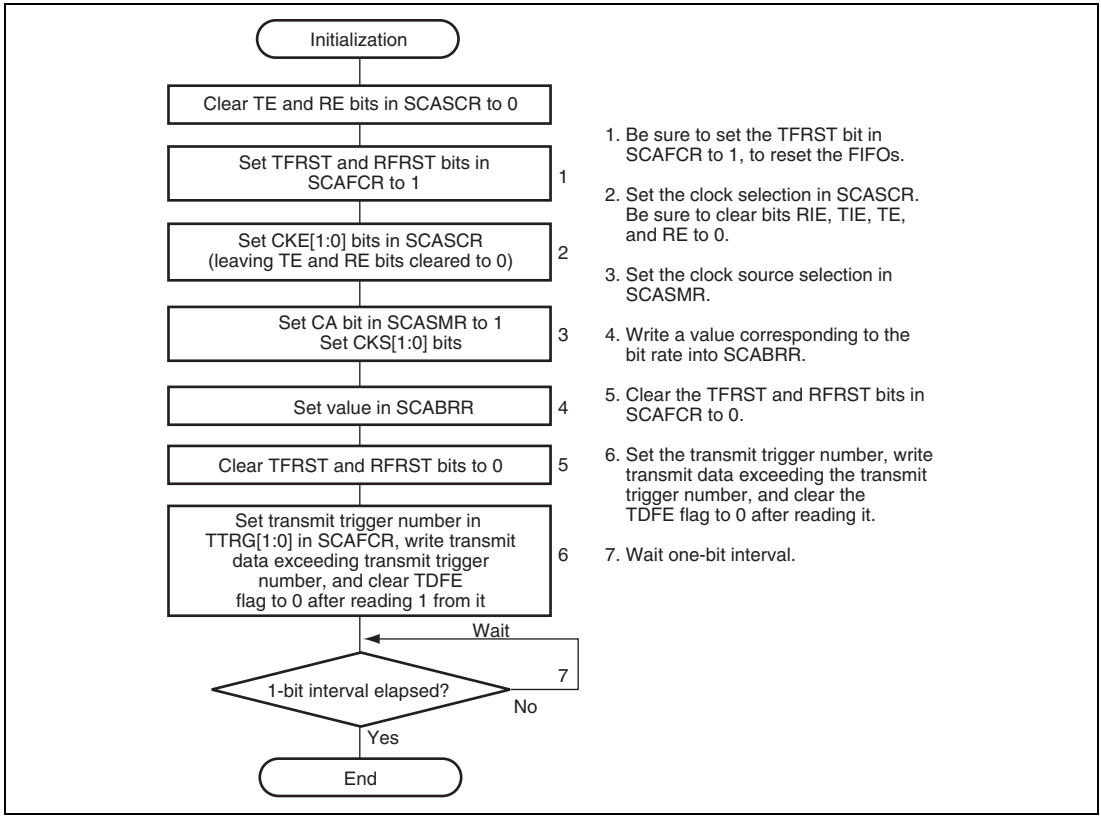


Figure 27.15 Sample SCIFA Initialization Flowchart (3)
(Simultaneous Transmission and Reception)

(b) Serial Data Transmission

Figures 27.16 and 22.17 show sample flowcharts for serial transmission.

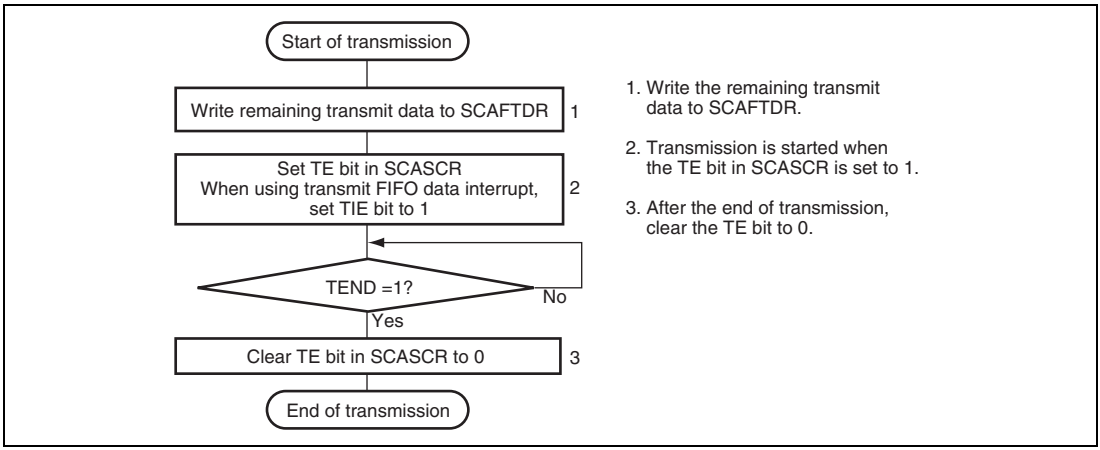


Figure 27.16 Sample Serial Transmission Flowchart (1)
(First Transmission after Initialization)

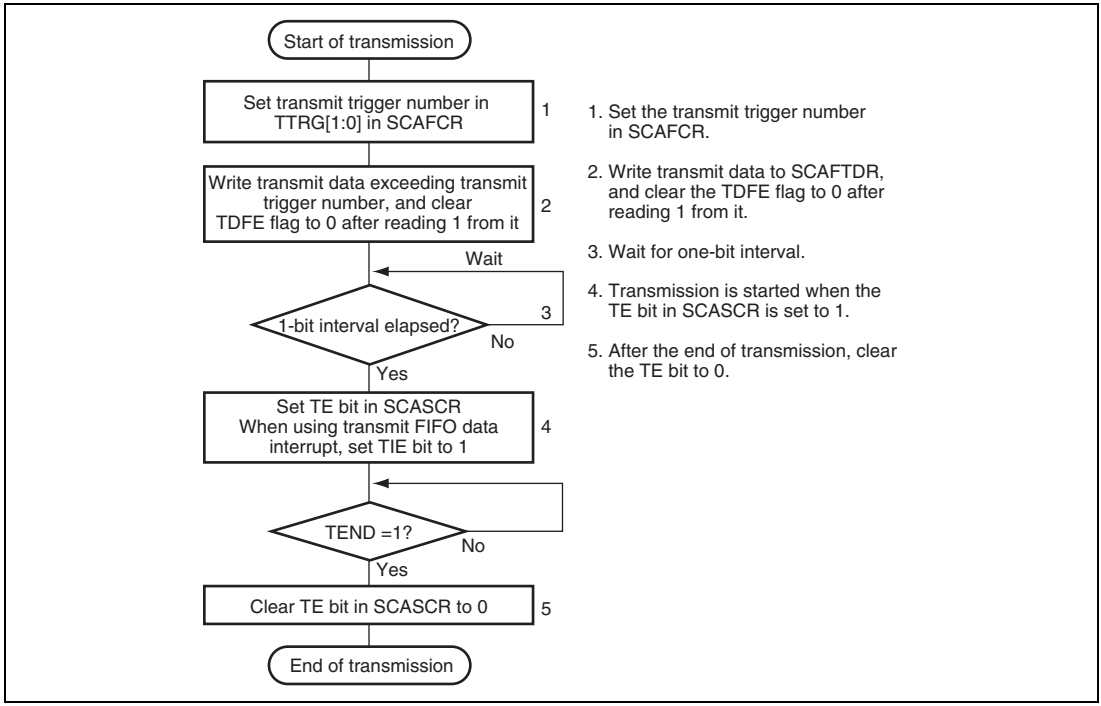


Figure 27.17 Sample Serial Transmission Flowchart (2)
(Second and Subsequent Transmission)

(c) Serial Data Reception

Figures 27.18 and 22.19 show sample flowcharts for serial reception.

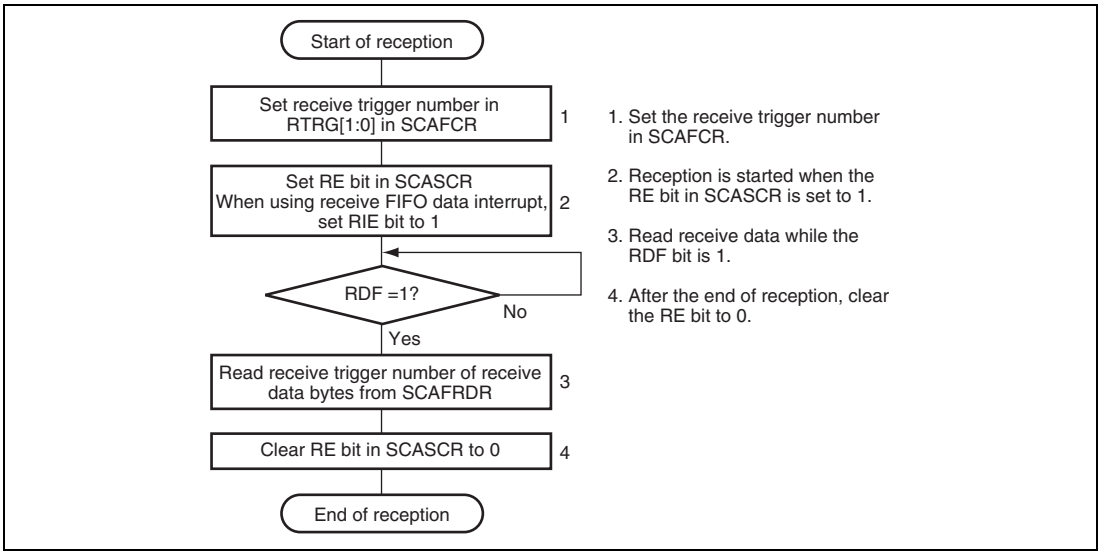


Figure 27.18 Sample Serial Reception Flowchart (1)
(First Reception after Initialization)

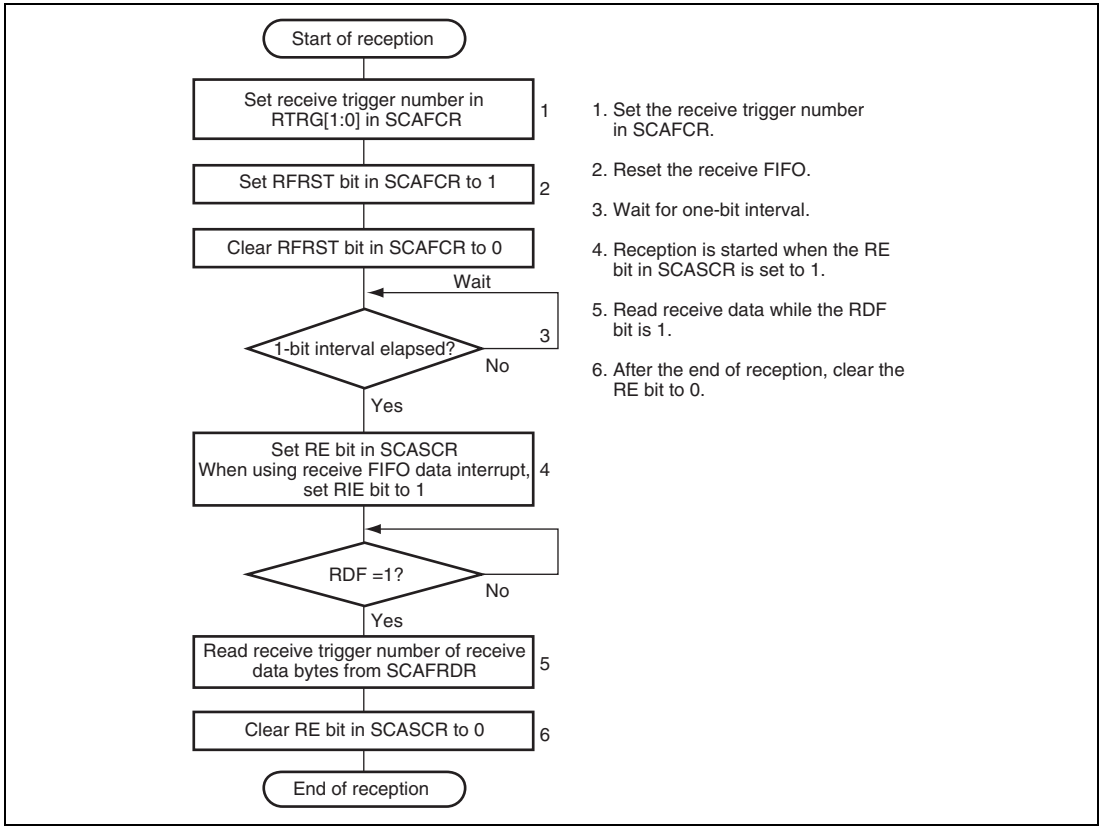


Figure 27.19 Sample Serial Reception Flowchart (2)
(Second and Subsequent Reception)

(d) Simultaneous Serial Data Transmission and Reception

Figures 27.20, and 22.21 show sample flowcharts for simultaneous serial transmission and reception.

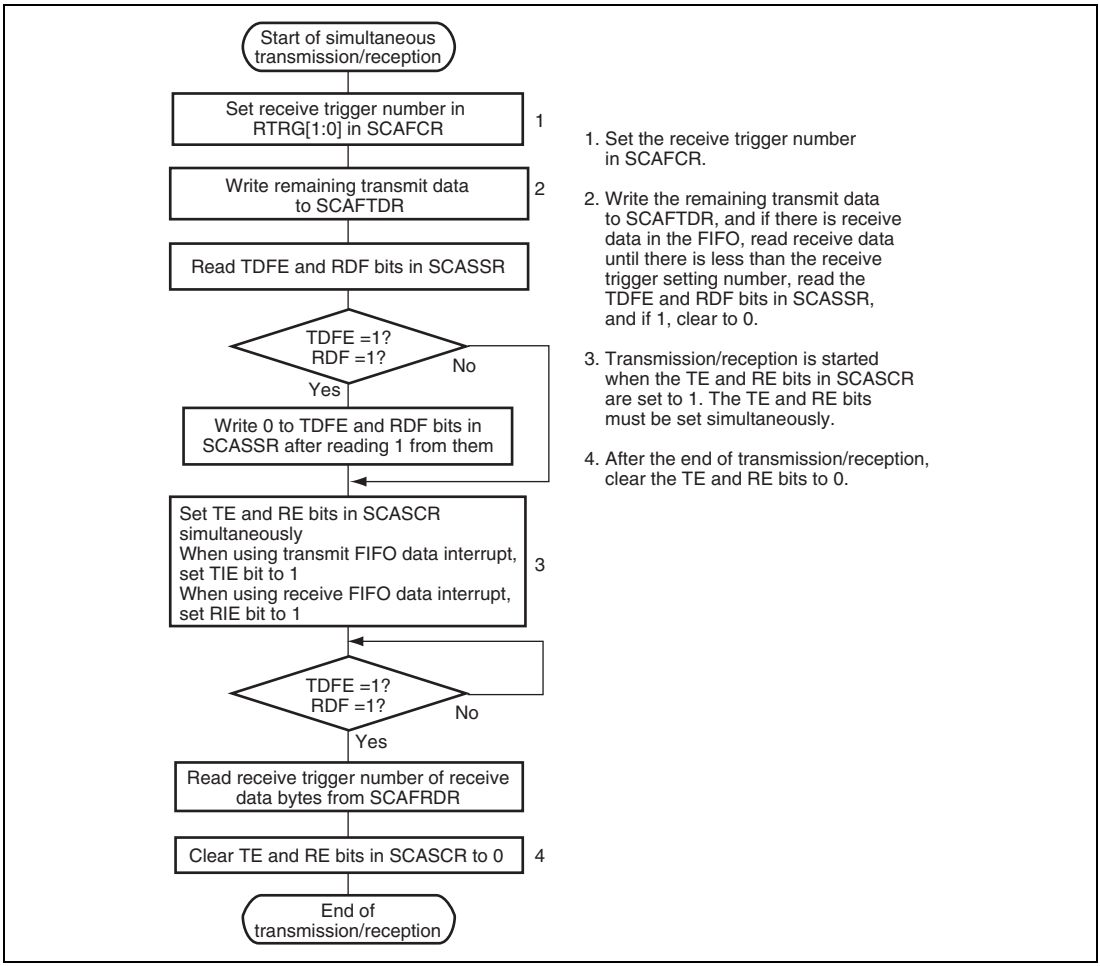
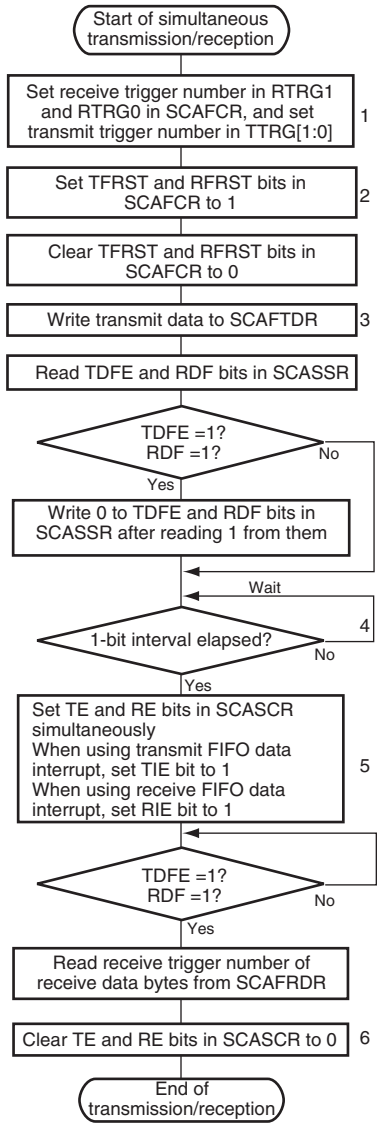


Figure 27.20 Sample Simultaneous Serial Transmission and Reception Flowchart (1) (First Transfer after Initialization)



1. Set the receive trigger number and transmit trigger number in SCAFCR.
2. Reset the receive FIFO and transmit FIFO.
3. Write transmit data to SCAFTDR, and if there is receive data in the FIFO, read receive data until there is less than the receive trigger setting number, read the TDFE and RDF bits in SCASSR, and if 1, clear to 0.
4. Wait for one-bit interval.
5. Transmission/reception is started when the TE and RE bits in SCASCR are set to 1. The TE and RE bits must be set simultaneously.
6. After the end of transmission/reception, clear the TE and RE bits to 0.

Figure 27.21 Sample Simultaneous Serial Transmission and Reception Flowchart (2) (Second and Subsequent Transfer)

27.5 Interrupt Sources and DMAC

In asynchronous mode, the SCIFA supports six interrupts: transmit-FIFO-data-empty, transmit data stop, receive-error, receive-FIFO-data-full, break receive, and received data ready. A common interrupt vector is assigned to each interrupt source.

In synchronous mode, the SCIFA supports two interrupts: transmit-FIFO-data-empty and receive-FIFO-data-full.

Table 27.7 shows the interrupt sources. The interrupt sources are enabled or disabled by means of the TIE, RIE, ERIE, BRIE, DRIE, and TSIE bits in SCASCR.

When the TDFE flag in SCASSR is set to 1, the transmit-FIFO-data-empty interrupt request is generated. When the TSF flag in SCASSR is set to 1, the transmit-data-stop interrupt request is generated. Activating the DMAC and transferring data can be performed by the transmit-FIFO-data-empty interrupt and data stop interrupt requests. The DMAC transfer request is automatically cleared when the number of data bytes written to SCAFTDR by the DMAC is increased more than that of setting transmit triggers.

When the RDF flag in SCASSR is set to 1, a receive-FIFO-data-full interrupt request is generated. Activating the DMAC and transferring data can be performed by the receive-FIFO-data-full interrupt request. The DMAC transfer request is automatically cleared when received data is read from SCAFRDR by the DMAC until the number of received data bytes in SCAFRDR is decreased less than that of receive triggers.

When executing the data transmission/reception using the DMAC, configure the DMAC first and enable it, then configure the SCIFA. The completion of the DMA transfer is the completion of transmission/reception.

An interrupt request is generated when the ER flag in SCASSR is set to1; the BRK flag in SCASSR is set to 1; the DR flag in SCASSR is set to 1; or the TSF flag in SCASSR is set to 1. A common interrupt vector is assigned to each interrupt source. The activation of DMAC and generation of an interrupt are not executed at the same time by the same source. The DMAC should be activated according to the following procedure.

1. Set the interrupt enable bit (TIE, RIE, or TDIE) corresponding to the generated interrupt source to 1.
2. Mask the corresponding interrupt request by the interrupt mask register of the interrupt controller.

Table 27.7 SCIFA Interrupt Sources

Interrupt Source	DMAC Activation
Interrupt initiated by receive error (ER) or break (BRK)	Not possible
Interrupt initiated by receive FIFO data full flag (RDF) or data ready flag (DR)	Possible*[1]
Interrupt initiated by receive FIFO data empty flag (TDFE) or transmit data stop flag (TSF)	Possible*[2]

Notes: 1. DMAC can be activated only by the receive-FIFO-data-full interrupt request.
2. DMAC can be activated only by the transmit-FIFO-data-empty interrupt request.

27.6 Usage Notes

Note the following points when using the SCIFA.

(1) SCAFTDR Writing and the TDFE Flag

The TDFE flag in the serial status register (SCASSR) is set when the number of transmit data bytes written in SCAFTDR has fallen below the transmission trigger number set by bits TTRG[1:0] in SCAFCR. After TDFE is set, transmit data up to the number of empty bytes in SCAFTDR can be written, allowing efficient continuous transmission.
However, if the number of data bytes written in SCAFTDR is less than or equal to the transmission trigger number, the TDFE flag will be set to 1 again after being cleared to 0. The TDFE flag should therefore be cleared to 0 after a number of data bytes exceeding the transmission trigger number has been written to SCAFTDR.
The number of transmit data bytes in SCAFTDR can be found in the bits T[6:0] of SCAFDR.

(2) SCAFRDR Reading and the RDF Flag

The RDF flag in SCASSR is set when the number of received data bytes in SCAFRDR has become equal to or greater than the reception trigger number set by bits RTRG[1:0] in SCAFCR. After RDF is set, received data equivalent to the trigger number can be read from SCAFRDR, allowing efficient continuous reception.
However, if the number of data bytes in SCAFRDR exceeds the trigger number, the RDF flag will be set to 1 again after being cleared to 0. The RDF flag should therefore be cleared to 0 when 1 has been written to RDF after all received data has been read.
The number of received data bytes in SCAFRDR can be found in the bits R[6:0] of SCAFDR.

(3) Break Detection and Processing

Break signals can be detected by reading the RXD pin directly when a framing error (FER) is detected. In the break state the input from the RXD pin consists of all 0s, so the FER flag is set and the parity error flag (PER) may also be set. Note that, although transfer of received data to SCAFRDR is halted in the break state, the SCIFA receiver continues to operate.

(4) Received data Sampling Timing and Receive Margin

An example with a sampling rate 1/16 is given. The SCIFA operates on a base clock with a frequency of 8 times the transfer rate. In reception, the SCIFA synchronizes internally with the fall of the start bit, which it samples on the base clock. Received data is latched at the rising edge of the eighth base clock pulse. The timing is shown in Figure 27.22.

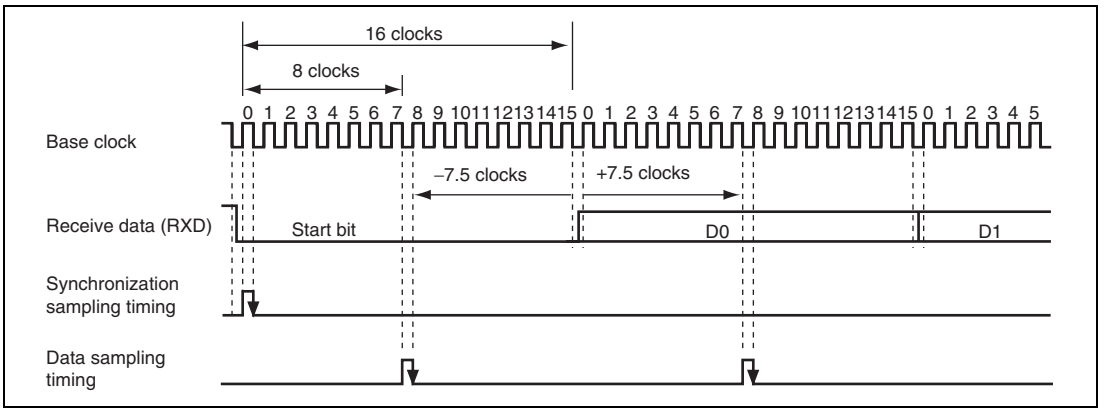


Figure 27.22 Received data Sampling Timing in Asynchronous Mode

The receive margin in asynchronous mode can therefore be expressed as shown in equation (1).

Equation 1:

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5)\,F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\% \quad \cdots\cdots\cdots\cdots (1)$$

M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty cycle (D = 0 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute deviation of clock frequency

From equation (1), if F = 0 and D = 0.5, the receive margin is 46.875%, as given by equation (2).

When D = 0.5 and F = 0:

$$M = (0.5 - 1/(2 \times 16)) \times 100\%$$
$$= 46.875\% \quad \cdots\cdots\cdots\cdots (2)$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

(5) Simultaneous serial transmission and reception in Synchronous mode

In simultaneous serial transmission and reception mode, receive operation continue until receive FIFO is full, even if transmit operation is terminated and TXD and SCK output is stopped.

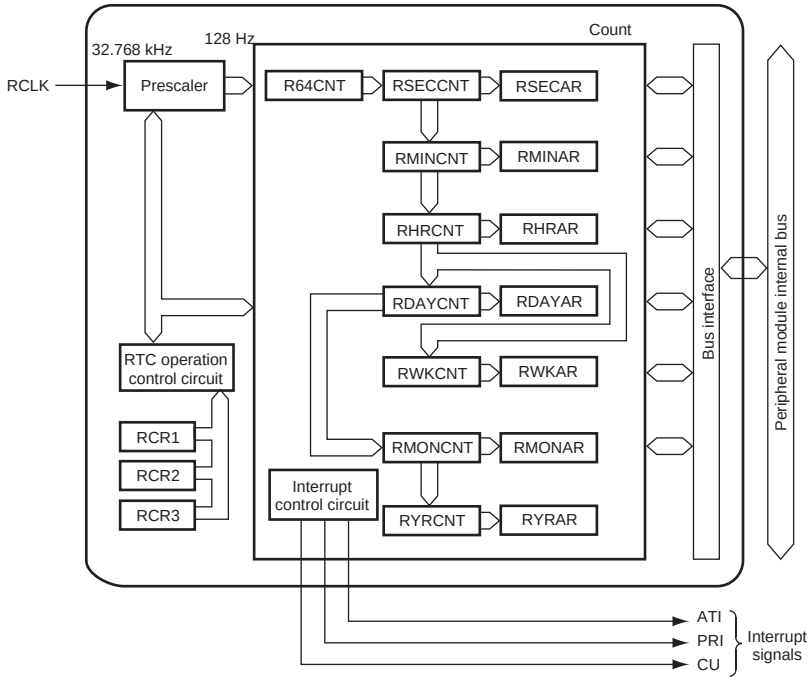
Section 28 Realtime Clock (RTC)

This LSI has a realtime clock (RTC).

28.1 Features

- Clock and calendar functions (BCD format): Seconds, minutes, hours, date, day of the week, month, and year
- 1-Hz to 64-Hz timer (binary format)
  64-Hz counter indicates the state of the RTC divider circuit between 64 Hz and 1 Hz
- Start/stop function
- 30-second adjust function
- Alarm interrupt: Frame comparison of seconds, minutes, hours, date, day of the week, month, and year can be used as conditions for the alarm interrupt
- Periodic interrupts: the interrupt cycle may be 1/256 second, 1/64 second, 1/16 second, 1/4 second, 1/2 second, 1 second, or 2 seconds
- Carry interrupt: indicates when a carry occurs or 64-Hz counter carry occurs during a 64-Hz counter read
- Automatic leap year adjustment

Figure 28.1 shows the block diagram of RTC.



[Legend]

RSECCNT:	Second counter (8 bits)	RSECAR:	Second alarm register (8 bits)
RMINCNT:	Minute counter (8 bits)	RMINAR:	Minute alarm registger (8 bits)
RHRCNT:	Hour counter (8 bits)	RHRAR:	Hour alarm register (8 bits)
RWKCNT:	Day of week counter (8 bits)	RWKAR:	Day of week alarm register (8 bits)
RDAYCNT:	Date counter (8 bits)	RDAYAR:	Date alarm register (8 bits)
RMONCNT:	Month counter (8 bits)	RMONAR:	Month alarm register (8 bits)
RYRCNT:	Year counter (16 bits)	RYRAR:	Year alarm register (16 bits)
R64CNT:	64-Hz counter (8 bits)	RCR2:	RTC control register 2 (8 bits)
RCR1:	RTC control register 1 (8 bits)	RCR3:	RTC control register 3 (8 bits)

Figure 28.1 RTC Block Diagram

28.2 Input/Output Pin

Table 28.1 shows the RTC pin configuration.

Table 28.1 Pin Configuration

Name	Abbreviation	I/O	Function
External clock for RTC	RCLK	Input	Inputs the external clock for the RTC.

28.3 Register Descriptions

Table 28.2 shows the register configuration. Table 28.3 shows the register states in each operating mode.

Table 28.2 Register Configuration of RTC

Name	Abbreviation	R/W	Address	Access Size
64-Hz counter	R64CNT	R	H'A465 FEC0	8
Second counter	RSECCNT	R/W	H'A465 FEC2	8
Minute counter	RMINCNT	R/W	H'A465 FEC4	8
Hour counter	RHRCNT	R/W	H'A465 FEC6	8
Day of week counter	RWKCNT	R/W	H'A465 FEC8	8
Date counter	RDAYCNT	R/W	H'A465 FECA	8
Month counter	RMONCNT	R/W	H'A465 FECC	8
Year counter	RYRCNT	R/W	H'A465 FECE	16
Second alarm register	RSECAR	R/W	H'A465 FED0	8
Minute alarm register	RMINAR	R/W	H'A465 FED2	8
Hour alarm register	RHRAR	R/W	H'A465 FED4	8
Day of week alarm register	RWKAR	R/W	H'A465 FED6	8
Date alarm register	RDAYAR	R/W	H'A465 FED8	8
Month alarm register	RMONAR	R/W	H'A465 FEDA	8
RTC control register 1	RCR1	R/W	H'A465 FEDC	8
RTC control register 2	RCR2	R/W	H'A465 FEDE	8
Year alarm register	RYRAR	R/W	H'A465 FEE0	16
RTC control register 3	RCR3	R/W	H'A465 FEE4	8

Table 28.3 Register State of RTC in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
R64CNT	Undefined	Retained	Retained	Retained	Retained	Retained
RSECCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RMINCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RHRCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RWKCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RDAYCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RMONCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RYRCNT	Undefined	Retained	Retained	Retained	Retained	Retained
RSECAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained
RMINAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained
RHRAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained
RWKAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained
RDAYAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained
RMONAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained
RCR1	Initialized/ Undefined	Initialized	Retained	Retained	Retained	Retained
RCR2	Initialized	Initialized/ Undefined	Retained	Retained	Retained	Retained
RYRAR	Undefined	Retained	Retained	Retained	Retained	Retained
RCR3	Initialized	Initialized	Retained	Retained	Retained	Retained

28.3.1 64-Hz Counter (R64CNT)

R64CNT indicates the state of the divider circuit between 64 Hz and 1 Hz.

Reading this register, when carry from 128-Hz divider stage is generated, sets the CF bit in the RTC control register 1 (RCR1) to 1 so that the carrying and reading 64 Hz counter are performed at the same time is indicated. In this case, the R64CNT should be read again after writing 0 to the CF bit in RCR1 since the read value is not valid.

After the RESET bit or ADJ bit in the RTC control register 2 (RCR2) is set to 1, the RTC divider circuit is initialized and R64CNT is initialized to H'00.

R64CNT is not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	—	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz
Initial value:	0	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R

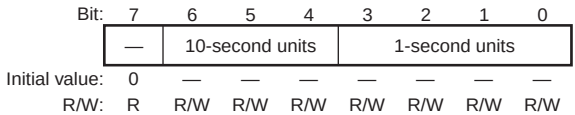
Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. Writing has no effect.
6	1 Hz	Undefined	R	Indicate the state of the divider circuit between 64 Hz and 1 Hz.
5	2 Hz	Undefined	R	
4	4 Hz	Undefined	R	
3	8 Hz	Undefined	R	
2	16 Hz	Undefined	R	
1	32 Hz	Undefined	R	
0	64 Hz	Undefined	R	

28.3.2 Second Counter (RSECCNT)

RSECCNT is used for setting/counting in the BCD-coded second section. The count operation is performed by a carry for each second of the 64-Hz counter.

The range of second can be set is 00 to 59 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag.

RSECCNT is not initialized by a power-on reset or manual reset, or in standby mode.



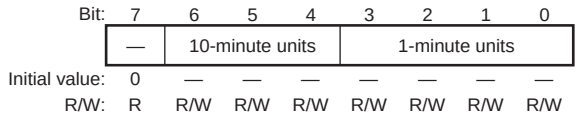
Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	—	Undefined	R/W	Counting Ten's Position of Seconds Counts on 0 to 5 for 60-seconds counting.
3 to 0	—	Undefined	R/W	Counting One's Position of Seconds Counts on 0 to 9 once per second. When a carry is generated, 1 is added to the ten's position.

28.3.3 Minute Counter (RMINCNT)

RMINCNT is used for setting/counting in the BCD-coded minute section. The count operation is performed by a carry for each minute of the second counter.

The range of minute can be set is 00 to 59 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag.

RMINCNT is not initialized by a power-on reset or manual reset, or in standby mode.



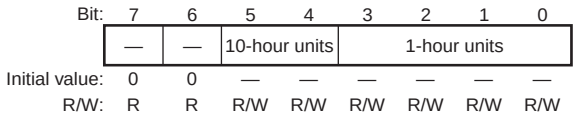
Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0.The write value should always be 0.
6 to 4	—	Undefined	R/W	Counting Ten's Position of Minutes Counts on 0 to 5 for 60-minutes counting.
3 to 0	—	Undefined	R/W	Counting One's Position of Minutes Counts on 0 to 9 once per second. When a carry is generated, 1 is added to the ten's position.

28.3.4 Hour Counter (RHRCNT)

RHRCNT is used for setting/counting in the BCD-coded hour section. The count operation is performed by a carry for each 1 hour of the minute counter.

The range of hour can be set is 00 to 23 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag..

RHRCNT is not initialized by a power-on reset or manual reset, or in standby mode.



Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. Though writing has no effect, the write value should always be 0.
5, 4	—	Undefined	R/W	Counting Ten's Position of Hours Counts on 0 to 2 for ten's position of hours.
3 to 0	—	Undefined	R/W	Counting One's Position of Hours Counts on 0 to 9 once per hour. When a carry is generated, 1 is added to the ten's position.

28.3.5 Day of Week Counter (RWKCNT)

RWKCNT is used for setting/counting day of week section. The count operation is performed by a carry for each day of the date counter.

The range for day of the week can be set is 0 to 6 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag..

RWKCNT is not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	Day-of-week code		
Initial value:	0	0	0	0	0	—	—	—
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. Though writing has n effect, the write value should always be 0.
2 to 0	—	Undefined	R/W	Day-of-Week Counting Day-of-week is indicated with a binary code. 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

28.3.6 Date Counter (RDAYCNT)

RDAYCNT is used for setting/counting in the BCD-coded date section. The count operation is performed by a carry for each day of the hour counter.

The range of date, which can be set, is 01 to 31 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag..

RDAYCNT is not initialized by a power-on reset or manual reset, or in standby mode.

The range of date changes with each month and in leap years. Please confirm the correct setting. Leap years are recognized by dividing the year counter values by 400, 100, and 4 and obtaining a fractional result of 0. The year counter value of 0000 is included in the leap year.

Bit:	7	6	5	4	3	2	1	0
	—	—	10-day units		1-day units			
Initial value:	0	0	—	—	—	—	—	—
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	—	Undefined	R/W	Counting Ten's Position of Dates
3 to 0	—	Undefined	R/W	Counting One's Position of Dates Counts on 0 to 9 once per date. When a carry is generated, 1 is added to the ten's position.

28.3.7 Month Counter (RMONCNT)

RMONCNT is used for setting/counting in the BCD-coded month section. The count operation is performed by a carry for each month of the date counter.

The range of month can be set is 01 to 12 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag..

RMONCNT is not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	10-month unit	1-month units			
Initial value:	0	0	0	—	—	—	—	—
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. Though writing has no effect, the write value should always be 0.
4	—	Undefined	R/W	Counting Ten's Position of Months
3 to 0	—	Undefined	R/W	Counting One's Position of Months Counts on 0 to 9 once per month. When a carry is generated, 1 is added to the ten's position.

28.3.8 Year Counter (RYRCNT)

RYRCNT is used for setting/counting in the BCD-coded year section. The count operation is performed by a carry for each year of the month counter.

The range for year, which can be set, is 0000 to 9999 (decimal). Errant operation will result if any other value is set. Carry out write processing after stopping the count operation with the START bit in RCR2, or by using a carry flag.

RYRCNT is not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1000-year units				100-year units				10-year units				1-year units			
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

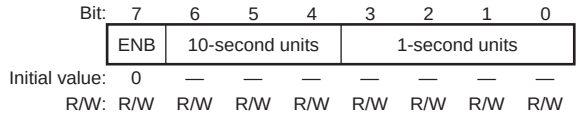
Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	Undefined	R/W	Counting Thousand's Position of Years
11 to 8	—	Undefined	R/W	Counting Hundred's Position of Years
7 to 4	—	Undefined	R/W	Counting Ten's Position of Years
3 to 0	—	Undefined	R/W	Counting One's Position of Years

28.3.9 Second Alarm Register (RSECAR)

RSECAR is an alarm register corresponding to the BCD-coded second counter RSECCNT of the RTC. When the ENB bit is set to 1, a comparison with the RSECCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The range of second alarm, which can be set, is 00 to 59 (decimal) plus ENB bit. Errant operation will result if any other value is set.

The ENB bit in RSECAR is initialized to 0 by a power-on reset. The remaining RSECAR fields are not initialized by a power-on reset or manual reset, or in standby mode.



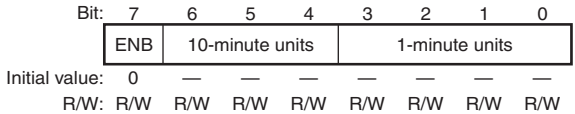
Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, a comparison with the RSECCNT value is performed.
6 to 4	—	Undefined	R/W	Ten's position of seconds setting value
3 to 0	—	Undefined	R/W	One's position of seconds setting value

28.3.10 Minute Alarm Register (RMINAR)

RMINAR is an alarm register corresponding to the minute counter RMINCNT. When the ENB bit is set to 1, a comparison with the RMINCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The range of minute alarm, which can be set, is 00 to 59 (decimal) plus ENB bit. Errant operation will result if any other value is set.

The ENB bit in RMINAR is initialized by a power-on reset. The remaining RMINAR fields are not initialized by a power-on reset or manual reset, or in standby mode.



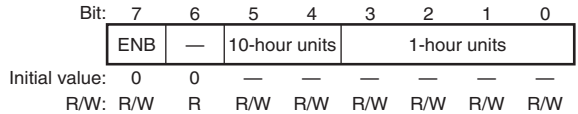
Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, a comparison with the RMINCNT value is performed.
6 to 4	—	Undefined	R/W	Ten's position of minutes setting value
3 to 0	—	Undefined	R/W	One's position of minutes setting value

28.3.11 Hour Alarm Register (RHRAR)

RHRAR is an alarm register corresponding to the BCD-coded hour counter RHRCNT of the RTC. When the ENB bit is set to 1, a comparison with the RHRCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The range of hour alarm, which can be set, is 00 to 23 (decimal) plus ENB bit. Errant operation will result if any other value is set.

The ENB bit in RHRAR is initialized by a power-on reset. The remaining RHRAR fields are not initialized by a power-on reset or manual reset, or in standby mode.



Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, a comparison with the RHRCNT value is performed.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	—	Undefined	R/W	Ten's position of hours setting value
3 to 0	—	Undefined	R/W	One's position of hours setting value

28.3.12 Day of Week Alarm Register (RWKAR)

RWKAR is an alarm register corresponding to the BCD-coded day of week counter RWKCNT. When the ENB bit is set to 1, a comparison with the RWKCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The range of day of the week alarm, which can be set, is 0 to 6 (decimal) plus ENB bit. Errant operation will result if any other value is set.

The ENB bit in RWKAR is initialized by a power-on reset. The remaining RWKAR fields are not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	ENB	—	—	—	—	Day-of-week code		
Initial value:	0	0	0	0	0	—	—	—
R/W:	R/W	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, a comparison with the RWKCNT value is performed.
6 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	—	Undefined	R/W	Day of week setting value

Code	0	1	2	3	4	5	6
Day	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday

28.3.13 Date Alarm Register (RDAYAR)

RDAYAR is an alarm register corresponding to the BCD-coded date counter RDAYCNT. When the ENB bit is set to 1, a comparison with the RDAYCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The range of date alarm, which can be set, is 01 to 31 (decimal) plus ENB bit. Errant operation will result if any other value is set.

The ENB bit in RDAYAR is initialized by a power-on reset. The remaining RDAYAR fields are not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	ENB	—	10-day units		1-day units			
Initial value:	0	0	—	—	—	—	—	—
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, a comparison with the RDAYCNT value is performed.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	—	Undefined	R/W	Ten's position of dates setting value
3 to 0	—	Undefined	R/W	One's position of dates setting value

28.3.14 Month Alarm Register (RMONAR)

RMONAR is an alarm register corresponding to the month counter RMONCNT. When the ENB bit is set to 1, a comparison with the RMONCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The range of month alarm, which can be set, is 01 to 12 (decimal) plus ENB bit. Errant operation will result if any other value is set.

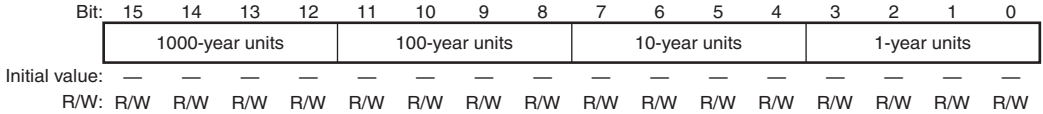
The ENB bit in RMONAR is initialized by a power-on reset. The remaining RMONAR fields are not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	ENB	—	—	10-month unit	1-month units			
Initial value:	0	0	0	—	—	—	—	—
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, a comparison with the RMONCNT value is performed.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	—	Undefined	R/W	Ten's position of months setting value
3 to 0	—	Undefined	R/W	One's position of months setting value

28.3.15 Year Alarm Register (RYRAR)

RYRAR is an alarm register corresponding to the BCD-coded year counter RYRCNT. The range of year alarm, which can be set, is 0000 to 9999 (decimal). Errant operation will result if any other value is set.

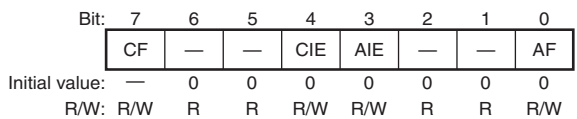


Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	Undefined	R/W	Thousand's position of years setting value
11 to 8	—	Undefined	R/W	Hundred's position of years setting value
7 to 4	—	Undefined	R/W	Ten's position of years setting value
3 to 0	—	Undefined	R/W	One's position of years setting value

28.3.16 RTC Control Register 1 (RCR1)

RCR1 is a register that affects carry flags and alarm flags. It also selects whether to generate interrupts for each flag.

RCR1 is initialized to H'00 by a power-on reset or a manual reset, and all bits are initialized to 0 except for the CF flag, which is undefined until the divider circuit is reset (set RESET and ADJ in RCR2 to 1). When using the CF flag, it must be initialized beforehand. This register is not initialized in standby mode.



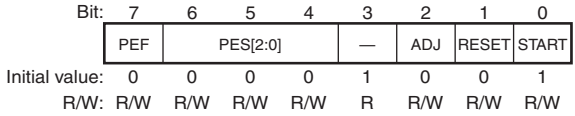
Bit	Bit Name	Initial Value	R/W	Description
7	CF	Undefined	R/W	Carry Flag Status flag that indicates that a carry has occurred. CF is set to 1 when a count-up to 64-Hz occurs at the second counter carry or 64-Hz counter read. A count register value read at this time cannot be guaranteed; another read is required. 0: No carry of 64-Hz counter by second counter or 64-Hz counter [Clearing condition] When 0 is written to CF 1: Carry of 64-Hz counter by second counter or 64 Hz counter [Setting condition] When the second counter or 64-Hz counter is read during a carry occurrence by the 64-Hz counter, or 1 is written to CF.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CIE	0	R/W	Carry Interrupt Enable Flag When the carry flag (CF) is set to 1, the CIE bit enables interrupts. 0: A carry interrupt is not generated when the CF flag is set to 1 1: A carry interrupt is generated when the CF flag is set to 1
3	AIE	0	R/W	Alarm Interrupt Enable Flag When the alarm flag (AF) is set to 1, the AIE bit allows interrupts. 0: An alarm interrupt is not generated when the AF flag is set to 1 1: An alarm interrupt is generated when the AF flag is set to 1

Bit	Bit Name	Initial Value	R/W	Description
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	AF	0	R/W	Alarm Flag The AF flag is set when the alarm time, which is set by an alarm register (ENB bit in RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, or RYRAR is set to 1), and counter match. 0: Alarm register and counter not match [Clearing condition] When 0 is written to AF. 1: Alarm register and counter match* [Setting condition] When alarm register (only a register with ENB bit set to 1) and counter match Note: * Writing 1 holds previous value.

28.3.17 RTC Control Register 2 (RCR2)

RCR2 is a register for periodic interrupt control, 30-second adjustment ADJ, divider circuit RESET, and RTC count control.

RCR2 is initialized to H'09 by a power-on reset. It is initialized except for START by a manual reset. It is not initialized in standby mode, and retains its contents.



Bit:	7	6	5	4	3	2	1	0
	PEF	PES[2:0]			—	ADJ	RESET	START
Initial value:	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PEF	0	R/W	Periodic Interrupt Flag Indicates interrupt generation with the period designated by the PES[2:0] bits. When set to 1, PEF generates periodic interrupts. 0: Interrupts not generated with the period designated by bits PES[2:0]. [Clearing condition] When 0 is written to PEF 1: Interrupts generated with the period designated by bits PES[2:0]. [Setting condition] When an interrupt is generated with the period designated by bits PES[2:0] or when 1 is written to the PEF flag
6 to 4	PES[2:0]	000	R/W	Interrupt Enable Flags These bits specify the periodic interrupt. 000: No periodic interrupts generated 001: Periodic interrupt generated every 1/256 second 010: Periodic interrupt generated every 1/64 second 011: Periodic interrupt generated every 1/16 second 100: Periodic interrupt generated every 1/4 second 101: Periodic interrupt generated every 1/2 second 110: Periodic interrupt generated every 1 second 111: Periodic interrupt generated every 2 seconds
3	—	1	R	Reserved These bits are always read as 1. The write value should always be 1.

Bit	Bit Name	Initial Value	R/W	Description
2	ADJ	0	R/W	30-Second Adjustment When 1 is written to the ADJ bit, times of 29 seconds or less will be rounded to 00 seconds and 30 seconds or more to 1 minute. The divider circuit (RTC prescaler and R64CNT) will be simultaneously reset. It is not necessary to write 0 to this bit because this bit changes to 0 automatically. This bit always reads 0. 0: Runs normally. 1: 30-second adjustment.
1	RESET	0	R/W	Reset When 1 is written, initializes the divider circuit. The divider circuit (RTC prescaler and R64CNT) will be reset when 1 is written to RESET bit. It is not necessary to write 0 to this bit because this bit changes to 0 automatically. This bit always reads 0. 0: Runs normally. 1: Divider circuit is reset.
0	START	1	R/W	Start Bit Halts and restarts the counter (clock). 0: Second/minute/hour/day/week/month/year counter halts. 1: Second/minute/hour/day/week/month/year counter runs normally.

28.3.18 RTC Control Register 3 (RCR3)

When the ENB bit is set to 1, RCR3 performs a comparison with the RYRCNT. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The ENB bit in RYRAR is initialized by a power-on reset. Remaining fields of RCR3 are not initialized by a power-on reset or manual reset, or in standby mode.

Bit:	7	6	5	4	3	2	1	0
	ENB	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	0	R/W	When this bit is set to 1, comparison of the year alarm register (RYRAR) and the year counter (RYRCNT) is performed.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

28.4 Operation

RTC usage is shown below.

28.4.1 Initial Settings of Registers after Power-On

All the registers should be set after the power is turned on.

28.4.2 Setting Time

Figure 28.2 shows how to set the time when the clock is stopped.

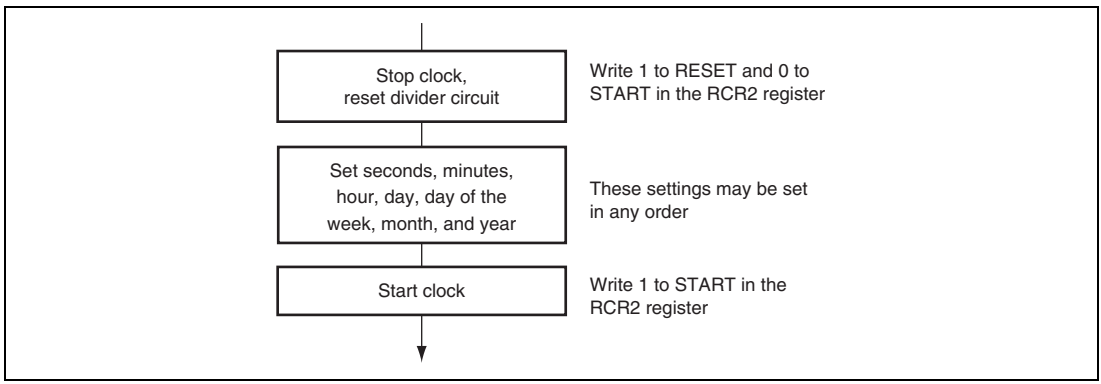


Figure 28.2 Setting Time

28.4.3 Reading Time

Figure 28.3 shows how to read the time.

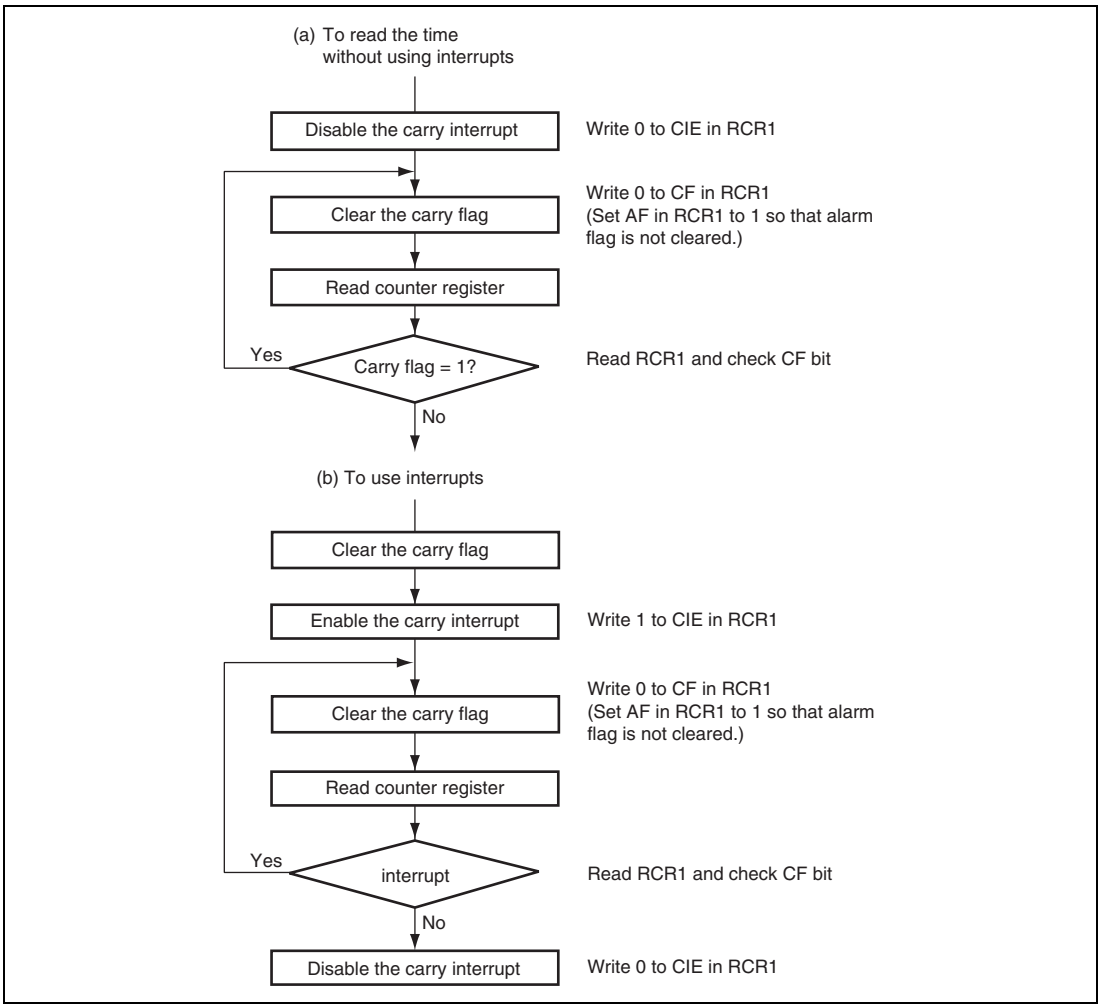


Figure 28.3 Reading Time

If a carry occurs while reading the time, the correct time will not be obtained, so it must be read again. Part (a) in Figure 28.3 shows the method of reading the time without using interrupts; part (b) in Figure 28.3 shows the method using carry interrupts. To keep programming simple, method (a) should normally be used.

28.4.4 Alarm Function

Figure 28.4 shows how to use the alarm function.

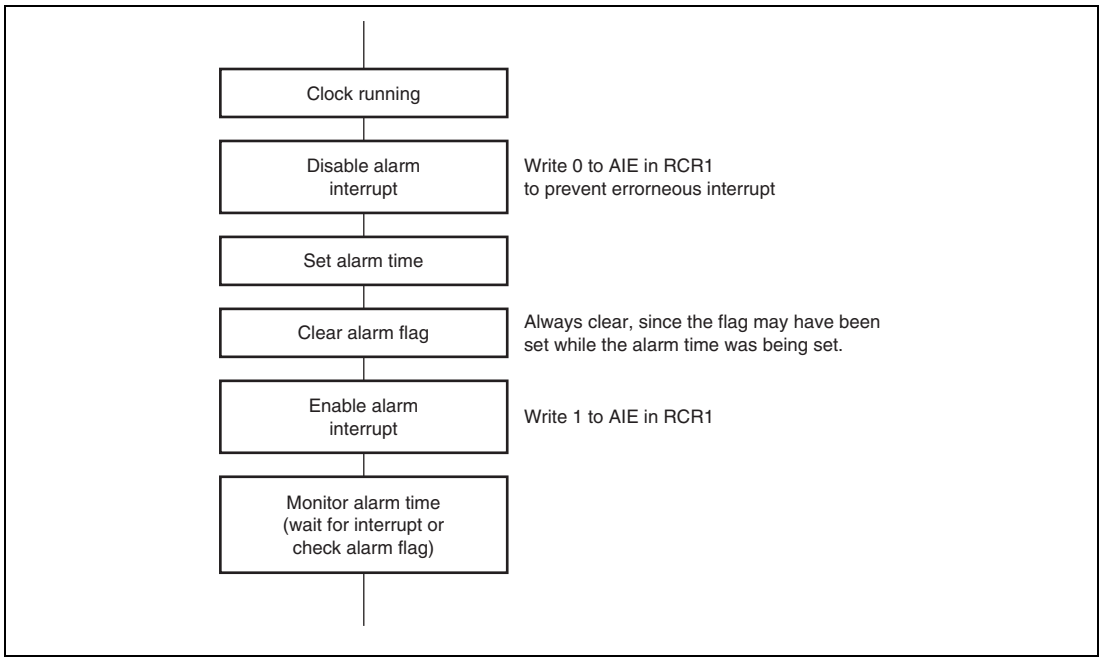


Figure 28.4 Using Alarm Function

Alarms can be generated using seconds, minutes, hours, day of the week, date, month, year, or any combination of these. Set the ENB bit in the register on which the alarm is placed to 1, and then set the alarm time in the lower bits. Clear the ENB bit in the register on which the alarm is not placed to 0.

When the clock and alarm times match, 1 is set in the AF bit in RCR1. Alarm detection can be checked by reading this bit, but normally it is done by interrupt. If 1 is set in the AIE bit in RCR1, an interrupt is generated when an alarm occurs.

The alarm flag is set when the clock and alarm times match. However, the alarm flag can be cleared by writing 0.

28.5 Usage Notes

28.5.1 Register Writing during RTC Count

The following RTC registers cannot be written to during an RTC count (while bit 0 = 1 in RCR2).

RSECCNT, RMINCNT, RHRCNT, RDAYCNT, RWKCNT, RMONCNT, RYRCNT

The RTC count must be stopped before writing to any of the above registers.

28.5.2 Use of Realtime Clock (RTC) Periodic Interrupts

The method of using the periodic interrupt function is shown in Figure 28.5.

A periodic interrupt can be generated periodically at the interval set by the flags PES[2:0] in RCR2. When the time set by the PES[2:0] has elapsed, the PEF is set to 1.

The PEF is cleared to 0 upon periodic interrupt generation or when the flags PES[2:0] are set. Periodic interrupt generation can be confirmed by reading this bit, but normally the interrupt function is used.

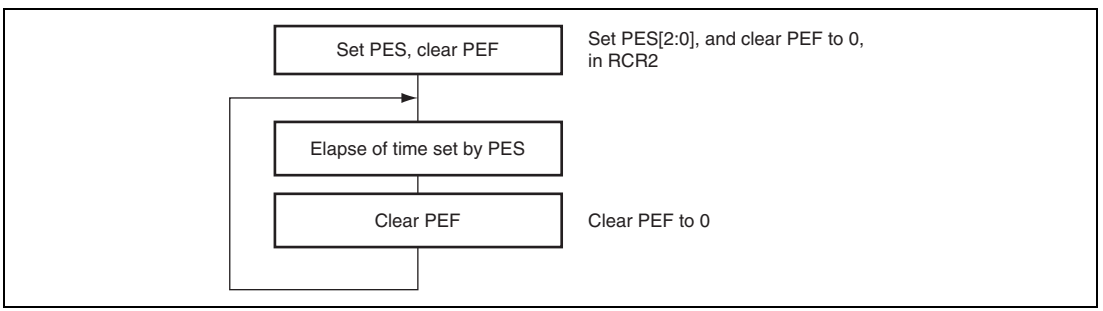


Figure 28.5 Using Periodic Interrupt Function

28.5.3 Transition to Standby Mode after Setting Register

When a transition to standby mode is made after registers in the RTC are set, sometimes counting is not performed correctly. In case the registers are set, be sure to make a transition to standby mode after waiting for two RTC clocks or more.

28.5.4 Usage of 30-Second Adjustment

Follow the procedure below to use the 30-second adjustment.

(1) Stop the clock operation. (Clear the START bit in the RCR2 register to 0.)

(2) After reading each counter (year, month, day, date, hour, minute), write back the read values to each counter.

(3) Set the 30-second adjustment. (Set the ADJ bit in the RCR2 register to 1.)

(4) Start the clock operation. (Set the START bit in the RCR2 register to 1.)

Section 29 IrDA Interface (IrDA)

The IrDA interface (IrDA) performs infrared data communication conforming to IrDA standard 1.2a through an external infrared transceiver unit connected to this LSI.

The IrDA includes a UART block to control data transmission and reception as well as an infrared transmit and receive (light-emit and light-receive) pulse modulator/demodulator block and a CRC engine block in front of the UART. The UART block controls serial data transmission and reception in the asynchronous mode. The infrared transmit and receive pulse modulator/demodulator block controls communication pulses and checks pulses received through infrared baseband modulation/demodulation conforming to IrDA standard 1.2a. The CRC engine block reads 8-bit input data and outputs a 16-bit CRC calculation result.

29.1 Features

The IrDA has the following UART features.

- Asynchronous serial communication
  - Data length: Eight bits
  - Stop bit: One bit
  - Parity bit: None
- Reception error detection: Overrun error and framing error
- Baud rate error correction: 16 decimal fractions can be selected.
- Baud rate count: Up to 65536 can be specified.

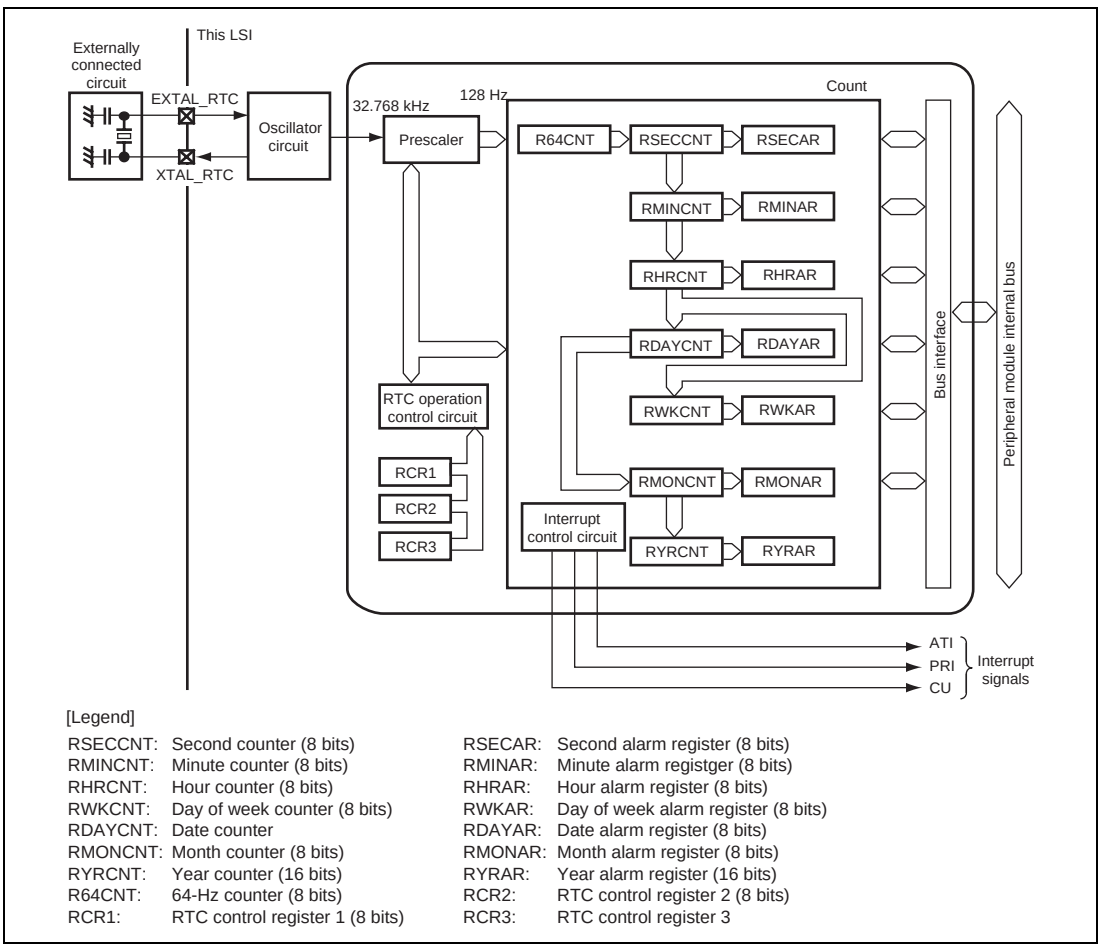
The IrDA has the following infrared transmit and receive pulse modulator/demodulator features.

- Infrared transmit (light-emit) pulse width: 1-bit width × 3/16 or 1.63 μs can be selected.
- Pulse width check: An out-of-standard pulse (insufficient or excess width) can be detected.
- 1.8432-MHz clock generator
  - Up to 16 can be specified for the integer part of the baud rate count.
  - The fractional part can be selected from 16 values.

The IrDA has the following CRC calculation features.

- Generator polynomial: $X^{16} + X^{12} + X^5 + 1$
- Data input
  - — Input in bytes
  - — CRC is calculated in 8-bit units starting from the lower bits.
- CRC output: 16-bit CRC is output.
- Maximum data length: 4096 bytes

Figure 29.1 shows a block diagram of the IrDA.



[Legend]

RSECCNT:	Second counter (8 bits)	RSECAR:	Second alarm register (8 bits)
RMINCNT:	Minute counter (8 bits)	RMINAR:	Minute alarm registger (8 bits)
RHRCNT:	Hour counter (8 bits)	RHRAR:	Hour alarm register (8 bits)
RWKCNT:	Day of week counter (8 bits)	RWKAR:	Day of week alarm register (8 bits)
RDAYCNT:	Date counter	RDAYAR:	Date alarm register (8 bits)
RMONCNT:	Month counter (8 bits)	RMONAR:	Month alarm register (8 bits)
RYRCNT:	Year counter (16 bits)	RYRAR:	Year alarm register (16 bits)
R64CNT:	64-Hz counter (8 bits)	RCR2:	RTC control register 2 (8 bits)
RCR1:	RTC control register 1 (8 bits)	RCR3:	RTC control register 3

Figure 29.1 Block Diagram of IrDA

29.2 Input/Output Pins

Table 29.1 shows the IrDA pin configuration.

Table 29.1 Pin Configuration

Pin Name	Abbreviation	I/O	Function
$\overline{\text{IrDA\_IN}}$	$\overline{\text{IRIN}}$	Input	Infrared receive (light-receive) pulse input (negative logic)
IrDA_OUT	IROUT	Output	Infrared transmit (light-emit) pulse output (positive logic)

29.3 Register Descriptions

Table 29.2 shows the IrDA register configuration. Table 29.3 shows the register states in each operating mode.

Table 29.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
DMA receive interrupt source clear register	IRIF_RINTCLR	W	H'A45D 0016	16/8
DMA transmit interrupt source clear register	IRIF_TINTCLR	W	H'A45D 0018	16/8
IrDA-SIR10 control register	IRIF_SIR0	R/W	H'A45D 0020	16/8
IrDA-SIR10 baud rate error correction register	IRIF_SIR1	R/W	H'A45D 0022	16/8
IrDA-SIR10 baud rate count set register	IRIF_SIR2	R/W	H'A45D 0024	16/8
IrDA-SIR10 status register	IRIF_SIR3	R	H'A45D 0026	16/8
Hardware frame processing set register	IRIF_SIR_FRM	R/W	H'A45D 0028	16/8
EOF value register	IRIF_SIR_EOF	R/W	H'A45D 002A	16/8
Flag clear register	IRIF_SIR_FLG	W	H'A45D 002C	16/8
UART status register 2	IRIF_UART_STS2	R/W	H'A45D 002E	16/8
UART control register	IRIF_UART0	R/W	H'A45D 0030	16/8
UART status register	IRIF_UART1	R	H'A45D 0032	16/8
UART mode register	IRIF_UART2	R/W	H'A45D 0034	16/8
UART transmit data register	IRIF_UART3	W	H'A45D 0036	16/8
UART receive data register	IRIF_UART4	R	H'A45D 0038	16/8
UART interrupt mask register	IRIF_UART5	R/W	H'A45D 003A	16/8
UART baud rate error correction register	IRIF_UART6	R/W	H'A45D 003C	16/8
UART baud rate count set register	IRIF_UART7	R/W	H'A45D 003E	16/8
CRC engine control register	IRIF_CRC0	R/W	H'A45D 0040	16/8
CRC engine input data register	IRIF_CRC1	W	H'A45D 0042	16/8
CRC engine calculation register	IRIF_CRC2	W	H'A45D 0044	16/8
CRC engine output data register 1	IRIF_CRC3	R	H'A45D 0046	16/8
CRC engine output data register 2	IRIF_CRC4	R	H'A45D 0048	16/8

Table 29.3 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
IRIF_RINTCLR	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_TINTCLR	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR2	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR3	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR_FRM	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR_EOF	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_SIR_FLG	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART_STS2	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART0	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART1	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART2	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART3	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART4	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART5	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART6	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_UART7	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_CRC0	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_CRC1	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_CRC2	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_CRC3	Initialized	Initialized	Retained	Retained	Initialized	Retained
IRIF_CRC4	Initialized	Initialized	Retained	Retained	Initialized	Retained

29.3.1 DMA Receive Interrupt Source Clear Register (IRIF_RINTCLR)

IRIF_RINTCLR is a register that clears a request for DMA transfer of received data.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDMAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	RDMAC [15:0]	H'0000	W	Clear of DMA Transfer Request for Received Data To clear a request, write any word data to this register.

29.3.2 DMA Transmit Interrupt Source Clear Register (IRIF_TINTCLR)

IRIF_TINTCLR is a register that clears a request for DMA transfer of transmit data.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDMAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TDMAC [15:0]	H'0000	W	Clear of DMA Transfer Request for Transmit Data To clear a request, write any word data to this register.

29.3.3 IrDA-SIR10 Control Register (IRIF_SIR0)

IRIF_SIR0 is a register that controls modulation/demodulation of infrared transmit and receive pulses.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IR TPW	IR ERRC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	IRTPW	0	R/W	Infrared Transmit (Light-Emit) Pulse Width Select Selects the pulse width for infrared transmission. 0: Outputs three cycles of the clock input through MSFCLK_IN. 1: Outputs three cycles of the 1.8432-MHz clock specified by IRIF_SIR1 and IRIF_SIR2.
0	IRERRC	0	R/W	Clear of Error Flag for Infrared Receive (Light-Receive) Pulse Width Clears the flag for an error in the pulse width of infrared reception. 0: Does not clear the error flag. 1: Clears the error flag. Note: This bit is automatically cleared to 0 immediately after set to 1: there is no need to write 0 to this bit.

29.3.4 IrDA-SIR10 Baud Rate Error Correction Register (IRIF_SIR1)

IRIF_SIR1 is a register that specifies error correction of the baud rate (the fractional part of the baud rate count) used in the infrared transmit and receive pulse modulator/demodulator block. This value is used in combination with the IRIF_SIR2 value to generate a 1.8432-MHz clock.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	IRBCA[3:0]				—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	IRBCA[3:0]	0000	R/W	Infrared Modulator/Demodulator Baud Rate Error Correction Set These bits specify error correction of the baud rate (the fractional part of the baud rate count) used in the infrared transmit and receive pulse modulator/demodulator block. The value shown to the right of each setting below is used as the fractional part for the baud rate count specified in IRIF_SIR2. Select an appropriate fractional value according to the user system operating specifications.

0000: 0.0000	1000: 0.5000
0001: 0.0625	1001: 0.5625
0010: 0.1250	1010: 0.6250
0011: 0.1875	1011: 0.6875
0100: 0.2500	1100: 0.7500
0101: 0.3125	1101: 0.8125
0110: 0.3750	1110: 0.8750
0111: 0.4375	1111: 0.9375

Note: These bits must not be modified during transmission or reception. If this is attempted, correct operation cannot be guaranteed.

3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

29.3.5 IrDA-SIR10 Baud Rate Count Set Register (IRIF_SIR2)

IRIF_SIR2 is a register that specifies the integer part of the baud rate count used in the infrared transmit and receive pulse modulator/demodulator block. This value is used in combination with the error correction value specified in IRIF_SIR1 to generate a 1.8432-MHz clock.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	IRBC[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	IRBC[3:0]	0000	R/W	Infrared Modulator/Demodulator Baud Rate Count Set These bits specify the integer part of the clock generation dividing count used in the infrared transmit and receive pulse modulator/demodulator block. Note: These bits must not be modified during transmission or reception. If this is attempted, correct operation cannot be guaranteed.

29.3.6 IrDA-SIR10 Status Register (IRIF_SIR3)

IRIF_SIR3 is a register that indicates an error in the width of the received infrared pulse during infrared pulse modulation/demodulation.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IRERR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	IRERR	0	R	Error Flag for Width of Received Infrared Pulse Indicates an error in the width of the received pulse during infrared pulse modulation/demodulation. 0: No error has occurred. 1: An error has occurred.

29.3.7 Hardware Frame Processing Set Register (IRIF_SIR_FRM)

IRIF_SIR_FRM is a register that specifies the processing of received data frames.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	EOFD	FRER	—	—	—	—	—	—	—	FRP
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	EOFD	1	R	EOF Detection Flag 0: EOF has been detected. 1: EOF has not been detected.
8	FRER	0	R	Frame Error Bit 0: No frame error has been detected. 1: A frame error has been detected.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FRP	0	R/W	Frame Processing Set 0: Disables EOF detection. 1: Enables EOF detection.

29.3.8 EOF Value Register (IRIF_SIR_EOF)

IRIF_SIR_EOF is a register that specifies the EOF value.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	EOF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	EOF[7:0]	H'C1	R/W	EOF Set These bits specify the EOF value to be detected.

29.3.9 Flag Clear Register (IRIF_SIR_FLG)

IRIF_SIR_FLG is a register that clears the frame error flag and EOF flag. Writing any data to the upper or lower eight bits of this register clears the corresponding flag.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FRERC[7:0]								EOFC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	FRERC[7:0]	H'00	W	Frame Error Flag Clear Writing any byte data to these bits (the upper eight bits of the register) clears the frame error flag.
7 to 0	EOFC[7:0]	H'00	W	EOF Error Flag Clear Writing any byte data to these bits (the lower eight bits of the register) clears the EOF error flag.

29.3.10 UART Status Register 2 (IRIF_UART_STS2)

IRIF_UART_STS2 is a register that indicates the operating status during data reception.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	IRSME	IROVE	IRFRE	IRPRE	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	IRSME	0	R/W	Receive Sum Error Flag 0: No receive sum error has occurred. 1: A receive sum error has occurred.
5	IROVE	0	R/W	Receive Overrun Error Flag 0: No receive overrun error has occurred. 1: A receive overrun error has occurred.
4	IRFRE	0	R/W	Receive Framing Error Flag 0: No receive framing error has occurred. 1: A receive framing error has occurred.
3	IRPRE	0	R/W	Receive Parity Error Flag 0: No receive parity error has occurred. 1: A receive parity error has occurred.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: Writing to this register clears all error flags.

29.3.11 UART Control Register (IRIF_UART0)

IRIF_UART0 is a register that controls data transmission and reception.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TBEC	RIE	TIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TBEC	0	W	Transmit Data Clear Clears the empty flag for the UART transmit buffer. Although the flag is cleared by writing 1 to this bit, the transmit data register is not cleared. This bit is always read as 0 even after 1 is written to it. 0: Does not clear the flag. 1: Clears the flag.
1	RIE	0	R/W	Receive Enable Starts or stops UART reception. If 0 is written to this bit during reception, the operation stops after one unit of data is received. 0: Stops reception. 1: Starts reception.
0	TIE	0	R/W	Transmit Enable Starts or stops UART transmission. If 0 is written to this bit during transmission, the operation stops after one unit of data is transmitted. 0. Stops transmission. 1: Starts transmission.

29.3.12 UART Status Register (IRIF_UART1)

IRIF_UART1 is a register that includes flags for indicating the UART operation status.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	URSME	UROVE	URFRE	URPRE	RBF	TSBE	TBE
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
8, 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	URSME	0	R	Receive Sum Error Flag This bit is set to 1 when any of the UART error flags (receive parity error flag, receive framing error flag, or receive overrun error flag) is set to 1, and is cleared to 0 when none of the UART error flags is set to 1. The error flag is cleared when the receive data register is read by the system. If the next data is received before the receive data register is read, the error flag is updated according to the latest received data status (previous received data error flags are overwritten). 0: No error has occurred. 1: An error has occurred.
5	UROVE	0	R	Receive Overrun Error Flag This bit is set to 1 when the next received data is stored in the UART receive data register before the previous received data is read from the register by the system, and is cleared to 0 when the receive data register is read by the system. (The previous received data is always overwritten with the latest received data.) 0: No error has occurred. 1: An error has occurred.

Bit	Bit Name	Initial Value	R/W	Description
4	URFRE	0	R	Receive Framing Error Flag This bit is cleared to 0 when the stop bits added behind the UART received data matches the stop bit length specified in the UART mode register, and is set to 1 when they do not match. The error flag is cleared when the receive data register is read by the system. If the next data is received before the receive data register is read, the error flag is updated according to the latest received data status (previous received data error flags are overwritten). 0: No error has occurred. 1: An error has occurred.
3	URPRE	0	R	Receive Parity Error Flag This bit is cleared to 0 when the parity in the UART received data matches the parity specified in the UART mode register, and is set to 1 when they do not match while parity check is enabled. The error flag is cleared when the receive data register is read by the system. If the next data is received before the receive data register is read, the error flag is updated according to the latest received data status (previous received data error flags are overwritten). 0: No error has occurred. 1: An error has occurred.
2	RBF	0	R	Receive Buffer Full Flag This bit is set to 1 when received data is stored in the UART receive data register (even if any of the receive parity error, receive framing error, and receive overrun error has occurred), and is cleared to 0 when the receive data register is read by the system. 0: No received data in the buffer. 1: Received data is in the buffer.
1	TSBE	1	R	Transmit Shift Buffer Empty Flag This bit is set to 1 when UART transmission is completed (the UART transmit shift buffer becomes empty), and is 0 during UART transmission 0: During transmission 1: Transmission completed

Bit	Bit Name	Initial Value	R/W	Description
0	TBE	1	R	Transmit Buffer Empty flag This bit is set to 1 when data is sent from the UART transmit data register to the transmit shift buffer (the transmit data register becomes empty) or when 1 is written to the transmit data clear bit in the UART control register, and is cleared to 0 when transmit data is written to the transmit data register. 0. Transmit data is in the register. 1: No transmit data is in the register.

29.3.13 UART Mode Register (IRIF_UART2)

IRIF_UART2 is a register that specifies the data format and transfer mode for serial data communication. The initial value must not be modified.

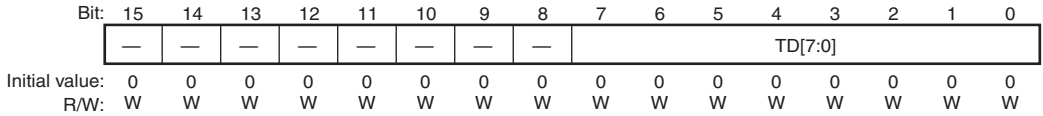
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BCT	CHR	STOP	PE	O/$\overline{E}$	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	BCT	0	R/W	Break Character Transmit Bit Specifies transmission of a break character by the UART. Setting this bit to 1 transmits a break character (UART transmit signal = L) and clearing to 0 specifies normal operation. 0: Normal operation. 1: Transmits a break character.
6	CHR	0	R/W	Character Length Select Selects the character length transmitted or received by the UART. 0: 8 bits 1: 7 bits

Bit	Bit Name	Initial Value	R/W	Description
5	STOP	0	R/W	Stop Bit Count Select Selects the number of stop bits added behind the data transmitted or received by the UART. Setting this bit to 1 selects two bits and clearing to 0 selects one bit. 0: 1 bit 1: 2 bits
4	PE	0	R/W	Parity Enable Enables or disables parity bit addition in transmission and parity bit checking in reception by the UART. 0: Disables parity addition and checking. 1: Enables parity addition and checking.
3	O/$\overline{E}$	0	R/W	Parity Mode Select Selects even or odd parity for data transmitted or received by the UART. 0: Even parity 1: Odd parity
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

29.3.14 UART Transmit Data Register (IRIF_UART3)

IRIF_UART3 is a register that stores transmit data.

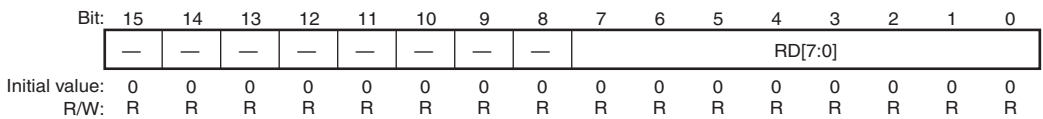


Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	W	Reserved The write value should always be 0.
7 to 0	TD[7:0]	H'00	W	UART Transmit Data The data to be transmitted should be specified here.

Note: Write to this register while the transmit buffer empty (TBE) flag in the UART status register (IRIF_UART1) is set to 1. If this register is written to while the flag is set to 0, undefined data may be transmitted depending on the timing.

29.3.15 UART Receive Data Register (IRIF_UART4)

IRIF_UART4 is a register that stores received data.



Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	RD[7:0]	H'00	R	UART Receive Data These bits store received data.

Note: Read this register while the receive buffer full (RBF) flag in the UART status register (IRIF_UART1) is set to 1. If this register is read while the flag is set to 0, undefined data may be read depending on the timing.

29.3.16 UART Interrupt Mask Register (IRIF_UART5)

IRIF_UART5 is a register that enables or disables UART interrupts.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RSEIM	—	—	—	RBFIM	TSBEIM	TBEIM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	RSEIM	0	R/W	Receive Sum Error Flag Interrupt Mask Enables or disables an interrupt by the receive sum error flag. 0: Disables an interrupt. 1: Enables an interrupt.
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	RBFIM	0	R/W	Receive Buffer Full Flag Interrupt Mask Enables or disables an interrupt by the receive buffer full flag. 0: Disables an interrupt. 1: Enables an interrupt.
1	TSBEIM	0	R/W	Transmit Shift Buffer Empty Flag Interrupt Mask Enables or disables an interrupt by the transmit shift buffer empty flag. When the flag is set to 1, interrupt processing is started. 0: Disables an interrupt. 1: Enables an interrupt.
0	TBEIM	0	R/W	Transmit Buffer Empty Flag Interrupt Mask Enables or disables an interrupt by the transmit buffer empty flag. When the flag is set to 1, interrupt processing is started. 0: Disables an interrupt. 1: Enables an interrupt.

Note: The TSBEIM and TBEIM flags must not both be set to 1 (enabled) at the same time.

29.3.17 UART Baud Rate Error Correction Register (IRIF_UART6)

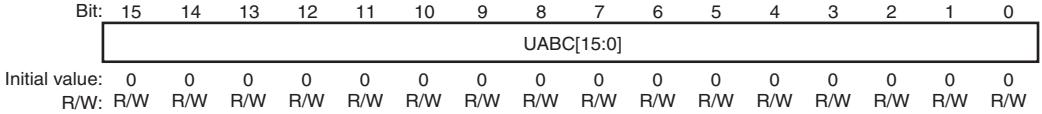
IRIF_UART6 is a register that specifies error correction of the baud rate for data communication.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UABCA[3:0]				—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	UABCA[3:0]	0000	R/W	Baud Rate Error Correction Set These bits specify error correction of the baud rate (the fractional part of the baud rate count) used by the UART in combination with the value specified in the UART baud rate count set register. The value shown to the right of each setting below is used as the fractional part for the baud rate count. Select an appropriate fractional value according to the user system operating specifications. 0000: 0.0000 1000: 0.5000 0001: 0.0625 1001: 0.5625 0010: 0.1250 1010: 0.6250 0011: 0.1875 1011: 0.6875 0100: 0.2500 1100: 0.7500 0101: 0.3125 1101: 0.8125 0110: 0.3750 1110: 0.8750 0111: 0.4375 1111: 0.9375
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

29.3.18 UART Baud Rate Count Set Register (IRIF_UART7)

IRIF_UART7 is a register that specifies the baud rate count for data communication.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	UABC[15:0]	H'0000	R/W	Baud Rate Count Set These bits specify the integer part of the baud rate count used by the UART in combination with the value specified in the UART baud rate error correction register. For details of baud rate setting, refer to section 29.4.1 (4), Baud Rate Setting for Data Transmission and Reception.

29.3.19 CRC Engine Control Register (IRIF_CRC0)

IRIF_CRC0 is a register that activates the CRC engine and counts the number of input data items.

Bit	Bit Name	Initial Value	R/W	Description
15	CRC_RST	0	W	CRC Engine Reset Clears all registers related to CRC calculation. After reset, this bit automatically returns to 0 and there is no need to write 0 to this bit. 0: Normal CRC calculation 1: CRC engine reset
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CRC_CT [11:0]	H'000	R	CRC Engine Input Data Count The number of data items that have been input to the CRC engine can be read. After the data count reaches 4096, it will wrap around to 0.

29.3.20 CRC Engine Input Data Register (IRIF_CRC1)

IRIF_CRC1 is a register that stores data to be input to the CRC engine.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CRC_IN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	W	Reserved The write value should always be 0.
7 to 0	CRC_IN[7:0]	H'00	W	CRC Engine Input Data The data to be input to the CRC engine should be specified here. The specified data is shifted into the CRC calculation register, starting with the LSB (CRC_IN0) of the data into the MSB (CRC_REG15) side of the calculation register. For details, see Figure 29.7.

29.3.21 CRC Engine Calculation Register (IRIF_CRC2)

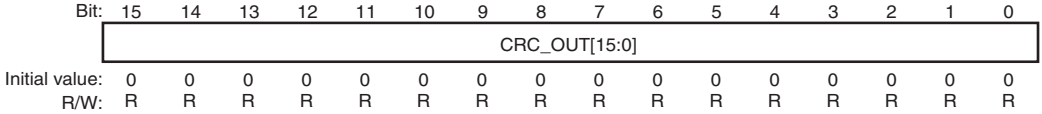
IRIF_CRC2 is a register used for CRC calculation. Generally, this register must be accessed only when the initial value for CRC calculation should be specified.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRC_REG[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CRC_REG [15:0]	H'0000	W	CRC Engine Calculation Data The initial value for CRC calculation should be specified here.

29.3.22 CRC Engine Output Data Register 1 (IRIF_CRC3)

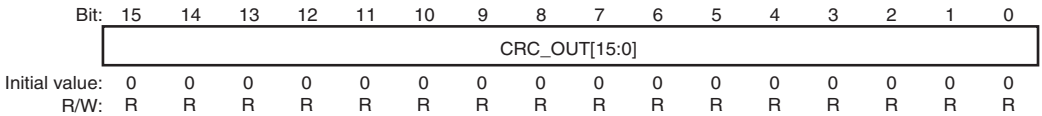
IRIF_CRC3 is a register that stores the CRC engine calculation result.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CRC_OUT [15:0]	H'0000	R	CRC Engine Output Data The result of calculation in the CRC engine can be read.

29.3.23 CRC Engine Output Data Register 2 (IRIF_CRC4)

IRIF_CRC4 is a register that stores the CRC engine calculation result. The bit order (LSB–MSB) of the IRIF_CRC3 value is inverted and stored in this register.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	CRC_OUT [15:0]	H'0000	R	CRC Engine Output Data The result of calculation in the CRC engine can be read. Bit 15 holds the LSB value and bit 0 holds the MSB value of the calculation result.

29.4 Operation

29.4.1 UART

The UART carries out serial communication in asynchronous mode.

(1) Data Format

Figure 29.2 shows the format of data that can be handled in the UART.

- ST bit (start bit)

  The ST bit indicates the beginning of data transmission or reception. A low-level signal of 1-bit duration is sent immediately before the data bits.
- Bits 0 to 7 (data bits)

  The data bits hold the transmit data specified in IRIF_UART3 or the received data to be stored in IRIF_UART4. Eight bits are used as the data bits for one character and the bits are transferred with the LSB first.
- SP bit (stop bit)

  The SP bit indicates the end of data transmission or reception. A high-level signal of 1-bit duration is sent immediately after the data bits.

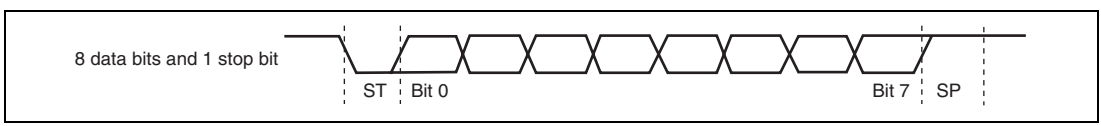


Figure 29.2 Data Transmission and Reception Format

(2) Data Transmission Timing

Figure 29.3 shows the data transmission timing controlled by the UART.

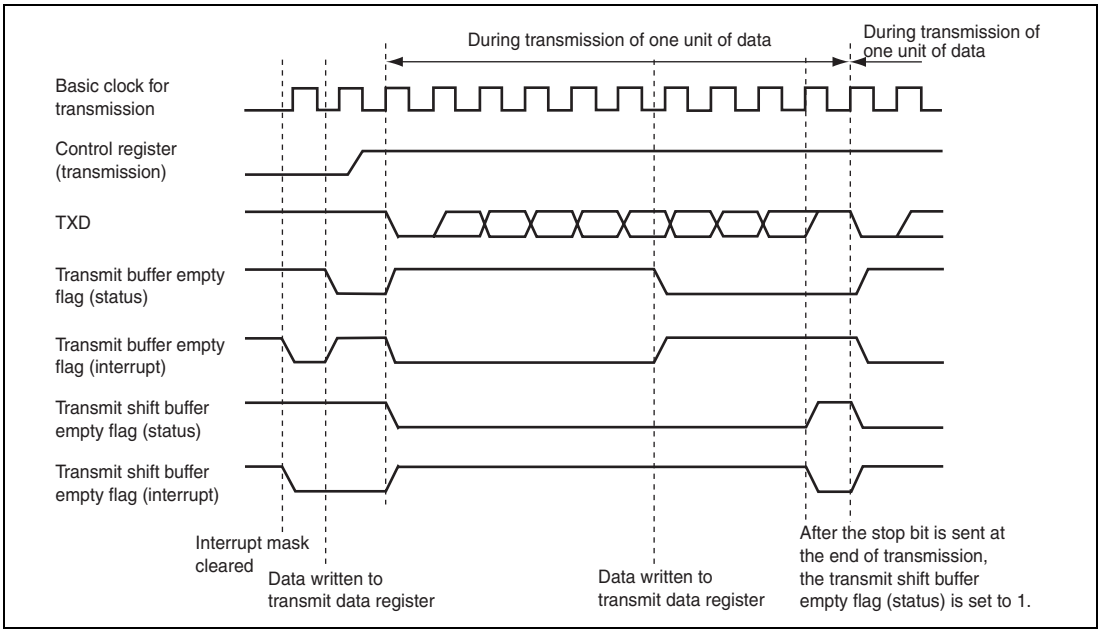
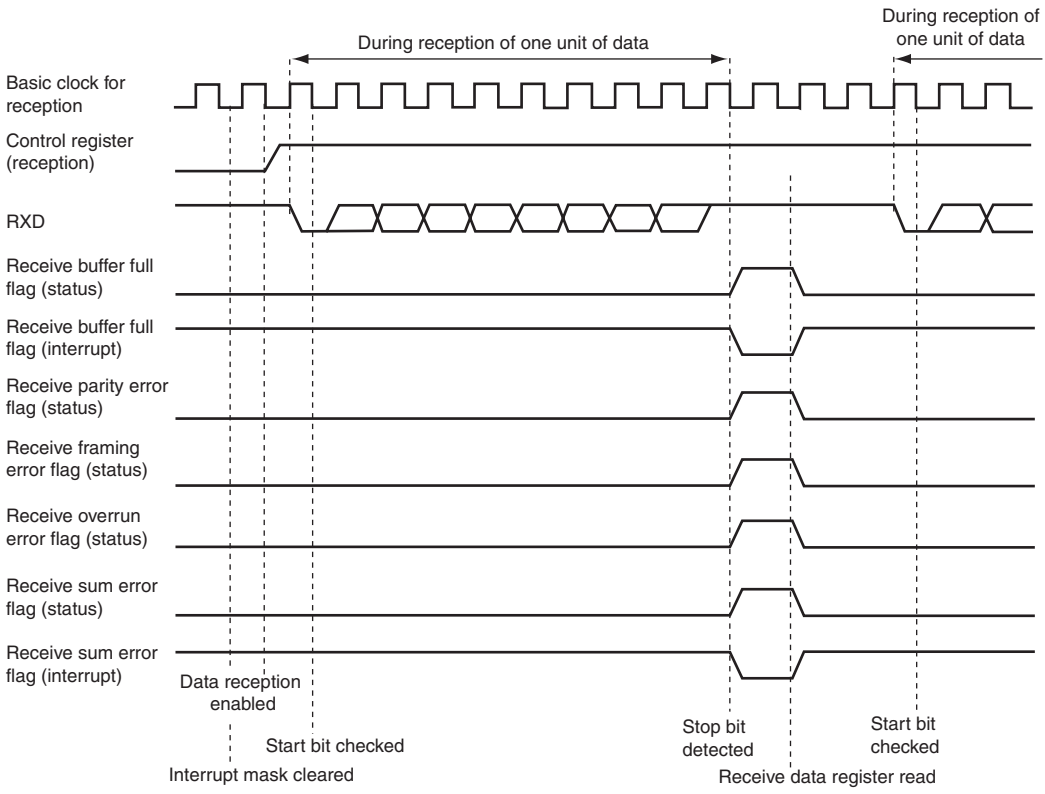


Figure 29.3 Data Transmission Timing

(3) Data Reception Timing

Figure 29.4 shows the data reception timing controlled by the UART.

When the last stop bit of the received data is detected, the received data is stored and the receive flags are set or cleared appropriately.



Note: The data reception control circuit recognizes the start bit when a falling edge is detected on the RXD input pin. After that, the control circuit samples the signal level at the center of the start bit duration, and if a low level is detected on the RXD input pin, the control circuit starts the reception sequence.

Figure 29.4 Data Reception Timing

(4) Baud Rate Setting for Data Transmission and Reception

The baud rate for UART data transmission and reception is calculated by the following equation.

$$\text{Baud rate [bps]} = \frac{\text{System clock (SCLK) [Hz]}}{\text{UABCA} + (\text{UABC} + 1) \times 16}$$

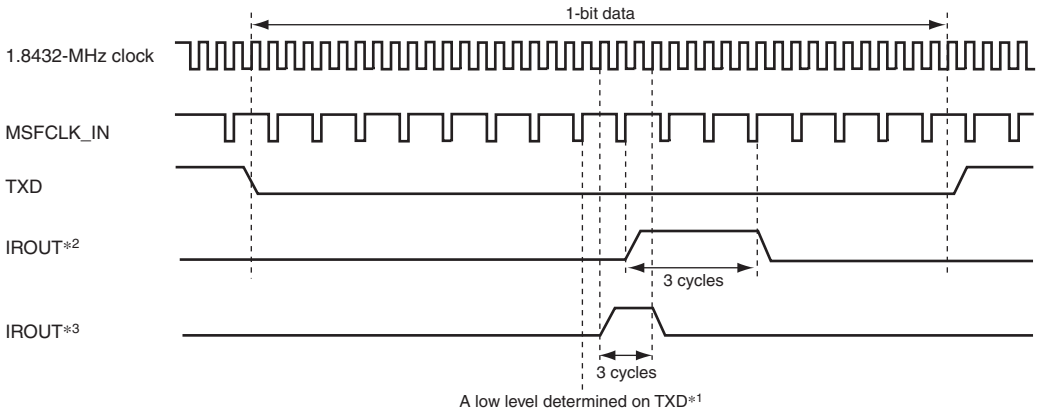
UABC: Baud rate counter value for data transmission and reception
(value specified in the UABC15 to UABC0 bits in IRIF_UART7)
UABCA: Baud rate counter value for data transmission and reception
(value selected with the UABCA3 to UABCA0 bits in IRIF_UART6)

The clock shown in the above equation is the clock input to the controller. Every time the integer part of the baud rate count is reloaded, the fractional part selected by the baud rate error correction register is accumulated. When an overflow occurs during the fractional part accumulation, 1 is added to the integer part and the resultant value is reloaded to the counter. This means that the error in the baud rate count is eliminated by adding 1 to the count when the accumulated error in the fractional part reaches 1.

29.4.2 Transmit and Receive Pulse Modulation and Demodulation

(1) Transmission of Infrared Light-Emit Pulse Data

The data transmitted from the UART is encoded into a waveform conforming to IrDA standard 1.0, and infrared transmit (light-emit) pulse data is sent to the infrared transceiver device. Figure 29.5 shows the encoding timing.

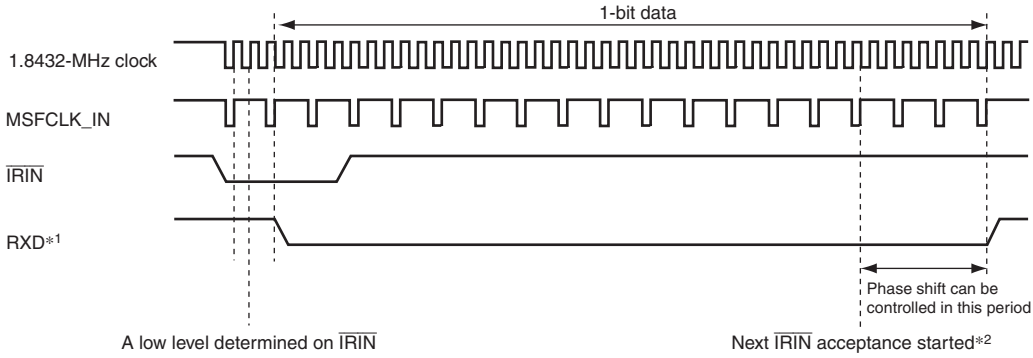


Notes: 1. A low level through TXD is determined when a low level is continuously detected for eight MSFCLK_IN cycles. After that, a pulse having a width equal to three cycles of the specified clock is not output.
2. This waveform shows the output timing at the IROUT pin when the infrared transmit pulse width select bit is set to 0 (three cycles of MSFCLK_IN).
3. This waveform shows the output timing at the IROUT pin when the infrared transmit pulse width select bit is set to 1 (three cycles of the 1.8432-MHz clock).

Figure 29.5 Timing for Encoding Infrared Transmit (Light-Emit) Pulse Data

(2) Reception of Infrared Light-Receive Pulse Data

The infrared receive (light-receive) pulse data is sent from the infrared transceiver and its waveform conforming to IrDA standard 1.0 is decoded and transferred to the UART. Figure 29.6 shows the decoding timing.



Notes: 1. When a low level is detected on the IRIN pin for at least two continuous cycles of the 1.8432-MHz clock (an internally generated clock), a low pulse having a width of 17 MSFCLK_IN cycles is output. This eliminates the phase shift (lag) in the IRIN input pulse for one MSFCLK_IN cycle.
2. The next IRIN input can be detected 14 MSFCLK_IN cycles after a low level is determined on the IRIN pin. Until 14 cycles have passed, a low level on the IRIN pin is not detected as an effective pulse. This eliminates the phase shift (lead) in the IRIN input pulse for one MSFCLK_IN cycle.

Figure 29.6 Timing for Decoding Infrared Receive (Light-Receive) Pulse Data

(3) Internal Clock Generation for Transmit and Receive Pulse Modulation and Demodulation

The 1.8432-MHz clock used in the transmit and receive pulse modulator/demodulator block is generated by the following equation.

$$\text{1.8432-MHz clock} = \frac{\text{System clock (SCLK) [Hz]}}{\text{IRBCA} + (\text{IRBC} + 1)}$$

IRBC: Baud rate counter value for infrared transmit and receive pulse modulation and demodulation (value specified in the IRBC3 to IRBC0 bits in IRIF_SIR2)
IRBCA: Baud rate error correction value for infrared transmit and receive pulse modulation and demodulation (value selected with the IRBCA3 to IRBCA0 bits in IRIF_SIR1)

The 1.8432-MHz clock is used to measure 1.63 μs and is necessary in the following operations.

- Generating a 1.63-μs infrared transmit (light-emit) pulse
- Recognizing an infrared receive (light-receive) pulse
- Detecting an error in the width (shorter than the standard) of an infrared receive (light-receive) pulse

The clock shown in the above equation is the clock input to the controller. Every time the integer part of the baud rate count is reloaded, the fractional part selected by the baud rate error correction register is accumulated. When an overflow occurs during fractional part accumulation, 1 is added to the integer part and the resultant value is reloaded to the counter. This means that the error in the baud rate count is eliminated by adding 1 to the count when the accumulated error in the fractional part reaches 1.

(4) Notes on Infrared Transmit and Receive Pulse Modulation and Demodulation

(a) Errors in the width of infrared receive (light-receive) pulses

The infrared receive pulse error flag (IRERR) is set to 1 when the width of an infrared receive (light-receive) pulse is determined as outside of the standard; that is, in the following cases.

- When a low level of an infrared receive pulse is detected for only one cycle of the 1.8432-MHz clock (shorter than the standard)
- When a low level of an infrared receive pulse is detected for five or more continuous cycles of the MSFCLK_IN clock (longer than the standard)
- When a high level of an infrared receive pulse is detected for only one cycle of the 1.8432-MHz clock (lacking a pulse)

Note that the following case is not detected as an error although this pulse width does not satisfy the standard.

- When a pulse is input for less than one cycle of the 1.8432-MHz clock

Note: When the pulse width is longer than the standard value, the error flag is set to 1 but the pulse is determined as effective and the modulator/demodulator performs the normal operation (outputs a low level through RXD). The above errors in the pulse width can be detected even during data reception after an infrared receive pulse is recognized.

(b) Interface with the infrared transceiver device

The polarity of the input signal to the infrared transceiver device is opposite to that of the output signal as follows.

- Infrared data transmit pin (IROUT): Positive logic output
- Infrared data receive pin ($\overline{\text{IRIN}}$): Negative logic input

(c) Register read and write

Do not write to IRIF_SIR0, IRIF_SIR1, or IRIF_SIR2 during data transmission or reception. If this is attempted, correct data communication cannot be guaranteed.

(d) Infrared transmit (light-emit) pulse width select bit

When a 1.8432-MHz clock is input through MSFCLK_IN (the data transmission and reception function operates at 115.2 kbps), do not set the infrared transmit (light-emit) pulse width select bit to 1. If this is attempted, correct transmit pulses might not be output.

(e) Pulse width

Do not set the baud rate count set register for infrared transmit and receive pulse modulation and demodulation (IRIF_SIR2) to H'0000. If this is attempted, the transmit pulse width may be less than the lower limit (pulse duration minimum: 1.41 μs) prescribed in the standard (Infrared Data Association Serial Infrared Physical Layer Specification Version 1.3).

29.4.3 CRC Engine

(1) CRC Engine Configuration

The CRC engine consists of an input data register, a byte counter, a CRC calculation register, and a CRC output register. Figure 29.7 shows the configuration of the CRC engine.

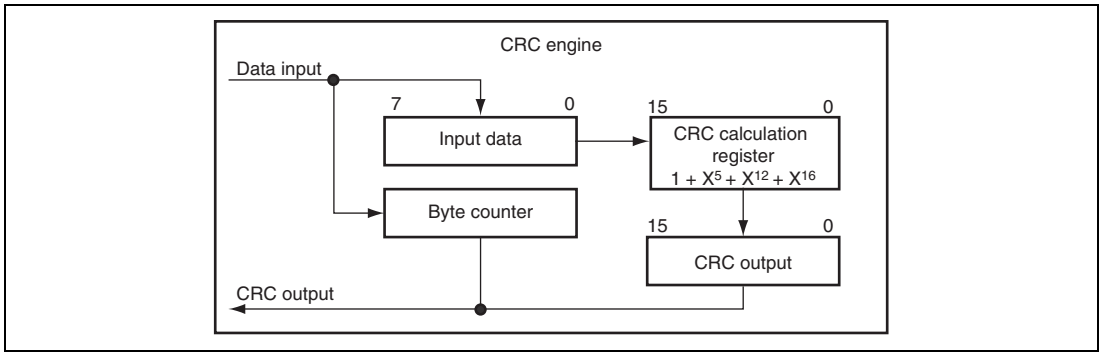


Figure 29.7 CRC Engine Configuration

(2) CRC Engine Operation

Writing 8-bit input data to the CRC engine starts CRC calculation in 8-bit units beginning with the lower bits to output a 16-bit calculation result. Figure 29.8 gives an overview of the CRC calculation.

The CRC generator polynomial is $1 + X^5 + X^{12} + X^{16}$, and the maximum data length is 4096 bytes.

An example of CRC calculation is shown here. After resetting the registers, write H'CC, H'F5, H'F1 and H'A7 as input data in that order. The resultant byte count will be 4 and H'51DF will be output as the CRC calculation result.

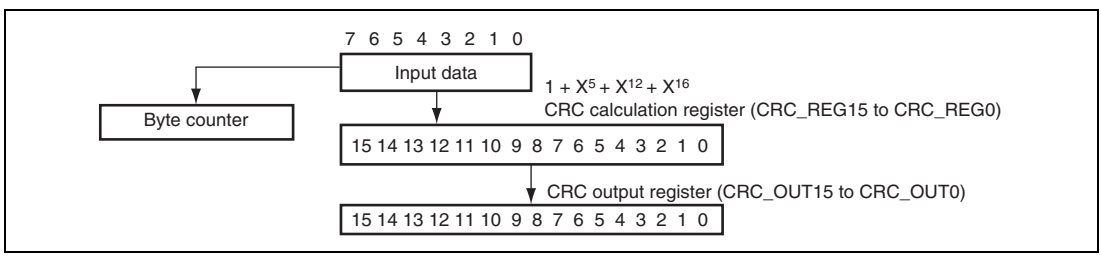


Figure 29.8 CRC Engine Operation

29.4.4 Communication Flow

(1) IrDA Transmission Flow

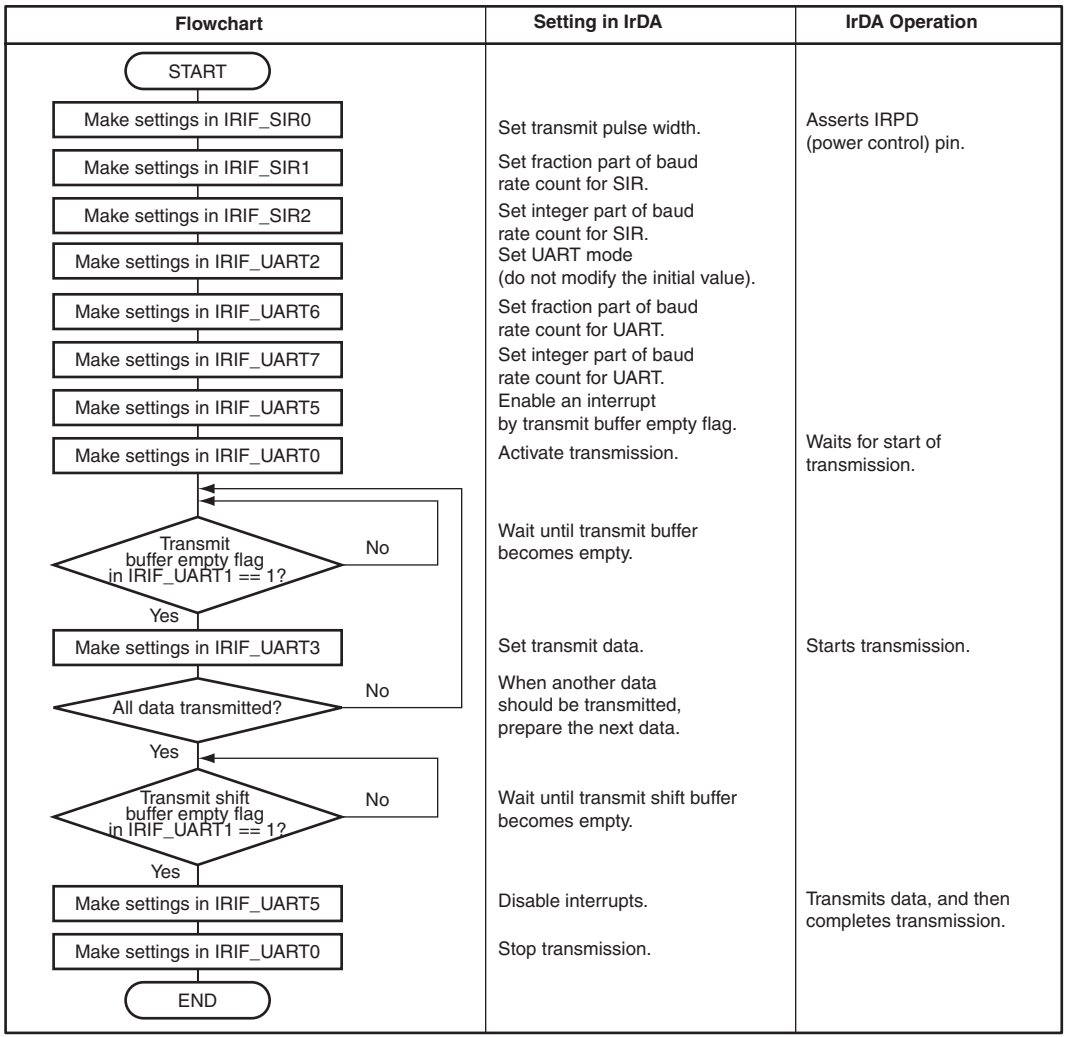


Figure 29.9 IrDA Transmission Flow

(2) IrDA Transmission (CRC Calculation) Flow

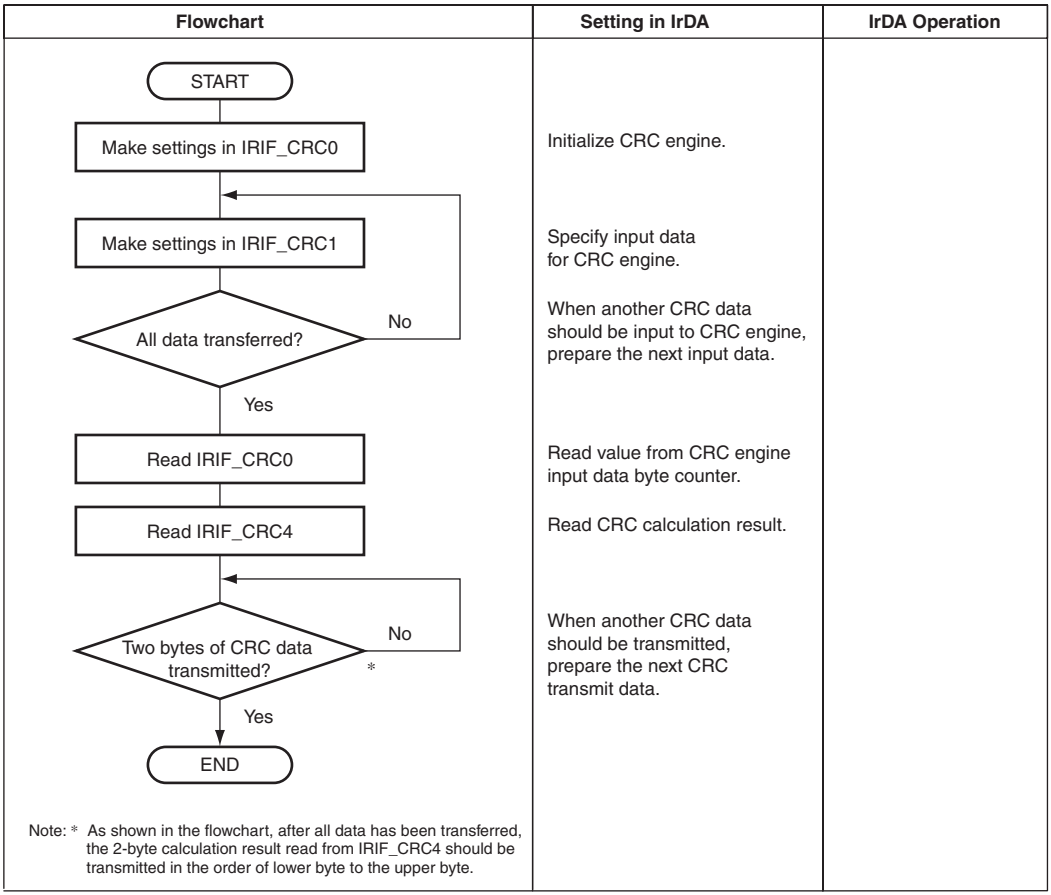


Figure 29.10 IrDA Transmission (CRC Calculation) Flow

(3) IrDA Reception Flow

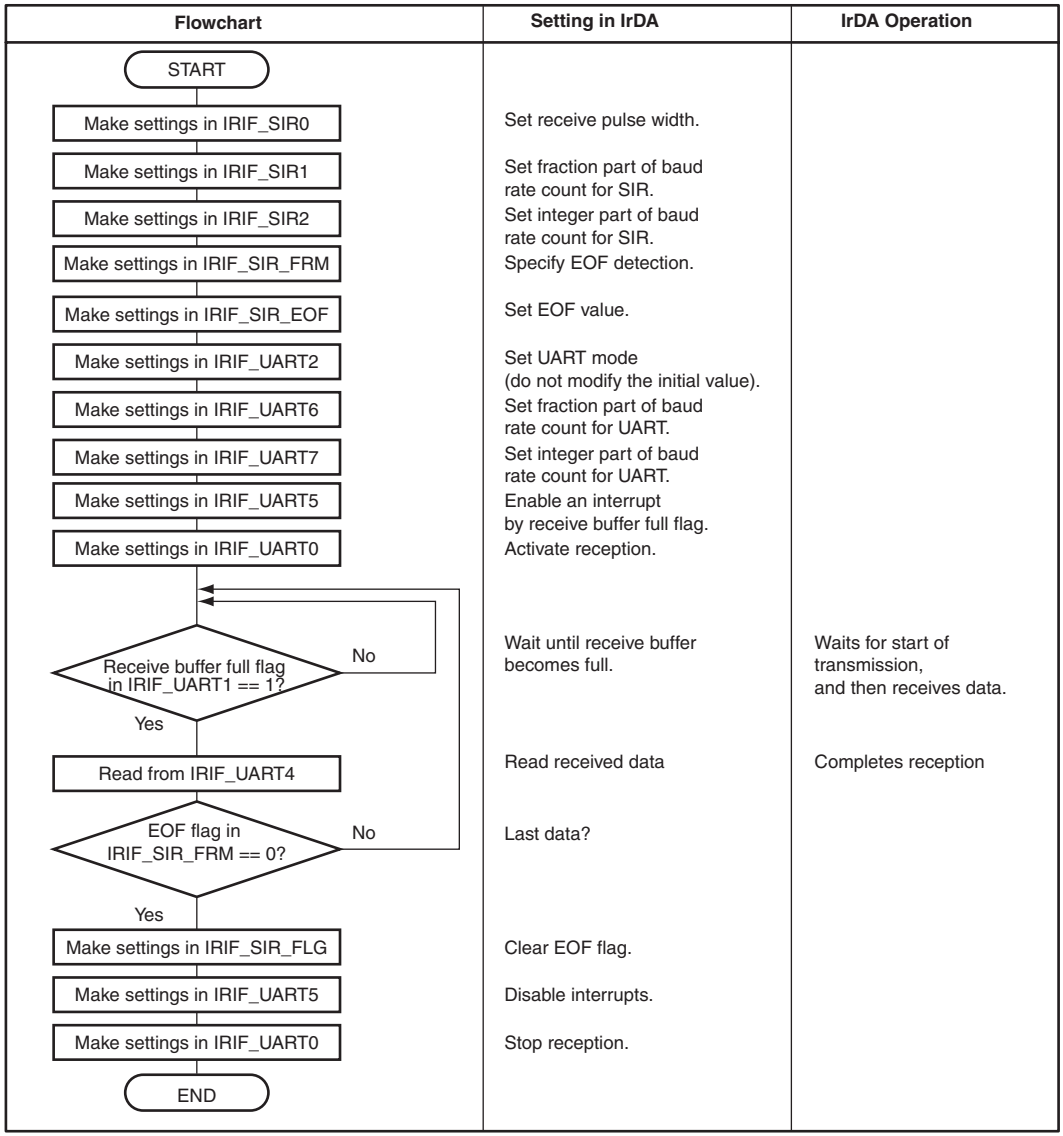


Figure 29.11 IrDA Reception Flow

(4) IrDA Reception (CRC Calculation) Flow

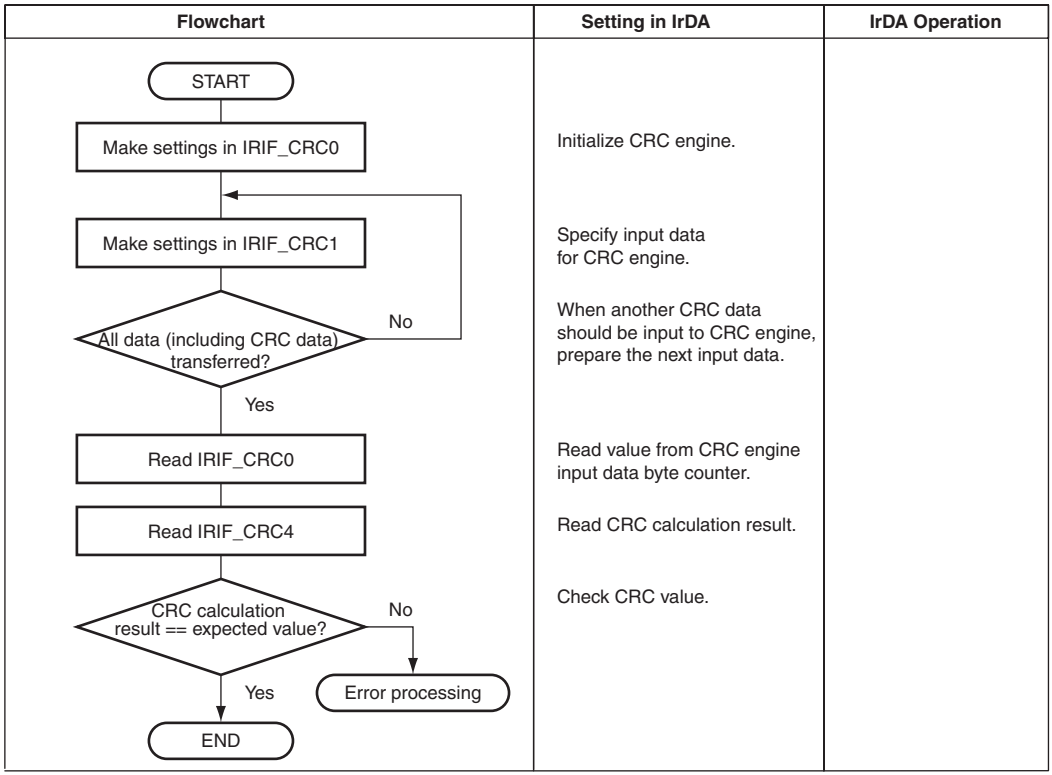


Figure 29.12 IrDA Reception (CRC Calculation) Flow

29.5 Notes on Data Transmission and Reception

(1) Access to Data Receive Buffer

After data reception is completed (the receive buffer becomes full), if read access to the receive buffer register is delayed and then attempted at exactly the same time as when the next data reception is completed, data may be lost without any error interrupt occurring.

Under usual conditions, if the next data reception is completed without reading the previously received data from the receive buffer register, a receive overrun error interrupt should be generated. However, if the previous data is read from the receive buffer register at exactly the same time as when the next data reception is completed, one read operation may be incorrectly recognized as two read operations. In this case, the read value is undefined and a receive overrun error interrupt may not occur.

This problem should be prevented by controlling the operating conditions so that no receive buffer overrun error occurs.

(2) Transmission Jitter

When the IrDA transmit pulse width is set to 1.63 μs and the baud rate is set within a range from 57.6 kbps to 19.2 kbps, the transmission jitter of an IrDA pulse may exceed the upper limit (rate tolerance: ±0.87%) prescribed in the IrDA standard (Infrared Data Association Serial Infrared Physical Layer Specification Version 1.3).

(3) Prohibited Value (H'0001) for IRIF_SIR0 at a Baud Rate of 115 kbps

Do not set IRIF_SIR0 to H'0001 when the baud rate is 115 kbps. If this is attempted, the transmit pulse width may be below the lower limit (pulse duration minimum: 1.41 μs) prescribed in the IrDA standard (Infrared Data Association Serial Infrared Physical Layer Specification Version 1.3).

Section 30 Key Scan Interface (KEYSC)

This LSI has a key scan interface (KEYSC) that can set the input or output bit numbers to be programmable.

30.1 Features

- On-chip chattering elimination circuit
- Chattering elimination time can be set to be programmable
- Measures to deal with multiple key presses
- Level/edge-selectable internal interrupts
- Canceling software standby and R/U-standby modes by the key input (level) interrupt.
- Input or output bit numbers can be set to be programmable

Figure 30.1 shows a block diagram of the key scan interface.

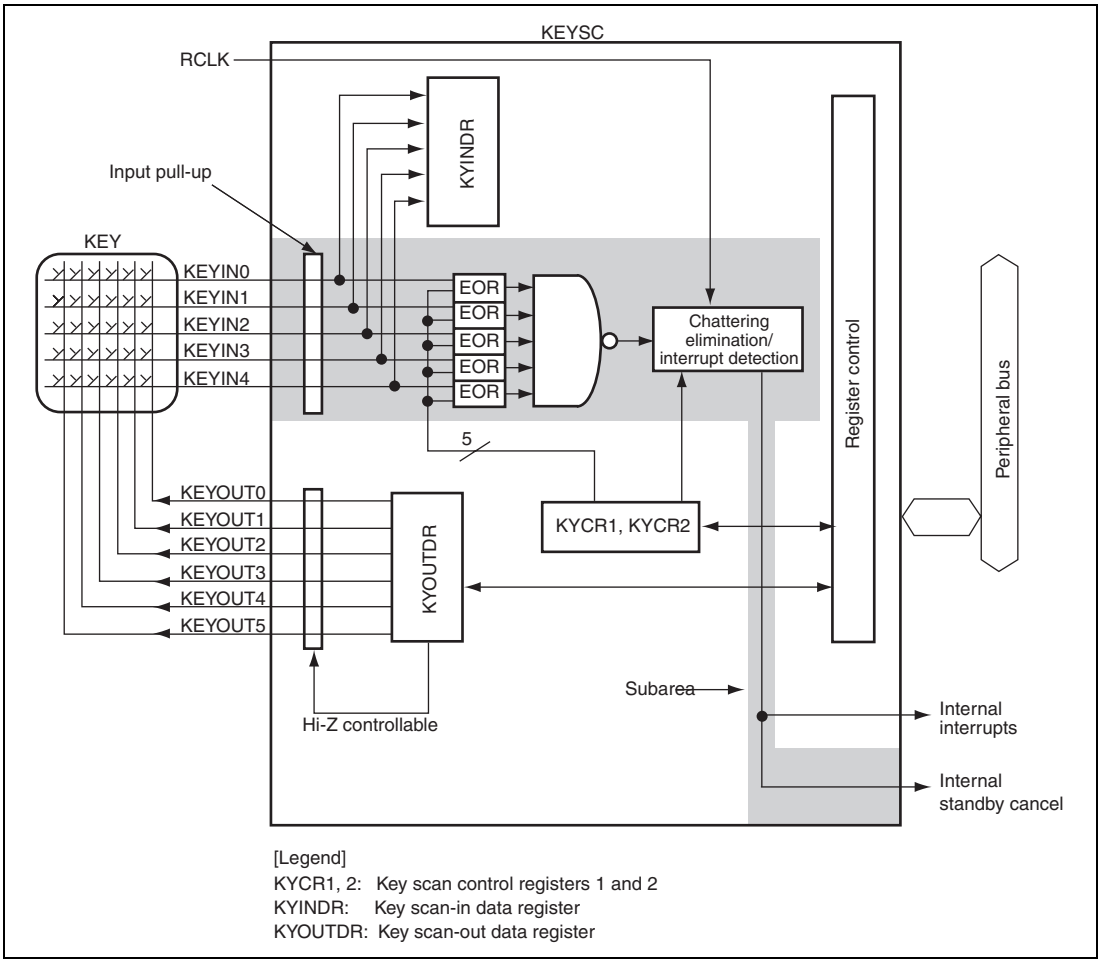


Figure 30.1 Block Diagram of Key Scan Interface (Key Pin Mode 1)

30.2 Input/Output Pins

The pin configuration of the key scan interface is listed in Table 30.1.

Table 30.1 Pin Configuration

Name	Abbreviation	I/O	Function
Input key scan interface 6 to 0	KEYIN6 to KEYIN0	Input	Key scan interface for input
Output key scan interface 5 to 0	KEYOUT5 to KEYOUT0	Output	Key scan interface for output

The KEYOUT5 and KEYOUT4 pins are multiplexed with the KEYIN5 and KEYIN6 pins respectively. Setting the KYMD1 and KYMD0 bits in the key scan control register 1 (KYCR1) selects either those functions. Table 30.2 shows the possible combinations between the KEYIN and KEYOUT pins.

Table 30.2 Multiplex Pin Setting

Name	KYMD1	KYMD0	KEYOUT5/KEYIN5 Pin	KEYOUT4/KEYIN6 Pin
Key pin mode 1	0	0	Selects KEYOUT5 pin	Selects KEYOUT4 pin
Key pin mode 2	0	1	Selects KEYIN5 pin	Selects KEYOUT4 pin
Key pin mode 3	1	0	Selects KEYIN5 pin	Selects KEYIN6 pin

30.3 Register Descriptions

Table 30.3 shows the KEYSC register configuration. Table 30.4 shows the register states in each operating mode.

Table 30.3 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Key scan control register 1	KYCR1	R/W	H'A44B 0000	16
Key scan control register 2	KYCR2	R/W	H'A44B 0004	16
Key scan-in data register	KYINDR	R	H'A44B 0008	16
Key scan-out data register	KYOUTDR	R/W	H'A44B 000C	16

Table 30.4 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
KYCR1	Initialized	Initialized	Retained	Retained	Initialized*2	Initialized	Retained
KYCR2	Initialized*1	Initialized*1	Retained	Retained	Initialized*3	Initialized*1	Retained
KYINDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
KYOUTDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Note: *1 Only the KEYIF bit retains the value before a reset.
*2 Only the KYOUT_IE bit retains.
*3 Only the KYCPU_IE[1:0] bit retains.

30.3.1 Key Scan Control Register 1 (KYCR1)

KYCR1 is a 16-bit readable/writable register that selects the functions of the key scan interface.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	KYMD[1:0]		KYOUT_IE	—	—	—	—	SCN[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	KYMD[1:0]	00	R/W	Key Pin Mode Set the KEYIN/KEYOUT pin to be used. 00: Use key pin mode 1 (KEYIN4 to KEYIN0, KEYOUT5 to KEYOUT0 are available) 01: Use key pin mode 2 (KEYIN5 to KEYIN0, KEYOUT4 to KEYOUT0 are available) 10: Use key pin mode 3 (KEYIN6 to KEYIN0, KEYOUT3 to KEYOUT0 are available) 11: Setting prohibited
7	KYOUT_IE	0	R/W	Key Scan Standby Mode Cancel Enable When various standby modes are canceled by the key input, this bit should be set to 1. 0: Disables various standby modes to be canceled by the key input. 1: Enables various standby modes to be canceled by the key input.
6 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	SCN[2:0]	000	R/W	Scan Timing Specify the chattering elimination time for the interrupt detection of the key scan interface. 000: RCLK/16 (0.49 ms) 001: RCLK/32 (0.98 ms) 010: RCLK/64 (1.95 ms) 011: RCLK/128 (3.91 ms) 100: RCLK/256 (7.81 ms) 101: RCLK/512 (15.63 ms) 110: RCLK/1024 (31.25 ms) 111: Chattering elimination circuit is invalid

30.3.2 Key Scan Control Register 2 (KYCR2)

KYCR2 is a 16-bit readable/writable register that selects the functions of the key scan interface.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	KYDIR6	KYDIR5	KYDIR4	KYDIR3	KYDIR2	KYDIR1	KYDIR0	—	—	KYCPU_IE[1:0]		—	—	—	KEYIF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	KYDIR6	0	R/W	Key Direction 6 The EOR result of the KEYIN6 pin and this bit is used to detect a KEYIN6 interrupt. Note: Values written at setting key pin mode 1 and key pin mode 2 are ignored.
13	KYDIR5	0	R/W	Key Direction 5 The EOR result of the KEYIN5 pin and this bit is used to detect a KEYIN5 interrupt. Note: Values written at setting key pin mode 1 are ignored.
12	KYDIR4	0	R/W	Key Direction 4 The EOR result of the KEYIN4 pin and this bit is used to detect a KEYIN4 interrupt.
11	KYDIR3	0	R/W	Key Direction 3 The EOR result of the KEYIN3 pin and this bit is used to detect a KEYIN3 interrupt.
10	KYDIR2	0	R/W	Key Direction 2 The EOR result of the KEYIN2 pin and this bit is used to detect a KEYIN2 interrupt.
9	KYDIR1	0	R/W	Key Direction 1 The EOR result of the KEYIN1 pin and this bit is used to detect a KEYIN1 interrupt.
8	KYDIR0	0	R/W	Key Direction 0 The EOR result of the KEYIN0 pin and this bit is used to detect a KEYIN0 interrupt.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	KYCPU_IE [1:0]	00	R/W	Key Internal Interrupt Enable Enable or disable internal interrupt requests to the CPU when there is key input. 00: Key interrupt requests disabled 01: Key interrupt requests enabled (level) 10: Key interrupt requests enabled (rising edge) 11: Key interrupt requests enabled (falling edge) Note: These bits should be set to 01 when various standby modes are canceled by the key input.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	KEYIF	Undefined (value before reset is retained)	R/W	Key Interrupt This flag is set when a key internal interrupt occurs. While this flag is 1, an internal interrupt request is sent to the CPU. (It is held at 1 until overwritten by 0.) 0: No key interrupt occurs 1: A key interrupt occurs*

Note: * Case of level interrupts: When there is even one instance in which the result of the exclusive OR (EOR) of the KEYIN6 to KEYIN0 pins and the KYDIR6 to KYDIR0 bits, respectively, is 0.

Case of edge interrupts: When there is even one instance in which the result of the exclusive OR (EOR) of the KEYIN6 to KEYIN0 pins and the KYDIR6 to KYDIR0 bits, respectively, is 0, and the changing edges for the state in which all results are 1.

30.3.3 Key Scan-In Data Register (KYINDR)

KYINDR is a 16-bit read-only register that stores the data of pins KEYIN6 to KEYIN0. Bits KYI6DT to KYI0DT correspond respectively to pins KEYIN6 to KEYIN0.

In order to use the key scan interface functions, the pin function controller should be used to set the pin function settings to the key scan interface pin side. Pins KEYIN6 to KEYIN0 are pulled-up in the internal LSI.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	KYI6DT	KYI5DT	KYI4DT	KYI3DT	KYI2DT	KYI1DT	KYI0DT
Initial value:	0	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0.
6	KYI6DT	Undefined*	R	State of Pin KEYIN6 (1 at high or 0 at low) Note: This bit is always read as 0 in key pin mode 1 and key pin mode 2.
5	KYI5DT	Undefined*	R	State of Pin KEYIN5 (1 at high or 0 at low) Note: This bit is always read as 0 in key pin mode 1.
4	KYI4DT	Undefined*	R	State of Pin KEYIN4 (1 at high or 0 at low)
3	KYI3DT	Undefined*	R	State of Pin KEYIN3 (1 at high or 0 at low)
2	KYI2DT	Undefined*	R	State of Pin KEYIN2 (1 at high or 0 at low)
1	KYI1DT	Undefined*	R	State of Pin KEYIN1 (1 at high or 0 at low)
0	KYI0DT	Undefined*	R	State of Pin KEYIN0 (1 at high or 0 at low)

Note: * Values read from these bits immediately after a reset are undefined because the data-bus function of the BSC is being enabled.

30.3.4 Key Scan-Out Data Register (KYOUTDR)

KYOUTDR is a 16-bit readable/writable register that stores the output data for pins KEYOUT5 to KEYOUT0. Bits KYO5DT to KYO0DT correspond respectively to pins KEYOUT5 to KEYOUT0.

In order to use the key scan interface functions, the pin function controller should be used to set the pin function settings to the key scan interface pin side.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	KYO5DT[1:0]		KYO4DT[1:0]		KYO3DT[1:0]		KYO2DT[1:0]		KYO1DT[1:0]		KYO0DT[1:0]	
Initial value:	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11, 10	KYO5DT[1:0]	11	R/W	KYOnDT: Control output data of pin KEYOUTn. (n = 0 to 5) 00: Low-level output 01: High-level output 10: High-impedance state 11: High-impedance state Note: KYO5DT: Values written at setting key pin mode 2 and key pin mode 3 are ignored. KYO4DT: Values written at setting key pin mode 3 are ignored.
9, 8	KYO4DT[1:0]	11	R/W	
7, 6	KYO3DT[1:0]	11	R/W	
5, 4	KYO2DT[1:0]	11	R/W	
3, 2	KYO1DT[1:0]	11	R/W	
1, 0	KYO0DT[1:0]	11	R/W	

30.4 Operation

30.4.1 Chattering Elimination

By setting the SCN bits in KYCR1, chattering elimination in key interrupt detection is possible. Through this chattering elimination, if a key is pressed continuously for longer than a length of time set by the SCN bits, a continuous key press is recognized. Thereafter, when a state in which a key is released continues for longer than a length of time set by the SCN bits, the key is recognized as having been released.

30.4.2 Detection of Multiple Key Presses

While the KYOUTDR setting is used to set the KEYOUT5 to KEYOUT0 pins to the low-level output in order one at a time, and with the other five pins at high impedance, multiple key presses can be detected by reading KYINDR.

30.4.3 Register Access

In order for the chattering elimination unit and various standby mode cancel notification unit of the key scan interface to operate even in U-standby mode, they exist in the sub area. The register settings necessary for operation are performed from the Vcc (main) area. Hence when updating the registers, the following limitations apply.

When updating the registers as described below, do not make a transition to U-standby mode until the settings are reflected in actual operation. Otherwise, the register settings may not be reflected in actual operation.

- KYCR1: Bits KYOUT_IE and SCN2 to SCN0
- KYCR2: Bits KYDIR4 to KYDIR0

After writing to a register, it can be read immediately, but in order to reflect it in actual operation, two cycles are required for RCLK.

30.5 Examples of Use

30.5.1 Level Interrupt (KYCPU_IE1 and KYCPU_IE0 = 01)

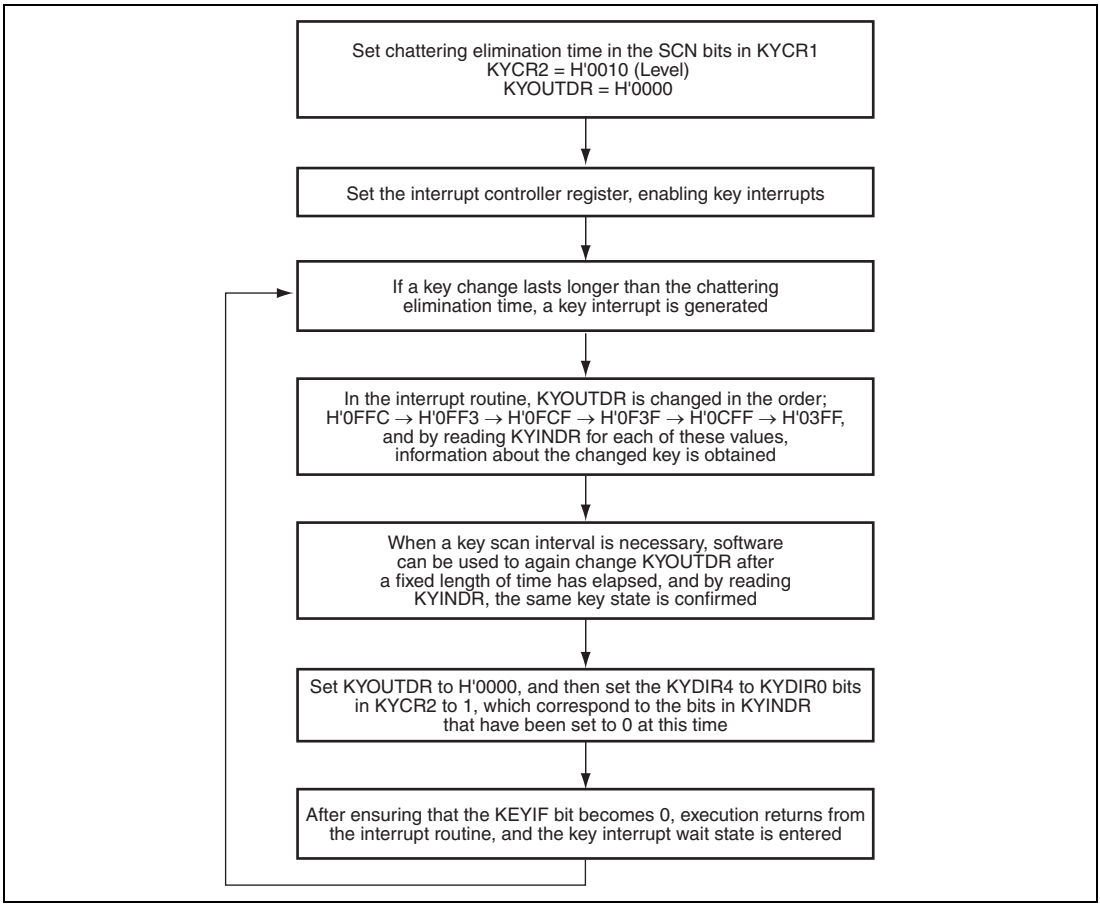


Figure 30.2 Example of Key Scan Using Level Interrupt (Key Pin Mode 1)

30.5.2 Edge Interrupt (KYCPU_IE1 and KYCPU_IE0 = 1x)

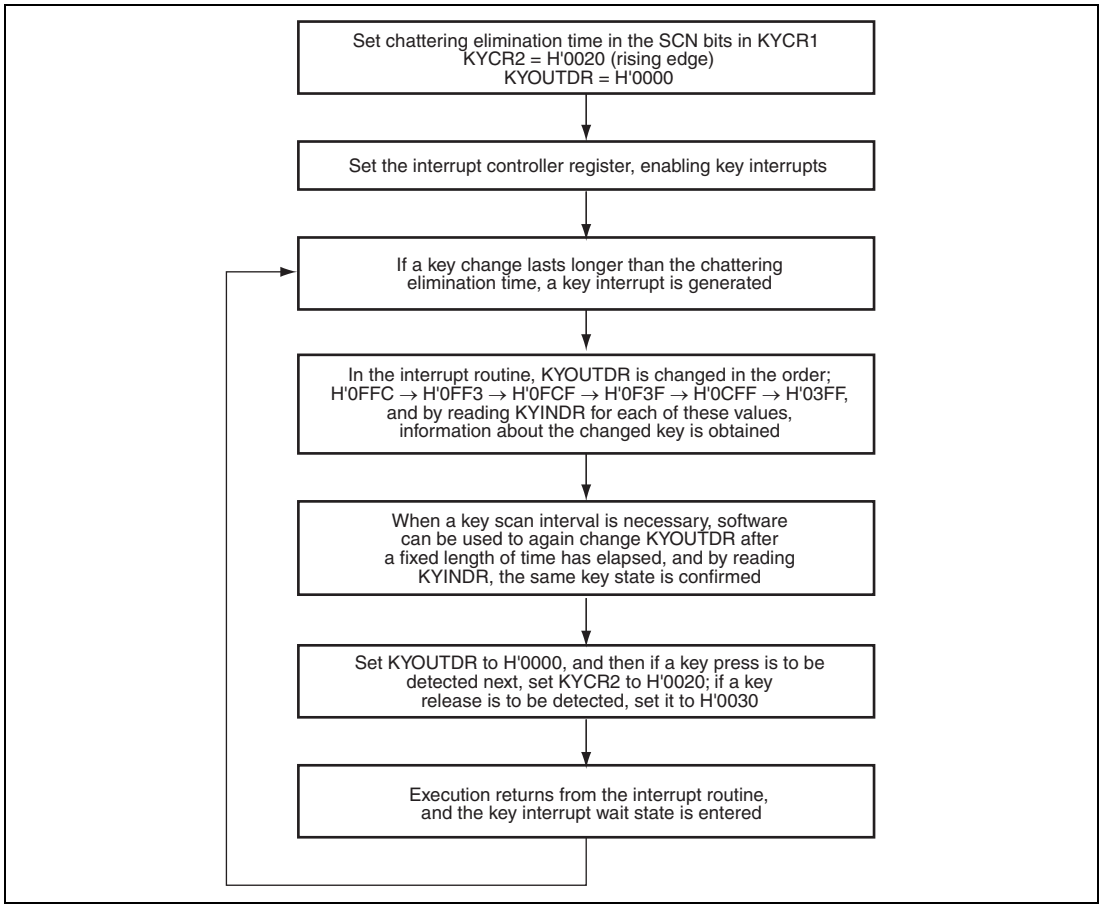


Figure 30.3 Example of Key Scan Using Edge Interrupt (Key Pin Mode 1)

Section 31 USB 2.0 Host/Function Module (USB0, USB1)

The USB 2.0 host/function module (USB0, USB1) is a USB controller which provides capabilities as a USB host controller and USB function controller function. This module supports high-speed transfer defined by USB (universal serial bus) Specification 2.0, full-speed transfer, and low-speed transfer when used as the host controller, and supports high-speed transfer and full-speed transfer when used as the function controller. This module has two USB transceivers and supports all of the transfer types defined by the USB specification.

USB0 and USB1 have a 16-kbyte buffer memory for data transfer, providing a maximum of ten pipes each other. Any endpoint numbers can be assigned to PIPE1 to PIPE9, based on the peripheral devices or user system for communication.

31.1 Features

(1) Host Controller and Function Controller Supporting USB High-Speed Operation

- Two sets of USB host controller and USB function controller are incorporated.
- The USB host controller and USB function controller can be switched by register settings.
- USB transceiver is incorporated.

(2) Reduced Number of External Pins and Space-Saving Installation

- The VBUS signal can be directly connected to the input pin of this module.
- On-chip D+ pull-up resistor (during USB function operation)
- On-chip D+ and D- pull-down resistor (during USB host operation)
- On-chip D+ and D- terminal resistor (during high-speed operation)
- On-chip D+ and D- output impedance (during full-speed operation)

(3) All Types of USB Transfers Supported

- Control transfer
- Bulk transfer
- Interrupt transfer (high bandwidth transfers not supported)
- Isochronous transfer (high bandwidth transfers not supported)

(4) Internal Bus Interfaces

- Two DMA interface channels are incorporated in USB0 and USB1 each other.

(5) Pipe Configuration

The following features exist in USB0 and USB1 each other.

- Up to 16 Kbytes of buffer memory for USB communications are supported
- Up to ten pipes can be selected (including the default control pipe)
- Programmable pipe configuration
- Endpoint numbers can be assigned flexibly to PIPE1 to PIPE9.
- Transfer conditions that can be set for each pipe:

PIPE0:	Control transfer (default control pipe: DCP), 64-byte fixed single buffer
PIPE1 and PIPE2:	Bulk transfers/isochronous transfer, continuous transfer mode, programmable buffer size (up to 2-kbytes: double buffer can be specified)
PIPE3 to PIPE5:	Bulk transfer, continuous transfer mode, programmable buffer size (up to 2-kbytes: double buffer can be specified)
PIPE6 to PIPE9:	Interrupt transfer, 64-byte fixed single buffer

(6) Features of the USB Host Controller

- High-speed transfer (480 Mbps), full-speed transfer (12 Mbps), and low-speed transfer (1.5 Mbps) are supported.
- Communications with multiple peripheral devices connected via a single HUB
- Automatic response to the reset handshake
- Automatic scheduling for SOF and packet transmissions
- Programmable intervals for isochronous and interrupt transfers

(7) Features of the USB Function Controller

- Both high-speed transfer (480 Mbps) and full-speed transfer (12 Mbps) are supported.
- Automatic recognition of high-speed operation or full-speed operation based on automatic response to the reset handshake
- Control transfer stage control function
- Device state control function
- Auto response function for SET_ADDRESS request
- NAK response interrupt function (NRDY)
- SOF interpolation function

(8) Other Features

- Transfer ending function using transaction count
- BRDY interrupt event notification timing change function (BFRE)
- Function that automatically clears the buffer memory after the data for the pipe specified at the DnFIFO (n = 0 or 1) port has been read (DCLRM)
- NAK setting function for response PID generated by end of transfer (SHTNAK)

31.2 Input/Output Pins

Table 31.1 shows the pin configuration of the USB0. Table 31.2 shows the pin configuration of the USB1.

Table 31.1 USB0 Pin Configuration

Pin Name	Name	I/O	Function
DP0	USB0 D+ data	I/O	D+ I/O of the USB0 on-chip transceiver This pin should be connected to the D+ pin of the USB bus.
DM0	USB0 D- data	I/O	D− I/O of the USB0 on-chip transceiver This pin should be connected to the D- pin of the USB bus.
VBUS0	VBUS0 input	Input	USB0 cable connection monitor pin This pin should be connected directly to the VBUS of the USB bus. Whether the VBUS is connected or disconnected can be detected. If this pin is not connected with the VBUS of the USB bus, it should be supplied with 5 V. It should be supplied with 5 V also when the host controller function is selected.
REFRIN0	Reference input	Input	Reference resistor connection pin This pin should be connected to AG33 through a 5.6 kΩ ±1% resistor.
XTAL_USB	Crystal input output pin (Clock input pin) (Common to USB1)	Output	These pins should be connected to crystal oscillators for the USB. The EXTAL_USB pin can be used for external clock input.
EXTAL_USB		Input	

Table 31.2 USB1 Pin Configuration

Pin Name	Name	I/O	Function
DP1	USB1 D+ data	I/O	D+ I/O of the USB1 on-chip transceiver This pin should be connected to the D+ pin of the USB bus.
DM1	USB1 D- data	I/O	D− I/O of the USB1 on-chip transceiver This pin should be connected to the D- pin of the USB bus.
VBUS1	VBUS1 input	Input	USB1 cable connection monitor pin This pin should be connected directly to the VBUS of the USB bus. Whether the VBUS is connected or disconnected can be detected. If this pin is not connected with the VBUS of the USB bus, it should be supplied with 5 V. It should be supplied with 5 V also when the host controller function is selected.
REFRIN1	Reference input	Input	Reference resistor connection pin This pin should be connected to AG33 through a 5.6 kΩ ±1% resistor.
XTAL_USB	Crystal input output pin	Output	These pins should be connected to crystal oscillators for the USB. The EXTAL_USB pin can be used for external clock input.
EXTAL_USB	(Clock input pin) (Common to USB0)	Input	

31.3 USB block diagram

Figure 31.1 shows USB block diagram.

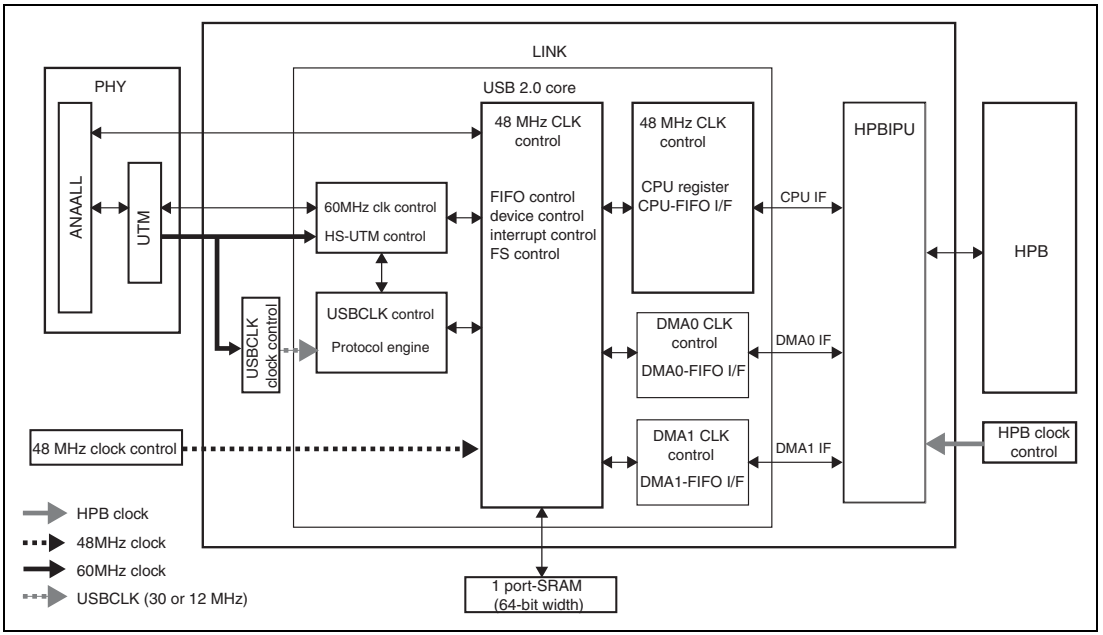


Figure 31.1 USB block diagram

31.4 Register Description

Table 31.3 shows the register configuration of the USB0. Table 31.4 shows the register configuration of the USB1. Table 31.5 shows the register state in each processing mode.

Table 31.3 USB0 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
System configuration control register_0	SYSCFG_0	R/W	H'A4D8 0000	16
CPU bus wait setting register_0	BUSWAIT_0	R/W	H'A4D8 0002	16
System configuration status register_0	SYSSTS_0	R	H'A4D8 0004	16
Device state control register_0	DVSTCTR_0	R/W	H'A4D8 0008	16
Test mode register_0	TESTMODE_0	R/W	H'A4D8 000C	16
CFIFO port register_0	CFIFO_0	R/W	H'A4D8 0014	8/16/32
D0FIFO port register_0	D0FIFO_0	R/W	H'A4D8 0100	8/16/32
D1FIFO port register_0	D1FIFO_0	R/W	H'A4D8 0120	8/16/32
CFIFO port select register_0	CFIFOSEL_0	R/W	H'A4D8 0020	16
CFIFO port control register_0	CFIFOCTR_0	R/W	H'A4D8 0022	16
D0FIFO port select register_0	D0FIFOSEL_0	R/W	H'A4D8 0028	16
D0FIFO port control register_0	D0FIFOCTR_0	R/W	H'A4D8 002A	16
D1FIFO port select register_0	D1FIFOSEL_0	R/W	H'A4D8 002C	16
D1FIFO port control register_0	D1FIFOCTR_0	R/W	H'A4D8 002E	16
Interrupt enable register 0_0	INTENB0_0	R/W	H'A4D8 0030	16
Interrupt enable register 1_0	INTENB1_0	R/W	H'A4D8 0032	16
BRDY interrupt enable register_0	BRDYENB_0	R/W	H'A4D8 0036	16
NRDY interrupt enable register_0	NRDYENB_0	R/W	H'A4D8 0038	16
BEMP interrupt enable register_0	BEMPENB_0	R/W	H'A4D8 003A	16
SOF output configuration register_0	SOFCFG_0	R/W	H'A4D8 003C	16
Interrupt status register 0_0	INTSTS0_0	R/W	H'A4D8 0040	16
Interrupt status register 1_0	INTSTS1_0	R/W	H'A4D8 0042	16

Register Name	Abbreviation	R/W	Address	Access Size
BRDY interrupt status register_0	BRDYSTS_0	R/W	H'A4D8 0046	16
NRDY interrupt status register_0	NRDYSTS_0	R/W	H'A4D8 0048	16
BEMP interrupt status register_0	BEMPSTS_0	R/W	H'A4D8 004A	16
Frame number register_0	FRMNUM_0	R/W	H'A4D8 004C	16
μFrame number register_0	UFRMNUM_0	R/W	H'A4D8 004E	16
USB address register_0	USBADDR_0	R	H'A4D8 0050	16
USB request type register_0	USBREQ_0	R/W	H'A4D8 0054	16
USB request value register_0	USBVAL_0	R/W	H'A4D8 0056	16
USB request index register_0	USBINDX_0	R/W	H'A4D8 0058	16
USB request length register_0	USBLENG_0	R/W	H'A4D8 005A	16
DCP configuration register_0	DCPCFG_0	R/W	H'A4D8 005C	16
DCP maximum packet size register_0	DCPMAXP_0	R/W	H'A4D8 005E	16
DCP control register_0	DCPCTR_0	R/W	H'A4D8 0060	16
Pipe window select register_0	PIPESEL_0	R/W	H'A4D8 0064	16
Pipe configuration register_0	PIPECFG_0	R/W	H'A4D8 0068	16
Pipe buffer setting register_0	PIPEBUF_0	R/W	H'A4D8 006A	16
Pipe maximum packet size register_0	PIPEMAXP_0	R/W	H'A4D8 006C	16
Pipe cycle control register_0	PIPEPERI_0	R/W	H'A4D8 006E	16
Pipe 1 control register_0	PIPE1CTR_0	R/W	H'A4D8 0070	16
Pipe 2 control register_0	PIPE2CTR_0	R/W	H'A4D8 0072	16
Pipe 3 control register_0	PIPE3CTR_0	R/W	H'A4D8 0074	16
Pipe 4 control register_0	PIPE4CTR_0	R/W	H'A4D8 0076	16
Pipe 5 control register_0	PIPE5CTR_0	R/W	H'A4D8 0078	16
Pipe 6 control register_0	PIPE6CTR_0	R/W	H'A4D8 007A	16
Pipe 7 control register_0	PIPE7CTR_0	R/W	H'A4D8 007C	16
Pipe 8 control register_0	PIPE8CTR_0	R/W	H'A4D8 007E	16
Pipe 9 control register_0	PIPE9CTR_0	R/W	H'A4D8 0080	16
Pipe 1 transaction counter enable register_0	PIPE1TRE_0	R/W	H'A4D8 0090	16
Pipe 1 transaction counter register_0	PIPE1TRN_0	R/W	H'A4D8 0092	16
Pipe 2 transaction counter enable register_0	PIPE2TRE_0	R/W	H'A4D8 0094	16

Register Name	Abbreviation	R/W	Address	Access Size
Pipe 2 transaction counter register_0	PIPE2TRN_0	R/W	H'A4D8 0096	16
Pipe 3 transaction counter enable register_0	PIPE3TRE_0	R/W	H'A4D8 0098	16
Pipe 3 transaction counter register_0	PIPE3TRN_0	R/W	H'A4D8 009A	16
Pipe 4 transaction counter enable register_0	PIPE4TRE_0	R/W	H'A4D8 009C	16
Pipe 4 transaction counter register_0	PIPE4TRN_0	R/W	H'A4D8 009E	16
Pipe 5 transaction counter enable register_0	PIPE5TRE_0	R/W	H'A4D8 00A0	16
Pipe 5 transaction counter register_0	PIPE5TRN_0	R/W	H'A4D8 00A2	16
Device address 0 configuration register_0	DEVADD0_0	R/W	H'A4D8 00D0	16
Device address 1 configuration register_0	DEVADD1_0	R/W	H'A4D8 00D2	16
Device address 2 configuration register_0	DEVADD2_0	R/W	H'A4D8 00D4	16
Device address 3 configuration register_0	DEVADD3_0	R/W	H'A4D8 00D6	16
Device address 4 configuration register_0	DEVADD4_0	R/W	H'A4D8 00D8	16
Device address 5 configuration register_0	DEVADD5_0	R/W	H'A4D8 00DA	16
Device address 6 configuration register_0	DEVADD6_0	R/W	H'A4D8 00DC	16
Device address 7 configuration register_0	DEVADD7_0	R/W	H'A4D8 00DE	16
Device address 8 configuration register_0	DEVADD8_0	R/W	H'A4D8 00E0	16
Device address 9 configuration register_0	DEVADD9_0	R/W	H'A4D8 00E2	16
Device address A configuration register_0	DEVADDA_0	R/W	H'A4D8 00E4	16
USB power control register	UPONCR0	R/W	H'A405 01D4	16

Table 31.4 USB1 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
System configuration control register_1	SYSCFG_1	R/W	H'A4D9 0000	16
CPU bus wait setting register_1	BUSWAIT_1	R/W	H'A4D9 0002	16
System configuration status register_1	SYSSTS_1	R	H'A4D9 0004	16
Device state control register_1	DVSTCTR_1	R/W	H'A4D9 0008	16
Test mode register_1	TESTMODE_1	R/W	H'A4D9 000C	16
CFIFO port register_1	CFIFO_1	R/W	H'A4D9 0014	8/16/32
D0FIFO port register_1	D0FIFO_1	R/W	H'A4D9 0100	8/16/32
D1FIFO port register_1	D1FIFO_1	R/W	H'A4D9 0120	8/16/32
CFIFO port select register_1	CFIFOSEL_1	R/W	H'A4D9 0020	16
CFIFO port control register_1	CFIFOCTR_1	R/W	H'A4D9 0022	16
D0FIFO port select register_1	D0FIFOSEL_1	R/W	H'A4D9 0028	16
D0FIFO port control register_1	D0FIFOCTR_1	R/W	H'A4D9 002A	16
D1FIFO port select register_1	D1FIFOSEL_1	R/W	H'A4D9 002C	16
D1FIFO port control register_1	D1FIFOCTR_1	R/W	H'A4D9 002E	16
Interrupt enable register 0_1	INTENB0_1	R/W	H'A4D9 0030	16
Interrupt enable register 1_1	INTENB1_1	R/W	H'A4D9 0032	16
BRDY interrupt enable register_1	BRDYENB_1	R/W	H'A4D9 0036	16
NRDY interrupt enable register_1	NRDYENB_1	R/W	H'A4D9 0038	16
BEMP interrupt enable register_1	BEMPENB_1	R/W	H'A4D9 003A	16
SOF output configuration register_1	SOFCFG_1	R/W	H'A4D9 003C	16
Interrupt status register 0_1	INTSTS0_1	R/W	H'A4D9 0040	16
Interrupt status register 1_1	INTSTS1_1	R/W	H'A4D9 0042	16
BRDY interrupt status register_1	BRDYSTS_1	R/W	H'A4D9 0046	16
NRDY interrupt status register_1	NRDYSTS_1	R/W	H'A4D9 0048	16
BEMP interrupt status register_1	BEMPSTS_1	R/W	H'A4D9 004A	16
Frame number register_1	FRMNUM_1	R/W	H'A4D9 004C	16
µFrame number register_1	UFRMNUM_1	R/W	H'A4D9 004E	16
USB address register_1	USBADDR_1	R	H'A4D9 0050	16
USB request type register_1	USBREQ_1	R/W	H'A4D9 0054	16
USB request value register_1	USBVAL_1	R/W	H'A4D9 0056	16

Register Name	Abbreviation	R/W	Address	Access Size
USB request index register_1	USBINDX_1	R/W	H'A4D9 0058	16
USB request length register_1	USBLENG_1	R/W	H'A4D9 005A	16
DCP configuration register_1	DCPCFG_1	R/W	H'A4D9 005C	16
DCP maximum packet size register_1	DCPMAXP_1	R/W	H'A4D9 005E	16
DCP control register_1	DCPCTR_1	R/W	H'A4D9 0060	16
Pipe window select register_1	PIPESEL_1	R/W	H'A4D9 0064	16
Pipe configuration register_1	PIPECFG_1	R/W	H'A4D9 0068	16
Pipe buffer setting register_1	PIPEBUF_1	R/W	H'A4D9 006A	16
Pipe maximum packet size register_1	PIPEMAXP_1	R/W	H'A4D9 006C	16
Pipe cycle control register_1	PIPEPERI_1	R/W	H'A4D9 006E	16
Pipe 1 control register_1	PIPE1CTR_1	R/W	H'A4D9 0070	16
Pipe 2 control register_1	PIPE2CTR_1	R/W	H'A4D9 0072	16
Pipe 3 control register_1	PIPE3CTR_1	R/W	H'A4D9 0074	16
Pipe 4 control register_1	PIPE4CTR_1	R/W	H'A4D9 0076	16
Pipe 5 control register_1	PIPE5CTR_1	R/W	H'A4D9 0078	16
Pipe 6 control register_1	PIPE6CTR_1	R/W	H'A4D9 007A	16
Pipe 7 control register_1	PIPE7CTR_1	R/W	H'A4D9 007C	16
Pipe 8 control register_1	PIPE8CTR_1	R/W	H'A4D9 007E	16
Pipe 9 control register_1	PIPE9CTR_1	R/W	H'A4D9 0080	16
Pipe 1 transaction counter enable register_1	PIPE1TRE_1	R/W	H'A4D9 0090	16
Pipe 1 transaction counter register_1	PIPE1TRN_1	R/W	H'A4D9 0092	16
Pipe 2 transaction counter enable register_1	PIPE2TRE_1	R/W	H'A4D9 0094	16
Pipe 2 transaction counter register_1	PIPE2TRN_1	R/W	H'A4D9 0096	16
Pipe 3 transaction counter enable register_1	PIPE3TRE_1	R/W	H'A4D9 0098	16
Pipe 3 transaction counter register_1	PIPE3TRN_1	R/W	H'A4D9 009A	16
Pipe 4 transaction counter enable register_1	PIPE4TRE_1	R/W	H'A4D9 009C	16
Pipe 4 transaction counter register_1	PIPE4TRN_1	R/W	H'A4D9 009E	16
Pipe 5 transaction counter enable register_1	PIPE5TRE_1	R/W	H'A4D9 00A0	16

Register Name	Abbreviation	R/W	Address	Access Size
Pipe 5 transaction counter register_1	PIPE5TRN_1	R/W	H'A4D9 00A2	16
Device address 0 configuration register_1	DEVADD0_1	R/W	H'A4D9 00D0	16
Device address 1 configuration register_1	DEVADD1_1	R/W	H'A4D9 00D2	16
Device address 2 configuration register_1	DEVADD2_1	R/W	H'A4D9 00D4	16
Device address 3 configuration register_1	DEVADD3_1	R/W	H'A4D9 00D6	16
Device address 4 configuration register_1	DEVADD4_1	R/W	H'A4D9 00D8	16
Device address 5 configuration register_1	DEVADD5_1	R/W	H'A4D9 00DA	16
Device address 6 configuration register_1	DEVADD6_1	R/W	H'A4D9 00DC	16
Device address 7 configuration register_1	DEVADD7_1	R/W	H'A4D9 00DE	16
Device address 8 configuration register_1	DEVADD8_1	R/W	H'A4D9 00E0	16
Device address 9 configuration register_1	DEVADD9_1	R/W	H'A4D9 00E2	16
Device address A configuration register_1	DEVADDA_1	R/W	H'A4D9 00E4	16
USB power control register 1	UPONCR1	R/W	H’A405 0192	16

Table 31.5 Register States in Each Operation Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	U-standby	R-Standby	Sleep
SYSCFG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BUSWAIT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SYSSTS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DVSTCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
TESTMODE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CFIFO	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
D0FIFO	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
D1FIFO	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CFIFOSEL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CFIFOCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
D0FIFOSEL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
D0FIFOCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
D1FIFOSEL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
D1FIFOCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
INTENB0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
INTENB1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BRDYENB	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
NRDYENB	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BEMPENB	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SOFCFG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
INTSTS0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
INTSTS1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BRDYSTS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
NRDYSTS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BEMPSTS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
FRMNUM	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
UFRMNUM	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	U-standby	R-Standby	Sleep
USBADDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
USBREQ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
USBVAL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
USBINDX	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
USBLENG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DCPCFG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DCPMAXP	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DCPCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPESEL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPECFG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPEBUF	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPEMAXP	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPEPERI	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE1CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE2CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE3CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE4CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE5CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE6CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE7CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE8CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE9CTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE1TRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE1TRN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE2TRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE2TRN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE3TRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE3TRN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE4TRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE4TRN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	U-standby	R-Stanby	Sleep
PIPE5TRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
PIPE5TRN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD6	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD7	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD8	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADD9	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
DEVADDA	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
UPONCR0	Initialized	Retained	Retained	Retained	Initialized	Retained	Retained
UPONCR1	Initialized	Retained	Retained	Retained	Initialized	Retained	Retained

31.4.1 System Configuration Control Register (SYSCFG)

SYSCFG is a register that enables high-speed operation, selects the host controller function or function controller function, controls the DP and DM pins, and enables operation of this module.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCKE	—	—	HSE	DCFM	DRPD	DPRPU	—	—	—	USBE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	SCKE	0	R/W	USB Module Clock Enable Stops or enables supplying 48-MHz clock signal to this module. 0: Stops supplying the clock signal to the USB module. 1: Enables supplying the clock signal to the USB module. When this bit is 0, only this register and the BUSWAIT register allow both writing and reading; the other registers in the USB module allows reading only.
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7	HSE	0	R/W	High-Speed Operation Enable 0: High-speed operation is disabled When the function controller function is selected: Only full-speed operation is enabled. When the host controller function is selected: Full-speed or low-speed operation is enabled. 1: High-speed operation is enabled (detected by this module) (1) When the host controller function is selected When HSE = 0, the USB port performs low-speed or full-speed operation. Set HSE to 0 when connection of a low-speed peripheral device to the USB port has been detected. When HSE = 1, this module executes the reset handshake protocol, and automatically allows the USB port to perform high-speed or full-speed operation according to the protocol execution result. This bit should be modified after detecting device connection (after detecting the ATTCH interrupt) and before executing a USB bus reset (before setting USBRESET to 1). (2) When the function controller function is selected When HSE = 0, this module performs full-speed operation. When HSE = 1, this module executes the reset handshake protocol, and automatically performs high-speed or full-speed operation according to the protocol execution result. This bit should be modified while DPRPU is 0.
6	DCFM	0	R/W	Controller Function Select Selects the host controller function or function controller function. 0: Function controller function is selected. 1: Host controller function is selected. This bit should be modified while DPRPU and DPRD are 0.

Bit	Bit Name	Initial Value	R/W	Description
5	DRPD	0	R/W	D+/D− Line Resistor Control Enables or disables pulling down D+ and D- lines when the host controller function is selected. 0: Pulling down the lines is disabled. 1: Pulling down the lines is enabled. This bit should be set to 1 if the host controller function is selected, and should be set to 0 if the function controller function is selected.
4	DPRPU	0	R/W	D+ Line Resistor Control Enables or disables pulling up D+ line when the function controller function is selected. 0: Pulling up the line is disabled. 1: Pulling up the line is enabled. Setting this bit to 1 when the function controller function is selected allows this module to pull up the D+ line to 3.3 V, thus notifying the USB host of connection. Modifying this bit from 1 to 0 allows this module to cancel pulling up the D+ line, thus notifying the USB host of disconnection. This bit should be set to 1 if the function controller function is selected, and should be set to 0 if the host controller function is selected.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	USBE	0	R/W	USB Module Operation Enable Enables or disables operation of this module. 0: USB module operation is disabled. 1: USB module operation is enabled. Modifying this bit from 1 to 0 initializes some register bits as listed in tables 31.6 and 31.7. This bit should be modified while SCKE is 1. When the host controller function is selected, this bit should be set to 1 after setting DPRD to 1, eliminating LNST bit chattering, and checking that the USB bus has been settled.

Table 31.6 Register Bits Initialized by Writing USBE = 0 (when Function Controller Function is Selected)

Register Name	Bit Name	Remarks
SYSSTS	LNST	The value is retained when the host controller function is selected.
DVSTCTR	RHST	
INTSTS0	DVSQ	The value is retained when the host controller function is selected.
USBADDR	USBADDR	The value is retained when the host controller function is selected.
USBREQ	BRequest, bmRequestType	The values are retained when the host controller function is selected.
USBVAL	wValue	The value is retained when the host controller function is selected.
USBINDX	wIndex	The value is retained when the host controller function is selected.
USBLENG	wLength	The value is retained when the host controller function is selected.

Table 31.7 Register Bits Initialized by Writing USBE = 0 (when Host Controller Function is Selected)

Register Name	Bit Name	Remarks
DVSTCTR	RHST	
FRMNUM	FRNM	The value is retained when the function controller function is selected.
UFRMNUM	UFRNM	The value is retained when the function controller function is selected.

31.4.2 CPU Bus Wait Setting Register (BUSWAIT)

BUSWAIT is a register that specifies the number of wait cycles to be inserted during an access from the CPU to this module. Although this register is initialized to H'000F, it should be set to H'0005 or more using the initialization routine so that it retains H'0005 or more while this module is operating.

This register can be modified even when the SCKE bit in SYSCFG is 0.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	BWAIT[3:0]			
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	BWAIT[3:0]	1111	R/W	CPU Bus Wait Specifies the number of wait cycles to be inserted during an access to this module. 0000: 0 wait cycle (2 access cycles) : : 0010: 2 wait cycle (4 access cycles) : : 0101: 5 wait cycle (7 access cycles) : : 1111: 15 wait cycles (access cycles) (initial value) Note: Be sure to set these bits to B'0101 or more, not to set less than B'0101 (B'0000 to B'0100) using the initialization routine.

31.4.3 System Configuration Status Register (SYSSTS)

SYSSTS is a register that monitors the line status (D + and D – lines) of the USB data bus.

This register is initialized by a power-on reset or a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LNST[1:0]	
Initial value :	0	0	0	0	0	1	0	0	0	0	0	0	0	0	—	—
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
9 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	LNST[1:0]	Undefined*	R	USB Data Line Status Monitor Indicates the status of the USB data bus lines (D+ and D-) as shown in table 31.8. These bits should be read after setting DPRPU to 1 to notify connection when the function controller function is selected; whereas after setting DRPD to 1 to enable pulling down the lines when the host controller function is selected.

Note: * Depends on the DP and DM pin status.

Table 31.8 USB Data Bus Line Status

LNST[1]	LNST[0]	During Low-Speed Operation (only when Host Controller Function is Selected)	During Full-Speed Operation	During High-Speed Operation	During Chirp Operation
0	0	SE0	SE0	Squelch	Squelch
0	1	K state	J state	Not squelch	Chirp J
1	0	J state	K state	Invalid	Chirp K
1	1	SE1	SE1	Invalid	Invalid

[Legend]

Chirp: The reset handshake protocol is being executed in high-speed operation enabled state (the HSE bit in SYSCFG is set to 1).

Squelch: SE0 or idle state

Not squelch: High-speed J state or high-speed K state

Chirp J: Chirp J state

Chirp K: Chirp K state

31.4.4 Device State Control Register (DVSTCTR)

DVSTCTR is a register that controls and confirms the state of the USB data bus.

This register is initialized by a power-on reset. After a USB bus reset, WKUP is initialized but RESUME is undefined.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WKUP	RWUPE	USBRST	RESUME	UACT	—	RHST[2:0]		
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R/W*	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	WKUP	0	R/W	Wakeup Output Enables or disables outputting the remote wakeup signal (resume signal) to the USB bus when the function controller function is selected. 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output. The module controls the output time of a remote wakeup signal. When this bit is set to 1, this module clears this bit to 0 after outputting the 10-ms K state. According to the USB specification, the USB bus idle state must be kept for 5 ms or longer before a remote wakeup signal is output. If this module writes 1 to this bit right after detection of suspended state, the K state will be output after 2 ms. Note: Do not write 1 to this bit, unless the device state is in the suspended state (the DVSQ bit in the INTSTS0 register is set to 1xx) and the USB host enables the remote wakeup signal. When this bit is set to 1, the internal clock must not be stopped even in the suspended state (write 1 to this bit while SCKE is 1). This bit should be set to 0 if the host controller function is selected.

Bit	Bit Name	Initial Value	R/W	Description
7	RWUPE	0	R/W	Wakeup Detection Enable Enables or disables the downstream port peripheral device to use the remote wakeup function (resume signal output) when the host controller function is selected. 0: Downstream port wakeup is disabled. 1: Downstream port wakeup is enabled. With this bit set to 1, on detecting the remote wakeup signal, this module detects the resume signal (K-state for 2.5 μs) from the downstream port device and performs the resume process (drives the port to the K-state). With this bit set to 0, this module ignores the detected remote wakeup signal (K-state) from the peripheral device connected to the downstream port. While this bit is 1, the internal clock should not be stopped even in the suspended state (SCKE should be set to 1). Also note that the USB bus should not be reset from the suspended state (USBRST should not be set to 1); it is prohibited by USB Specification 2.0. This bit should be set to 0 if the function controller function is selected.

Bit	Bit Name	Initial Value	R/W	Description
6	USBRST	0	R/W	Bus Reset Output Controls the USB bus reset signal output when the host controller function is selected. 0: USB bus reset signal is not output. 1: USB bus reset signal is output. When the host controller function is selected, setting this bit to 1 allows this module to drive the USB port to SE0 to reset the USB bus. Here, this module performs the reset handshake protocol if the HSE bit is 1. This module continues outputting SE0 while USBRST is 1 (until software sets USBRST to 0). USBRST should be 1 (= USB bus reset period) for the time defined by USB Specification 2.0. Writing 1 to this bit during communication (UACT = 1) or during the resume process (RESUME = 1) prevents this module from starting the USB bus reset process until both UACT and RESUME become 0. Write 1 to the UACT bit simultaneously with the end of the USB bus reset process (writing 0 to USBRST). This bit should be set to 0 if the function controller function is selected.
5	RESUME	0	R/W	Resume Output Controls the resume signal output when the host controller function is selected. 0: Resume signal is not output. 1: Resume signal is output. Setting this bit to 1 allows this module to drive the port to the K-state and output the resume signal. This module continues outputting K-state while RESUME is 1 (until software sets RESUME to 0). RESUME should be 1 (= resume period) for the time defined by USB Specification 2.0. This bit should be set to 1 in the suspended state. Write 1 to the UACT bit simultaneously with the end of the resume process (writing 0 to RESUME). This bit should be set to 0 if the function controller function is selected.

Bit	Bit Name	Initial Value	R/W	Description
4	UACT	0	R/W	USB Bus Enable Enables operation of the USB bus (controls the SOF or μSOF packet transmission to the USB bus) when the host controller function is selected. 0: Downstream port is disabled (SOF/μSOF transmission is disabled). 1: Downstream port is enabled (SOF/μSOF transmission is enabled). With this bit set to 1, this module puts the USB port to the USB-bus enabled state and performs SOF output and data transmission and reception. This module starts outputting SOF/μSOF within 1 (μ) frame after software has written 1 to UACT. With this bit set to 0, this module enters the idle state after outputting SOF/μSOF. This module sets this bit to 0 on any of the following conditions. • A DTCH interrupt is detected during communication (while UACT = 1). • An EOFERR interrupt is detected during communication (while UACT = 1). Writing 1 to this bit should be done at the end of the USB reset process (writing 0 to USBRST) or at the end of the resume process from the suspended state (writing 0 to RESUME). This bit should be set to 0 if the function controller function is selected.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	RHST[2:0]	000	R	Reset Handshake Indicates the status of the reset handshake. (1) When the host controller function is selected 000: Communication speed not determined (powered state or no connection) 1xx: Reset handshake in progress 001: Low-speed connection 010: Full-speed connection 011: High-speed connection These bits indicate 100 after software has written 1 to USBRST. If HSE has been set to 1, these bits indicate 111 as soon as this module detects Chirp-K from the peripheral device. This module fixes the value of the RHST bits when software writes 0 to USBRST and this module completes SE0 driving. (2) When the function controller function is selected 000: Communication speed not determined 100: Reset handshake in progress 010: Full-speed connection 011: High-speed connection If HSE has been set to 1, these bits indicate 100 as soon as this module detects the USB bus reset. Then, these bits indicate 011 as soon as this module outputs Chirp-K and detects Chirp-JK from the USB host three times. If the connection speed is not fixed to high speed within 2.5 ms after Chirp-K output, these bits indicate 010. If HSE has been set to 0, these bits indicate 010 as soon as this module detects the USB bus reset. A DVST interrupt is generated as soon as this module detects the USB bus reset and then the value of the RHST bits is fixed to 010 or 011.

Note: * Only 1 can be written.

31.4.5 Test Mode Register (TESTMODE)

TESTMODE is a register that controls the USB test signal output during high-speed operation.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST[3:0]			
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	UTST[3:0]	0000	R/W	Test Mode This module outputs the USB test signals during the high-speed operation, when these bits are written appropriate value. Table 31.9 shows test mode operation of this module.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	UTST[3:0]	0000	R/W	(1) When the host controller function is selected These bits can be set after writing 1 to DRPD. This module outputs waveforms to the USB port for which both DPRD and UACT have been set to 1. This module also performs high-speed termination for the USB port. • Procedure for setting the UTST bits 1. Power-on reset. 2. Start the clock supply (Set SCKE to 1 after the crystal oscillation and the PLL for USB are settled). 3. Set DCFM and DPRD to 1 (setting HSE to 1 is not required). 4. Set USBE to 1. 5. Set the UTST bits to the appropriate value according to the test specifications. 6. Set the UACT bit to 1. • Procedure for modifying the UTST bits 1. (In the state after executing step 6 above) Set UACT and USBE to 0. 2. Set USBE to 1. 3. Set the UTST bits to the appropriate value according to the test specifications. 4. Set the UACT bit to 1. When these bits are set to Test_SE0_NAK (1011), this module does not output the SOF packet to the port even when 1 has been set to UACT for the port. When these bits are set to Test_Force_Enable (1101), this module outputs the SOF packet to the port for which 1 has been set to UACT. In this test mode, this module does not perform hardware control consequent to detection of high-speed disconnection (detection of the DTCH interrupt). When setting the UTST bits, the PID bits for all the pipes should be set to NAK. To return to normal USB communication after a test mode has been set and executed, a power-on reset should be applied.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	UTST[3:0]	0000	R/W	(2) When the function controller function is selected The appropriate value should be set to these bits according to the SetFeature request from the USB host during high-speed communication. This module does not make a transition to the suspended state while these bits are 0001 to 0100.

Table 31.9 Test Mode Operation

	UTST Bit Setting	
Test Mode	**When Function Controller Function is Selected**	**When Host Controller Function is Selected**
Normal operation	0000	0000
Test_J	0001	1001
Test_K	0010	1010
Test_SE0_NAK	0011	1011
Test_Packet	0100	1100
Test_Force_Enable	—	1101
Reserved	0101 to 0111	1110 to 1111

31.4.6 FIFO Port Registers (CFIFO, D0FIFO, D1FIFO)

CFIFO, D0FIFO and D1FIFO are port registers that are used to read data from the FIFO buffer memory and writing data to the FIFO buffer memory.

There are three FIFO ports: the CFIFO, D0FIFO and D1FIFO ports. Each FIFO port is configured of a port register (CFIFO, D0FIFO, D1FIFO) that handles reading of data from the FIFO buffer memory and writing of data to the FIFO buffer memory, a select register (CFIFOSEL, D0FIFOSEL, D1FIFOSEL) that is used to select the pipe assigned to the FIFO port, and a control register (CFIFOCTR, D0FIFOCTR, D1FIFOCTR).

Each FIFO port has the following features.

- The DCP FIFO buffer should be accessed through the CFIFO port.
- Accessing the FIFO buffer using DMA transfer should be performed through the D0FIFO or D1FIFO port.
- The D1FIFO and D0FIFO ports can be accessed also by the CPU.
- When using functions specific to the FIFO port, the pipe number (selected pipe) specified by the CURPIPE bits cannot be changed (when the DMA transfer function is used, etc.).
- Registers configuring a FIFO port do not affect other FIFO ports.
- The same pipe should not be assigned to two or more FIFO ports.
- There are two FIFO buffer states: the access right is on the CPU side and it is on the SIE side. When the FIFO buffer access right is on the SIE side, the FIFO buffer cannot be accessed from the CPU.

These registers are initialized by a power-on reset.

Bit :	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFOPORT[31:16]															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFOPORT[15:0]															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT [31:0]	All 0	R/W	FIFO Port Accessing these bits allow reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer. These bits can be accessed only while the FRDY bit in each control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR) is 1. The valid bits in this register depend on the settings of the MBW bits (access bit width setting) and BIGEND bit (endian setting) as shown in tables 31.10 to 31.12.

Table 31.10 Endian Operation in 32-Bit Access (when MBW = 10)

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	N + 3 address	N + 2 address	N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	N + 2 address	N + 3 address

Table 31.11 Endian Operation in 16-Bit Access (when MBW = 01)

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	Writing: invalid, reading: prohibited*		N + 1 address	N + 0 address
1	N + 0 address	N + 1 address	Writing: invalid, reading: prohibited*	

Note: * Reading data from the invalid bits in a word or byte unit is prohibited.

Table 31.12 Endian Operation in 8-Bit Access (when MBW = 00)

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	Writing: invalid, reading: prohibited*			N + 0 address
1	N + 0 address	Writing: invalid, reading: prohibited*		

Note: * Reading data from the invalid bits in a word or byte unit is prohibited.

Note that as address mapping selects big endian in this module, in little endian mode FIFO has to be accessed shifting the address as follows.

Define as FPA = FIFO port address

32-bit Access

Access bit	D31 to D0
Access address	FPA + 0

When the number of access byte is not equal to a multiple of 4, follow the instruction listed below.

- A multiple of 4 + 1:
  The last one byte access is a byte (8-bit) access to FPA + 3
- A multiple of 4 + 2;
  The last two bytes access are a word (16-bit) access to FPA + 2
- A multiple of 4 + 3;
  The last three bytes access are a word (16-bit) access to FPA + 2 and a byte (8-bit) access to FPA + 1

16-bit access

Access bit	D31 to D16	D15 to D0
Access address	Writing: invalid Reading: prohibited	FPA + 2

When the number of access bytes is equal to an odd number, the last one byte access is a byte (8-bit) access to FPA + 3.

8-bit access

Access bit	D31 to D24, D23 to D16, D15 to D8	D7 to D0
Access address	Writing: invalid Reading: prohibited	FPA + 3

31.4.7 FIFO Port Select Registers (CFIFOSEL, D0FIFOSEL, D1FIFOSEL)

CFIFOSEL, D0FIFOSEL and D1FIFOSEL are registers that assign the pipe to the FIFO port, and control access to the corresponding port.

The same pipe should not be specified by the CURPIPE bits in CFIFOSEL, D0FIFOSEL and D1FIFOSEL. When the CURPIPE bits in D0FIFOSEL and D1FIFOSEL are cleared to B'000, no pipe is selected.

The pipe number should not be changed while the DMA transfer is enabled.

These registers are initialized by a power-on reset.

(1) CFIFOSEL

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW[1:0]		—	BIGEND	—	—	ISEL	—	CURPIPE[3:0]			
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W*	R	R	R/W	R/W	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read Count Mode Specifies the read mode for the value in the DTLN bits in CFIFOCTR. 0: The DTLN bit is cleared when all of the receive data has been read from the CFIFO. (In double buffer mode, the DTLN bit value is cleared when all the data has been read from a single plane.) 1: The DTLN bit is decremented when the receive data is read from the CFIFO.

Bit	Bit Name	Initial Value	R/W	Description
14	REW	0	R/W*	Buffer Pointer Rewind Specifies whether or not to rewind the buffer pointer. 0: The buffer pointer is not rewound. 1: The buffer pointer is rewound. When the selected pipe is in the receiving direction, setting this bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double buffer mode, re-reading the currently-read FIFO buffer plane from the first data is allowed). Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1. To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11, 10	MBW[1:0]	00	R/W	CFIFO Port Access Bit Width Specifies the bit width for accessing the CFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited When the selected pipe is in the receiving direction, once reading data is started after setting these bits, these bits should not be modified until all the data has been read. When the selected pipe is in the receiving direction, set the CURPIPE and MBW bits simultaneously. When the selected pipe is in the transmitting direction, the bit width cannot be changed from the 8-bit width to the 16-/32-bit width or from the 16-bit width to the 32-bit width while data is being written to the buffer memory. The odd number of bytes can also be written through byte-access control even when 8- or 16-bit width is selected.

Bit	Bit Name	Initial Value	R/W	Description
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	BIGEND	0	R/W	CFIFO Port Endian Control Specifies the byte endian for the CFIFO port. 0: Little endian 1: Big endian
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	ISEL	0	R/W	CFIFO Port Access Direction When DCP is Selected 0: Reading from the buffer memory is selected 1: Writing to the buffer memory is selected After writing to this bit with the DCP being a selected pipe, read this bit to check that the written value agrees with the read value before proceeding to the next process. Even if an attempt is made to modify the setting of this bit during access to the FIFO buffer, the current access setting is retained until the access is completed. Then, the modification becomes effective thus enabling continuous access. Set this bit and the CURPIPE bits simultaneously.
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CURPIPE[3:0]	0000	R/W	CFIFO Port Access Pipe Specification Specifies the pipe number using which data is read or written through the CFIFO port. 0000: DCP 0001: Pipe 1 0010: Pipe 2 0011: Pipe 3 0100: Pipe 4 0101: Pipe 5 0110: Pipe 6 0111: Pipe 7 1000: Pipe 8 1001: Pipe 9 Other than above: Setting prohibited After writing to these bits, read these bits to check that the written value agrees with the read value before proceeding to the next process. Do not set the same pipe number to the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL. Even if an attempt is made to modify the setting of these bits during access to the FIFO buffer, the current access setting is retained until the access is completed. Then, the modification becomes effective thus enabling continuous access.

Note: * Only 0 can be read.

(2) D0FIFOSEL, D1FIFOSEL

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW[1:0]		—	BIG END	—	—	—	—	CURPIPE[3:0]			
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W*	R/W	R/W	R/W	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	Read Count Mode Specifies the read mode for the value in the DTLN bits in DnFIFOCTR. 0: The DTLN bit is cleared when all of the receive data has been read from the DnFIFO. (In double buffer mode, the DTLN bit value is cleared when all the data has been read from a single plane.) 1: The DTLN bit is decremented when the receive data is read from the DnFIFO. When accessing DnFIFO with the BFRE bit set to 1, set this bit to 0.
14	REW	0	R/W*	Buffer Pointer Rewind Specifies whether or not to rewind the buffer pointer. 0: The buffer pointer is not rewound. 1: The buffer pointer is rewound. When the selected pipe is in the receiving direction, setting this bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double buffer mode, re-reading the currently-read FIFO buffer plane from the first data is allowed). Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1. When accessing DnFIFO with the BFRE bit set to 1, do not set this bit to 1 in the state in which the short packet data has been read out. To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.

Bit	Bit Name	Initial Value	R/W	Description
13	DCLRM	0	R/W	Auto Buffer Memory Clear Mode Accessed after Specified Pipe Data is Read Enables or disables the buffer memory to be cleared automatically after data has been read out using the selected pipe. 0: Auto buffer clear mode is disabled. 1: Auto buffer clear mode is enabled. With this bit set to 1, this module sets BCLR to 1 for the FIFO buffer of the selected pipe on receiving a zero-length packet while the FIFO buffer assigned to the selected pipe is empty, or on receiving a short packet and reading the data while BFRE is 1. When using this module with the BRDYM bit set to 1, set this bit to 0.
12	DREQE	0	R/W	DMA Transfer Request Enable Enables or disables the DMA transfer request to be issued. 0: Request disabled 1: Request enabled Before setting this bit to 1 to enable the DMA transfer request to be issued, set the CURPIPE bits. Before modifying the CURPIPE bit setting, set this bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
11, 10	MBW[1:0]	All 0	R/W	FIFO Port Access Bit Width Specifies the bit width for accessing the DnFIFO port. 00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited When the selected pipe is in the receiving direction, once reading data is started after setting these bits, these bits should not be modified until all the data has been read. When the selected pipe is in the receiving direction, set the CURPIPE and MBW bits simultaneously. When the selected pipe is in the transmitting direction, the bit width cannot be changed from the 8-bit width to the 16-/32-bit width or from the 16-bit width to the 32-bit width while data is being written to the buffer memory. The odd number of bytes can be written through byte-access control even when 8- or 16-bit width is selected.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	BIGEND	0	R/W	FIFO Port Endian Control Specifies the byte endian for the DnFIFO port. 0: Little endian 1: Big endian
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	CURPIPE[3:0]	0000	R/W	FIFO Port Access Pipe Specification Specifies the pipe number using which data is read or written through the D0FIFO/D1FIFO port. 0000: No pipe specified 0001: Pipe 1 0010: Pipe 2 0011: Pipe 3 0100: Pipe 4 0101: Pipe 5 0110: Pipe 6 0111: Pipe 7 1000: Pipe 8 1001: Pipe 9 Other than above: Setting prohibited After writing to these bits, read these bits to check that the written value agrees with the read value before proceeding to the next process. Do not set the same pipe number to the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL. Even if an attempt is made to modify the setting of these bits during access to the FIFO buffer, the current access setting is retained until the access is completed. Then, the modification becomes effective thus enabling continuous access.

Note: * Only 0 can be read.

31.4.8 FIFO Port Control Registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR)

CFIFOCTR, D0FIFOCTR and D1FIFOCTR are registers that determine whether or not writing to the buffer memory has been finished, the buffer accessed from the CPU has been cleared, and the FIFO port is accessible. CFIFOCTR, D0FIFOCTR, and D1FIFOCTR are used for the corresponding FIFO ports.

These registers are initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—	DTLN[11:0]											
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W*2	R/W*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	BVAL	0	R/W*2	Buffer Memory Valid Flag This bit should be set to 1 when data has been completely written to the FIFO buffer on the CPU side for the pipe selected using the CURPIPE bits (selected pipe). 0: Invalid 1: Writing ended When the selected pipe is in the transmitting direction, set this bit to 1 in the following cases. Then, this module switches the FIFO buffer from the CPU side to the SIE side, enabling transmission. • To transmit a short packet, set this bit to 1 after data has been written. • To transmit a zero-length packet, set this bit to 1 before data is written to the FIFO buffer. • Set this bit to 1 after the number of data bytes has been written for the pipe in continuous transfer mode, where the number is a natural integer multiple of the maximum packet size and less than the buffer size. When the data of the maximum packet size has been written for the pipe in continuous transfer mode, this module sets this bit to 1 and switches the FIFO buffer from the CPU side to the SIE side, enabling transmission. Writing 1 to this bit should be done while FRDY indicates 1 (set by this module). When the selected pipe is in the receiving direction, do not set this bit to 1.

Bit	Bit Name	Initial Value	R/W	Description
14	BCLR	0	R/W*[1]	CPU Buffer Clear This bit should be set to 1 to clear the FIFO buffer on the CPU side for the selected pipe. 0: Invalid 1: Clears the buffer memory on the CPU side. When double buffer mode is set for the FIFO buffer assigned to the selected pipe, this module clears only one plane of the FIFO buffer even when both planes are read-enabled. When the selected pipe is the DCP, setting BCLR to 1 allows this module to clear the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or SIE side. To clear the buffer on the SIE side, set the PID bits for the DCP to NAK before setting BCLR to 1. When the selected pipe is in the transmitting direction, if 1 is written to BVAL and BCLR bits simultaneously, this module clears the data that has been written before it, enabling transmission of a zero-length packet. When the selected pipe is not the DCP, writing 1 to this bit should be done while FRDY indicates 1 (set by this module).
13	FRDY	0	R	FIFO Port Ready Indicates whether the FIFO port can be accessed by the CPU (DMAC). 0: FIFO port access is disabled. 1: FIFO port access is enabled. In the following cases, this module sets FRDY to 1 but data cannot be read via the FIFO port because there is no data to be read. In these cases, set BCLR to 1 to clear the FIFO buffer, and enable transmission and reception of the next data. • A zero-length packet is received when the FIFO buffer assigned to the selected pipe is empty. • A short packet is received and the data is completely read while BFRE is 1.

Bit	Bit Name	Initial Value	R/W	Description
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11 to 0	DTLN[11:0]	H'000	R	Receive Data Length Indicates the length of the receive data. While the FIFO buffer is being read, these bits indicate the different values depending on the RCNT bit value as described below. • RCNT = 0: This module sets these bits to indicate the length of the receive data until the CPU (DMAC) has read all the received data from a single FIFO buffer plane. While BFRE is 1, these bits retain the length of the receive data until BCLR is set to 1 even after all the data has been read. • RCNT = 1: This module decrements the value indicated by these bits each time data is read from the FIFO buffer. (The value is decremented by one when MBW is 0, and by two when MBW is 1.) This module sets these bits to 0 when all the data has been read from one FIFO buffer plane. However, in double buffer mode, if data has been received in one FIFO buffer plane before all the data has been read from the other plane, this module sets these bits to indicate the length of the receive data in the former plane when all the data has been read from the latter plane. When RCNT is 1, reading these bits while the FIFO buffer is being read returns the latest value within 150 ns after the FIFO port read cycle.

Notes: 1. Only 0 can be read and 1 can be written to.
2. Only 1 can be written to.

31.4.9 Interrupts Enable Register 0 (INTENB0)

INTENB0 is a register that specifies the various interrupt masks. On detecting the interrupt corresponding to the bit in this register to which software has set 1, this module generates the USB interrupt.

This module sets 1 to each status bit in INTSTS0 when a detection condition of the corresponding interrupt source has been satisfied regardless of the set value in INTENB0 (regardless of whether the interrupt output is enabled or disabled).

While the status bit in INTSTS0 corresponding to the interrupt source indicates 1, this module generates the USB interrupt when software modifies the corresponding interrupt enable bit in INTENB0 from 0 to 1.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBSE	0	R/W	VBUS Interrupts Enable Enables or disables the USB interrupt output when the VBINT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
14	RSME	0	R/W	Resume Interrupts Enable Enables or disables the USB interrupt output when the RESM interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
13	SOFE	0	R/W	Frame Number Update Interrupts Enable Enables or disables the USB interrupt output when the SOFR interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
12	DVSE	0	R/W	Device State Transition Interrupts Enable* Enables or disables the USB interrupt output when the DVST interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
11	CTRE	0	R/W	Control Transfer Stage Transition Interrupts Enable* Enables or disables the USB interrupt output when the CTRT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
10	BEMPE	0	R/W	Buffer Empty Interrupts Enable Enables or disables the USB interrupt output when the BEMP interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
9	NRDYE	0	R/W	Buffer Not Ready Response Interrupts Enable Enables or disables the USB interrupt output when the NRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
8	BRDYE	0	R/W	Buffer Ready Interrupts Enable Enables or disables the USB interrupt output when the BRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * The RSME, DVSE, and CTRE bits can be set to 1 only when the function controller function is selected; do not set these bits to 1 to enable the corresponding interrupt output when the host controller function is selected.

31.4.10 Interrupt Enable Register 1 (INTENB1)

INTENB1 is a register that specifies the various interrupt masks when the host controller function is selected. On detecting the interrupt corresponding to the bit in this register to which software has set 1, this module generates the USB interrupt.

This module sets 1 to each status bit in INTSTS1 when a detection condition of the corresponding interrupt source has been satisfied regardless of the set value in INTENB1 (regardless of whether the interrupt output is enabled or disabled).

While the status bit in INTSTS1 corresponding to the interrupt source indicates 1, this module generates the USB interrupt when software modifies the corresponding interrupt enable bit in INTENB1 from 0 to 1.

When the function controller function is selected, the interrupts should not be enabled.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHGE	—	DTCHE	ATTCHE	—	—	—	—	EOFERRE	SIGNE	SACKE	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	BCHGE	0	R/W	USB Bus Change Interrupt Enable Enables or disables the USB interrupt output when the BCHG interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	DTCHE	0	R/W	Disconnection Detection Interrupt Enable Enables or disables the USB interrupt output when the DTCH interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
11	ATTCHE	0	R/W	Connection Detection Interrupt Enable Enables or disables the USB interrupt output when the ATTCH interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
10 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	EOFERRE	0	R/W	EOF Error Detection Interrupt Enable Enables or disables the USB interrupt output when the EOFERR interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
5	SIGNE	0	R/W	Setup Transaction Error Interrupt Enable Enables or disables the USB interrupt output when the SIGN interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
4	SACKE	0	R/W	Setup Transaction Normal Response Interrupt Enable Enables or disables the USB interrupt output when the SACK interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: The INTENB1 register bits can be set to 1 only when the host controller function is selected; do not set these bits to 1 to enable the corresponding interrupt output when the function controller function is selected.

31.4.11 BRDY Interrupt Enable Register (BRDYENB)

BRDYENB is a register that enables or disables the BRDY bit in INTSTS0 to be set to 1 when the BRDY interrupt is detected for each pipe.

On detecting the BRDY interrupt for the pipe corresponding to the bit in this register to which software has set 1, this module sets 1 to the corresponding PIPEBRDY bit in BRDYSTS and the BRDY bit in INTSTS0, and generates the BRDY interrupt.

While at least one PIPEBRDY bit in BRDYSTS indicates 1, this module generates the BRDY interrupt when software modifies the corresponding interrupt enable bit in BRDYENB from 0 to 1.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BRDYE	PIPE8 BRDYE	PIPE7 BRDYE	PIPE6 BRDYE	PIPE5 BRDYE	PIPE4 BRDYE	PIPE3 BRDYE	PIPE2 BRDYE	PIPE1 BRDYE	PIPE0 BRDYE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PIPE9BRDYE	0	R/W	BRDY interrupt Enable for PIPE9 0: Interrupt output disabled 1: Interrupt output enabled
8	PIPE8BRDYE	0	R/W	BRDY interrupt Enable for PIPE8 0: Interrupt output disabled 1: Interrupt output enabled
7	PIPE7BRDYE	0	R/W	BRDY interrupt Enable for PIPE7 0: Interrupt output disabled 1: Interrupt output enabled
6	PIPE6BRDYE	0	R/W	BRDY interrupt Enable for PIPE6 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
5	PIPE5BRDYE	0	R/W	BRDY interrupt Enable for PIPE5 0: Interrupt output disabled 1: Interrupt output enabled
4	PIPE4BRDYE	0	R/W	BRDY interrupt Enable for PIPE4 0: Interrupt output disabled 1: Interrupt output enabled
3	PIPE3BRDYE	0	R/W	BRDY interrupt Enable for PIPE3 0: Interrupt output disabled 1: Interrupt output enabled
2	PIPE2BRDYE	0	R/W	BRDY interrupt Enable for PIPE2 0: Interrupt output disabled 1: Interrupt output enabled
1	PIPE1BRDYE	0	R/W	BRDY interrupt Enable for PIPE1 0: Interrupt output disabled 1: Interrupt output enabled
0	PIPE0BRDYE	0	R/W	BRDY interrupt Enable for PIPE0 0: Interrupt output disabled 1: Interrupt output enabled

31.4.12 NRDY Interrupt Enable Register (NRDYENB)

NRDYENB is a register that enables or disables the NRDY bit in INTSTS0 to be set to 1 when the NRDY interrupt is detected for each pipe.

On detecting the NRDY interrupt for the pipe corresponding to the bit in this register to which software has set 1, this module sets 1 to the corresponding PIPENRDY bit in NRDYSTS and the NRDY bit in INTSTS0, and generates the NRDY interrupt.

While at least one PIPENRDY bit in NRDYSTS indicates 1, this module generates the NRDY interrupt when software modifies the corresponding interrupt enable bit in NRDYENB from 0 to 1.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 NRDYE	PIPE8 NRDYE	PIPE7 NRDYE	PIPE6 NRDYE	PIPE5 NRDYE	PIPE4 NRDYE	PIPE3 NRDYE	PIPE2 NRDYE	PIPE1 NRDYE	PIPE0 NRDYE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PIPE9NRDYE	0	R/W	NRDY Interrupt Enable for PIPE9 0: Interrupt output disabled 1: Interrupt output enabled
8	PIPE8NRDYE	0	R/W	NRDY Interrupt Enable for PIPE8 0: Interrupt output disabled 1: Interrupt output enabled
7	PIPE7NRDYE	0	R/W	NRDY Interrupt Enable for PIPE7 0: Interrupt output disabled 1: Interrupt output enabled
6	PIPE6NRDYE	0	R/W	NRDY Interrupt Enable for PIPE6 0: Interrupt output disabled 1: Interrupt output enabled
5	PIPE5NRDYE	0	R/W	NRDY Interrupt Enable for PIPE5 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
4	PIPE4NRDYE	0	R/W	NRDY Interrupt Enable for PIPE4 0: Interrupt output disabled 1: Interrupt output enabled
3	PIPE3NRDYE	0	R/W	NRDY Interrupt Enable for PIPE3 0: Interrupt output disabled 1: Interrupt output enabled
2	PIPE2NRDYE	0	R/W	NRDY Interrupt Enable for PIPE2 0: Interrupt output disabled 1: Interrupt output enabled
1	PIPE1NRDYE	0	R/W	NRDY Interrupt Enable for PIPE1 0: Interrupt output disabled 1: Interrupt output enabled
0	PIPE0NRDYE	0	R/W	NRDY Interrupt Enable for PIPE0 0: Interrupt output disabled 1: Interrupt output enabled

31.4.13 BEMP Interrupt Enable Register (BEMPENB)

BEMPENB is a register that enables or disables the BEMP bit in INTSTS0 to be set to 1 when the BEMP interrupt is detected for each pipe.

On detecting the BEMP interrupt for the pipe corresponding to the bit in this register to which software has set 1, this module sets 1 to the corresponding PIPEBEMP bit in BEMPSTS and the BEMP bit in INTSTS0, and generates the BEMP interrupt.

While at least one PIPEBEMP bit in BEMPSTS indicates 1, this module generates the BEMP interrupt when software modifies the corresponding interrupt enable bit in BEMPENB from 0 to 1.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BEMPE	PIPE8 BEMPE	PIPE7 BEMPE	PIPE6 BEMPE	PIPE5 BEMPE	PIPE4 BEMPE	PIPE3 BEMPE	PIPE2 BEMPE	PIPE1 BEMPE	PIPE0 BEMPE
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PIPE9BEMPE	0	R/W	BEMP Interrupt Enable for PIPE9 0: Interrupt output disabled 1: Interrupt output enabled
8	PIPE8BEMPE	0	R/W	BEMP Interrupt Enable for PIPE8 0: Interrupt output disabled 1: Interrupt output enabled
7	PIPE7BEMPE	0	R/W	BEMP Interrupt Enable for PIPE7 0: Interrupt output disabled 1: Interrupt output enabled
6	PIPE6BEMPE	0	R/W	BEMP Interrupt Enable for PIPE6 0: Interrupt output disabled 1: Interrupt output enabled
5	PIPE5BEMPE	0	R/W	BEMP Interrupt Enable for PIPE5 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
4	PIPE4BEMPE	0	R/W	BEMP Interrupt Enable for PIPE4 0: Interrupt output disabled 1: Interrupt output enabled
3	PIPE3BEMPE	0	R/W	BEMP Interrupt Enable for PIPE3 0: Interrupt output disabled 1: Interrupt output enabled
2	PIPE2BEMPE	0	R/W	BEMP Interrupt Enable for PIPE2 0: Interrupt output disabled 1: Interrupt output enabled
1	PIPE1BEMPE	0	R/W	BEMP Interrupt Enable for PIPE1 0: Interrupt output disabled 1: Interrupt output enabled
0	PIPE0BEMPE	0	R/W	BEMP Interrupt Enable for PIPE0 0: Interrupt output disabled 1: Interrupt output enabled

31.4.14 SOF Control Register (SOFCFG)

SOFCFG is a register that specifies the transaction-enabled time and BRDY interrupt status clear timing.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TRNENSEL	—	BRDYM	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0*	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TRNENSEL	0	R/W	Transaction-Enabled Time Select Selects the transaction-enabled time either for full- or low-speed communication, where is the time in which this module issues tokens in a frame via the port. 0: For non-low-speed communication 1: For low-speed communication This bit is valid only when the host controller function is selected. Even when the host controller function is selected, the setting of this bit has no effect on the transaction-enabled time during high-speed communication. This bit should be set to 0 when the function controller function is selected.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	BRDYM	0	R/W	BRDY Interrupt Status Clear Timing for each Pipe Specifies the timing for clearing the BRDY interrupt status for each pipe. 0: Software clears the status. 1: This module clears the status when data has been read from the FIFO buffer or data has been written to the FIFO buffer.

Bit	Bit Name	Initial Value	R/W	Description
5	—	0*	R	Reserved This bit is reserved. The previously read value should be written to this bit. Note: Although this bit is initialized to 0 by a power-on reset, be sure to set this bit to 1 using the initialization routine of this module.
4 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * Although this bit is initialized to 0 by a power-on reset, be sure to set this bit to 1 using the initialization routine of this module.

31.4.15 Interrupt Status Register 0 (INTSTS0)

INTSTS0 is a register that indicates the status of the various interrupts detected.

This register is initialized by a power-on reset. By a USB bus reset, the DVSQ2 to DVSQ0 bits are initialized.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
Initial value :	0	0	0	0	0	0	0	0	*3	*2	*2	*2	0	0	0	0
R/W :	R/W*7	R/W*7	R/W*7	R/W*7	R/W*7	R	R	R	R	R	R	R	R/W*7	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBINT	0	R/W*7	VBUS Interrupt Status*4*5 0: VBUS interrupts not generated 1: VBUS interrupts generated This module sets this bit to 1 on detecting a level change (high to low or low to high) in the VBUS pin input value. This module sets the VBSTS bit to indicate the VBUS pin input value. When the VBUS interrupt is generated, use software to repeat reading the VBSTS bit until the same value is read three or more times, and eliminate chattering.

Bit	Bit Name	Initial Value	R/W	Description
14	RESM	0	R/W*[7]	Resume Interrupt Status*[4]*[5]*[6] 0: Resume interrupts not generated 1: Resume interrupts generated When the function controller function is selected, this module sets this bit to 1 on detecting the falling edge of the signal on the DP pin in the suspended state (DVSQ = 1XX). When the host controller function is selected, the read value is invalid.
13	SOFR	0	R/W*[7]	Frame Number Refresh Interrupt Status*[4] 0: SOF interrupts not generated 1: SOF interrupts generated (1) When the host controller function is selected This module sets this bit to 1 on updating the frame number when software has set the UACT bit to 1. (This interrupt is detected every 1 ms.) (2) When the function controller function is selected This module sets this bit to 1 on updating the frame number. (This interrupt is detected every 1 ms.) This module can detect an SOFR interrupt through the internal interpolation function even when a damaged SOF packet is received from the USB host.
12	DVST	0/1*[1]	R/W*[7]	Device State Transition Interrupt Status*[4]*[6] 0: Device state transition interrupts not generated 1: Device state transition interrupts generated When the function controller function is selected, this module updates the DVSQ value and sets this bit to 1 on detecting a change in the device state. When this interrupt is generated, clear the status before this module detects the next device state transition. When the host controller function is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
11	CTRT	0	R/W*[7]	Control Transfer Stage Transition Interrupt Status*[4]*[6] 0: Control transfer stage transition interrupts not generated 1: Control transfer stage transition interrupts generated When the function controller function is selected, this module updates the CTSQ value and sets this bit to 1 on detecting a change in the control transfer stage. When this interrupt is generated, clear the status before this module detects the next control transfer stage transition. When the host controller function is selected, the read value is invalid.
10	BEMP	0	R	Buffer Empty Interrupt Status 0: BEMP interrupts not generated 1: BEMP interrupts generated This module sets this bit to 1 when at least one PIPEBEMP bit in BEMPSTS is set to 1 among the PIPEBEMP bits corresponding to the PIPEBEMPE bits in BEMPENB to which 1 has been set (when this module detects the BEMP interrupt status in at least one pipe among the pipes for which software enables the BEMP interrupt output). For the conditions for PIPEBEMP status assertion, refer to (3) BEMP Interrupts under section 31.5.2, Interrupt Functions. This module clears this bit to 0 when software writes 0 to all the PIPEBEMP bits corresponding to the PIPEBEMPE bits to which 1 has been set. This bit cannot be cleared to 0 even if software writes 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
9	NRDY	0	R	Buffer Not Ready Interrupt Status 0: NRDY interrupts not generated 1: NRDY interrupts generated This module sets this bit to 1 when at least one PIPENRDY bit in NRDYSTS is set to 1 among the PIPENRDY bits corresponding to the PIPENRDYE bits in NRDYENB to which 1 has been set (when this module detects the NRDY interrupt status in at least one pipe among the pipes for which software enables the NRDY interrupt output). For the conditions for PIPENRDY status assertion, refer to (2) NRDY Interrupts under section 31.5.2, Interrupt Functions. This module clears this bit to 0 when software writes 0 to all the PIPENRDY bits corresponding to the PIPENRDYE bits to which 1 has been set. This bit cannot be cleared to 0 even if software writes 0 to this bit.
8	BRDY	0	R	Buffer Ready Interrupt Status Indicates the BRDY interrupt status. 0: BRDY interrupts not generated 1: BRDY interrupts generated This module sets this bit to 1 when at least one PIPEBRDY bit in BRDYSTS is set to 1 among the PIPEBRDY bits corresponding to the PIPEBRDYE bits in BRDYENB to which 1 has been set (when this module detects the BRDY interrupt status in at least one pipe among the pipes for which software enables the BRDY interrupt output). For the conditions for PIPEBRDY status assertion, refer to (1) BRDY Interrupts under section 31.5.2, Interrupt Functions. This module clears this bit to 0 when software writes 0 to all the PIPEBRDY bits corresponding to the PIPEBRDYE bits to which 1 has been set. This bit cannot be cleared to 0 even if software writes 0 to this bit.
7	VBSTS	0/1*[3]	R	VBUS Input Status 0: The VBUS pin is low level. 1: The VBUS pin is high level.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	DVSQ[2:0]	000/001*[2]	R	Device State 000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state When the host controller function is selected, the read value is invalid.
3	VALID	0	R/W*[7]	USB Request Reception 0: Not detected 1: Setup packet reception When the host controller function is selected, the read value is invalid.
2 to 0	CTSQ[2:0]	000	R	Control Transfer Stage 000: Idle or setup stage 001: Control read data stage 010: Control read status stage 011: Control write data stage 100: Control write status stage 101: Control write (no data) status stage 110: Control transfer sequence error 111: Setting prohibited When the host controller function is selected, the read value is invalid.

Notes:
1. This bit is initialized to B'0 by a power-on reset and B'1 by a USB bus reset.
2. These bits are initialized to B'000 by a power-on reset and B'001 by a USB bus reset.
3. This bit is 1 when the level of the VBUS pin input is high and 0 when low.
4. To clear the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 only to the bits to be cleared; write 1 to the other bits. Do not write 0 to the status bits indicating 0.
5. A change in the status indicated by the VBINT and RESM bits can be detected even while the clock supply is stopped (while SCKE is 0), and the interrupts are output when the corresponding interrupt enable bits are enabled. Clearing the status through software should be done after enabling the clock supply.
6. A change in the status of the RESM, DVST, and CTRT bits occurs only when the function controller function is selected; disable the corresponding interrupt enable bits (set to 0) when the host controller function is selected.
7. Only 0 can be written to.

31.4.16 Interrupt Status Register 1 (INTSTS1)

INTSTS1 is a register that is used to confirm interrupt status.

Interrupt generation can be confirmed simply by referencing one of the registers: INTSTS0 when the function controller function is selected and INTSTS1 when the host controller function is selected.

The various interrupts indicated by the bits in this register should be enabled only when the host controller function is selected.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHG	—	DTCH	ATTCH	—	—	—	—	EOF ERR	SIGN	SACK	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W*1	R	R/W*1	R/W*1	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	BCHG	0	R/W*[1]	USB Bus Change Interrupt Status Indicates the status of the USB bus change interrupt. 0: BCHG interrupts not generated 1: BCHG interrupts generated This module detects the BCHG interrupt when a change in the full-speed or low-speed signal level occurs on the USB port (a change from J-state, K-state, or SE0 to J-state, K-state, or SE0), and sets this bit to 1. Here, if software has set the corresponding interrupt enable bit to 1, this module generates the interrupt. This module sets the LNST bits in SYSSTS0 to indicate the current input state of the USB port. When the BCHG interrupt is generated, use software to repeat reading the LNST bits until the same value is read three or more times, and eliminate chattering. A change in the USB bus state can be detected even while the internal clock supply is stopped. When the function controller function is selected, the read value is invalid.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
12	DTCH	0	R/W*[1]	USB Disconnection Detection Interrupt Status Indicates the status of the USB disconnection detection interrupt when the host controller function is selected. 0: DTCH interrupts not generated 1: DTCH interrupts generated This module detects the DTCH interrupt on detecting USB bus disconnection, and sets this bit to 1. Here, if software has set the corresponding interrupt enable bit to 1, this module generates the interrupt. This module detects bus disconnection based on USB Specification 2.0. After detecting the DTCH interrupt, this module controls hardware as described below (irrespective of the set value of the corresponding interrupt enable bit). Software should terminate all the pipes in which communications are currently carried out for the USB port and make a transition to the wait state for bus connection to the USB port (wait state for ATTCH interrupt generation). • Modifies the UACT bit for the port in which a DTCH interrupt has been detected to 0. • Puts the port in which a DTCH interrupt has been generated into the idle state. When the function controller function is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
11	ATTCH	0	R/W*[1]	ATTCH Interrupt Status Indicates the status of the ATTCH interrupt when the host controller function is selected. 0: ATTCH interrupts not generated 1: ATTCH interrupts generated This module detects the ATTCH interrupt on detecting J-state or K-state of the full-speed or low-speed level signal for 2.5 μs, and sets this bit to 1. Here, if software has set the corresponding interrupt enable bit to 1, this module generates the interrupt. Specifically, this module detects the ATTCH interrupt on any of the following conditions. • K-state, SE0, or SE1 changes to J-state, and J-state continues 2.5 μs. • J-state, SE0, or SE1 changes to K-state, and K-state continues 2.5 μs. When the function controller function is selected, the read value is invalid.
10 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	EOFERR	0	R/W*[1]	EOF Error Detection Interrupt Status Indicates the status of the EOFERR interrupt when the host controller function is selected. 0: EOFERR interrupt not generated 1: EOFERR interrupt generated This module detects the EOFERR interrupt on detecting that communication is not completed at the EOF2 timing prescribed by USB Specification 2.0, and sets this bit to 1. Here, if software has set the corresponding interrupt enable bit to 1, this module generates the EOFERR interrupt. After detecting the EOFERR interrupt, this module controls hardware as described below (irrespective of the set value of the corresponding interrupt enable bit). Software should terminate all the pipes in which communications are currently carried for the USB port and perform re-enumeration of the USB port. • Modifies the UACT bit for the port in which an EOFERR interrupt has been detected to 0. • Puts the port in which an EOFERR interrupt has been generated into the idle state. When the function controller function is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
5	SIGN	0	R/W*[1]	Setup Transaction Error Interrupt Status Indicates the status of the setup transaction error interrupt when the host controller function is selected. 0: SIGN interrupts not generated 1: SIGN interrupts generated This module detects the SIGN interrupt when ACK response is not returned from the peripheral device three consecutive times during the setup transactions issued by this module, and sets this bit to 1. Here, if software has set the corresponding interrupt enable bit to 1, this module generates the SIGN interrupt. Specifically, this module detects the SIGN interrupt when any of the following response conditions occur for three consecutive setup transactions. • Timeout is detected when the peripheral device has returned no response. • A damaged ACK packet is received. • A handshake other than ACK (NAK, NYET, or STALL) is received. When the function controller function is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
4	SACK	0	R/W*[1]	Setup Transaction Normal Response Interrupt Status Indicates the status of the setup transaction normal response interrupt when the host controller function is selected. 0: SACK interrupts not generated 1: SACK interrupts generated This module detects the SACK interrupt when ACK response is returned from the peripheral device during the setup transactions issued by this module, and sets this bit to 1. Here, if software has set the corresponding interrupt enable bit to 1, this module generates the SACK interrupt.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Notes: 1. To clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits.

2. A change in the status indicated by the BCHG bit can be detected even while the clock supply is stopped (while SCKE is 0), and the interrupt is output when the corresponding interrupt enable bit is enabled. Clearing the status through software should be done after enabling the clock supply.
No interrupts other than BCHG can be detected while the clock supply is stopped (while SCKE is 0).

31.4.17 BRDY Interrupt Status Register (BRDYSTS)

BRDYSTS is a register that indicates the BRDY interrupt status for each pipe.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BRDY	PIPE8 BRDY	PIPE7 BRDY	PIPE6 BRDY	PIPE5 BRDY	PIPE4 BRDY	PIPE3 BRDY	PIPE2 BRDY	PIPE1 BRDY	PIPE0 BRDY
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PIPE9BRDY	0	R/W*1	BRDY Interrupt Status for PIPE9*2 0: Interrupts not generated 1: Interrupts generated
8	PIPE8BRDY	0	R/W*1	BRDY Interrupt Status for PIPE8*2 0: Interrupts not generated 1: Interrupts generated
7	PIPE7BRDY	0	R/W*1	BRDY Interrupt Status for PIPE7*2 0: Interrupts not generated 1: Interrupts generated
6	PIPE6BRDY	0	R/W*1	BRDY Interrupt Status for PIPE6*2 0: Interrupts not generated 1: Interrupts generated
5	PIPE5BRDY	0	R/W*1	BRDY Interrupt Status for PIPE5*2 0: Interrupts not generated 1: Interrupts generated
4	PIPE4BRDY	0	R/W*1	BRDY Interrupt Status for PIPE4*2 0: Interrupts not generated 1: Interrupts generated
3	PIPE3BRDY	0	R/W*1	BRDY Interrupt Status for PIPE3*2 0: Interrupts not generated 1: Interrupts generated

Bit	Bit Name	Initial Value	R/W	Description
2	PIPE2BRDY	0	R/W*[1]	BRDY Interrupt Status for PIPE2*[2] 0: Interrupts not generated 1: Interrupts generated
1	PIPE1BRDY	0	R/W*[1]	BRDY Interrupt Status for PIPE1*[2] 0: Interrupts not generated 1: Interrupts generated
0	PIPE0BRDY	0	R/W*[1]	BRDY Interrupt Status for PIPE0*[2] 0: Interrupts not generated 1: Interrupts generated

Notes: 1. When BRDYM is 0, to clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits.
2. When BRDYM is 0, clearing this bit should be done before accessing the FIFO.

31.4.18 NRDY Interrupt Status Register (NRDYSTS)

NRDYSTS is a register that indicates the NRDY interrupt status for each pipe.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 NRDY	PIPE8 NRDY	PIPE7 NRDY	PIPE6 NRDY	PIPE5 NRDY	PIPE4 NRDY	PIPE3 NRDY	PIPE2 NRDY	PIPE1 NRDY	PIPE0 NRDY
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PIPE9NRDY	0	R/W*	NRDY Interrupt Status for PIPE9 0: Interrupts not generated 1: Interrupts generated
8	PIPE8NRDY	0	R/W*	NRDY Interrupt Status for PIPE8 0: Interrupts not generated 1: Interrupts generated

Bit	Bit Name	Initial Value	R/W	Description
7	PIPE7NRDY	0	R/W*	NRDY Interrupt Status for PIPE7 0: Interrupts not generated 1: Interrupts generated
6	PIPE6NRDY	0	R/W*	NRDY Interrupt Status for PIPE6 0: Interrupts not generated 1: Interrupts generated
5	PIPE5NRDY	0	R/W*	NRDY Interrupt Status for PIPE5 0: Interrupts not generated 1: Interrupts generated
4	PIPE4NRDY	0	R/W*	NRDY Interrupt Status for PIPE4 0: Interrupts not generated 1: Interrupts generated
3	PIPE3NRDY	0	R/W*	NRDY Interrupt Status for PIPE3 0: Interrupts not generated 1: Interrupts generated
2	PIPE2NRDY	0	R/W*	NRDY Interrupt Status for PIPE2 0: Interrupts not generated 1: Interrupts generated
1	PIPE1NRDY	0	R/W*	NRDY Interrupt Status for PIPE1 0: Interrupts not generated 1: Interrupts generated
0	PIPE0NRDY	0	R/W*	NRDY Interrupt Status for PIPE0 0: Interrupts not generated 1: Interrupts generated

Note: * To clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits.

31.4.19 BEMP Interrupt Status Register (BEMPSTS)

BEMPSTS is a register that indicates the BEMP interrupt status for each pipe.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIPE9 BEMP	PIPE8 BEMP	PIPE7 BEMP	PIPE6 BEMP	PIPE5 BEMP	PIPE4 BEMP	PIPE3 BEMP	PIPE2 BEMP	PIPE1 BEMP	PIPE0 BEMP
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	PIPE9BEMP	0	R/W*	BEMP Interrupts for PIPE9 0: Interrupts not generated 1: Interrupts generated
8	PIPE8BEMP	0	R/W*	BEMP Interrupts for PIPE8 0: Interrupts not generated 1: Interrupts generated
7	PIPE7BEMP	0	R/W*	BEMP Interrupts for PIPE7 0: Interrupts not generated 1: Interrupts generated
6	PIPE6BEMP	0	R/W*	BEMP Interrupts for PIPE6 0: Interrupts not generated 1: Interrupts generated
5	PIPE5BEMP	0	R/W*	BEMP Interrupts for PIPE5 0: Interrupts not generated 1: Interrupts generated
4	PIPE4BEMP	0	R/W*	BEMP Interrupts for PIPE4 0: Interrupts not generated 1: Interrupts generated
3	PIPE3BEMP	0	R/W*	BEMP Interrupts for PIPE3 0: Interrupts not generated 1: Interrupts generated

Bit	Bit Name	Initial Value	R/W	Description
2	PIPE2BEMP	0	R/W*	BEMP Interrupts for PIPE2 0: Interrupts not generated 1: Interrupts generated
1	PIPE1BEMP	0	R/W*	BEMP Interrupts for PIPE1 0: Interrupts not generated 1: Interrupts generated
0	PIPE0BEMP	0	R/W*	BEMP Interrupts for PIPE0 0: Interrupts not generated 1: Interrupts generated

Note: * To clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits.

31.4.20 Frame Number Register (FRMNUM)

FRMNUM is a register that determines the source of isochronous error notification and indicates the frame number.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	—	FRNM[10:0]										
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W*	R/W*	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	OVRN	0	R/W*	Overrun/Underrun Detection Status Indicates whether an overrun/underrun error has been detected in the pipe during isochronous transfer. 0: No error 1: An error occurred Software can clear this bit to 0 by writing 0 to the bit. Here, 1 should be written to the other bits in this register. (1) When the host controller function is selected This module sets this bit to 1 on any of the following conditions. • For the isochronous transfer pipe in the transmitting direction, the time to issue an OUT token comes before all the transmit data has been written to the FIFO buffer. • For the isochronous transfer pipe in the receiving direction, the time to issue an IN token comes when no FIFO buffer planes are empty. (2) When the function controller function is selected This module sets this bit to 1 on any of the following conditions. • For the isochronous transfer pipe in the transmitting direction, the IN token is received before all the transmit data has been written to the FIFO buffer. • For the isochronous transfer pipe in the receiving direction, the OUT token is received when no FIFO buffer planes are empty.

Bit	Bit Name	Initial Value	R/W	Description
14	CRCE	0	R/W*	Receive Data Error Indicates whether a CRC error or bit stuffing error has been detected in the pipe during isochronous transfer. 0: No error 1: An error occurred Software can clear this bit to 0 by writing 0 to the bit. Here, 1 should be written to the other bits in this register. (1) When the host controller function is selected On detecting a CRC error, this module generates the internal NRDY interrupt request. (2) When the function controller function is selected On detecting a CRC error, this module does not generate the internal NRDY interrupt request.
13 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	FRNM[10:0]	H'000	R	Frame Number This module sets these bits to indicate the latest frame number, which is updated every time an SOF packet is issued or received (every 1 ms). Repeat reading these bits until the same value is read twice.

Note: * Only 0 can be written to

31.4.21 μFrame Number Register (UFRMNUM)

UFRMNUM is a register that indicates the μframe number.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM[2:0]		
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	UFRNM[2:0]	000	R	μFrame The μframe number can be confirmed. This module sets these bits to indicate the μframe number during high-speed operation. During operation other than high-speed operation, this module sets these bits to B'000. Repeat reading these bits until the same value is read twice.

31.4.22 USB Address Register (USBADDR)

USBADDR is a register that indicates the USB address. This register is valid only when the function controller function is selected. When the host controller function is selected, peripheral device addresses should be set using the DEVSEL bits in PIPEMAXP.

This register is initialized by a power-on reset or a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR[6:0]						
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	USBADDR [6:0]	H'00	R	USB Address When the function controller function is selected, these bits indicate the USB address assigned by the host when the SET_ADDRESS request is successfully processed. On detecting the USB reset, this module sets these bits to H'00. When the host controller function is selected, these bits are invalid.

31.4.23 USB Request Type Register (USBREQ)

USBREQ is a register that stores setup requests for control transfers. When the function controller function is selected, the values of bRequest and bmRequestType that have been received are stored. When the host controller function is selected, the values of bRequest and bmRequestType to be transmitted are set.

This register is initialized by a power-on reset or a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BREQUEST[7:0]								BMREQUESTTYPE[7:0]							
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	BREQUEST [7:0]	H'00	R/W*	Request These bits store the USB request bRequest value. (1) When the host controller function is selected The USB request data value for the setup transaction to be transmitted should be set in these bits. Do not modify these bits while SUREQ is 1. (2) When the function controller function is selected Indicates the USB request data value received during the setup transaction. Writing to these bits is invalid.
7 to 0	BMREQUEST-TYPE[7:0]	H'00	R/W*	Request Type These bits store the USB request bmRequestType value. (1) When the host controller function is selected The USB request type value for the setup transaction to be transmitted should be set in these bits. Do not modify these bits while SUREQ is 1. (2) When the function controller function is selected Indicates the USB request type value received during the setup transaction. Writing to these bits is invalid.

Note: * When the function controller function is selected, these bits can only be read, and writing to these bits is invalid. When the host controller function is selected, these bits can be read and written to.

31.4.24 USB Request Value Register (USBVAL)

USBVAL is a register that stores setup requests for control transfers. When the function controller function is selected, the value of wValue that has been received is stored. When the host controller function is selected, the value of wValue to be transmitted is set.

This register is initialized by a power-on reset or a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WVALUE[15:0]															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	WVALUE[15:0]	H'0000	R/W*	Value These bits store the USB request wValue value. (1) When the host controller function is selected The USB request wValue value for the setup transaction to be transmitted should be set in these bits. Do not modify these bits while SUREQ is 1. (2) When the function controller function is selected Indicates the USB request wValue value received during the setup transaction. Writing to these bits is invalid.

Note: * When the function controller function is selected, these bits can only be read, and writing to these bits is invalid. When the host controller function is selected, these bits can be read and written to.

31.4.25 USB Request Index Register (USBINDX)

USBINDEX is a register that stores setup requests for control transfers. When the function controller function is selected, the value of wIndex that has been received is stored. When the host controller function is selected, the value of wIndex to be transmitted is set.

This register is initialized by a power-on reset or a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WINDEX[15:0]															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	WINDEX[15:0]	H'0000	R/W*	Index These bits store the USB request wIndex value. (1) When the host controller function is selected The USB request wIndex value for the setup transaction to be transmitted should be set in these bits. Do not modify these bits while SUREQ is 1. (2) When the function controller function is selected Indicates the USB request wIndex value received during the setup transaction. Writing to these bits is invalid.

Note: * When the function controller function is selected, these bits can only be read, and writing to these bits is invalid. When the host controller function is selected, these bits can be read and written to.

31.4.26 USB Request Length Register (USBLENG)

USBLENG is a register that stores setup requests for control transfers. When the function controller function is selected, the value of wLength that has been received is stored. When the host controller function is selected, the value of wLength to be transmitted is set.

This register is initialized by a power-on reset or a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WLENGTH[15:0]															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*	R/W*

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	WLENGTH [15:0]	H'0000	R/W*	Length These bits store the USB request wLength value. (1) When the host controller function is selected The USB request wLength value for the setup transaction to be transmitted should be set in these bits. Do not modify these bits while SUREQ is 1. (2) When the function controller function is selected Indicates the USB request wLength value received during the setup transaction. Writing to these bits is invalid.

Note: * When the function controller function is selected, these bits can only be read, and writing to these bits is invalid. When the host controller function is selected, these bits can be read and written to.

31.4.27 DCP Configuration Register (DCPCFG)

DCPCFG is a register that specifies the data transfer direction for the default control pipe (DCP).

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DIR	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DIR	0	R/W	Transfer Direction When the host controller function is selected, this bit sets the transfer direction of data stage. 0: Data receiving direction 1: Data transmitting direction When the function controller function is selected, this bit should be cleared to 0.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

31.4.28 DCP Maximum Packet Size Register (DCPMAXP)

DCPMAXP is a register that specifies the maximum packet size for the DCP.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEVSEL[3:0]				—	—	—	—	—	MXPS[6:0]						
Initial value :	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	DEVSEL[3:0]	0000	R/W	Device Select When the host controller function is selected, these bits specify the communication target peripheral device address. 0000: Address 0000 0001: Address 0001 : : 1001: Address 1001 1010: Address 1010 Other than above: Setting prohibited These bits should be set after setting the address to the DEVADDn register corresponding to the value to be set in these bits. For example, before setting DEVSEL to 0010, the address should be set to the DEVADD2 register. These bits should be set while CSSTS is 0, PID is NAK, and SUREQ is 0. Before modifying these bits after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. When the function controller function is selected, these bits should be set to B'0000.

Bit	Bit Name	Initial Value	R/W	Description
11 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	MXPS[6:0]	H'40	R/W	Maximum Packet Size Specifies the maximum data payload (maximum packet size) for the DCP. These bits are initialized to H'40 (64 bytes). These bits should be set to the value based on the USB Specification. These bits should be set while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. While MXPS is 0, do not write to the FIFO buffer or do not set PID to BUF.

31.4.29 DCP Control Register (DCPCTR)

DCPCTR is a register that is used to confirm the buffer memory status, change and confirm the data PID sequence bit, and set the response PID for the DCP.

This register is initialized by a power-on reset. The CCPL and PID[1:0] bits are initialized by a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	SUREQ	CSCLR	CSSTS	SUREQ CLR	—	—	SQCLR	SQSET	SQMON	PBUSY	PINGE	—	CCPL	PID[1:0]	
Initial value :	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W :	R	R/W*2	R/W*1	R	R/W*1	R	R	R/W*1	R/W*1	R	R	R/W	R	R/W*1	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer Status Indicates whether DCP FIFO buffer access is enabled or disabled. 0: Buffer access is disabled. 1: Buffer access is enabled. The meaning of the BSTS bit depends on the ISEL bit setting as follows. • When ISEL = 0, BSTS indicates whether the received data can be read from the buffer. • When ISEL = 1, BSTS indicates whether the data to be transmitted can be written to the buffer.
14	SUREQ	0	R/W*[2]	SETUP Token Transmission Transmits the setup packet by setting this bit to 1 when the host controller function is selected. 0: Invalid 1: Transmits the setup packet. After completing the setup transaction process, this module generates either the SACK or SIGN interrupt and clears this bit to 0. This module also clears this bit to 0 when software sets the SUREQCLR bit to 1. Before setting this bit to 1, set the DEVSEL bits, USBREQ register, USBVAL register, USBINDX register, and USBLENG register appropriately to transmit the desired USB request in the setup transaction. Before setting this bit to 1, check that the PID bits for the DCP are set to NAK. After setting this bit to 1, do not modify the DEVSEL bits, USBREQ register, USBVAL register, USBINDX register, or USBLENG register until the setup transaction is completed (SUREQ = 1). Write 1 to this bit only when transmitting the setup token; for the other purposes, write 0. When the function controller function is selected, be sure to write 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
13	CSCLR	0	R/W*[1]	C-SPLIT Status Clear for Split Transaction When the host controller function is selected, setting this bit to 1 clears the CSSTS bit to 0 for the transfer using the split transaction. In this case, the next DCP transfer restarts with the S-SPLIT. 0: Invalid 1: Clears the CSSTS bit to 0. When software sets this bit to 1, this module clears the CSSTS bit to 0. For the transfer using the split transaction, to restart the next transfer with the S-SPLIT forcibly, set this bit to 1 through software. However, for the normal split transaction, this module automatically clears the CSSTS bit to 0 upon completion of the C-SPLIT; therefore, clearing the CSSTS bit through software is not necessary. Controlling the CSSTS bit through this bit must be done while UACT is 0 and thus communication is halted or while no transfer is being performed with bus disconnection detected. Setting this bit to 1 while CSSTS is 0 has no effect. When the function controller function is selected, be sure to write 0 to this bit.
12	CSSTS	0	R	COMPLETE SPLIT (C-SPLIT) Status of Split Transaction Indicates the C-SPLIT status of the split transaction when the host controller function is selected. 0: START-SPLIT (S-SPLIT) transaction being processed or the device not using the split transaction being processed 1: C-SPLIT transaction being processed This module sets this bit to 1 upon start of the C-SPLIT and clears this bit to 0 upon detection of C-SPLIT completion. When the function controller function is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
11	SUREQCLR	0	R/W*[1]	SUREQ Bit Clear When the host controller function is selected, setting this bit to 1 clears the SUREQ bit to 0. 0: Invalid 1: Clears the SUREQ bit to 0. This bit always indicates 0. Set this bit to 1 through software when communication has stopped with SUREQ being 1 during the setup transaction. However, for normal setup transactions, this module automatically clears the SUREQ bit to 0 upon completion of the transaction; therefore, clearing the SUREQ bit through software is not necessary. Controlling the SUREQ bit through this bit must be done while UACT is 0 and thus communication is halted or while no transfer is being performed with bus disconnection detected. When the function controller function is selected, be sure to write 0 to this bit.
10, 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W*[1]	Toggle Bit Clear Specifies DATA0 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: Invalid 1: Specifies DATA0. This bit always indicates 0. Do not set the SQCLR and SQSET bits to 1 simultaneously. Set this bit to 1 while CSCTS is 0, PID is NAK, and CURPIPE bits are not yet set. Before setting this bit to 1 after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
7	SQSET	0	R/W*[1]	Toggle Bit Set Specifies DATA1 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: Invalid 1: Specifies DATA1. Do not set the SQCLR and SQSET bits to 1 simultaneously. Set this bit to 1 while CSCTS is 0, PID is NAK, and CURPIPE bits are not yet set. Before setting this bit to 1 after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
6	SQMON	1	R	Sequence Toggle Bit Monitor Indicates the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: DATA0 1: DATA1 This module allows this bit to toggle upon normal completion of the transaction. However, this bit is not allowed to toggle when a DATA-PID disagreement occurs during the transfer in the receiving direction. When the function controller function is selected, this module sets this bit to 1 (specifies DATA1 as the expected value) upon normal reception of the setup packet. When the function controller function is selected, this module does not reference to this bit during the IN/OUT transaction of the status stage, and does not allow this bit to toggle upon normal completion.
5	PBUSY	0	R	Pipe Busy This bit indicates whether DCP is used or not for the transaction when USB changes the PID bits from BUF to NAK. 0: DCP is not used for the transaction. 1: DCP is used for the transaction. This module modifies this bit from 0 to 1 upon start of the USB transaction for the pertinent pipe, and modifies the bit from 1 to 0 upon completion of one transaction. Reading this bit after software has set PID to NAK allows checking that modification of the pipe settings is possible. For details, refer to (1) Pipe Control Register Switching Procedures under section 31.5.3, Pipe Control.

Bit	Bit Name	Initial Value	R/W	Description
4	PINGE	0	R/W	PING Token Issue Enable When the host controller function is selected, setting this bit to 1 allows this module to issue the PING token during transfers in the transmitting direction and start a transfer in the transmitting direction with the PING transaction. 0: Disables issuing PING token. 1: Enables normal PING operation. When having detected the ACK handshake during PING transactions, this module performs the OUT transaction as the next transaction. When having detected the NAK handshake during OUT transactions, this module performs the PING transaction as the next transaction. When the host controller function is selected, setting this bit to 0 through software prevents this module from issuing the PING token during transfers in the transmitting direction and only allows this module to perform OUT transactions for the transfers in the transmitting direction. These bits should be modified while CSSTS is 0 and PID is NAK. Before setting this bit to 1 after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. When the function controller function is selected, be sure to write 0 to this bit.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	CCPL	0	R/W*[1]	Control Transfer End Enable When the function controller function is selected, setting this bit to 1 enables the status stage of the control transfer to be completed. 0: Invalid 1: Completion of control transfer is enabled. When software sets this bit to 1 while the corresponding PID bits are set to BUF, this module completes the control transfer stage. Specifically, during control read transfer, this module transmits the ACK handshake in response to the OUT transaction from the USB host, and outputs the zero-length packet in response to the IN transaction from the USB host during control write or no-data control transfer. However, on detecting the SET_ADDRESS request, this module operates in auto response mode from the setup stage up to the status stage completion irrespective of the setting of this bit. This module modifies this bit from 1 to 0 on receiving the new setup packet. Software cannot write 1 to this bit while VALID is 1. When the host controller function is selected, be sure to write 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
1,0	PID[1:0]	00	R/W	Response PID Controls the response type of this module during control transfer. 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response (1) When the host controller function is selected Modify the setting of these bits from NAK to BUF using the following procedure. • When the transmitting direction is set Write all the transmit data to the FIFO buffer while UACT is 1 and PID is NAK, and then set PID to BUF. After PID has been set to BUF, this module executes the OUT transaction (or PING transaction). • When the receiving direction is set Check that the FIFO buffer is empty (or empty the buffer) while UACT is 1 and PID is NAK, and then set PID to BUF. After PID has been set to BUF, this module executes the IN transaction. This module modifies the setting of these bits as follows. • This module sets PID to STALL (11) on receiving the data of the size exceeding the maximum packet size when software has set PID to BUF. • This module sets PID to NAK on detecting a receive error such as a CRC error three consecutive times. • This module also sets PID to STALL (11) on receiving the STALL handshake.

Bit	Bit Name	Initial Value	R/W	Description
1,0	PID[1:0]	00	R/W	Even if software modifies the PID bits to NAK after this module has issued S-SPLIT of the split transaction for the selected pipe (while CSSTS indicates 1), this module continues the transaction until C-SPLIT completes. On completion of C-SPLIT, this module sets PID to NAK. (2) When the function controller function is selected This module modifies the setting of these bits as follows. • This module modifies PID to NAK on receiving the setup packet. Here, this module sets VALID to 1. Software cannot modify the setting of PID until software sets VALID to 0. • This module sets PID to STALL (11) on receiving the data of the size exceeding the maximum packet size when software has set PID to BUF. • This module sets PID to STALL (1x) on detecting the control transfer sequence error. • This module sets PID to NAK on detecting the USB bus reset. This module does not reference to the setting of the PID bits while the SET_ADDRESS request is processed (auto processing).

Notes: 1. This bit is always read as 0. Only 1 can be written to.
2. Only 1 can be written to.

31.4.30 Pipe Window Select Register (PIPESEL)

PIPE1 to PIPE 9 should be set using PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPEnCTR, PIPEnTRE, and PIPEnTRN. After selecting the pipe using PIPESEL, functions of the pipe should be set using PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI. PIPEnCTR, PIPEnTRE, and PIPEnTRN can be set regardless of the pipe selection in PIPESEL.

For a power-on reset and a USB bus reset, the corresponding bits for not only the selected pipe but all of the pipes are initialized.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL[3:0]			
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	PIPESEL[3:0]	0000	R/W	Pipe Window Select Selects the pipe number corresponding to the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers which data is written to or read from. 0000: No pipe selected 0001: PIPE1 0010: PIPE2 0011: PIPE3 0100: PIPE4 0101: PIPE5 0110: PIPE6 0111: PIPE7 1000: PIPE8 1001: PIPE9 Other than above: Setting prohibited Selecting a pipe number through these bits allows writing to and reading from the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers that correspond to the selected pipe number. When PIPESEL = 0000, 0 is read from all of the bits in PIPECFG, PIPEBUF, PIPEMAXP, PIPEERI and PIPEnCTR. Writing to these bits is invalid.

31.4.31 Pipe Configuration Register (PIPECFG)

PIPECFG is a register that specifies the transfer type, buffer memory access direction, and endpoint numbers for PIPE1 to PIPE9. It also selects continuous or non-continuous transfer mode, single or double buffer mode, and whether to continue or disable pipe operation at the end of transfer.

This register is initialized by a power-on reset. Only the TYPE[1:0] bits are initialized by a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE[1:0]		—	—	—	BFRE	DBLB	CNTMD	SHT NAK	—	—	DIR	EPNUM[3:0]			
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TYPE[1:0]	00	R/W	Transfer Type Selects the transfer type for the pipe selected by the PIPESEL bits (selected pipe) • PIPE1 and PIPE2 00: Pipe not used 01: Bulk transfer 10: Setting prohibited 11: Isochronous transfer • PIPE3 to PIPE5 00: Pipe not used 01: Bulk transfer 10: Setting prohibited 11: Setting prohibited • PIPE6 to PIPE9 00: Pipe not used 01: Setting prohibited 10: Interrupt transfer 11: Setting prohibited Before setting PID to BUF for the selected pipe (before starting USB communication using the selected pipe), be sure to set these bits to the value other than 00. Modify these bits while the PID bits for the selected pipe are set to NAK. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
13 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	BFRE	0	R/W	BRDY Interrupt Operation Specification Specifies the BRDY interrupt generation timing from this module to the CPU with respect to the selected pipe. 0: BRDY interrupt upon transmitting or receiving of data 1: BRDY interrupt upon completion of reading of data When software has set this bit to 1 and the selected pipe is in the receiving direction, this module detects the transfer completion and generates the BRDY interrupt on having read the pertinent packet. When the BRDY interrupt is generated with the above conditions, software needs to write 1 to BCLR. The FIFO buffer assigned to the selected pipe is not enabled for reception until 1 is written to BCLR. When software has set this bit to 1 and the selected pipe is in the transmitting direction, this module does not generate the BRDY interrupt. For details, refer to (1) BRDY Interrupt under section 31.5.2, Interrupt Functions. Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
9	DBLB	0	R/W	Double Buffer Mode Selects either single or double buffer mode for the FIFO buffer used by the selected pipe. 0: Single buffer 1: Double buffer This bit is valid when PIPE1 to PIPE5 are selected. When software has set this bit to 1, this module assigns two planes of the FIFO buffer size specified by the BUFSIZE bits in PIPEBUF to the selected pipe. Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this module. (BUFSIZE + 1) * 64 * (DBLB + 1) [bytes] When software has set this bit to 1 and the selected pipe is in the transmitting direction, this module does not generate the BRDY interrupt. For details, refer to (1) BRDY Interrupt under section 31.5.2, Interrupt Functions. Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
8	CNTMD	0	R/W	Continuous Transfer Mode Specifies whether to use the selected pipe in continuous transfer mode. 0: Non-continuous transfer mode 1: Continuous transfer mode This bit is valid when PIPE1 to PIPE5 are selected by the PIPESEL bits and bulk transfer is selected (TYPE = 01). Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
7	SHTNAK	0	R/W	Pipe Disabled at End of Transfer Specifies whether to modify PID to NAK upon the end of transfer when the selected pipe is in the receiving direction. 0: Pipe continued at the end of transfer 1: Pipe disabled at the end of transfer This bit is valid when the selected pipe is PIPE1 to PIPE5 in the receiving direction. When software has set this bit to 1 for the selected pipe in the receiving direction, this module modifies the PID bits corresponding to the selected pipe to NAK on determining the end of the transfer. This module determines that the transfer has ended on any of the following conditions. • A short packet (including a zero-length packet) is successfully received. • The transaction counter is used and the number of packets specified by the counter is successfully received. Modify these bits while CSSTS is 0 and PID is NAK. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. This bit should be cleared to 0 for the pipe in the transmitting direction.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	DIR	0	R/W	Transfer Direction Specifies the transfer direction for the selected pipe. 0: Receiving direction 1: Sending direction When software has set this bit to 0, this module uses the selected pipe in the receiving direction, and when software has set this bit to 1, this module uses the selected pipe in the transmitting direction. Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. To modify these bits after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously through software to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
3 to 0	EPNUM[3:0]	0000	R/W	Endpoint Number These bits specify the endpoint number for the selected pipe. Setting 0000 means unused pipe. Modify these bits while CSSTS is 0 and PID is NAK. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. Do not make the settings such that the combination of the set values in the DIR and EPNUM bits should be the same for two or more pipes (EPNUM = 0000 can be set for all the pipes).

31.4.32 Pipe Buffer Setting Register (PIPEBUF)

PIPEBUF is a register that specifies the buffer size and buffer number for PIPE1 to PIPE9.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BUFSIZE[4:0]					—	—	BUFNMB[7:0]							
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14 to 10	BUFSIZE[4:0]	H'00	R/W	Buffer Size Specifies the size of the buffer for the pipe selected by the PIPESEL bits (selected pipe) in terms of blocks, where one block comprises 64 bytes. 00000 (H'00): 64 bytes 00001 (H'01): 128 bytes : : 11111 (H'1F): 2 kbytes When software has set the DBLB bit to 1, this module assigns two planes of the FIFO buffer size specified by the BUFSIZE bits to the selected pipe. Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this module. (BUFSIZE + 1) * 64 * (DBLB + 1) [bytes] The valid value for these bits depends on the selected pipe. • PIPE1 to PIPE5: Any value from H'00 to H'1F is valid. • PIPE6 to PIPE9: H'00 should be set. When used with CNTMD = 1, set an integer multiple of the maximum packet size to the BUFSIZE bits. Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	BUFNMB[7:0]	H'00	R/W	Buffer Number These bits specify the FIFO buffer number for the selected pipe (from H'04 to H'7F). When the selected pipe is one of PIPE1 to PIPE5, any value can be set to these bits according to the user system. BUFNUMB = H'00 to H'03 are used exclusively for DCP. BUFNMB = H'04 is used exclusively for PIPE6. When PIPE6 is not used, H'04 can be used for other pipes. When PIPE6 is selected, writing to these bits is invalid and H'04 is automatically assigned by this module. BUFNMB = H'05 is used exclusively for PIPE7. When PIPE7 is not used, H'05 can be used for other pipes. When PIPE7 is selected, writing to these bits is invalid and H'05 is automatically assigned by this module. BUFNUMB = H'06 is used exclusively for PIPE8. When PIPE8 is not used, H'06 can be used for other pipes. When PIPE8 is selected, writing to these bits is invalid and H'06 is automatically assigned by this module. BUFNUMB = H'07 is used exclusively for PIPE9. When PIPE9 is not used, H'07 can be used for other pipes. When PIPE9 is selected, writing to these bits is invalid and H'07 is automatically assigned by this module. Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

31.4.33 Pipe Maximum Packet Size Register (PIPEMAXP)

PIPEMAXP is a register that specifies the maximum packet size for PIPE1 to PIPE9.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEVSEL[3:0]				—	MXPS[10:0]										
Initial value :	0	0	0	0	0	*	*	*	*	*	*	*	*	*	*	*
R/W :	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	DEVSEL[3:0]	0000	R/W	Device Select When the host controller function is selected, these bits specify the USB address of the communication target peripheral device. 0000: Address 0000 0001: Address 0001 0010: Address 0010 : : 1010: Address 1010 Other than above: Setting prohibited These bits should be set after setting the address to the DEVADDn register corresponding to the value to be set in these bits. For example, before setting DEVSEL to 0010, the address should be set to the DEVADD2 register. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. When the function controller function is selected, these bits should be set to B'0000.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	MXPS[10:0]	*	R/W	Maximum Packet Size Specifies the maximum data payload (maximum packet size) for the selected pipe. The valid value for these bits depends on the pipe as follows. PIPE1, PIPE2: 1 byte (H'001) to 1,024 bytes (H'400) PIPE3 to PIPE5: 8 bytes (H'008), 16 bytes (H'010), 32 bytes (H'020), 64 bytes (H'040), and 512 bytes (H'200) (Bits 2 to 0 are not provided.) PIPE6 to PIPE9: 1 byte (H'001) to 64 bytes (H'040) These bits should be set to the appropriate value for each transfer type based on the USB Specification. For split transactions using the isochronous pipe, these bits should be set to 188 bytes or less. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. While MXPS is 0, do not write to the FIFO buffer or set PID to BUF.

Note: * The initial value of MXPS is H'000 when no pipe is selected with the PIPESEL bits in PIPESEL and H'040 when a pipe is selected with the PIPESEL bit in PIPESEL.

31.4.34 Pipe Timing Control Register (PIPEPERI)

PIPEPERI is a register that selects whether the buffer is flushed or not when an interval error occurred during isochronous IN transfer, and sets the interval error detection interval for PIPE1 to PIPE9.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV[2:0]		
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	IFIS	0	R/W	Isochronous IN Buffer Flush Specifies whether to flush the buffer when the pipe selected by the PIPESEL bits (selected pipe) is used for isochronous IN transfers. 0: The buffer is not flushed. 1: The buffer is flushed. When the function controller function is selected and the selected pipe is for isochronous IN transfers, this module automatically clears the FIFO buffer when this module fails to receive the IN token from the USB host within the interval set by the IITV bits in terms of (μ) frames. In double buffer mode (DBLB = 1), this module only clears the data in the plane used earlier. This module clears the FIFO buffer on receiving the SOF packet immediately after the (μ) frame in which this module has expected to receive the IN token. Even if the SOF packet is corrupted, this module also clears the FIFO buffer at the right timing to receive the SOF packet by using the internal interpolation. When the host controller function is selected, set this bit to 0. When the selected pipe is not for the isochronous transfer, set this bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
11 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	IITV[2:0]	000	R/W	Interval Error Detection Interval Specifies the interval error detection timing for the selected pipe in terms of frames, which is expressed as n-th power of 2 (n is the value to be set). As described later, the detailed functions are different in host controller mode and in function controller mode. Modify these bits while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. Before modifying these bits after USB communication has been completed with these bits set to a certain value, set PID to NAK and then set ACLRM to 1 to initialize the interval timer. The IITV bits are invalid for PIPE3 to PIPE5; set these bits to 000 for these pipes.

31.4.35 PIPEn Control Registers (PIPEnCTR) (n = 1 to 9)

PIPEnCTR is a register that is used to confirm the buffer memory status for the corresponding pipe, change and confirm the data PID sequence bit, determine whether auto response mode is set, determine whether auto buffer clear mode is set, and set a response PID for PIPE1 to PIPE9. This register can be set regardless of the pipe selection in PIPESEL.

This register is initialized by a power-on reset. PID[1:0] are initialized by a USB bus reset.

(1) PIPEnCTR (n = 1 to 5)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INBUFM	CSCLR	CSSTS	—	AT REPM	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID[1:0]	
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R/W*2	R	R	R/W	R/W	R/W*1	R/W*1	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer Status Indicates the FIFO buffer status for the pertinent pipe. 0: Buffer access is disabled. 1: Buffer access is enabled. The meaning of this bit depends on the settings of the DIR, BFRE, and DCLRM bits as shown in table 31.13.

Bit	Bit Name	Initial Value	R/W	Description
14	INBUFM	0	R	IN Buffer Monitor Indicates the pertinent FIFO buffer status when the pertinent pipe is in the transmitting direction. 0: There is no data to be transmitted in the buffer memory. 1: There is data to be transmitted in the buffer memory. When the pertinent pipe is in the transmitting direction (DIR = 1), this module sets this bit to 1 when software (or DMAC) completes writing data to at least one FIFO buffer plane. This module sets this bit to 0 when this module completes transmitting the data from the FIFO buffer plane to which all the data has been written. In double buffer mode (DBLB = 1), this module sets this bit to 0 when this module completes transmitting the data from the two FIFO buffer planes before software (or DMAC) completes writing data to one FIFO buffer plane. This bit indicates the same value as the BSTS bit when the pertinent pipe is in the receiving direction (DIR = 0).

Bit	Bit Name	Initial Value	R/W	Description
13	CSCLR	0	R/W*[2]	C-SPLIT Status Clear Bit When the host controller function is selected, setting this bit to 1 through software allows this module to clear the CSSTS bit to 0. 0: Writing invalid 1: Clears the CSSTS bit to 0. For the transfer using the split transaction, to restart the next transfer with the S-SPLIT forcibly, set this bit to 1 through software. However, for the normal split transaction, this module automatically clears the CSSTS bit to 0 upon completion of the C-SPLIT; therefore, clearing the CSSTS bit through software is not necessary. Controlling the CSSTS bit through this bit must be done while UACT is 0 and thus communication is halted or while no transfer is being performed with bus disconnection detected. Setting this bit to 1 while CSSTS is 0 has no effect. When the function controller function is selected, be sure to write 0 to this bit.
12	CSSTS	0	R	CSSTS Status Bit Indicates the C-SPLIT status of the split transaction when the host controller function is selected. 0: START-SPLIT (S-SPLIT) transaction being processed or the transfer not using the split transaction in progress 1: C-SPLIT transaction being processed This module sets this bit to 1 upon start of the C-SPLIT and clears this bit to 0 upon detection of C-SPLIT completion. Indicates the valid value only when the host controller function is selected.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10	ATREPM	0	R/W	Auto Response Mode Enables or disables auto response mode for the pertinent pipe. 0: Auto response disabled 1: Auto response enabled When the function controller function is selected and the pertinent pipe is for bulk transfer, this bit can be set to 1. When this bit is set to 1, this module responds to the token from the USB host as described below. (1) When the pertinent pipe is for bulk IN transfer (TYPE = 01 and DIR = 1) When ATREPM = 1 and PID = BUF, this module transmits a zero-length packet in response to the IN token. This module updates (allows toggling of) the sequence toggle bit (DATA-PID) each time this module receives the ACK from the USB host (in a single transaction, IN token is received, zero-length packet is transmitted, and then ACK is received.). In this case, this module does not generate the BRDY or BEMP interrupt. (2) When the pertinent pipe is for bulk OUT transfer (TYPE = 01 and DIR = 0) When ATREPM = 1 and PID = BUF, this module returns NAK in response to the OUT (or PING) token and generates the NRDY interrupt. Modify this bit while CSSTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
10	ATREPM	0	R/W	For USB communication in auto response mode, set this bit to 1 while the FIFO buffer is empty. Do not write to the FIFO buffer during USB communication in auto response mode. When the pertinent pipe is for isochronous transfer, be sure to set this bit to 0. When the host controller function is selected, set this bit to 0.
9	ACLRM	0	R/W	Auto Buffer Clear Mode Enables or disables automatic buffer clear mode for the pertinent pipe. 0: Disabled 1: Enabled (all buffers are initialized) To delete the information in the FIFO buffer assigned to the pertinent pipe completely, write 1 and then 0 to this bit continuously. Table 31.14 shows the information cleared by writing 1 and 0 to this bit continuously and the cases in which clearing the information is necessary. Modify this bit while CSSTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W*[1]	Toggle Bit Clear This bit should be set to 1 to clear the expected value (to set DATA0 as the expected value) of the sequence toggle bit for the next transaction of the pertinent pipe. 0: Invalid 1: Specifies DATA0. Setting this bit to 1 through software allows this module to set DATA0 as the expected value of the sequence toggle bit of the pertinent pipe. This module always sets this bit to 0. When the host controller function is selected, setting this bit to 1 for the pipe for bulk OUT transfer, this module starts the next transfer of the pertinent pipe with the PING token. Set the SQCLR bit to 1 while CSCTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
7	SQSET	0	R/W*[1]	Toggle Bit Set This bit should be set to 1 to set DATA1 as the expected value of the sequence toggle bit for the next transaction of the pertinent pipe. 0: Invalid 1: Specifies DATA1. Setting this bit to 1 through software allows this module to set DATA1 as the expected value of the sequence toggle bit of the pertinent pipe. This module always sets this bit to 0. Set the SQSET bit to 1 while CSSTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
6	SQMON	0	R	Toggle Bit Confirmation Indicates the expected value of the sequence toggle bit for the next transaction of the pertinent pipe. 0: DATA0 1: DATA1 When the pertinent pipe is not for the isochronous transfer, this module allows this bit to toggle upon normal completion of the transaction. However, this bit is not allowed to toggle when a DATA-PID disagreement occurs during the receiving transfer.

Bit	Bit Name	Initial Value	R/W	Description
5	PBUSY	0	R	Pipe Busy This bit indicates whether the relevant pipe is used or not for the transaction. 0: The relevant pipe is not used for the transaction. 1: The relevant pipe is used for the transaction. This module modifies this bit from 0 to 1 upon start of the USB transaction for the pertinent pipe, and modifies the bit from 1 to 0 upon completion of one transaction. Reading this bit after software has set PID to NAK allows checking that modification of the pipe settings is possible. For details, refer to (1) Pipe Control Register Switching Procedures under section 31.5.3, Pipe Control.
4 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	Response PID Specifies the response type for the next transaction of the pertinent pipe. 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response The default setting of these bits is NAK. Modify the setting to BUF to use the pertinent pipe for USB transfer. Tables 31.15 and 31.16 show the basic operation (operation when there are no errors in the transmitted and received packets) of this module depending on the PID bit setting. After modifying the setting of these bits through software from BUF to NAK during USB communication using the pertinent pipe, check that PBUSY is 1 to see if USB communication using the pertinent pipe has actually entered the NAK state. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. This module modifies the setting of these bits as follows. • This module sets PID to NAK on recognizing the completion of the transfer when the pertinent pipe is in the receiving direction and software has set the SHTNAK bit for the selected pipe to 1. • This module sets PID to STALL (11) on receiving the data packet with the payload exceeding the maximum packet size of the pertinent pipe. • This module sets PID to NAK on detecting a USB bus reset when the function controller function is selected. • This module sets PID to NAK on detecting a receive error such as a CRC error three consecutive times when the host controller function is selected.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	• This module sets PID to STALL (11) on receiving the STALL handshake when the host controller function is selected. To specify each response type, set these bits as follows. • To make a transition from NAK (00) to STALL, set 10. • To make a transition from BUF (01) to STALL, set 11. • To make a transition from STALL (11) to NAK, set 10 and then 00. • To make a transition from STALL to BUF, set 00 (NAK) and then 01 (BUF).

Notes: 1. Only 0 can be read and 1 can be written to.
2. Only 1 can be written to.

Table 31.13 Meaning of BSTS Bit

DIR Bit	BFRE Bit	DCLRM Bit	Meaning of BSTS Bit
0	0	0	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
		1	Setting prohibited
	1	0	1: The received data can be read from the FIFO buffer. 0: Software has set BCLR to 1 after the received data has been completely read from the FIFO buffer.
		1	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
1	0	0	1: The transmit data can be written to the FIFO buffer. 0: The transmit data has been completely written to the FIFO buffer.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

Table 31.14 Information Cleared by this Module by Setting ACLRM = 1

No.	Information Cleared by ACLRM Bit Manipulation	Cases in which Clearing the Information is Necessary
1	All the information in the FIFO buffer assigned to the pertinent pipe (all the information in two FIFO buffer planes in double buffer mode)	
2	The interval count value when the pertinent pipe is for isochronous transfer	When the interval count value is to be reset
3	Values of the internal flags related to the BFRE bit	When the BFRE setting is modified
4	FIFO buffer toggle control	When the DBLB setting is modified
5	Values of the internal flags related to the transaction count	When the transaction count function is forcibly terminated

Table 31.15 Operation of This Module depending on PID Setting (when Host Controller Function is Selected)

PID	Transfer Type	Transfer Direction (DIR Bit)	Operation of This Module
00 (NAK)	Operation does not depend on the setting.	Operation does not depend on the setting.	Does not issue tokens.
01 (BUF)	Bulk or interrupt	Operation does not depend on the setting.	Issues tokens while UACT is 1 and the FIFO buffer corresponding to the pertinent pipe is ready for transmission and reception. Does not issue tokens while UACT is 0 or the FIFO buffer corresponding to the pertinent pipe is not ready for transmission or reception.
	Isochronous	Operation does not depend on the setting.	Issues tokens irrespective of the status of the FIFO buffer corresponding to the pertinent pipe.
10 (STALL) or 11 (STALL)	Operation does not depend on the setting.	Operation does not depend on the setting.	Does not issue tokens.

Table 31.16 Operation of This Module depending on PID Setting (when Function Controller Function is Selected)

PID	Transfer Type	Transfer Direction (DIR Bit)	Operation of This Module
00 (NAK)	Bulk or interrupt	Operation does not depend on the setting.	Returns NAK in response to the token from the USB host. For the operation when ATREPM is 1, refer to the description of the ATREPM bit.
	Isochronous	Operation does not depend on the setting.	Returns nothing in response to the token from the USB host.
01 (BUF)	Bulk	Receiving direction (DIR = 0)	Receives data and returns ACK in response to the OUT token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Returns NAK if not ready. Returns ACK in response to the PING token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Returns NYET if not ready.
	Interrupt	Receiving direction (DIR = 0)	Receives data and returns ACK in response to the OUT token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Returns NAK if not ready.
	Bulk or interrupt	Transmitting direction (DIR = 1)	Transmits data in response to the token from the USB host if the corresponding FIFO buffer is ready for transmission. Returns NAK if not ready.
	Isochronous	Receiving direction (DIR = 0)	Receives data in response to the OUT token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Discards data if not ready.

PID	Transfer Type	Transfer Direction (DIR Bit)	Operation of This Module
01 (BUF)	Isochronous	Transmitting direction (DIR = 1)	Transmits data in response to the token from the USB host if the corresponding FIFO buffer is ready for transmission. Transmits the zero-length packet if not ready.
10 (STALL) or 11 (STALL)	Bulk or interrupt	Operation does not depend on the setting.	Returns STALL in response to the token from the USB host.
	Isochronous	Operation does not depend on the setting	Returns nothing in response to the token from the USB host.

(2) PIPEnCTR (n = 6 to 9)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR	SQSET	SQMON	PBUSY	—	—	—	PID[1:0]	
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R/W*1	R/W	R	R	R/W	R/W*1	R/W*1	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	Buffer Status Indicates the FIFO buffer status for the pertinent pipe. 0: Buffer access is disabled. 1: Buffer access is enabled. The meaning of this bit depends on the settings of the DIR, BFRE, and DCLRM bits as shown in table 31.13.
14	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13	CSCLR	0	R/W*[1]	C-SPLIT Status Clear Bit Setting this bit to 1 allows this module to clear the CSSTS bit of the pertinent pipe to 0. 0: Writing invalid 1: Clears the CSSTS bit to 0. For the transfer using the split transaction, to restart the next transfer with the S-SPLIT forcibly, set this bit to 1 through software. However, for the normal split transaction, this module automatically clears the CSSTS bit to 0 upon completion of the C-SPLIT; therefore, clearing the CSSTS bit through software is not necessary. Controlling the CSSTS bit through this bit must be done while UACT is 0 thus communication is halted or while no transfer is being performed with bus disconnection detected. Setting this bit to 1 while CSSTS is 0 has no effect. When the function controller function is selected, be sure to write 0 to this bit.

Bit	Bit Name	Initial Value	R/W	Description
12	CSSTS	0	R/W	CSSTS Status Bit Indicates the C-SPLIT status of the split transaction when the host controller function is selected. 0: START-SPLIT (S-SPLIT) transaction being processed or the transfer not using the split transaction in progress 1: C-SPLIT transaction being processed This module sets this bit to 1 upon start of the C-SPLIT and clears this bit to 0 upon detection of C-SPLIT completion. Indicates the valid value only when the host controller function is selected.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	ACLRM	0	R/W	Auto Buffer Clear Mode*3*4 Enables or disables automatic buffer clear mode for the pertinent pipe. 0: Disabled 1: Enabled (all buffers are initialized) To delete the information in the FIFO buffer assigned to the pertinent pipe completely, write 1 and then 0 to this bit continuously. Table 31.17 shows the information cleared by writing 1 and 0 to this bit continuously and the cases in which clearing the information is necessary. Modify this bit while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W*[1]	Toggle Bit Clear*[3]*[4] This bit should be set to 1 to clear the expected value (to set DATA0 as the expected value) of the sequence toggle bit for the next transaction of the pertinent pipe. 0: Invalid 1: Specifies DATA0. Setting this bit to 1 through software allows this module to set DATA0 as the expected value of the sequence toggle bit of the pertinent pipe. This module always sets this bit to 0. When the host controller function is selected, setting this bit to 1 for the pipe for bulk OUT transfer, this module starts the next transfer of the pertinent pipe with the PING token. Set the SQCLR bit to 1 while CSCTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

Bit	Bit Name	Initial Value	R/W	Description
7	SQSET	0	R/W*[1]	Toggle Bit Set*[3]*[4] This bit should be set to 1 to set DATA1 as the expected value of the sequence toggle bit for the next transaction of the pertinent pipe. 0: Invalid 1: Specifies DATA1. Setting this bit to 1 through software allows this module to set DATA1 as the expected value of the sequence toggle bit of the pertinent pipe. This module always sets this bit to 0. Set the SQSET bit to 1 while CSSTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.
6	SQMON	0	R	Toggle Bit Confirmation Indicates the expected value of the sequence toggle bit for the next transaction of the pertinent pipe. 0: DATA0 1: DATA1 When the pertinent pipe is not for the isochronous transfer, this module allows this bit to toggle upon normal completion of the transaction. However, this bit is not allowed to toggle when a DATA-PID disagreement occurs during the receiving transfer.

Bit	Bit Name	Initial Value	R/W	Description
5	PBUSY	0	R	Pipe Busy This bit indicates whether the relevant pipe is used or not for the transaction. 0: The relevant pipe is not used for the transaction. 1: The relevant pipe is used for the transaction. This module modifies this bit from 0 to 1 upon start of the USB transaction for the pertinent pipe, and modifies the bit from 1 to 0 upon completion of one transaction. Reading this bit after software has set PID to NAK allows checking that modification of the pipe settings is possible. For details, refer to (1) Pipe Control Register Switching Procedures under section 31.5.3, Pipe Control.
4 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	Response PID Specifies the response type for the next transaction of the pertinent pipe. 00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response The default setting of these bits is NAK. Modify the setting to BUF to use the pertinent pipe for USB transfer. Tables 31.15 and 31.16 show the basic operation (operation when there are no errors in the transmitted and received packets) of this module depending on the PID bit setting. After modifying the setting of these bits through software from BUF to NAK during USB communication using the pertinent pipe, check that PBUSY is 1 to see if USB communication using the pertinent pipe has actually entered the NAK state. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. This module modifies the setting of these bits as follows. • This module sets PID to NAK on recognizing the completion of the transfer when the pertinent pipe is in the receiving direction and software has set the SHTNAK bit for the selected pipe to 1. • This module sets PID to STALL (11) on receiving the data packet with the payload exceeding the maximum packet size of the pertinent pipe. • This module sets PID to NAK on detecting a USB bus reset when the function controller function is selected.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	• This module sets PID to NAK on detecting a receive error such as a CRC error three consecutive times when the host controller function is selected. • This module sets PID to STALL (11) on receiving the STALL handshake when the host controller function is selected. To specify each response type, set these bits as follows. • To make a transition from NAK (00) to STALL, set 10. • To make a transition from BUF (01) to STALL, set 11. • To make a transition from STALL (11) to NAK, set 10 and then 00. • To make a transition from STALL to BUF, set 00 (NAK) and then 01 (BUF).

Notes: 1. Only 0 can be read and 1 can be written to.
2. Only 1 can be written to.
3. The ACLRM, SQCLR, or SQSET bits should be set while CSSTS is 0 and PID is NAK and before the pipe is selected by the CURPIPE bits.
4. Before modifying ACLRM, SQCLR, or SQSET bits after modifying the PID bits from BUF to NAK, it should be checked that CSSTS and PBUSY for the selected pipe are 0. However, if the PID bits have been modified to NAK through hardware control, checking PBUSY is not necessary.

Table 31.17 Information Cleared by this Module by Setting ACLRM = 1

No.	Information Cleared by ACLRM Bit Manipulation	Cases in which Clearing the Information is Necessary
1	All the information in the FIFO buffer assigned to the pertinent pipe	
2	When the host controller function is selected, the interval count value when the pertinent pipe is for isochronous transfer	When the interval count value is to be reset
3	Values of the internal flags related to the BFRE bit	When the BFRE setting is modified
4	Values of the internal flags related to the transaction count	When the transaction count function is forcibly terminated

31.4.36 PIPEn Transaction Counter Enable Registers (PIPEnTRE) (n = 1 to 5)

PIPEnTRE is a register that enables or disables the transaction counter corresponding to PIPE1 to PIPE5, and clears the transaction counter.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	TRENB	0	R/W	Transaction Counter Enable Enables or disables the transaction counter. 0: The transaction counter is disabled. 1: The transaction counter is enabled. For the pipe in the receiving direction, setting this bit to 1 after setting the total number of the packets to be received in the TRNCNT bits through software allows this module to control hardware as described below on having received the number of packets equal to the set value in the TRNCNT bits. • In continuous transmission/reception mode (CNTMD = 1), this module switches the FIFO buffer to the CPU side even if the FIFO buffer is not full on completion of reception. • While SHTNAK is 1, this module modifies the PID bits to NAK for the corresponding pipe on having received the number of packets equal to the set value in the TRNCNT bits. • While BFRE is 1, this module asserts the BRDY interrupt on having received the number of packets equal to the set value in the TRNCNT bits and then reading out the last received data. For the pipe in the transmitting direction, set this bit to 0. When the transaction counter is not used, set this bit to 0. When the transaction counter is used, set the TRNCNT bits before setting this bit to 1. Set this bit to 1 before receiving the first packet to be counted by the transaction counter.
8	TRCLR	0	R/W	Transaction Counter Clear Clears the current value of the transaction counter corresponding to the pertinent pipe and then sets this bit to 0. 0: Invalid 1: The current counter value is cleared.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: Modify each bit in this register while CSSTS is 0 and PID is NAK. Before modifying each bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary.

31.4.37 PIPEn Transaction Counter Registers (PIPEnTRN) (n = 1 to 5)

PIPEnTRN is a transaction counter corresponding to PIPE1 to PIPE5.

This register is initialized by a power-on reset, but retains the set value by a USB bus reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRNCNT[15:0]															
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT[15:0]	All 0	R/W	Transaction Counter When written to: Specifies the number of transactions to be transferred through DMA. When read from: Indicates the specified number of transactions if TRENB is 0. Indicates the number of currently counted transaction if TRENB is 1.

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT[15:0]	All 0	R/W	This module increments the value of these bits by one when all of the following conditions are satisfied on receiving the packet. • TRENB is 1. • (TRNCNT set value ≠ current counter value + 1) on receiving the packet. • The payload of the received packet agrees with the set value in the MXPS bits. This module clears the value of these bits to 0 when any of the following conditions are satisfied. • All the following conditions are satisfied. TRENB is 1. (TRNCNT set value = current counter value + 1) on receiving the packet. The payload of the received packet agrees with the set value in the MXPS bits. • All the following conditions are satisfied. TRENB is 1. This module has received a short packet. • All the following conditions are satisfied. TRENB is 1. Software has set the TRCLR bit to 1. For the pipe in the transmitting direction, set these bits to 0. When the transaction counter is not used, set these bits to 0. Modify these bits while CSSTS is 0, PID is NAK, and TRENB is 0. Before modifying these bits after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, checking PBUSY through software is not necessary. To modify the value of these bits, set TRCLR to 1 before setting TRENB to 1.

31.4.38 Device Address n Configuration Registers (DEVADDn) (n = 0 to A)

DEVADDn is a register that specifies the address and port number of the hub to which the communication target peripheral device is connected and that also specifies the transfer speed of the peripheral device for PIPE0 to PIPE9.

When the host controller function is selected, this register should be set before starting communication using each pipe.

The bits in this register should be modified while no valid pipes are using the settings of this register. Valid pipes refer to the ones satisfying both of condition 1 and 2 below.

1. This register is selected by the DEVSEL bits as the communication target.
2. The PID bits are set to BUF for the selected pipe or the selected pipe is the DCP with SUREQ being 1.

This register is initialized by a power-on reset.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	UPPHUB[3:0]				HUBPORT[2:0]			USBSPD[1:0]		—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 11	UPPHUB[3:0]	0000	R/W	Address of Hub to which Communication Target is Connected Specifies the USB address of the hub to which the communication target peripheral device is connected. 0000: The peripheral device is directly connected to the port of this LSI. 0001 to 1010: USB address of the hub 1011 to 1111: Setting prohibited When the host controller function is selected, this module refers to the setting of these bits to generate packets for split transactions. When the function controller function is selected, set these bits to 0000.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	HUBPORT[2:0]	000	R/W	Port Number of Hub to which Communication Target is Connected Specifies the port number of the hub to which the communication target peripheral device is connected. 000: The peripheral device is directly connected to the port of this LSI. 001 to 111: Port number of the hub When the host controller function is selected, this module refers to the setting of these bits to generate packets for split transactions. When the function controller function is selected, set these bits to 000.
7, 6	USBSPD[1:0]	00	R/W	Transfer Speed of the Communication Target Device Specifies the USB transfer speed of the communication target peripheral device. 00: DEVADDn is not used. 01: Low speed 10: Full speed 11: High speed When the host controller function is selected, this module refers to the setting of these bits to generate packets. When the function controller function is selected, set these bits to 00.
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

31.4.39 USB Power Control Registers (UPONCRn) USB0 is n =0, USB1 is n =1.

UPONCRn is a register that controls Power of USB0, USB1. It makes possible to make Power of USB0 or USB1 OFF. For details, refer to (1) Power Control Procedures under section 31.5.1, System Control.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	UPONn[1:0]		—	—	—	—	—	—	—	—	—
Initial value :	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W :	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10,9	UPONn[1:0]	00	R/W	USB power control (n = 0 or 1) 00: Power off enable 01: Reserved 10: Reserved 11: Power on (During using USB)
8 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

31.5 Operation

31.5.1 System Control

This section describes the register operations that are necessary to the initial settings of this module (USB0 or USB1), and the registers necessary for power consumption control.

(1) Power Control Procedures

After Power ON of USB0 or USB1, it is necessary to set the following register bits before using USB0 or USB1.

In case of USB0;

- Set the bit10, 9 (initial value 0) of UPONCR0 register (Address: H'A40501D4) to 1.
- Except the above bits (bit15 to 11, 8 to 0) is reserved (initial value 0). The write value should always be 0.
- After using USB0, set bit10, 9 to 0 before USB0 Power OFF.

(Note) After power-on, set the bites 10 to 9 (initial value 0) to 1 when supplying power to USB0 even when USB0 is not being used.

In case of USB1;

- Set the bit10, 9 (initial value 0) of UPONCR1 register (Address: H' A4050192•to 1.
- Except the above bits (bit15 to 11, 8 to 0) is reserved (initial value 0). The write value should always be 0.
- After using USB1, set bit10, 9 to 0 before USB1 Power OFF.

(Note) Even if the USB0 module is not to be used, always set bites 9 and 10 (initial value 0) to 1 after power is supplied to USB0.

(2) Resets

Table 31.18 lists the types of controller resets. For the initialized states of the registers following the reset operations, see section 31.4, Register Description.

Table 31.18 Types of Reset

Name	Operation
Power-on reset	Low level input from the $\overline{\text{RESETP}}$ pin. Note: Power-on resets described in this manual using the $\overline{\text{RESETP}}$ pin
USB bus reset	Automatically detected by this module from the D+ and D− lines when the function controller function is selected

(3) Controller Function Selection

This module can select the host controller function or function controller function using the DCFM bit in SYSCFG. Changing the DCFM bit should be done in the initial settings immediately after a power-on reset or in the D+ pull-up disabled (DPRPU = 0) and D+ /D– pull-down disabled (DRPD = 0) state.

(4) Enabling High-Speed Operation

This module can select a USB communication speed (communication bit rate) using software. When the host controller function is selected, either of the high-speed operation or full-speed/low-speed operation can be selected. In order to enable the high-speed operation for this module, the HSE bit in SYSCFG should be set to 1. If high-speed mode has been enabled, this module executes the reset handshake protocol, and the USB communication speed is set automatically. The results of the reset handshake can be confirmed using the RHST bit in DVSTCTR.

If high-speed operation has been disabled, this module operates at full-speed or low-speed. If the function controller function is also selected, this module operates at full-speed.

Changing the HSE bit should be done between the ATTCH interrupt detection and bus reset execution when the host controller function is selected, or with the D+ line pull-up disabled (DPRPU = 0) when the host controller function is selected.

(5) USB Data Bus Resistor Control

Figure 31.2 shows a diagram of the connections between this module and the USB connectors.

This module incorporates a pull-up resistor for the D+ signal and a pull-down resistor for the D+ and D- signals. These signals can be pulled up or down using the DPRPU and DRPD bits in SYSCFG.

This module controls the terminal resistor for the D+ and D- signals during high-speed operation and the output resistor for the signals during full-speed operation. This module automatically switches the resistor after connection with the host controller or peripheral device by means of reset handshake, suspended state and resume detection.

When the function controller function is selected and the DPRPU bit in SYSCFG is cleared to 0 during communication with the host controller, the pull-up resistor (or the terminal resistor) of the USB data line is disabled, making it possible to notify the USB host of the device disconnection.

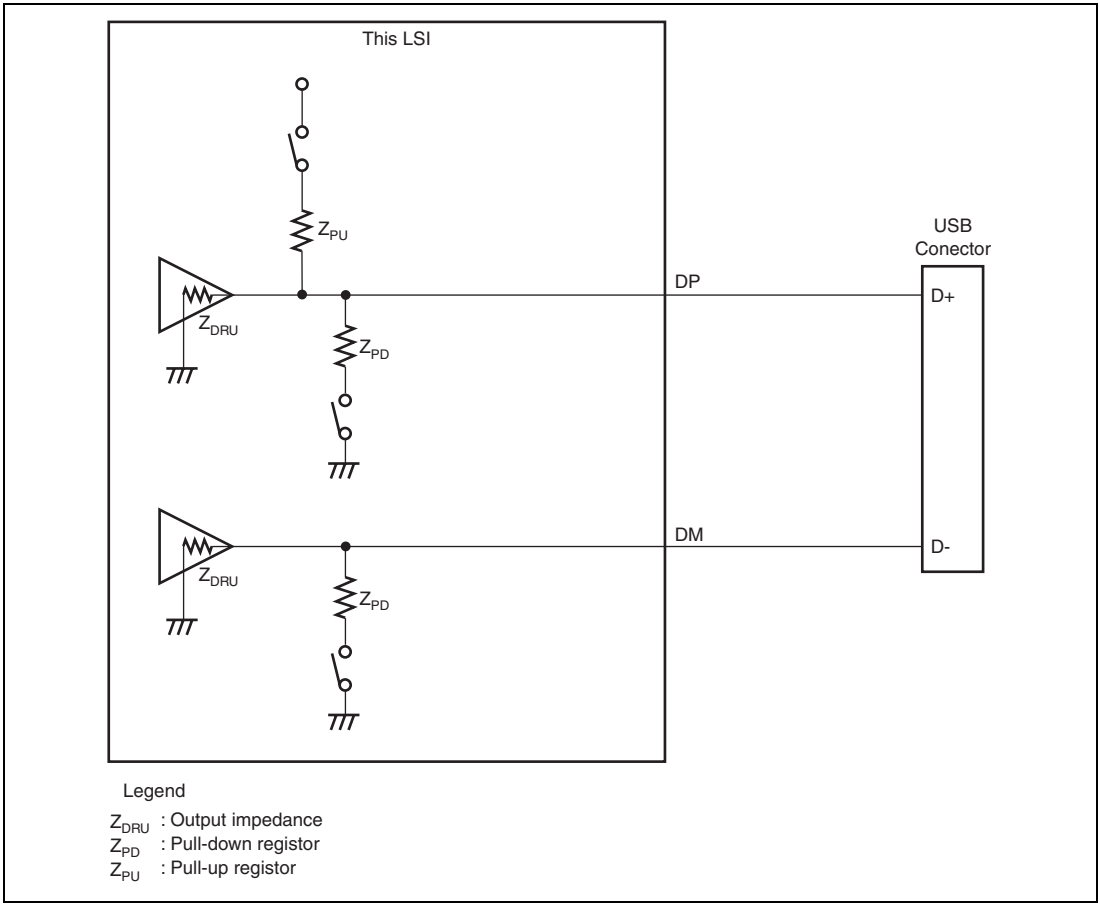


Figure 31.2 UBS Connector Connection

31.5.2 Interrupt Functions

Table 31.19 lists the interrupt generation conditions for this module.

When an interrupt generation condition is satisfied and the interrupt output is enabled using the corresponding interrupt enable register, this module issues a USB interrupt request to the INTC.

Table 31.19 Interrupt Generation Conditions

Bit	Interrupt Name	Cause of Interrupt	Function That Generates the Interrupt	Related Status
VBINT	VBUS interrupt	When a change in the state of the VBUS input pin has been detected (low to high or high to low)	Host, function	VBSTS
RESM	Resume interrupt	When a change in the state of the USB bus has been detected in the suspended state (J-state to K-state or J-state to SE0)	Function	—
SOFR	Frame number update interrupt	When the host controller function is selected: • When an SOF packet with a different frame number has been transmitted When the function controller function is selected: • SOFRM = 0: When an SOF packet with a different frame number is received • SOFRM = 1: When the SOF with the μframe number 0 cannot be received due to a corruption of a packet	Host, function	—

Bit	Interrupt Name	Cause of Interrupt	Function That Generates the Interrupt	Related Status
DVST	Device state transition interrupt	When a device state transition is detected • A USB bus reset detected • The suspend state detected • SET_ADDRESS request received • SET_CONFIGURATION request received	Function	DVSQ
CTRT	Control transfer stage transition interrupt	When a stage transition is detected in control transfer • Setup stage completed • Control write transfer status stage transition • Control read transfer status stage transition • Control transfer completed • A control transfer sequence error occurred	Function	CTSQ
BEMP	Buffer empty interrupt	• When transmission of all of the data in the buffer memory has been completed • When an excessive maximum packet size error has been detected	Host, Function	BEMPSTS.PIPEBEMP

Bit	Interrupt Name	Cause of Interrupt	Function That Generates the Interrupt	Related Status
NRDY	Buffer not ready interrupt	When the host controller function is selected: • When STALL is received from the peripheral side for the issued token • When a response cannot be received correctly from the peripheral side for the issued token (No response is returned three consecutive times or a packet reception error occurred three consecutive times.) • When an overrun/underrun occurred during isochronous transfer When the function controller function is selected: • When NAK is returned for an IN/OUT/PING token. • When a CRC error or a bit stuffing error occurred during data reception in isochronous transfer • When an overrun/underrun occurred during data reception in isochronous transfer	Host, function	NRDYSTS.PIPENRDY
BRDY	Buffer ready interrupt	When the buffer is ready (reading or writing is enabled)	Host, function	BRDYSYS PIPEBRDY
BCHG	Bus change interrupt	When a change of USB bus state is detected	Host, function	—
DTCH	Device disconnection detection	When disconnection of a peripheral device is detected.	Host	—

Bit	Interrupt Name	Cause of Interrupt	Function That Generates the Interrupt	Related Status
ATTCH	Device connection detection	When J-state or K-state is detected on the USB port for 2.5 μs. Used for checking whether a peripheral device is connected.	Host	—
EOFERR	EOF error detection	When EOF error of a peripheral device is detected	Host	—
SACK	Normal setup operation	When the normal response (ACK) for the setup transaction is received	Host	—
SIGN	Setup error	When a setup transaction error (no response or ACK packet corruption) is detected three consecutive times.	Host	—

Note: All the bits without register name indication are in INTSTS0.

Figure 31.3 shows a diagram relating to interrupts of this module.

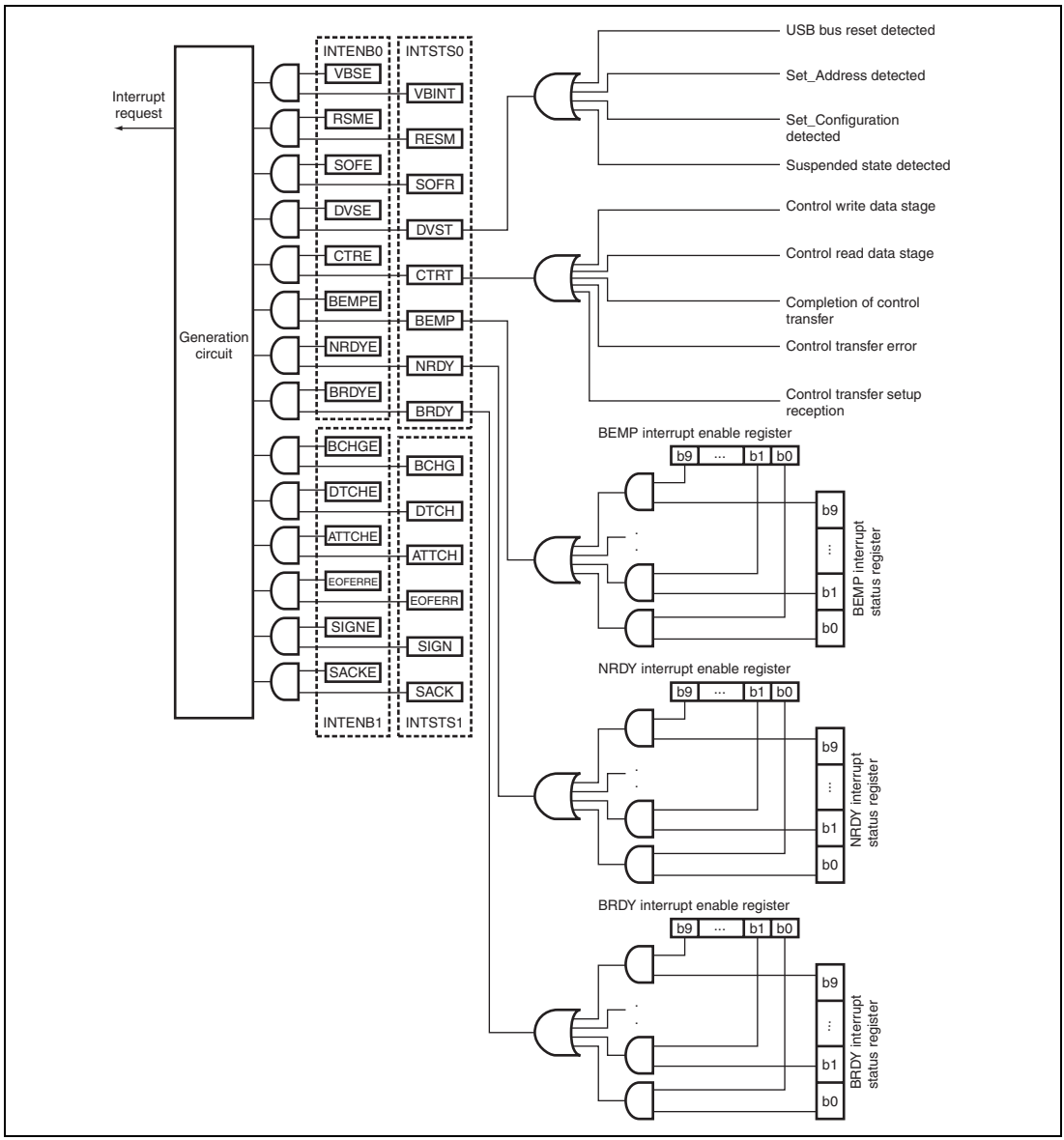


Figure 31.3 Items Relating to Interrupts

(1) BRDY Interrupt

The BRDY interrupt is generated when either of the host controller function or function controller function is selected. The following shows the conditions under which this module sets 1 to a corresponding bit in BRDYSTS. Under this condition, this module generates BRDY interrupt, if software sets the PIPEBRDYE bit in BRDYENB that corresponds to the pipe to 1 and the BRDYE bit in INTENB0 to 1.

The conditions for generating and clearing the BRDY interrupt depend on the settings of the BRDYM bit and BFRE bit for the pertinent pipe as described below.

(a) When the BRDYM bit is 0 and BFRE bit is 0

With these settings, the BRDY interrupt indicates that the FIFO port is accessible.

On any of the following conditions, this module generates the internal BRDY interrupt request trigger and sets 1 to the PIPEBRDY bit corresponding to the pertinent pipe.

(i) For the pipe in the transmitting direction:

- When software changes the DIR bit from 0 to 1.
- When packet transmission is completed using the pertinent pipe when write-access from the CPU to the FIFO buffer for the pertinent pipe is disabled (when the BSTS bit is read as 0).

  In continuous transmission/reception mode, the request trigger is generated on completion of transmitting data of one plane of the FIFO buffer.
- When one FIFO buffer is empty on completion of writing data to the other FIFO buffer in double buffer mode.

  The request trigger is not generated until completion of writing data to the currently-written FIFO buffer plane even if transmission to the other FIFO buffer is completed.
- When the hardware flushes the buffer of the pipe for isochronous transfers.
- When 1 is written to the ACLRM bit, which causes the FIFO buffer to make transition from the write-disabled to write-enabled state.

The request trigger is not generated for the DCP (that is, during data transmission for control transfers).

(ii) For the pipe in the receiving direction:

— When packet reception is completed successfully thus enabling the FIFO buffer to be read when read-access from the CPU to the FIFO buffer for the pertinent pipe is disabled (when the BSTS bit is read as 0).

The request trigger is not generated for the transaction in which DATA-PID disagreement occurs.

In continuous transmission/reception mode, the request trigger is not generated when the data is of the specified maximum packet size and the buffer has available space.

When a short packet is received, the request trigger is generated even if the FIFO buffer has available space.

When the transaction counter is used, the request trigger is generated on receiving the specified number of packets. In this case, the request trigger is generated even if the FIFO buffer has available space.

— When one FIFO buffer is read-enabled on completion of reading data from the other FIFO buffer in double buffer mode.

The request trigger is not generated until completion of reading data from the currently-read FIFO buffer plane even if reception by the other FIFO buffer is completed.

When the function controller function is selected, the BRDY interrupt is not generated in the status stage of control transfers.
The PIPEBRDY interrupt status of the pertinent pipe can be cleared to 0 by writing 0 to the corresponding PIPEBRDY interrupt status bit in the BRDYSTS register through software. In this case, 1s should be written to the PIPEBRDY interrupt status bits for the other pipes.
Be sure to clear the BRDY status before accessing the FIFO buffer.

(b) When the BRDYM bit is 0 and the BFRE bit is 1

With these settings, this module generates the BRDY interrupt on completion of reading all the data for a single transfer using the pipe in the receiving direction, and sets 1 to the PIPEBRDY bit corresponding to the pertinent pipe.

On any of the following conditions, this module determines that the last data for a single transfer has been received.

- When a short packet including a zero-length packet is received.
- When the transaction counter register (TRNCNT bits) is used and the number of packets specified by the TRNCNT bits are completely received.

When the pertinent data is completely read out after any of the above determination conditions has been satisfied, this module determines that all the data for a single transfer has been completely read out.

When a zero-length packet is received when the FIFO buffer is empty, this module determines that all the data for a single transfer has been completely read out upon passing the zero-length packet data to the CPU. In this case, to start the next transfer, write 1 to the BCLR bit in the corresponding FIFOCTR register through software.

With these settings, this module does not detect the BRDY interrupt for the pipe in the transmitting direction.

The PIPEBRDY interrupt status of the pertinent pipe can be cleared to 0 by writing 0 to the corresponding PIPEBRDY interrupt status bit through software. In this case, 1s should be written to the PIPEBRDY interrupt status bits for the other pipes.

In this mode, the BFRE bit setting should not be modified until all the data for a single transfer has been processed. When it is necessary to modify the BFRE bit before completion of processing, all the FIFO buffers for the pertinent pipe should be cleared using the ACLRM bit.

(c) When the BRDYM bit is 1 and the BFRE bit is 0

With these settings, the PIPEBRDY values are linked to the BSTS bit settings for each pipe. In other words, the BRDY interrupt status bits (PIPEBRDY) are set to 1 or 0 by this module depending on the FIFO buffer status.

(i) For the pipe in the transmitting direction:

The BRDY interrupt status bits are set to 1 when the FIFO buffer is write-enabled and are set to 0 when write-disabled.
However, the BRDY interrupt is not generated if the DCP in the transmitting direction is write-enabled.

(ii) For the pipe in the receiving direction:

The BRDY interrupt status bits are set to 1 when the FIFO buffer is read-enabled and are set to 0 when all the data have been read (read-disabled).
When a zero-length packet is received when the FIFO buffer is empty, the pertinent bit is set to 1 and the BRDY interrupt is continuously generated until BCLR = 1 is written through software. With this setting, the PIPEBRDY bit cannot be cleared to 0 through software. When BRDYM is set to 1, all of the BFRE bits (for all pipes) should be cleared to 0.

Figure 31.4 shows the timing at which the BRDY interrupt is generated.

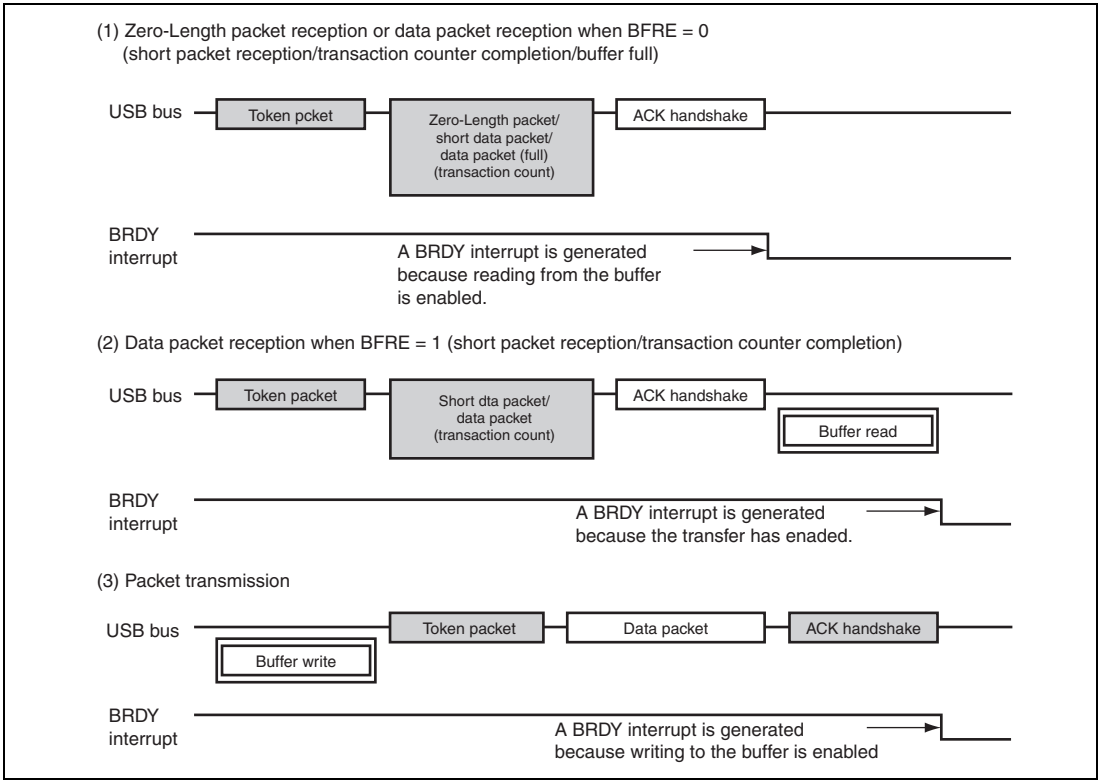


Figure 31.4 Timing at which a BRDY Interrupt is Generated

(2) NRDY Interrupt

On generating the internal NRDY interrupt request for the pipe whose PID bits are set to BUF by software, this module sets the corresponding PIPENRDY bit in NRDYSTS to 1. If the corresponding bit in NRDYENB is set to 1 by software, this module sets the NRDY bit in INTSTS0 to 1, allowing the USB interrupt to be generated.

The following describes the conditions on which this module generates the internal NRDY interrupt request for a given pipe.

However, the internal NRDY interrupt request is not generated during setup transaction execution when the host controller function is selected. During setup transactions when the host controller function is selected, the SACK or SIGN interrupt is detected.

The internal NRDY interrupt request is not generated during status stage execution of the control transfer when the function controller function is selected.

(a) When the host controller function is selected and when the connection is used in which no split transactions occur

(i) For the pipe in the transmitting direction:

On any of the following conditions, this module detects the NRDY interrupt.

- For the pipe for isochronous transfers, when the time to issue an OUT token comes in a state in which there is no data to be transmitted in the FIFO buffer.

  In this case, this module transmits a zero-length packet following the OUT token, setting the corresponding PIPENRDY bit and the OVRN bit to 1.
- During communications other than setup transactions using the pipe for the transfers other than isochronous transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the peripheral device (when timeout is detected before detection of the handshake packet from the peripheral device) and 2) an error is detected in the packet from the peripheral device.

  In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK.
- During communications other than setup transactions, when the STALL handshake is received from the peripheral device (including the STALL handshake in response to PING in addition to the STALL handshake in response to OUT).

  In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to STALL (11).

(ii) For the pipe in the receiving direction

— For the pipe for isochronous transfers, when the time to issue an IN token comes in a state in which there is no space available in the FIFO buffer.

In this case, this module discards the received data for the IN token, setting the PIPENRDY bit of the corresponding pipe and the OVRN bit to 1.

When a packet error is detected in the received data for the IN token, this module also sets the CRCE bit to 1.

— For the pipe for the transfers other than isochronous transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the peripheral device for the IN token issued by this module (when timeout is detected before detection of the DATA packet from the peripheral device) and 2) an error is detected in the packet from the peripheral device.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK.

— For the pipe for isochronous transfers, when no response is returned from the peripheral device for the IN token (when timeout is detected before detection of the DATA packet from the peripheral device) or an error is detected in the packet from the peripheral device.

In this case, this module sets the corresponding PIPENRDY bit to 1. (The setting of the PID bits of the corresponding pipe to NAK is not modified.)

— For the pipe for isochronous transfers, when a CRC error or a bit stuffing error is detected in the received data packet.

In this case, this module sets the corresponding PIPENRDY bit and CRCE bit to 1.

— When the STALL handshake is received.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to STALL.

(b) When the host controller function is selected and when the connection is used in which split transactions occur

(i) For the pipe in the transmitting direction:

— For the pipe for isochronous transfers, when the time to issue an OUT token comes in a state in which there is no data to be transmitted in the FIFO buffer.

In this case, this module transmits a zero-length packet following the OUT token, setting the corresponding PIPENRDY bit and the OVRN bit to 1 at the issuance of the start-split transaction (S-SPLIT).

— For the pipe for the transfers other than isochronous transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the HUB for the S-SPLIT or complete-split transaction (C-SPLIT) (when timeout is detected before detection of the handshake packet from the HUB) and 2) an error is detected in the packet from the HUB.

In this case, this module sets the PIPENRDY bit of the corresponding pipe to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK.

If the NRDY interrupt is detected when the C-SPLIT is issued, this module clears the CSSTS bit to 0.

— When the STALL handshake is received in response to the C-SPLIT.

In this case, this module sets the corresponding PIPENRDY bit to 1, modifies the setting of the PID bits of the corresponding pipe to STALL (11) and clears the CSSTS bit to 0.

This interrupt is not detected for SETUP transactions.

— When the NYET is received in response to the C-SPLIT and the microframe number = 4.

In this case, this module sets the corresponding PIPENRDY bit to 1 and clears the CSSTS bit to 0 (does not modify the setting of the PID bits).

(ii) For the pipe in the receiving direction:

— For the pipe for isochronous transfers, when the time to issue an IN token comes in a state in which there is no space available in the FIFO buffer.

In this case, this module discards the received data for the IN token, setting the corresponding PIPENRDY bit and the OVRN bit to 1 at the issuance of the S-SPLIT.

— During bulk-pipe transfers or the transfers other than SETUP transactions with the DCP, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the HUB for the IN token issued by this module at the issuance of S-SPLIT or C-SPLIT (when timeout is detected before detection of the DATA packet from the HUB) and 2) an error is detected in the packet from the HUB.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK. When the condition is generated during the C-SPLIT transaction, this module clears the CSSTS bit to 0.

— During the C-SPLIT transaction for the pipe for isochronous transfers or interrupt transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the HUB for the IN token issued by this module (when timeout is detected before detection of the DATA packet from the HUB) and 2) an error is detected in the packet from the HUB.

On generating this condition for the pipe for interrupt transfers, this module sets the corresponding PIPENRDY bit to 1, modifies the setting of the PID bits of the corresponding pipe to NAK and clears the CSSTS bit to 0.

On generating this condition for the pipe for isochronous transfers, this module sets the corresponding PIPENRDY bit to 1 and CRCE bit to 1, and clears the CSSTS bit to 0 (does not modify the setting of the PID bits).

— During the C-SPLIT transaction, when the STALL handshake is received for the pipe for the transfers other than isochronous transfers.

In this case, this module sets the corresponding PIPENRDY bit to 1, modifies the setting of the PID bits of the corresponding pipe to STALL (11) and clears the CSSTS bit to 0.

— During the C-SPLIT transaction, when the NYET handshake is received for the pipe for the isochronous transfers or interrupt transfers and the microframe number = 4.

In this case, this module sets the corresponding PIPENRDY bit to 1 and CRCE bit to 1, and clears the CSSTS bit to 0 (does not modify the setting of the PID bits).

(c) When the function controller function is selected

(i) For the pipe in the transmitting direction:

— On receiving an IN token when there is no data to be transmitted in the FIFO buffer.

In this case, this module generates a NRDY interrupt request at the reception of the IN token, setting the PIPENRDY bit to 1. For the pipe for the isochronous transfers in which an interrupt is generated, this module transmits a zero-length packet, setting the OVRN bit to 1.

(ii) For the pipe in the receiving direction:

— On receiving an OUT token when there is no space available in the FIFO buffer.

For the pipe for the isochronous transfers in which an interrupt is generated, this module generates a NRDY interrupt request, setting the PIPENRDY bit to 1 and OVRN bit to 1.

For the pipe for the transfers other than isochronous transfers in which an interrupt is generated, this module generates a NRDY interrupt request when a NAK handshake is transferred after the data following the OUT token was received, setting the PIPENRDY bit to 1.

However, during re-transmission (due to DATA-PID disagreement), the NRDY interrupt request is not generated. In addition, if an error occurs in the DATA packet, the NRDY interrupt request is not generated.

— On receiving a PING token when there is no space available in the FIFO buffer.

In this case, this module generates a NRDY interrupt request at the reception of the PING token, setting the PIPENRDY bit to 1.

— For the pipe for isochronous transfers, when a token is not received normally within an interval frame.

In this case, this module generates a NRDY interrupt request, setting the PIPENRDY bit to 1.

Figure 31.5 shows the timing at which an NRDY interrupt is generated when the function controller function is selected.

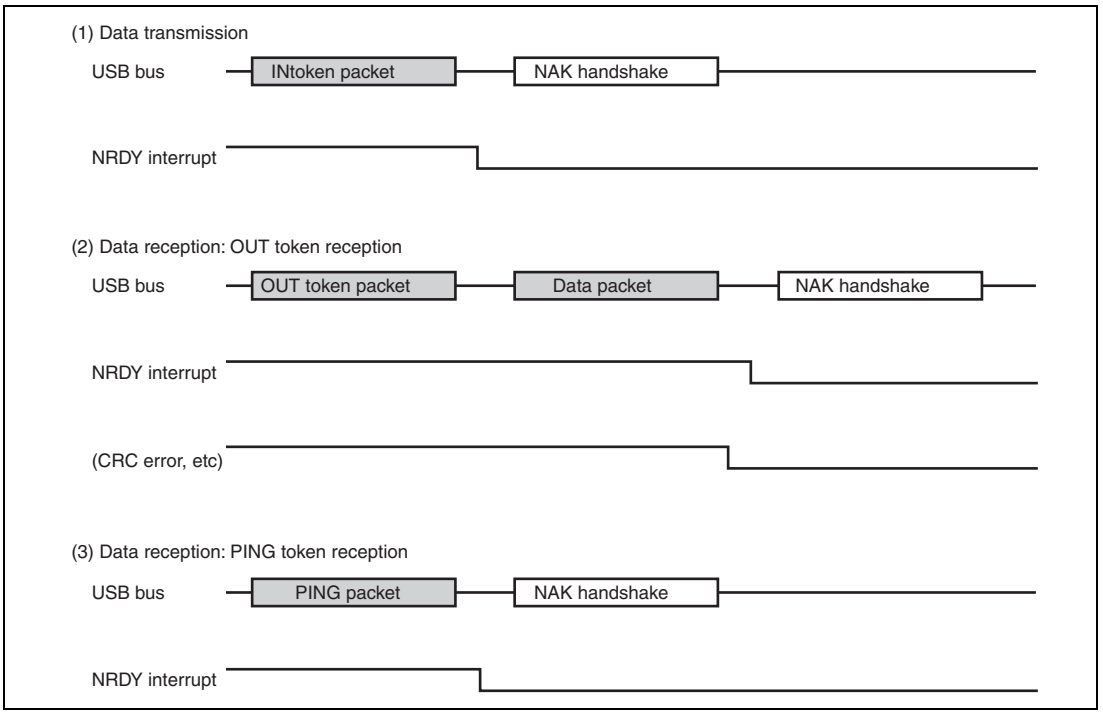


Figure 31.5 Timing at which NRDY Interrupt is Generated when Function Controller Function is Selected

(3) BEMP Interrupt

On generating the BEMP interrupt for the pipe whose PID bits are set to BUF by software, this module sets the corresponding PIPEBEMP bit in BEMPSTS to 1. If the corresponding bit in BEMPENB is set to 1 by software, this module sets the BEMP bit in INTSTS0 to 1, allowing the USB interrupt to be generated.

The following describes the conditions on which this module generates the internal BEMP interrupt request.

(a) For the pipe in the transmitting direction, when the FIFO buffer of the corresponding pipe is empty on completion of transmission (including zero-length packet transmission). In single buffer mode, the internal BEMP interrupt request is generated simultaneously with the BRDY interrupt for the pipe other than DCP. However, the internal BEMP interrupt request is not generated on any of the following conditions.

— When software (DMAC) has already started writing data to the FIFO buffer of the CPU on completion of transmitting data of one plane in double buffer mode.
— When the buffer is cleared (emptied) by setting the ACLRM or BCLR bit to 1.
— When IN transfer (zero-length packet transmission) is performed during the control transfer status stage in function controller mode.

(b) For the pipe in the receiving direction:
When the successfully-received data packet size exceeds the specified maximum packet size. In this case, this module generates the BEMP interrupt request, setting the corresponding PIPEBEMP bit to 1, and discards the received data and modifies the setting of the PID bits of the corresponding pipe to STALL (11).
Here, this module returns no response when used as the host controller, and returns STALL response when used as the function controller.
However, the internal BEMP interrupt request is not generated on any of the following conditions.

— When a CRC error or bit stuffing error is detected in the received data.
— When a setup transaction is being performed.Writing 0 to the PIPEBEMP bit clears the status; writing 1 to the PIPEBEMP bit has no effect.

Figure 31.6 shows the timing at which a BEMP interrupt is generated when the function controller function has been selected.

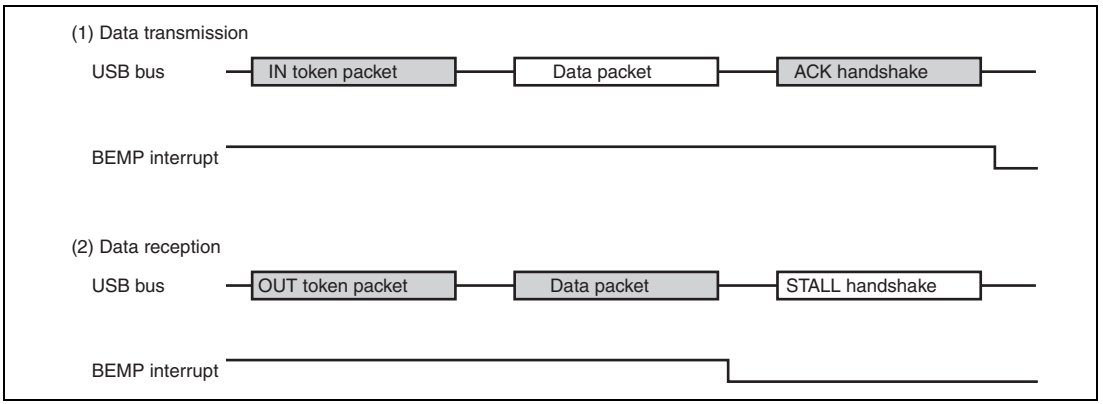


Figure 31.6 Timing at which BEMP Interrupt is Generated when Function Controller Function is Selected

(4) Device State Transition Interrupt

Figure 31.7 shows a diagram of this module device state transitions. This module controls device states and generates device state transition interrupts. However, recovery from the suspended state (resume signal detection) is detected by means of the resume interrupt. The device state transition interrupts can be enabled or disabled individually using INTENB0. The device state that made a transition can be confirmed using the DVSQ bit in INTSTS0.

To make a transition to the default state, the device state transition interrupt is generated after the reset handshake protocol has been completed.

Device state can be controlled only when the function controller function is selected. Also, the device state transition interrupts can be generated only when the function controller function is selected.

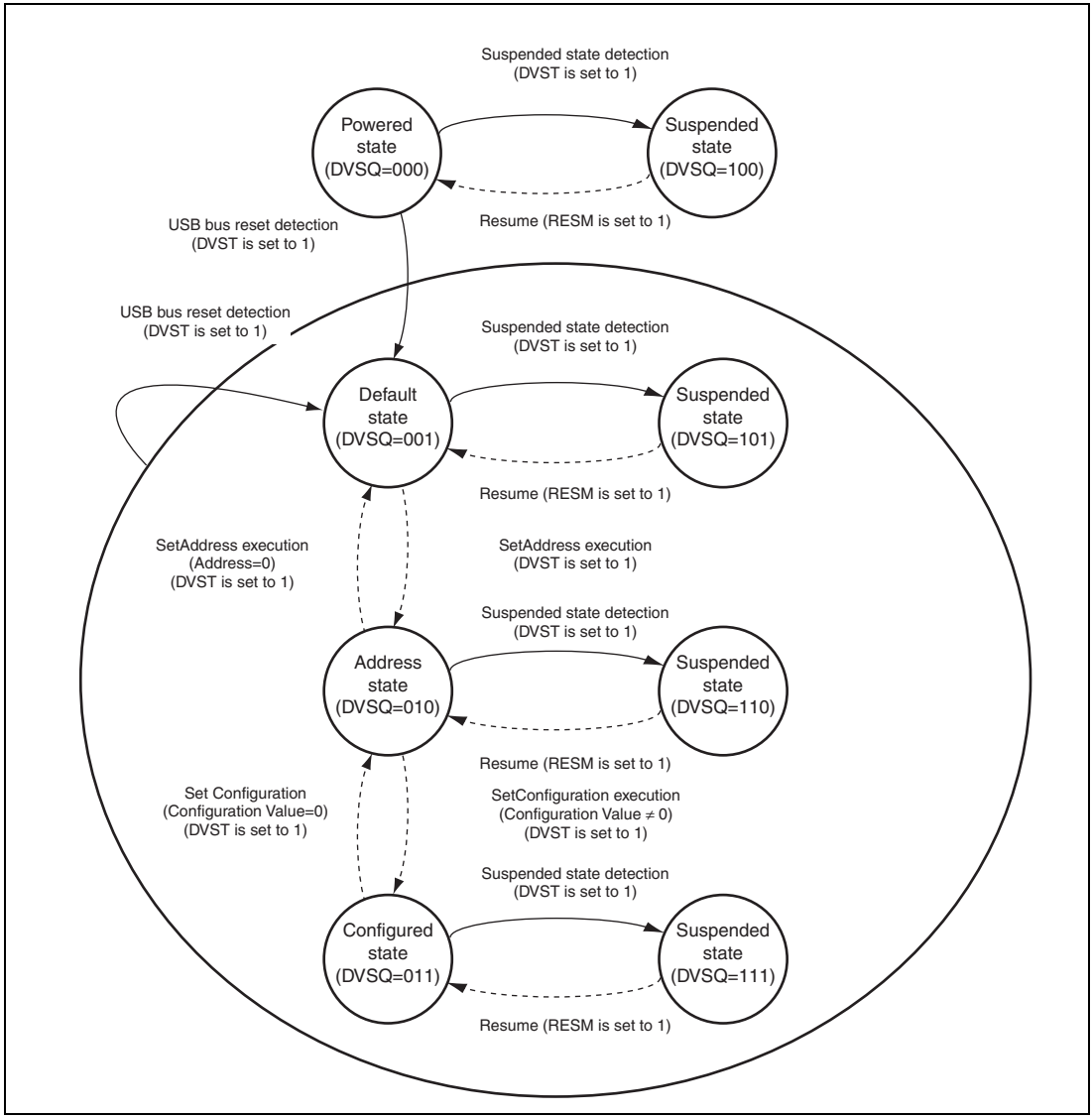


Figure 31.7 Device State Transitions

(5) Control Transfer Stage Transition Interrupt

Figure 31.8 shows a diagram of how this module handles the control transfer stage transition. This module controls the control transfer sequence and generates control transfer stage transition interrupts. Control transfer stage transition interrupts can be enabled or disabled individually using INTENB0. The transfer stage that made a transition can be confirmed using the CTSQ bit in INTSTS0.

The control transfer stage transition interrupts are generated only when the function controller function is selected.

The control transfer sequence errors are described below. If an error occurs, the PID bit in DCPCTR is set to B'1x (STALL).

(a) During control read transfers

- At the IN token of the data stage, an OUT or PING token is received when there have been no data transfers at all.
- An IN token is received at the status stage
- A packet is received at the status stage for which the data packet is DATAPID = DATA0

(b) During control write transfers

- At the OUT token of the data stage, an IN token is received when there have been no ACK response at all
- A packet is received at the data stage for which the first data packet is DATAPID = DATA0
- At the status stage, an OUT or PING token is received

(c) During no-data control transfers

- At the status stage, an OUT or PING token is received

At the control write transfer stage, if the number of receive data exceeds the wLength value of the USB request, it cannot be recognized as a control transfer sequence error. At the control read transfer status stage, packets other than zero-length packets are received by an ACK response and the transfer ends normally.

When a CTRT interrupt occurs in response to a sequence error (SERR = 1), the CTSQ = 110 value is retained until CTRT = 0 is written from the system (the interrupt status is cleared). Therefore, while CTSQ = 110 is being held, the CTRT interrupt that ends the setup stage will not be

generated even if a new USB request is received. (This module retains the setup stage end, and after the interrupt status has been cleared by software, a setup stage end interrupt is generated.)

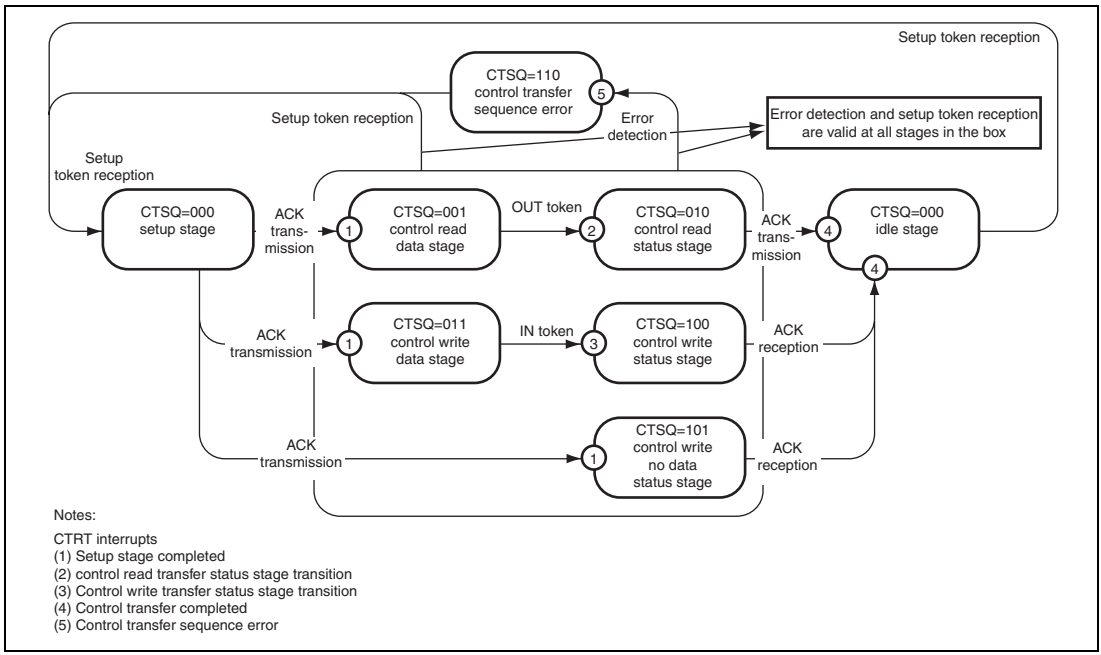


Figure 31.8 Control Transfer Stage Transitions

(6) Frame Update Interrupt

Figure 31.9 shows an example of the SOFR interrupt output timing of this module. With the host controller function selected, an interrupt is generated at the timing at which the frame number is updated. With the function controller function selected, the SOFR interrupt is generated when the frame number is updated.

When the function controller function is selected, this module updates the frame number and generates an SOFR interrupt if it detects a new SOF packet during full-speed operation. During high-speed operation, however, this module does not update the frame number, or generates no SOFR interrupt until the module enters the μSOF locked state. Also, the SOF interpolation function is not activated. The μSOF lock state is the state in which μSOF packets with different frame numbers are received twice continuously without error occurrence.

The conditions under which the μSOF lock monitoring begins and stops are as follows.

1. Conditions under which μSOF lock monitoring begins
   USBE = 1

2. Conditions under which µSOF lock monitoring stops
   USBE = 0, a USB bus reset is received, or suspended state is detected.

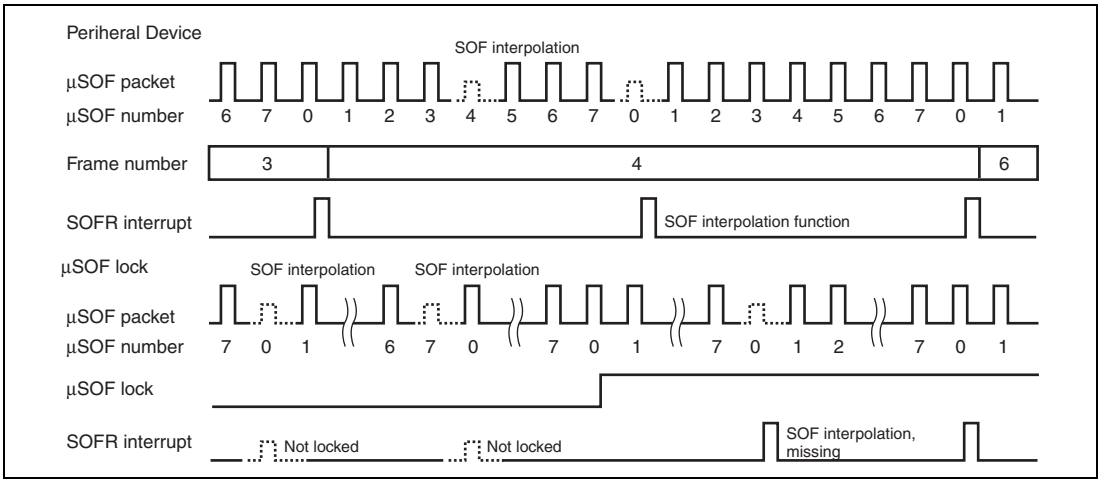


Figure 31.9 Example of SOFR Interrupt Output Timing

(7) VBUS Interrupt

If there has been a change in the VBUS pin, the VBUS interrupt is generated. The level of the VBUS pin can be checked with the VBSTS bit in INTSTS0. Whether the host controller is connected or disconnected can be confirmed using the VBUS interrupt. However, if the system is activated with the host controller connected, the first VBUS interrupt is not generated because there is no change in the VBUS pin.

(8) Resume Interrupt

The RESM interrupt is generated when the device state is the suspended state, and the USB bus state has changed (from J-state to K-state, or from J-state to SE0). Recovery from the suspended state is detected by means of the resume interrupt.

(9) BCHG Interrupt

The BCHG interrupt is generated when the USB bus state has changed. The BCHG interrupt can be used to detect whether or not the peripheral device is connected when the host controller function has been selected and can also be used to detect a remote wakeup. The BCHG interrupt is generated regardless of whether the host controller function or function controller function has been selected.

(10) DTCH Interrupt

The DTCH interrupt is generated if disconnection of the USB bus is detected when the host controller function has been selected. This module detects bus disconnection based on USB Specification 2.0.

After detecting the DTCH interrupt, this module controls hardware as described below (irrespective of the set value of the corresponding interrupt enable bit). Software should terminate all the pipes in which communications are currently carried out for the pertinent port and make a transition to the wait state for bus connection to the pertinent port (wait state for ATTCH interrupt generation).

(a) Modifies the UACT bit for the port in which a DTCH interrupt has been detected to 0.

(b) Puts the port in which a DTCH interrupt has been generated into the idle state.

Note: It may take 5ms at the longest that the DTCH interrupt occurs after disconnecting the cable. During the period, the NRDY interrupt may occur earlier.

(11) SACK Interrupt

The SACK interrupt is generated when an ACK response for the transmitted setup packet has been received from the peripheral device with the host controller function selected. The SACK interrupt can be used to confirm that the setup transaction has been completed successfully.

(12) SIGN Interrupt

The SIGN interrupt is generated when an ACK response for the transmitted setup packet has not been correctly received from the peripheral device three consecutive times with the host controller function selected. The SIGN interrupt can be used to detect no ACK response transmitted from the peripheral device or corruption of an ACK packet.

(13) ATTCH Interrupt

The ATTCH interrupt is generated when J-state or K-state of the full-speed or low-speed level signal is detected on the USB port for 2.5 μs in host controller mode. To be more specific, the ATTCH interrupt is detected on any of the following conditions.

(a) When K-state, SE0, or SE1 changes to J-state, and J-state continues 2.5 μs.

(b) When J-state, SE0, or SE1 changes to K-state, and K-state continues 2.5 μs.

(14) EOFERR Interrupt

The EOFERR interrupt is generated when it is detected that communication is not completed at the EOF2 timing prescribed by USB Specification 2.0.

After detecting the EOFERR interrupt, this module controls hardware as described below (irrespective of the set value of the corresponding interrupt enable bit). Software should terminate all the pipes in which communications are currently carried out for the pertinent port and perform re-enumeration of the pertinent port.

(a) Modifies the UACT bit for the port in which an EOFERR interrupt has been detected to 0.

(b) Puts the port in which an EOFERR interrupt has been generated into the idle state.

31.5.3 Pipe Control

Table 31.20 lists the pipe setting items of this module. With USB data transfer, data transmission has to be carried out using the logic pipe called the endpoint. This module has ten pipes that are used for data transfer.

Settings should be entered for each of the pipes in conjunction with the specifications of the system.

Table 31.20 Pipe Setting Items

Register Name	Bit Name	Setting Contents	Remarks
DCPCFG PIPECFG	TYPE	Specifies the transfer type	PIPE1 to PIPE9: Can be set
	BFRE	Selects the BRDY interrupt mode	PIPE1 to PIPE5: Can be set
	DBLB	Selects a double buffer	PIPE1 to PIPE5: Can be set
	CNTMD	Selects continuous transfer or non-continuous transfer	PIPE1 and PIPE2: Can be set (only when bulk transfer has been selected). PIPE3 to PIPE5: Can be set
	DIR	Selects transfer direction	IN or OUT can be set
	EPNUM	Endpoint number	PIPE1 to PIPE9: Can be set A value other than 0000 should be set when the pipe is used.
	SHTNAK	Selects disabled state for pipe when transfer ends	PIPE1 and PIPE2: Can be set (only when bulk transfer has been selected) PIPE3 to PIPE5: Can be set
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Cannot be set (fixed at 256 bytes) PIPE1 to PIPE5: Can be set (a maximum of 2 kbytes can be specified) PIPE6 to PIPE9: Cannot be set (fixed at 64 bytes)
	BUFNMB	Buffer memory number	DCP: Cannot be set (areas fixed at H'0 to H'3) PIPE1 to PIPE5: Can be set (can be specified in areas H'8 to H'7F) PIPE6 to PIPE9: Cannot be set (areas fixed at H'4 to H'7)

Register Name	Bit Name	Setting Contents	Remarks
DCPMAXP PIPEMAXP	DEVSEL	Selects a device	Referenced only when the host controller function is selected.
	MXPS	Maximum packet size	Compliant with the USB standard.
PIPEPERI	IFIS	Buffer flush	PIPE1 and PIPE2: Can be set (only when isochronous transfer has been selected) PIPE3 to PIPE5: Cannot be set PIPE6 to PIPE9: Can be set (only when the host controller function has been selected)
	IITV	Interval counter	PIPE1 and PIPE2: Can be set (only when isochronous transfer has been selected) PIPE3 to PIPE5: Cannot be set PIPE6 to PIPE9: Can be set (only when the host controller function has been selected)
DCPCTR PIPEnCTR	BSTS	Buffer status	For the DCP, receive buffer status and transmit buffer status are switched with the ISEL bit.
	INBUFM	IN buffer monitor	Mounted for PIPE3 to PIPE5.
	SUREQ	SETUP request	Can be set only for the DCP. Can be controlled only when the host controller function has been selected.
	SUREQCLR	SUREQ clear	Can be set only for the DCP. Can be controlled only when the host controller function has been selected.
	CSCLR	CSSTS clear	Can be controlled only when the host controller function has been selected.
	CSSTS	SPLIT status indication	Can be referenced only when the host controller function has been selected.
	ATREPM	Auto response mode	PIPE1 to PIPE5: Can be set Can be controlled only when the function controller function has been selected.

Register Name	Bit Name	Setting Contents	Remarks
DCPCTR PIPEnCTR	ACLRM	Auto buffer clear	PIPE1 to PIPE9: Can be set
	SQCLR	Sequence clear	Clears the data toggle bit
	SQSET	Sequence set	Sets the data toggle bit
	SQMON	Sequence monitor	Monitors the data toggle bit
	PBUSY	Pipe busy status	
	PID	Response PID	See section 31.5.3 (6), Response PID
PIPEnTRE	TRENB	Transaction counter enable	PIPE1 to PIPE5: Can be set
	TRCLR	Current transaction counter clear	PIPE1 to PIPE5: Can be set
PIPEnTRN	TRNCNT	Transaction counter	PIPE1 to PIPE5: Can be set

(1) Pipe control register switching procedures

The following bits in the pipe control registers can be modified only when USB communication is disabled (PID = NAK):

Registers that Should Not be Set in the USB Communication Enabled (PID = BUF) State

- Bits in DCPCFG and DCPMAXP
- The SQCLR and SQSET bits in DCPCTR
- Bits in PIPECFG, PIPEBUF, PIPEMAXP and PIPEPERI
- The ATREPM, ACLRM, SQCLR and SQSET bits in PIPExCTR
- Bits in PIPExTRE and PIPExTRN

In order to modify the above bits from the USB communication enabled (PID = BUF) state, follow the procedure shown below:

1. Generate a bit modification request with the pipe control register.
2. Modify the PID corresponding to the pipe to NAK.
3. Wait until the corresponding CSSTS bit is cleared to 0 (only when the host controller function has been selected).
4. Wait until the corresponding PBUSY bit is cleared to 0.

5. Modify the bits in the pipe control register.

The following bits in the pipe control registers can be modified only when the pertinent information has not been set by the CURPIPE bits in CFIFOSEL, D0FIFOSEL and D1FIFOSEL.

Registers that Should Not be Set When CURPIPE in FIFO-PORT is set.

- Bits in DCPCFG and DCPMAXP
- Bits in PIPECFG, PIPEBUF, PIPEMAXP and PIPEPERI

In order to modify pipe information, the CURPIPE bits should be set to the pipes other than the pipe to be modified. For the DCP, the buffer should be cleared using BCLR after the pipe information is modified.

(2) Transfer Types

The TYPE bit in PIPECFG is used to specify the transfer type for each pipe. The transfer types that can be set for the pipes are as follows.

1. DCP: No setting is necessary (fixed at control transfer).
2. PIPE1 and PIPE2: These should be set to bulk transfer or isochronous transfer.
3. PIPE3 to PIPE5: These should be set to bulk transfer.
4. PIPE6 to PIPE9: These should be set to interrupt transfer.

(3) Endpoint Number

The EPNUM bit in PIPECFG is used to set the endpoint number for each pipe. The DCP is fixed at endpoint 0. The other pipes can be set from endpoint 1 to endpoint 15.

1. DCP: No setting is necessary (fixed at end point 0).
2. PIPE1 to PIPE9: The endpoint numbers from 1 to 15 should be selected and set.
   These should be set so that the combination of the DIR bit and EPNUM bit is unique.

(4) Maximum Packet Size Setting

The MXPS bit in DCPMAXP and PIPEMAXP is used to specify the maximum packet size for each pipe. DCP and PIPE1 to PIPE5 can be set to any of the maximum pipe sizes defined by the USB specification. For PIPE6 to PIPE9, 64 bytes are the upper limit of the maximum packet size. The maximum packet size should be set before beginning the transfer (PID = BUF).

1. DCP: 64 should be set when using high-speed operation.
2. DCP: Select and set 8, 16, 32, or 64 when using full-speed operation.
3. PIPE1 to PIPE5: 512 should be set when using high-speed bulk transfer.
4. PIPE1 to PIPE5: Select and set 8, 16, 32, or 64 when using full-speed bulk transfer.
5. PIPE1 and PIPE2: Set a value between 1 and 1024 when using high-speed isochronous transfer.
6. PIPE1 and PIPE2: Set a value between 1 and 1023 when using full-speed isochronous transfer.
7. PIPE6 to PIPE9: Set a value between 1 and 64.

The high bandwidth transfers used with interrupt transfers and isochronous transfers are not supported.

(5) Transaction Counter (For PIPE1 to PIPE5 in Reading Direction)

When the specified number of transactions have been completed in the data packet receiving direction, this module recognizes that the transfer has ended. The transaction counter function is available when the pipes assigned to the D0FIFO/D1FIFO port have been set in the direction of reading data from the buffer memory. Two transaction counters are provided: one is the TRNCNT register that specifies the number of transactions to be executed and the other is the current counter that internally counts the number of executed transactions. When the current counter value matches the number of the transactions specified in TRNCNT, reading the buffer memory is enabled. The current counter of the transaction counter function is initialized by the TRCLR bit, so that the transactions can be counted again starting from the beginning. The information read from TRNCNT differs depending on the setting of the TRENB bit.

- TRENB = 0: The specified transaction counter value can be read.
- TRENB = 1: The current counter value indicating the internally counted number of executed transactions can be read.

When operating the TRCLR bit, the following should be noted.

- If the transactions are being counted and PID = BUF, the current counter cannot be cleared.
- If there is any data left in the buffer, the current counter cannot be cleared.

(6) Response PID

The PID bits in DCPCTR and PIPEnCTR are used to set the response PID for each pipe.

The following shows this module operation with various response PID settings:

(a) Response PID settings when the host controller function is selected:

The response PID is used to specify the execution of transactions.

(i) NAK setting: Using pipes is disabled. No transaction is executed.

(ii) BUF setting: Transactions are executed based on the status of the buffer memory. For OUT direction: If there are transmit data in the buffer memory, an OUT token is issued.
For IN direction: If there is an area to receive data in the buffer memory, an IN token is issued.

(iii) STALL setting: Using pipes is disabled. No transaction is executed.

Setup transactions for the DCP are set with the SUREQ bit.

(b) Response PID settings when the function controller function is selected:

The response PID is used to specify the response to transactions from the host.

(i) NAK setting: The NAK response is always returned in response to the generated transaction.

(ii) BUF setting: Responses are made to transactions based on the status of the buffer memory.

(iii) STALL setting: The STALL response is always returned in response to the generated transaction.

For setup transactions, an ACK response is always returned, regardless of the PID setting, and the USB request is stored in the register.

This module may carry out writing to the PID bits, depending on the results of the transaction.

(c) When the host controller function has been selected and the response PID is set by hardware:

(i) NAK setting: In the following cases, PID = NAK is set and issuing of tokens is automatically stopped:

- When a transfer other than isochronous transfer has been performed and the NRDY interrupt is generated. (For details, see descriptions of the NRDY interrupt.)
- If a short packet is received when the SHTNAK bit in PIPECFG has been set to 1 for bulk transfer.
- If the transaction counter ended when the SHTNAK bit has been set to 1 for bulk transfer.

(ii) BUF setting: There is no BUF writing by this module.

(iii) STALL setting: In the following cases, PID = STALL is set and issuing of tokens is automatically stopped:

- When STALL is received in response to the transmitted token.
- When the size of the receive data packet exceeds the maximum packet size.

(d) When the function controller function has been selected and the response PID is set by hardware:

(i) NAK setting: In the following cases, PID = NAK is set and NAK is always returned in response to transactions:

- When the SETUP token is received normally (DCP only).
- If the transaction counter ended or a short packet is received when the SHTNAK bit in PIPECFG has been set to 1 for bulk transfer.

(ii) BUF setting: There is no BUF writing by this module.

(iii) STALL setting: In the following cases, PID = STALL is set and STALL is always returned in response to transactions:

- When the size of the receive data packet exceeds the maximum packet size.
- When a control transfer sequence error has been detected (DCP only).

(7) Data PID Sequence Bit

This module automatically toggles the sequence bit in the data PID when data is transferred normally in the control transfer data stage, bulk transfer and interrupt transfer. The sequence bit of the data PID that was transmitted can be confirmed with the SQMON bit in DCPCTR and PIPEnCTR. When data is transmitted, the sequence bit switches at the timing at which the ACK handshake is received. When data is received, the sequence bit switches at the timing at which the ACK handshake is transmitted. The SQCLR bit in DCPCTR and the SQSET bit in PIPEnCTR can be used to change the data PID sequence bit.

When the function controller function has been selected and control transfer is used, this module automatically sets the sequence bit when a stage transition is made. DATA0 is returned when the setup stage is ended and DATA1 is returned in a status stage. Therefore, software settings are not required. However, when the host controller function has been selected and control transfer is used, the sequence bit should be set by software at the stage transition.

For the Clearfeature request transmission or reception, the data PID sequence bit should be set by software, regardless of whether the host controller function or function controller function is selected.

With pipes for which isochronous transfer has been set, sequence bit operation cannot be carried out using the SQSET bit.

(8) Response PID = NAK Function

This module has a function that disables pipe operation (PID response = NAK) at the timing at which the final data packet of a transaction is received (this module automatically distinguishes this based on reception of a short packet or the transaction counter) by setting the SHTNAK bit in PIPECFG to 1.

When a double buffer is being used for the buffer memory, using this function enables reception of data packets in transfer units. If pipe operation has disabled, the pipe has to be set to the enabled state again (PID response = BUF) using software.

This function can be used only when bulk transfers are used.

(9) Auto Transfer MODE

With the pipes for bulk transfer (PIPE1 to PIPE5), when the ATREPM bit in PIPEnCTR is set to 1, a transition is made to auto response mode. During an OUT transfer (DIR = 0), OUT-NAK mode is entered, and during an IN transfer (DIR = 1), null auto response mode is entered.

(a) OUT-NAK Mode

With the pipes for bulk OUT transfer, NAK is returned in response to an OUT or PING token and an NRDY interrupt is output when the ATREPM bit is set to 1. To make a transition from normal mode to OUT-NAK mode, OUT-NAK mode should be specified in the pipe operation disabled state (response PID = NAK) before enabling pipe operation (response PID = BUF). After pipe operation has been enabled, OUT-NAK mode becomes valid. However, if an OUT token is received immediately before pipe operation is disabled, the token data is normally received, and an ACK is retuned to the host.

To make a transition from OUT-NAK mode to normal mode, OUT-NAK mode should be canceled in the pipe operation disabled state (response PID = NAK) before enabling pipe operation (response PID = BUF). In normal mode, reception of OUT data is enabled and an ACK is returned in response to a PING token if the buffer is ready to receive data.

(b) Null Auto Response Mode

With the pipes for bulk IN transfer, zero-length packets are continuously transmitted when the ATREPM bit is set to 1.

To make a transition from normal mode to null auto response mode, null auto response mode should be set in the pipe operation disabled state (response PID = NAK) before enabling pipe operation (response PID = BUF). After pipe operation has been enabled, null auto response mode becomes valid. Before setting null auto response mode, INBUFM = 0 should be confirmed because the mode can be set only when the buffer is empty. If the INBUFM bit is 1, the buffer should be emptied with the ACLRM bit. While a transition to null auto response mode is being made, data should not be written from the FIFO port.

To make a transition from null auto response mode to normal mode, pipe operation disabled state (response PID = NAK) should be retained for the period of zero-length packet transmission (full-speed: 10 μs, high-speed: 3 μs) before canceling null auto response mode. In normal mode, data can be written from the FIFO port; therefore, packet transmission to the host is enabled by enabling pipe operation (response PID = BUF).

31.5.4 FIFO Buffer Memory

(1) FIFO Buffer Memory Allocation

Figure 31.10 shows an example of a FIFO buffer memory map for this module. The FIFO buffer memory is an area shared by the CPU and this module. In the FIFO buffer memory status, there are times when the access right to the buffer memory is allocated to the user system (CPU side), and times when it is allocated to this module (SIE side).

The buffer memory sets independent areas for each pipe. In the memory areas, 64 bytes comprise one block, and the memory areas are set using the first block number of the number of blocks (specified using the BUFNMB and BUFSIZE bits in PIPEBUF).

Independent buffer memory areas should be set for each pipe. Each memory area can be set using the first block number and the number of blocks (specified using the BUFNMB and BUFSIZE bits in PIPEBUF), where one block comprises 64 bytes.

When continuous transfer mode has been selected using the CNTMD bit in PIPEnCFG, the BUFSIZE bits should be set so that the buffer memory size should be an integral multiple of the maximum packet size. When double buffer mode has been selected using the DBLB bit in PIPEnCFG, two planes of the memory area specified using the BUFSIZE bits in PIPEBUF can be assigned to a single pipe.

Moreover, three FIFO ports are used for access to the buffer memory (reading and writing data). A pipe is assigned to the FIFO port by specifying the pipe number using the CURPIPE bit in C/DnFIFOSEL.

The buffer statuses of the various pipes can be confirmed using the BSTS bit in DCPCTR and the INBUFM bit in PIPEnCTR. Also, the access right of the FIFO port can be confirmed using the FRDY bit in CFIFOCTR or DnFIFOCTR.

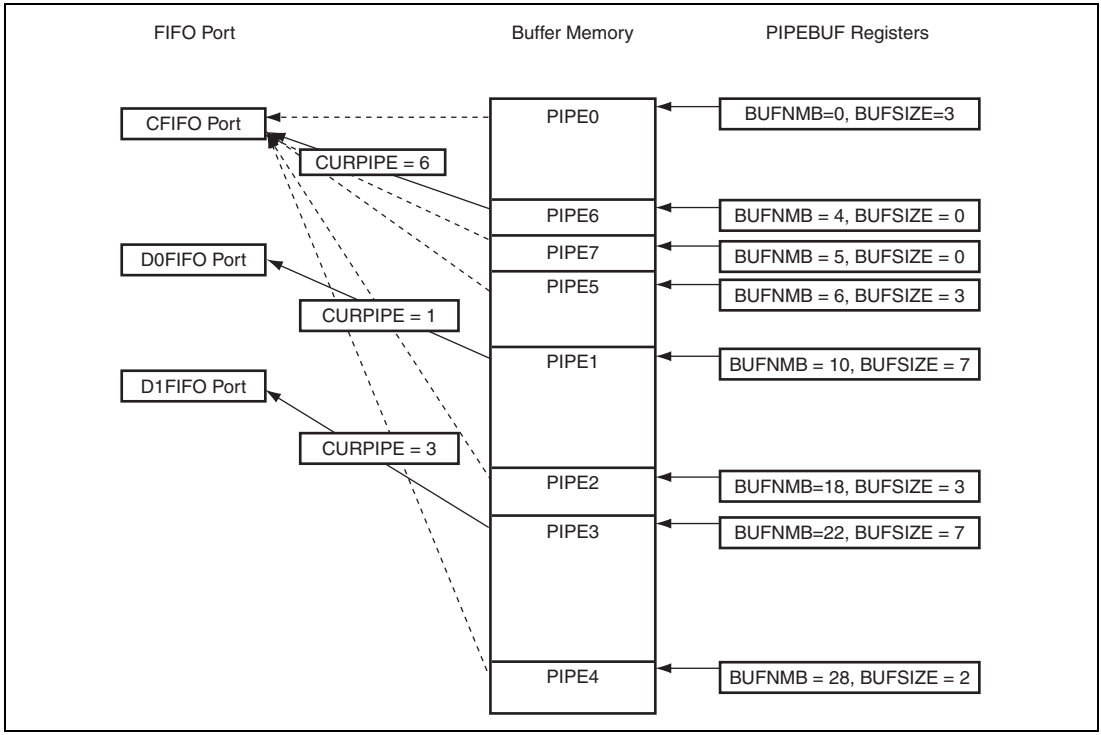


Figure 31.10 Example of a Buffer Memory Map

(a) Buffer Status

Tables 31.21 and 31.22 show the buffer status. The buffer memory status can be confirmed using the BSTS bit in DCPCTR and the INBUFM bit in PIPEnCTR. The access direction for the buffer memory can be specified using either the DIR bit in PIPEnCFG or the ISEL bit in CFIFOSEL (when DCP is selected).

The INBUFM bit is valid for PIPE1 to PIPE5 in the sending direction.

For an IN pipe uses double buffer, software can refer the BSTS bit to monitor the buffer memory status of CPU side and the INBUFM bit to monitor the buffer memory status of SIE side. In the case like the BEMP interrupt may not shows the buffer empty status because the CPU (DMAC) writes data slowly, software can use the INBUFM bit to confirm the end of sending.

Table 31.21 Buffer Status Indicated by the BSTS Bit

ISEL or DIR	BSTS	Buffer Memory State
0 (receiving direction)	0	There is no received data, or data is being received. Reading from the FIFO port is inhibited.
0 (receiving direction)	1	There is received data, or a zero-length packet has been received. Reading from the FIFO port is allowed. However, because reading is not possible when a zero-length packet is received, the buffer must be cleared.
1 (transmitting direction)	0	The transmission has not been finished. Writing to the FIFO port is inhibited.
1 (transmitting direction)	1	The transmission has been finished. CPU write is allowed.

Table 31.22 Buffer Status Indicated by the INBUFM Bit

IDIR	INBUFM	Buffer Memory State
0 (receiving direction)	Invalid	Invalid
1 (transmitting direction)	0	The transmission has been finished. There is no waiting data to be transmitted.
1 (transmitting direction)	1	The FIFO port has written data to the buffer. There is data to be transmitted

(b) FIFO Buffer Clearing

Table 31.23 shows the clearing of the FIFO buffer memory by this module. The buffer memory can be cleared using the three bits indicated below.

Table 31.23 List of Buffer Clearing Methods

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR DnFIFOCTR	DnFIFOSEL	PIPEnCTR
Function	Clears the buffer memory on the CPU side	In this mode, after the data of the specified pipe has been read, the buffer memory is cleared automatically.	This is the auto buffer clear mode, in which all of the received packets are discarded.
Clearing method	Cleared by writing 1	1: Mode valid 0: Mode invalid	1: Mode valid 0: Mode invalid

(c) Buffer Areas

Table 31.24 shows the FIFO buffer memory map of this controller. The buffer memory has special fixed areas to which pipes are assigned in advance, and user areas that can be set by the user.

The buffer for the DCP is a special fixed area that is used both for control read transfers and control write transfers.

The PIPE6 to PIPE9 area is assigned in advance, but the area for pipes that are not being used can be assigned to PIPE1 to PIPE5 as a user area.

The settings should ensure that the various pipes do not overlap. Note that each area is twice as large as the setting value in the double buffer.

Also, the buffer size should not be specified using a value that is less than the maximum packet size.

Table 31.24 Buffer Memory Map

Buffer Memory Number	Buffer Size	Pipe Setting	Note
H'0	64 bytes	Fixed area only for the DCP	Single buffer, continuous transfers enabled
H'1 to H'3	—	Prohibited to be used	—
H'4	64 bytes	Fixed area for PIPE6	Single buffer
H'5	64 bytes	Fixed area for PIPE7	Single buffer
H'6	64 bytes	Fixed area for PIPE8	Single buffer
H'7	64 bytes	Fixed area for PIPE9	Single buffer
H'8 to H'FF	Up to 15872 bytes	PIPE1 to PIPE5 user area	Double buffer can be set, continuous transfers enabled

(d) Auto Buffer Clear Mode Function

With this module, all of the received data packets are discarded if the ACLRM bit in PIPEnCTR is set to 1. If a normal data packet has been received, the ACK response is returned to the host controller. This function can be set only in the buffer memory reading direction.

Also, if the ACLRM bit is set to 1 and then to 0, the buffer memory of the selected pipe can be cleared regardless of the access direction.

An access cycle of at least 100 ns is required between ACLRM = 1 and ACLRM = 0.

(e) Buffer Memory Specifications (Single/Double Setting)

Either a single or double buffer can be selected for PIPE1 to PIPE5, using the DBLB bit in PIPEnCFG. The double buffer is a function that assigns two memory areas specified with the BUFSIZE bit in PIPEBUF to the same pipe. Figure 31.11 shows an example of buffer memory settings for this module.

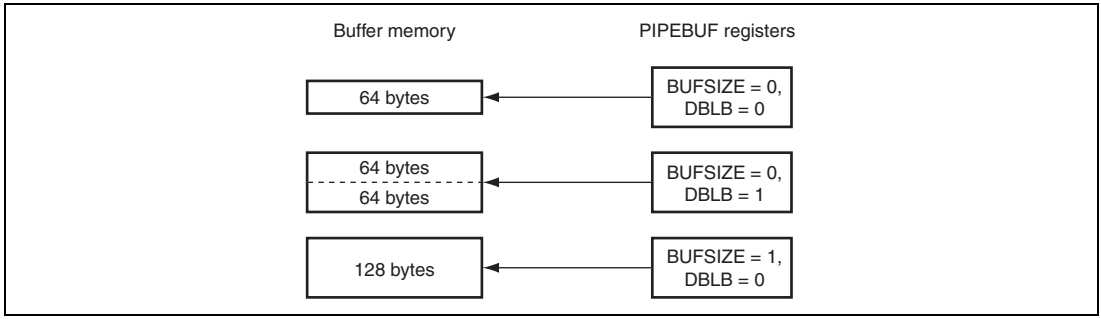


Figure 31.11 Example of Buffer Memory Settings

(f) Buffer Memory Operation (Continuous Transfer Setting)

Either the continuous transfer mode or the non-continuous transfer mode can be selected, using the CNTMD bit in PIPEnCFG. This selection is valid for PIPE1 to PIPE5.

The continuous transfer mode function is a function that sends and receives multiple transactions in succession. When the continuous transfer mode is set, data can be transferred without interrupts being issued to the CPU, up to the buffer sizes assigned for each of the pipes.

In the continuous sending mode, the data being written is divided into packets of the maximum packet size and sent. If the data being sent is less than the buffer size (short packet, or the integer multiple of the maximum packet size is less than the buffer size), BVAL = 1 must be set after the data being sent has been written.

In the continuous reception mode, interrupts are not issued during reception of packets up to the buffer size, until the transaction counter has ended, or a short packet is received.

Table 31.25 describes the relationship between the transfer mode settings by CNTMD bit and the timings at which reading data or transmitting data from the FIFO buffer is enabled.

Table 31.25 Relationship between Transfer Mode Settings by CNTMD Bit and Timings at which Reading Data or Transmitting Data from FIFO Buffer is Enabled

Continuous or Non-Continuous Transfer Mode	When Reading Data or Transmitting Data is Enabled
Non-continuous transfer (CNTMD = 0)	In the receiving direction (DIR = 0), reading data from the FIFO buffer is enabled when: • This module receives one packet.
	In the transmitting direction (DIR = 1), transmitting data from the FIFO buffer is enabled when: • Software (or DMAC) writes data of the maximum packet size to the FIFO buffer. or • Software (or DMAC) writes data of the short packet size (including 0-byte data) to the FIFO buffer and then writes 1 to BVAL.
Continuous transfer (CNTMD = 1)	In the receiving direction (DIR = 0), reading data from the FIFO buffer is enabled when: • The number of the data bytes received in the FIFO buffer assigned to the selected pipe becomes the same as the number of assigned data bytes ((BUFSIZE + 1) * 64). • This module receives a short packet other than a zero-length packet. • This module receives a zero-length packet when data is already stored in the FIFO buffer assigned to the selected pipe. or • This module receives the number of packets equal to the transaction counter value specified for the selected pipe by software.
	In the transmitting direction (DIR = 1), transmitting data from the FIFO buffer is enabled when: • The number of the data bytes written to the FIFO buffer by software (or DMAC) becomes the same as the number of data bytes in a single FIFO buffer plane assigned to the selected pipe. or • Software (or DMAC) writes to the FIFO buffer the number of data bytes less than the size of a single FIFO buffer plane (including 0-byte data) assigned to the selected pipe and then writes 1 to BVAL.

Figure 31.12 shows an example of buffer memory operation for this module.

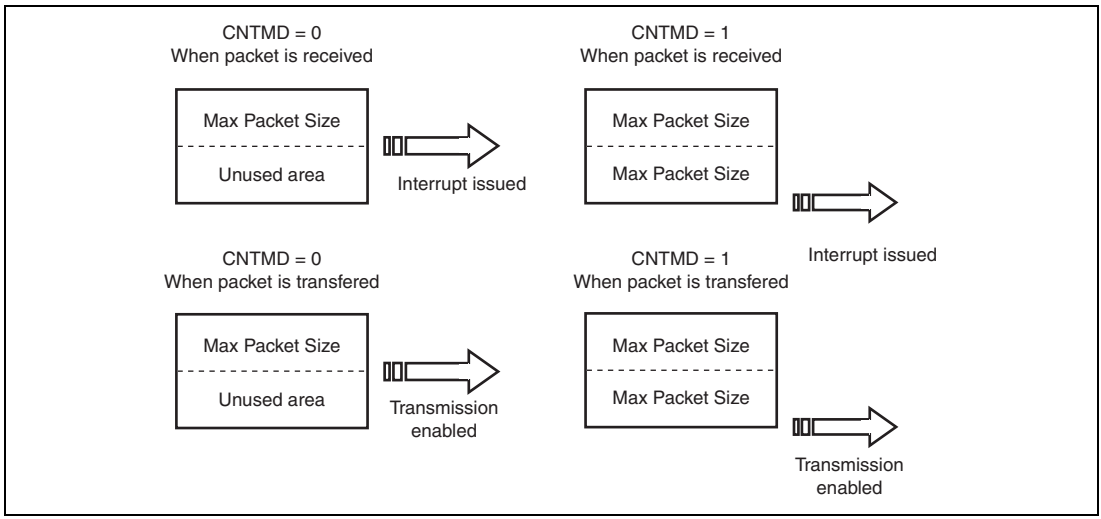


Figure 31.12 Example of Buffer Memory Operation

(2) FIFO Port Functions

Table 31.26 shows the settings for the FIFO port functions of this module. In write access, writing data until the buffer is full (or the maximum packet size for non-continuous transfers) automatically enables sending of the data. To enable sending of data before the buffer is full (or before the maximum packet size for non-continuous transfers), the BVAL bit in C/DnFIFOCTR must be set to end the writing. Also, to send a zero-length packet, the BCLR bit in the same register must be used to clear the buffer and then the BVAL bit set in order to end the writing.

In read access, reception of new packets is automatically enabled if all of the data has been read. Data cannot be read when a zero-length packet is being received (DTLN = 0), so the BCLR bit in the register must be used to release the buffer. The length of the data being received can be confirmed using the DTLN bit in C/DnFIFOCTR.

Table 31.26 FIFO Port Function Settings

Register Name	Bit Name	Function	Note
C/DnFIFOSEL	RCNT	Selects DTLN read mode	
	REW	Buffer memory rewind (re-read, rewrite)	
	DCLRM	Automatically clears data received for a specified pipe after the data has been read	For DnFIFO only
	DREQE	Enables DMA transfers	For DnFIFO only
	MBW	FIFO port access bit width	
	BIGEND	Selects FIFO port endian	
	ISEL	FIFO port access direction	For DCP only
	CURPIPE	Selects the current pipe	
C/DnFIFOCTR	BVAL	Ends writing to the buffer memory	
	BCLR	Clears the buffer memory on the CPU side	
	DTLN	Checks the length of received data	

(a) FIFO Port Selection

Table 31.27 shows the pipes that can be selected with the various FIFO ports. The pipe to be accessed is selected using the CURPIPE bit in C/DnFIFOSEL. After the pipe is selected, whether the CURPIPE value for the pipe which was written last can be correctly read should be checked. (If the previous pipe number is read, it indicates that the pipe modification is being executed by this module.) Then, the FIFO port can be accessed after FRDY = 1 is checked .

Also, the bus width to be accessed should be selected using the MBW bit. The buffer memory access direction conforms to the DIR bit in PIPEnCFG. The ISEL bit determines this only for the DCP.

Table 31.27 FIFO Port Access Categorized by Pipe

Pipe	Access Method	Port that can be Used
DCP	CPU access	CFIFO port register
PIPE1 to PIPE9	CPU access	CFIFO port register
		D0FIFO/D1FIFO port register
	DMA access	D0FIFO/D1FIFO port register

(b) REW Bit

It is possible to temporarily stop access to the pipe currently being accessed, access a different pipe, and then continue processing using the current pipe once again. The REW bit in C/DnFIFOSEL is used for this.

If a pipe is selected when the REW bit is set to 1 and at the same time the CURPIPE bit in C/DnFIFOSEL is set, the pointer used for reading from and writing to the buffer memory is reset, and reading or writing can be carried out from the first byte. Also, if a pipe is selected with 0 set for the REW bit, data can be read and written in continuation of the previous selection, without the pointer used for reading from and writing to the buffer memory being reset.

To access the FIFO port, FRDY = 1 must be ensured after selecting a pipe.

(3) DMA Transfers (D0FIFO/D1FIFO port)

(a) Overview of DMA Transfers

For pipes 1 to 9, the FIFO port can be accessed using the DMAC. When accessing the buffer for the pipe targeted for DMA transfer is enabled, a DMA transfer request is issued.

The unit of transfer to the FIFO port should be selected using the MBW bit in DnFIFOSEL and the pipe targeted for the DMA transfer should be selected using the CURPIPE bit. The selected pipe should not be changed during the DMA transfer.

(b) DnFIFO Auto Clear Mode (D0FIFO/D1FIFO Port Reading Direction)

If 1 is set for the DCLRM bit in DnFIFOSEL, the module automatically clears the buffer memory of the selected pipe when reading of the data from the buffer memory has been completed.

Table 31.28 shows the packet reception and buffer memory clearing processing for each of the various settings. As shown, the buffer clear conditions depend on the value set to the BFRE bit. Using the DCLRM bit eliminates the need for the buffer to be cleared by software even if a situation occurs that necessitates clearing of the buffer. This makes it possible to carry out DMA transfers without involving software.

This function can be set only in the buffer memory reading direction.

Table 31.28 Packet Reception and Buffer Memory Clearing Processing

Buffer Status When Packet is Received	Register Setting			
	DCLRM = 0		DCLRM = 1	
	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Doesn't need to be cleared	Doesn't need to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Zero-length packet reception	Needs to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Normal short packet reception	Doesn't need to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Transaction count ended	Doesn't need to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared

31.5.5 Control Transfers (DCP)

Data transfers of the data stage of control transfers are done using the default control pipe (DCP).

The DCP buffer memory is a 256-byte single buffer, and is a fixed area that is shared for both control reading and control writing. The buffer memory can be accessed through the CFIFO port.

(1) Control Transfers when the Host Controller Function is Selected

(a) Setup Stage

USQREQ, USBVAL, USBINDX, and USBLENG are the registers that are used to transmit a USB request for setup transactions. Writing setup packet data to the registers and writing 1 to the SUREQ bit in DCPCTR transmits the specified data for setup transactions. Upon completion of transactions, the SUREQ bit is cleared to 0. The above USB request registers should not be modified while SUREQ = 1. The device address for setup transactions is specified using the DEVSEL bits in DCPMAXP.

When the data for setup transactions has been sent, a SIGN or SACK interrupt request is generated according to the response received from the peripheral device (SIGN1 or SACK bits in INTSTS1), by means of which the result of the setup transactions can be confirmed.

A data packet of DATA0 (USB request) is transmitted as the data packet for the setup transactions regardless of the setting of the SQMON bit in DCPCTR.

(b) Data Stage

Data transfers are done using the DCP buffer memory.

The access direction of the DCP buffer memory should be specified using the ISEL bit in CFIFOSEL.

For the first data packet of the data stage, the data PID must be transferred as DATA1. Transaction is done by setting the data PID = DATA1 and the PID bit = BUF using the SQSET bit in DCPCFG. Completion of data transfer is detected using the BRDY and BEMP interrupts.

Setting continuous transfer mode allows data transfers over multiple packets. Note that when continuous transfer mode is set for the receiving direction, the BRDY interrupt is not generated until the buffer becomes full or a short packet is received (the integer multiple of the maximum packet size, and less than 256 bytes).

For control write transfers, when the number of data bytes to be sent is the integer multiple of the maximum packet size, software must control so as to send a zero-length packet at the end.

(c) Status Stage

Zero-length packet data transfers are done in the direction opposite to that in the data stage. As with the data stage, data transfers are done using the DCP buffer memory. Transactions are done in the same manner as the data stage.

For the data packets of the status stage, the data PID must be transferred as DATA1. The data PID should be set to DATA1 using the SQSET bit in DCPCFG.

For reception of a zero-length packet, the received data length must be confirmed using the DTLN bits in CFIFOCTR after the BRDY interrupt is generated, and the buffer memory must then be cleared using the BCLR bit.

(2) Control Transfers when the Function Controller Function is Selected

(a) Setup Stage

This module always sends an ACK response in response to a setup packet that is normal with respect to this module. The operation of this module operates in the setup stage is noted below.

(i) When a new USB request is received, this module sets the following registers:

- Set the VALID bit in INTSTS0 to 1.
- Set the PID bit in DCPCTR to NAK.
- Set the CCPL bit in DCPCTR to 0.

(ii) When a data packet is received right after the SETUP packet, the USB request parameters are stored in USBREQ, USBVAL, USBINDX, and USBLENG.

Response processing with respect to the control transfer should always be carried out after first setting VALID = 0. In the VALID = 1 state, PID = BUF cannot be set, and the data stage cannot be terminated.

Using the function of the VALID bit, this module is able to interrupt the processing of a request currently being processed if a new USB request is received during a control transfer, and can send a response in response to the newest request.

Also, this module automatically judges the direction bit (bit 8 of the bmRequestType) and the request data length (wLength) of the USB request that was received, and then distinguishes between control read transfers, control write transfers, and no-data control transfers, and controls the stage transition. For a wrong sequence, the sequence error of the control transfer stage transition interrupt is generated, and the software is notified. For information on the stage control of this module, see figure 31.8.

(b) Data Stage

Data transfers corresponding to USB requests that have been received should be done using the DCP. Before accessing the DCP buffer memory, the access direction should be specified using the ISEL bit in CFIFOSEL.

If the data being transferred is larger than the size of the DCP buffer memory, the data transfer should be carried out using the BRDY interrupt for control write transfers and the BEMP interrupt for control read transfers.

With control write transfers during high-speed operation, the NYET handshake response is carried out based on the state of the buffer memory.

(c) Status Stage

Control transfers are terminated by setting the CCPL bit to 1 with the PID bit in DCPCTR set to PID = BUF.

After the above settings have been entered, this module automatically executes the status stage in accordance with the data transfer direction determined at the setup stage. The specific procedure is as follows.

(i) For control read transfers:

This module sends a zero-length packet and receives an ACK response from the USB host.

(ii) For control write transfers and no-data control transfers:

The zero-length packet is received from the USB host, and this module sends an ACK response.

(d) Control Transfer Auto Response Function

This module automatically responds to a normal SET_ADDRESS request. If any of the following errors occur in the SET_ADDRESS request, a response from the software is necessary.

(i) Any transfer other than a control read transfer: bmRequestType ≠ H'00

(ii) If a request error occurs: wIndex ≠ H'00

(ii) For any transfer other than a no-data control transfer: wLength ≠ H'00

(iv) If a request error occurs: wValue > H'7F

(v) Control transfer of a device state error: DVSQ = 011 (Configured)

For all requests other than the SET_ADDRESS request, a response is required from the corresponding software.

31.5.6 Bulk Transfers (PIPE1 to PIPE5)

The buffer memory specifications for bulk transfers (single/double buffer setting, or continuous/non-continuous transfer mode setting) can be selected. The maximum size that can be set for the buffer memory is 2 kbytes. The buffer memory state is controlled by this module, with a response sent automatically for a PING packet/NYET handshake.

(1) PING Packet Control when the Host Controller Function is Selected

This module automatically sends a PING packet in the OUT direction.

On receiving an ACK handshake in the initial state in which PING packet sending mode is set, this module sends an OUT packet as noted below. Reception of an NAK or NYET handshake returns this module to PING packet sending mode. This control also applies to the control transfers in the data stage and status stage.

1. Sets OUT data sending mode.
2. Sends a PING packet.
3. Receives an ACK handshake.
4. Sends an OUT data packet.
5. Receives an ACK handshake.

(Repeats steps 4 and 5.)

6. Sends an OUT data packet.
7. Receives an NAK/NYET handshake.
8. Sends a PING packet.

This module is returned to PING packet sending mode by a power-on reset, receiving a NYET/NAK handshake, setting or clearing the sequence toggle bits (SQSET and SQCLR), and setting the buffer clear bit (ACLRM) in PIPEnCTR.

(2) NYET Handshake Control when the Function Controller Function is Selected

Table 31.29 shows the NYET handshake responses of this module. The NYET response of this module is made in conformance with the conditions noted below. When a short packet is received, however, the response will be an ACK response instead of a NYET packet response. The same applies to the data stages of control write transfers.

Table 31.29 NYET Handshake Responses

Value Set for PID Bit in DCPCTR	Buffer Memory State	Token	Response	Note
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/ PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY1	OUT/PING	ACK	If an OUT token is received, a data packet is received.
	RCV-BRDY2	OUT	NYET	Notifies whether a data packet can be received
	RCV-BRDY2	OUT (Short)	ACK	Notifies whether a data packet can be received
	RCV-BRDY2	PING	ACK	Notifies that a data packet can be received
	RCV-NRDY	OUT/PING	NAK	Notifies that a data packet cannot be received
	TRN-BRDY	IN	DATA0/DATA1	A data packet is transmitted
	TRN-NRDY	IN	NAK	TRN-NRDY

[Legend]

RCV-BRDY1: When an OUT/PING token is received, there is space in the buffer memory for two or more packets.

RCV-BRDY2: When an OUT token is received, there is only enough space in the buffer memory for one packet.

RCV-NRDY: When a PING token is received, there is no space in the buffer memory.

TRN-BRDY: When an IN token is received, there is data to be sent in the buffer memory.

TRN-NRDY: When an IN token is received, there is no data to be sent in the buffer memory.

31.5.7 Interrupt Transfers (PIPE6 to PIPE9)

When the function controller function is selected, this module carries out interrupt transfers in accordance with the timing controlled by the host controller. For interrupt transfers, PING packets are ignored (no responses are sent), and the ACK, NAK, and STALL responses are carried out without an NYET handshake response being made.

When the host controller function is selected, this module can set the timing of issuing a token using the interval timer. At this time, this module issues an OUT token even in the OUT direction, without issuing a PING token.

This module does not support high bandwidth transfers of interrupt transfers.

(1) Interval Counter during Interrupt Transfers when the Host Controller Function is Selected

For interrupt transfers, intervals between transactions are set in the IITV bits in PIPEPERI. This controller issues an interrupt transfer token based on the specified intervals.

(a) Counter Initialization

This controller initializes the interval counter under the following conditions.

(i) Power-on reset:

The IITV bits are initialized.

(ii) Buffer memory initialization using the ACLRM bit:

The IITV bits are not initialized but the count value is. Setting the ACLRM bit to 0 starts counting from the value set in the IITV bits.

Note that the interval counter is not initialized in the following case.

(iii) USB bus reset, USB suspended:

The IITV bits are not initialized. Setting 1 to the UACT bit starts counting from the value before entering the USB bus reset state or USB suspended state.

(b) Operation when Transmission/Reception is Impossible at Token Issuance Timing

This module cannot issue tokens even at token issuance timing in the following cases. In such a case, this module attempts transactions at the subsequent interval.

(i) When the PID is set to NAK or STALL.

(ii) When the buffer memory is full at the token sending timing in the receiving (IN) direction.

(iii) When there is no data to be sent in the buffer memory at the token sending timing in the sending (OUT) direction.

31.5.8 Isochronous Transfers (PIPE1 and PIPE2)

This module has the following functions pertaining to isochronous transfers.

- Notification of isochronous transfer error information
- Interval counter (specified by the IITV bit)
- Isochronous IN transfer data setup control (IDLY function)
- Isochronous IN transfer buffer flush function (specified by the IFIS bit)

This module does not support the High Bandwidth transfers of isochronous transfers.

(1) Error Detection with Isochronous Transfers

This module has a function for detecting the error information noted below, so that when errors occur in isochronous transfers, software can control them. Tables 31.30 and 31.31 show the priority in which errors are confirmed and the interrupts that are generated.

(i) PID errors

- If the PID of the packet being received is illegal

(ii) CRC errors and bit stuffing errors

- If an error occurs in the CRC of the packet being received, or the bit stuffing is illegal

(iii) Maximum packet size exceeded

- The maximum packet size exceeded the set value.

(iv) Overrun and underrun errors

- When host controller function is selected:
  — When using isochronous IN transfers (reception), the IN token was received but the buffer memory is not empty.

— When using isochronous OUT transfers (transmission), the OUT token was transmitted, but the data was not in the buffer memory.

- When function controller function is selected:
  — When using isochronous IN transfers (transmission), the IN token was received but the data was not in the buffer memory.
  — When using isochronous OUT transfers (reception), the OUT token was received, but the buffer memory was not empty.

(v) Interval errors

When function controller function is selected, Interval errors occur in following cases.

- During an isochronous IN transfer, the token could not be received during the interval frame.
- During an isochronous OUT transfer, the OUT token was received during frames other than the interval frame.

Table 31.30 Error Detection when a Token is Received

Detection Priority	Error	Generated Interrupt and Status
1	PID errors	No interrupts are generated in both cases when the host controller function is selected and the function controller function is selected (ignored as a corrupted packet).
2	CRC error and bit stuffing errors	No interrupts generated in both cases when the host controller function is selected and the function controller function is selected (ignored as a corrupted packet).
3	Overrun and underrun errors	An NRDY interrupt is generated to set the OVRN bit in both cases when host controller function is selected and function controller function is selected. When the host controller function is selected, no tokens are transmitted. When the function controller function is selected, a zero-length packet is transmitted in response to IN token. However, no data packets are received in response to OUT token.
4	Interval errors	An NRDY interrupt is generated when the function controller function is selected. It is not generated when the host controller function is selected.

Table 31.31 Error Detection when a Data Packet is Received

Detection Priority Order	Error	Generated Interrupt and Status
1	PID errors	No interrupts are generated (ignored as a corrupted packet)
2	CRC error and bit stuffing errors	An NRDY interrupt is generated to set the CRCE bit in both cases when the host controller function is selected and the function controller function is selected.
3	Maximum packet size exceeded error	A BEMP interrupt is generated to set the PID bits to STALL in both cases when the host controller function is selected and the function controller function is selected.

(2) DATA-PID

This module does not support High Bandwidth transfers. When the function controller function is selected, this module operates as follows in response to the received PID.

(a) IN direction

— DATA0: Sent as data packet PID
— DATA1: Not sent
— DATA2: Not sent
— mDATA: Not sent

(b) OUT direction (when using full-speed operation)

— DATA0: Received normally as data packet PID
— DATA1: Received normally as data packet PID
— DATA2: Packets are ignored
— mDATA: Packets are ignored

(c) OUT direction (when using high-speed operation)

— DATA0: Received normally as data packet PID
— DATA1: Received normally as data packet PID
— DATA2: Received normally as data packet PID
— mDATA: Received normally as data packet PID

(3) Interval Counter

The isochronous interval can be set using the IITV bits in PIPEPERI. The interval counter enables the functions shown in table 31.32 when the function controller function is selected. When the host controller function is selected, this module generates the token issuance timing. When the host controller function is selected, the interval counter operation is the same as the interrupt transfer operation.

Table 31.32 Functions of the Interval Counter when the Function Controller Function is Selected

Transfer Direction	Function	Conditions for Detection
IN	IN buffer flush function	When an IN token cannot be normally received in the interval frame during an isochronous IN transfer
OUT	Notifies that a token not being received	When an OUT token cannot be normally received in the interval frame during an isochronous OUT transfer

The interval count is carried out when an SOF is received or for interpolated SOFs, so the isochronism can be maintained even if an SOF is damaged. The frame interval that can be set is the 2^{IITV} frame or 2^{IITV} μ frames.

(a) Counter Initialization when the Function Controller Function is Selected

This module initializes the interval counter under the following conditions.

(i) Power-on reset

The IITV bit is initialized.

(ii) Buffer memory initialization using the ACLRM bit

The IITV bits are not initialized but the count value is. Setting the ACLRM bit to 0 starts counting from the value set in the IITV bits.

After the interval counter has been initialized, the counter is started under the following conditions 1 or 2 when a packet has been transferred normally.

1. An SOF is received following transmission of data in response to an IN token, in the PID = BUF state.
2. An SOF is received after data following an OUT token is received in the PID = BUF state.

The interval counter is not initialized under the conditions noted below.

1. When the PID bit is set to NAK or STALL

   The interval timer does not stop. This module attempts the transactions at the subsequent interval.
2. The USB bus reset or the USB is suspended

   The IITV bit is not initialized. When the SOF has been received, the counter is restarted from the value prior to the reception of the SOF.

(b) Interval Counting and Transfer Control when the Host Controller Function is Selected

This module controls the interval between token issuance operations based on the IITV bit settings. Specifically, this module issues a token for a selected pipe once every 2IITV (μ) frames.

This module counts the interval every 1-ms frame for the pipes used for communications with the full-speed or low-speed peripheral devices connected to a high-speed HUB.

This module starts counting the token issuance interval at the (μ) frame following the (μ) frame in which software has set the PID bits to BUF.

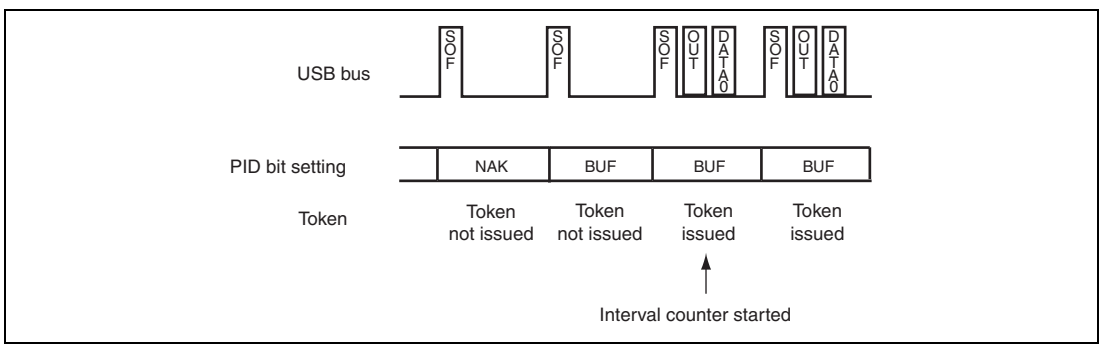


Figure 31.13 Token Issuance when IITV = 0

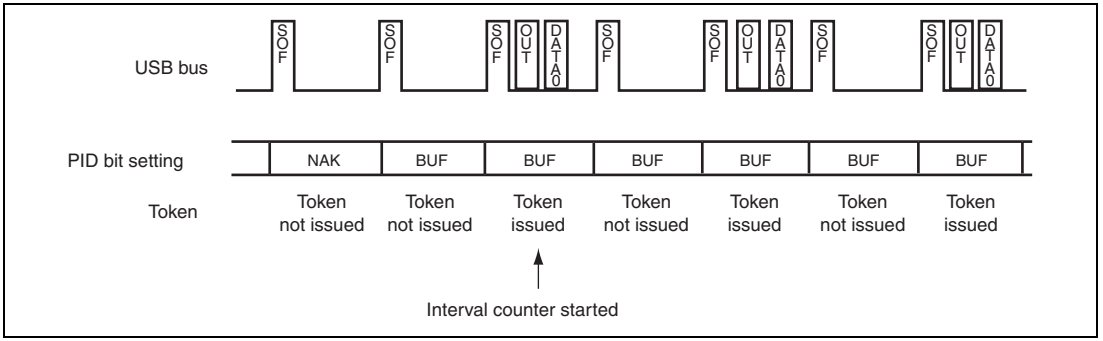


Figure 31.14 Token Issuance when IITV = 1

When the selected pipe is for isochronous transfers, this module carries out the operation below in addition to controlling token issuance interval. This module issues a token even when the NRDY interrupt generation condition is satisfied.

(i) When the selected pipe is for isochronous IN transfers

This module generates the NRDY interrupt when this module issues the IN token but does not receive a packet successfully from a peripheral device (no response or packet error).
This module sets the OVRN bit to 1 generating the NRDY interrupt when the time to issue an IN token comes in a state in which this module cannot receive data because the FIFO buffer is full (due to the fact that software (DMAC) is too slow to read data from the FIFO buffer),

(ii) When the selected pipe is for isochronous OUT transfers

This module sets the OVRN bit to 1 generating the NRDY interrupt and transmitting a zero-length packet when the time to issue an OUT token comes in a state in which there is no data to be transmitted in the FIFO buffer (because software (DMAC) is too slow to write data to the FIFO buffer).
The token issuance interval is reset on any of the following conditions.

— When a hardware-reset is applied to this module (here, the IITV bits are also cleared to 0).
— When software sets the ACLRM bit to 1.

(c) Interval Counting and Transfer Control when the Function Controller Function is Selected

(i) When the selected pipe is for isochronous OUT transfers

This module generates the NRDY interrupt when this module fails to receive a data packet within the interval set by the IITV bits in terms of (μ) frames.

This module generates the NRDY interrupt when this module fails to receive a data packet because of a CRC error or other errors contained in the packet, or because of the FIFO buffer being full.

This module generates the NRDY interrupt on receiving an SOF packet. Even if the SOF packet is corrupted, the internal interpolation is used and allows the interrupt to be generated at the timing to receive the SOF packet.

However, when the IITV bits are set to the value other than 0, this module generates the NRDY interrupt on receiving an SOF packet for every interval after starting interval counting operation. When the PID bits are set to NAK by software after starting the interval timer, this module does not generate the NRDY interrupt on receiving an SOF packet.

The interval counting starts at the different timing depending on the IITV bit setting as follows.

— When IITV = 0: The interval counting starts at the (μ) frame following the (μ) frame in which software has set the PID bits for the selected pipe to BUF.

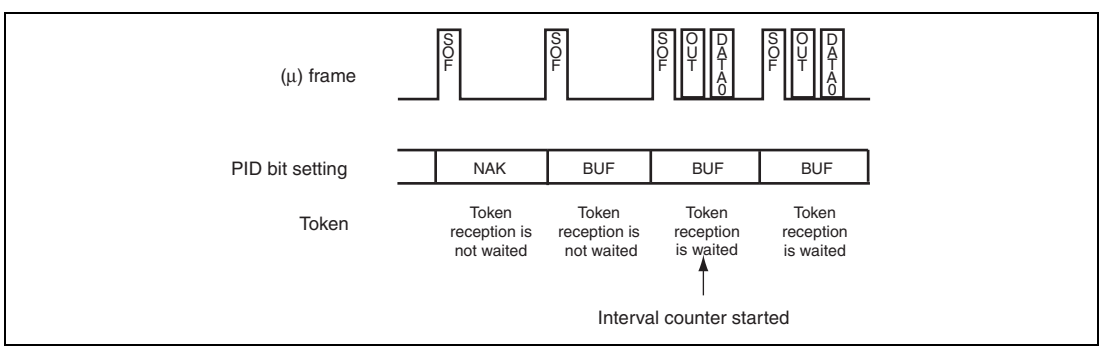


Figure 31.15 Relationship between (μ) Frames and Expected Token Reception when IITV = 0

— When IITV ≠ 0: The interval counting starts on completion of successful reception of the first data packet after the PID bits for the selected pipe have been modified to BUF.

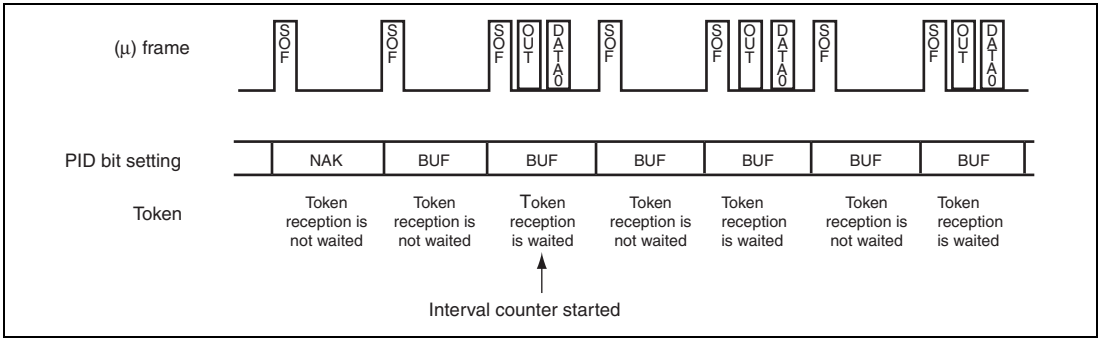


Figure 31.16 Relationship between (μ) Frames and Expected Token Reception when IITV ≠ 0

(ii) When the selected pipe is for isochronous IN transfers

The IFIS bit should be 1 for this use. When IFIS = 0, this module transmits a data packet in response to the received IN token irrespective of the IITV bit setting.

When IFIS = 1, this module clears the FIFO buffer when this module fails to receive an IN token within the interval set by the IITV bits in terms of (μ) frames in a state in which there is data to be transmitted in the FIFO buffer.

This module also clears the FIFO buffer when this module fails to receive an IN token successfully because of a bus error such as a CRC error contained in the token.

This module clears the FIFO buffer on receiving an SOF packet. Even if the SOF packet is corrupted, the internal interpolation is used and allows the FIFO buffer to be cleared at the timing to receive the SOF packet.

The interval counting starts at the different timing depending on the IITV bit setting (similar to the timing during OUT transfers).

The interval is counted on any of the following conditions in function controller mode.

— When a hardware-reset is applied to this module (here, the IITV bits are also cleared to 0).
— When software sets the ACLRM bit to 1.
— When this module detects a USB reset.

(4) Setup of Data to be Transmitted using Isochronous Transfer when the Function Controller Function is Selected

With isochronous data transmission using this module in function controller function, after data has been written to the buffer memory, a data packet can be sent with the next frame in which an SOF packet is detected. This function is called the isochronous transfer transmission data setup function, and it makes it possible to designate the frame from which transmission began.

If a double buffer is used for the buffer memory, transmission will be enabled for only one of the two buffers even after the writing of data to both buffers has been completed, that buffer memory being the one to which the data writing was completed first. For this reason, even if multiple IN tokens are received, the only buffer memory that can be sent is one packet's worth of data.

When an IN token is received, if the buffer memory is in the transmission enabled state, this module transmits the data. If the buffer memory is not in the transmission enabled state, however, a zero-length packet is sent and an underrun error occurs.

Figure 31.17 shows an example of transmission using the isochronous transfer transmission data setup function with this module, when IITV = 0 (every frame) has been set. Sending of a zero-length packet is displayed in the figure as Null, in a shaded box.

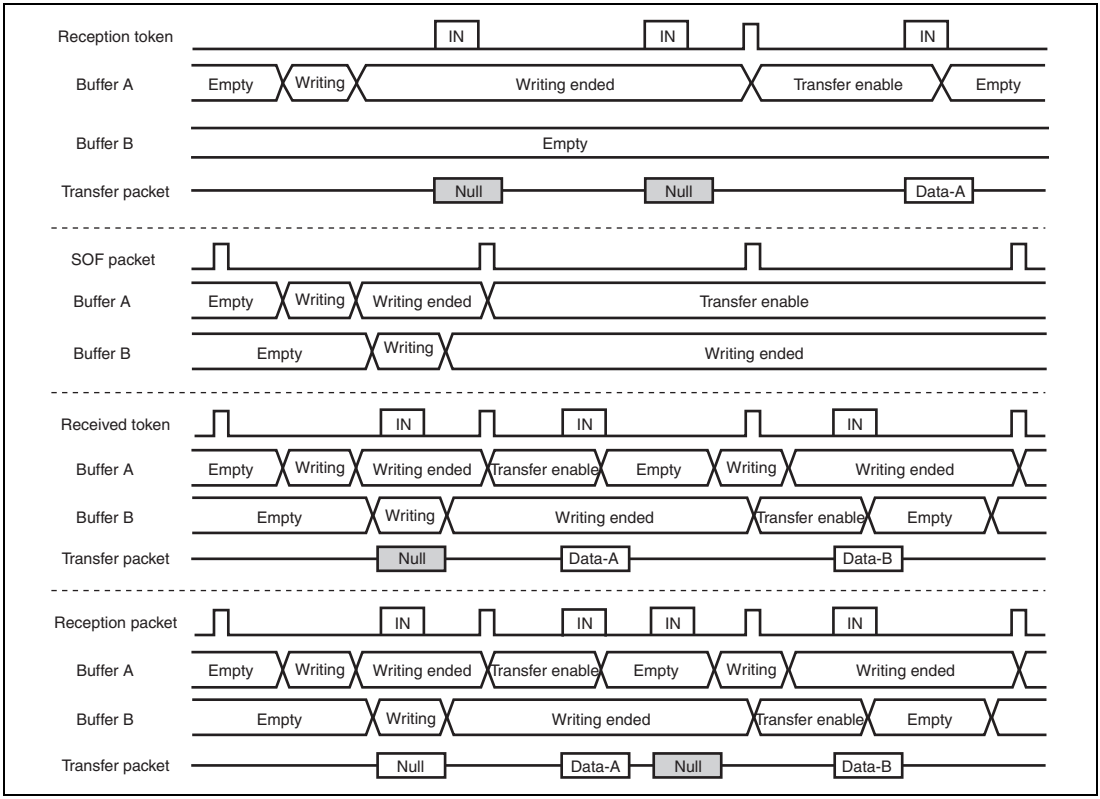


Figure31.17 Example of Data Setup Function Operation

(5) Isochronous Transfer Transmission Buffer Flush when the Function Controller Function is Selected

If an SOF packet or a μSOF packet is received without receiving an IN token in the interval frame during isochronous data transmission, this module operates as if an IN token had been corrupted, and clears the buffer for which transmission is enabled, putting that buffer in the writing enabled state.

If a double buffer is being used and writing to both buffers has been completed, the buffer memory that was cleared is seen as the data having been sent at the same interval frame, and transmission is enabled for the buffer memory that is not discarded with SOF or μSOF packets reception.

The timing at which the operation of the buffer flush function varies depending on the value set for the IITV bit.

(a) If IITV = 0

The buffer flush operation starts from the next frame after the pipe becomes valid.

(b) In any cases other than IITV = 0

The buffer flush operation is carried out subsequent to the first normal transaction.

Figure 31.18 shows an example of the buffer flush function of this module. When an unanticipated token is received prior to the interval frame, this module sends the written data or a zero-length packet according to the buffer state.

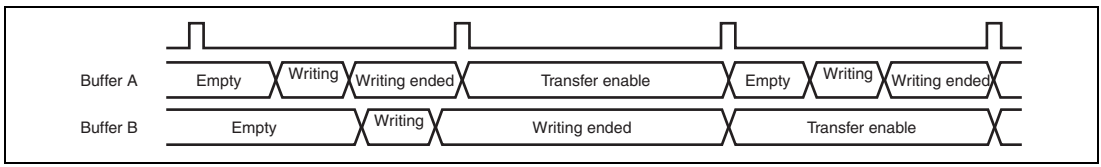


Figure 31.18 Example of Buffer Flush Function Operation

Figure 31.19 shows an example of this module generating an interval error. There are five types of interval errors, as shown below. The interval error is generated at the timing indicated by (1) in the figure, and the IN buffer flush function is activated.

If an interval error occurs during an IN transfers, the buffer flush function is activated; and if it occurs during an OUT transfer, an NRDY interrupt is generated.

The OVRN bit should be used to distinguish between NRDY interrupts such as received packet errors and overrun errors.

In response to tokens that are shaded in the figure, responses occur based on the buffer memory status.

1. IN direction:
   — If the buffer is in the transmission enabled state, the data is transferred as a normal response.
   — If the buffer is in the transmission disabled state, a zero-length packet is sent and an underrun error occurs.
2. OUT direction:
   — If the buffer is in the reception enabled state, the data is received as a normal response.
   — If the buffer is in the reception disabled state, the data is discarded and an overrun error occurs.

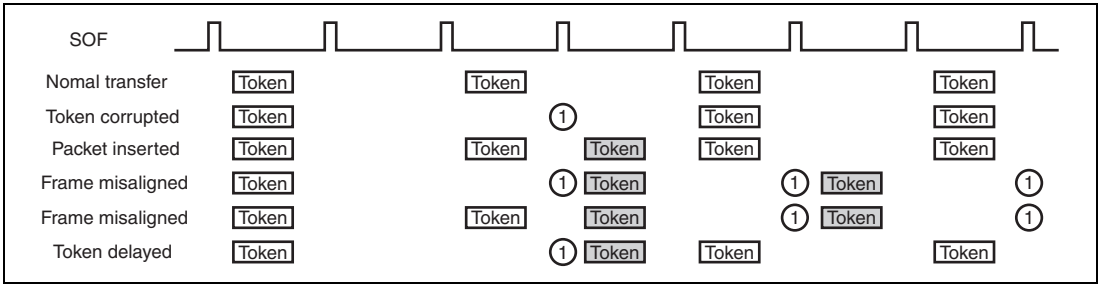


Figure 31.19 Example of an Interval Error Being Generated when IITV = 1

31.5.9 SOF Interpolation Function

When the function controller function is selected and if data could not be received at intervals of 1 ms (when using full-speed operation) or 125 μs (when using high-speed operation) because an SOF packet was corrupted or missing, this module interpolates the SOF. The SOF interpolation operation begins when the USBE and SCKE bits in SYSCFG have been set to 1 and an SOF packet is received. The interpolation function is initialized under the following conditions.

- Power-on reset
- USB bus reset
- Suspended state detected

Also, the SOF interpolation operates under the following specifications.

- 125 μs/1 ms conforms to the results of the reset handshake protocol.
- The interpolation function is not activated until an SOF packet is received.
- After the first SOF packet is received, either 125 μs or 1 ms is counted with an internal clock of 48 MHz, and interpolation is carried out.
- After the second and subsequent SOF packets are received, interpolation is carried out at the previous reception interval.
- Interpolation is not carried out in the suspended state or while a USB bus reset is being received. (With suspended transitions in high-speed operation, interpolation continues for 3 ms after the last packet is received.)

This module supports the following functions based on the SOF detection. These functions also operate normally with SOF interpolation, if the SOF packet was corrupted.

- Refreshing of the frame number and the micro-frame number
- SOFR interrupt timing and μSOF lock
- Isochronous transfer interval count

If an SOF packet is missing when full-speed operation is being used, the FRNM bit in FRMNUM is not refreshed.

If a μSOF packet is missing during high-speed operation, the UFRNM bit in UFRMNUM is refreshed.

However, if a μSOF packet for which the μFRNM = 000 is missing, the FRNM bit is not refreshed. In this case, the FRNM bit is not refreshed even if successive μSOF packets other than μFRNM = 000 are received normally.

31.5.10 Pipe Schedule

(1) Conditions for Generating a Transaction

When the host controller function is selected and UACT has been set to 1, this module generates a transaction under the conditions noted in table 31.33.

Table 31.33 Conditions for Generating a Transaction

	Conditions for Generation				
Transaction	**DIR**	**PID**	**IITV0**	**Buffer State**	**SUREQ**
Setup	—*1	—*1	—*1	—*1	1 setting
Control transfer data stage, status stage, bulk transfer	IN	BUF	Invalid	Receive area exists	—*1
	OUT	BUF	Invalid	Send data exists	—*1
Interrupt transfer	IN	BUF	Valid	Receive area exists	—*1
	OUT	BUF	Valid	Send data exists	—*1
Isochronous transfer	IN	BUF	Valid	*2	—*1
	OUT	BUF	Valid	*3	—*1

Notes: 1. Symbols (—) in the table indicate that the condition is one that is unrelated to the generating of tokens. "Valid" indicates that, for interrupt transfers and isochronous transfers, the condition is generated only in transfer frames that are based on the interval counter. "Invalid" indicates that the condition is generated regardless of the interval counter.
2. This indicates that a transaction is generated regardless of whether or not there is a receive area. If there was no receive area, however, the received data is destroyed.
3. This indicates that a transaction is generated regardless of whether or not there is any data to be sent. If there was no data to be sent, however, a zero-length packet is sent.

(2) Transfer Schedule

This section describes the transfer scheduling within a frame of this module. After the module sends an SOF, the transfer is carried out in the sequence described below.

(a) Execution of periodic transfers

A pipe is searched in the order of Pipe 1 → Pipe 2 → Pipe 6 → Pipe 7 → Pipe 8 → Pipe 9, and then, if the pipe is one for which an isochronous or interrupt transfer transaction can be generated, the transaction is generated.

(b) Setup transactions for control transfers

The DCP is checked, and if a setup transaction is possible, it is sent.

(c) Execution of bulk and control transfer data stages and status stages

A pipe is searched in the order of DCP → Pipe 1 → Pipe 2 → Pipe 3 → Pipe 4 → Pipe 5, and then, if the pipe is one for which a bulk or control transfer data stage or a control transfer status stage transaction can be generated, the transaction is generated.
If a transfer is generated, processing moves to the next pipe transaction regardless of whether the response from the peripheral device is ACK or NAK. Also, if there is time for the transfer to be done within the frame, step 3 is repeated.

(3) USB Communication Enabled

Setting the UACT bit of the DVSTCTR register to 1 initiates sending of an SOF or μSOF, and makes it possible to generate a transaction.

Setting the UACT bit to 0 stops the sending of the SOF or μSOF and initiates a suspend state. If the setting of the UACT bit is changed from 1 to 0, processing stops after the next SOF or μSOF is sent.

31.6 Usage Notes

31.6.1 USB External Circuit

The example of the USB external circuit when using this module as an USB function is shown in figure 31.20.

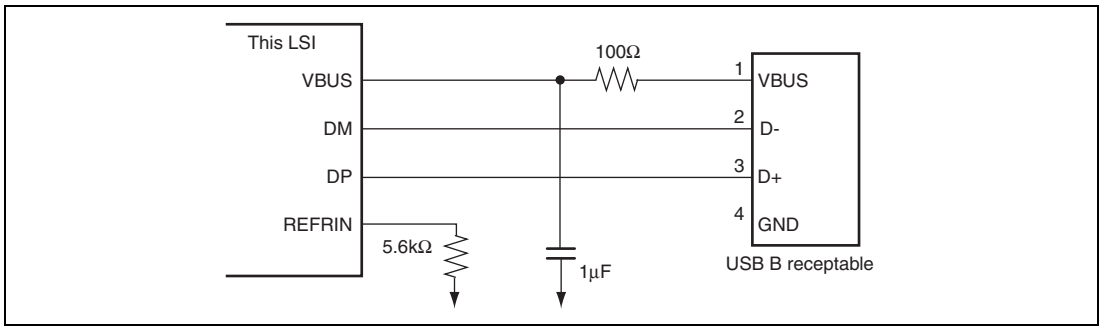


Figure 31.20 Example of USB External Circuit

The example of the USB external circuit when USB module is used as a host is shown in figure 31.21. The circuit to control 5 V power supply by using the port etc. is required though the detection of VBUS connection/disconnection is unnecessary for the USB host.

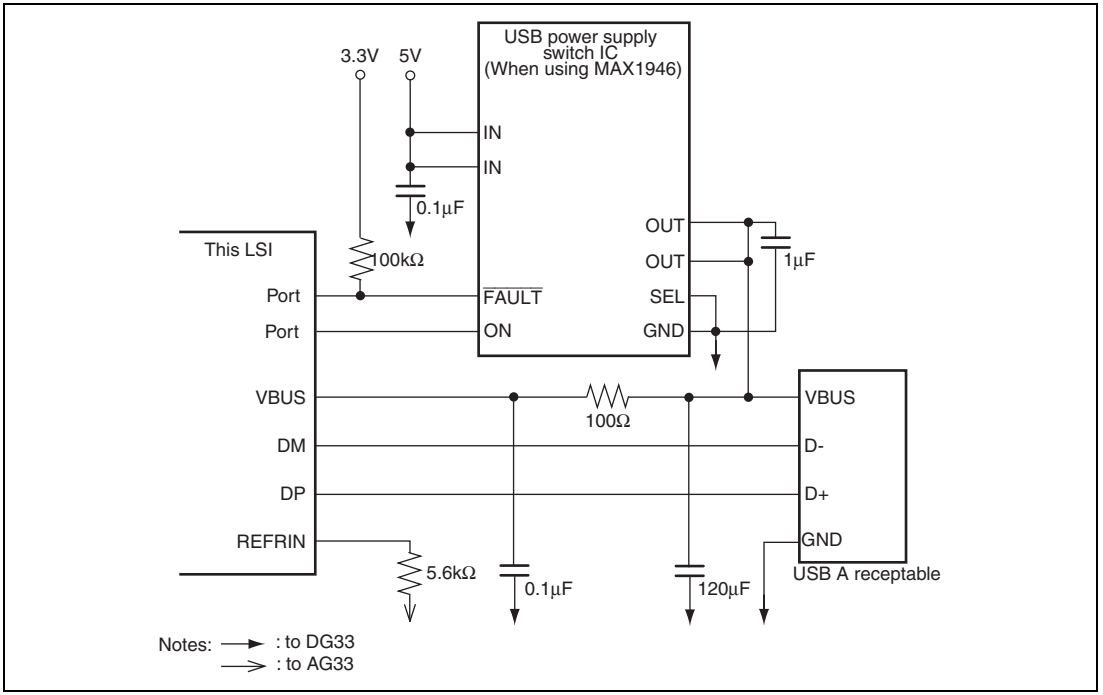


Figure 31.21 Example of USB External Circuit when USB module is used as host

Note: The external circuit shown above is an example based on the USB Specification. The correct operation cannot be guaranteed in any system. The board designs vary widely depending on the intended use.
Sufficient evaluation for the compatibility of the mounted devices should be required.

Section 32 I²C Bus Interface (IIC)

This LSI has I²C bus interface of two channels, I²C0 and I²C1.

Each I²C bus interface uses only one data line (SDA) and one clock line (SCL) to transfer data, saving board and connector space.

32.1 Features

- Start and stop conditions generated automatically
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Data transfer conforming to the I²C format
- Wait function

  A wait can be inserted by driving the SCL pin low after data transfer, excluding acknowledgement.

  The wait can be cleared by clearing the interrupt flag.
- I²C module corresponds to single master bus only

  This module is always in master mode. Since the slave mode is not incorporated, operation stops with bus open during loss of arbitration in data transfer.
- Four interrupt sources
  - Data transfer enable
  - Wait state
  - Non-acknowledge detection
  - Arbitration lost (operation stops with bus open when bus conflict is detected)
- Data transfer speed
  - Standard mode (100 kHz) and high-speed mode (400 kHz)
  - SCL clock can be set by clock control register setting
- Clock synchronous processing of SCL line

  A hazard (spike noise) generated in the high-count period by SCL is detected as an arbitration loss.

Figure 32.1 shows a block diagram of the I²C bus interface.

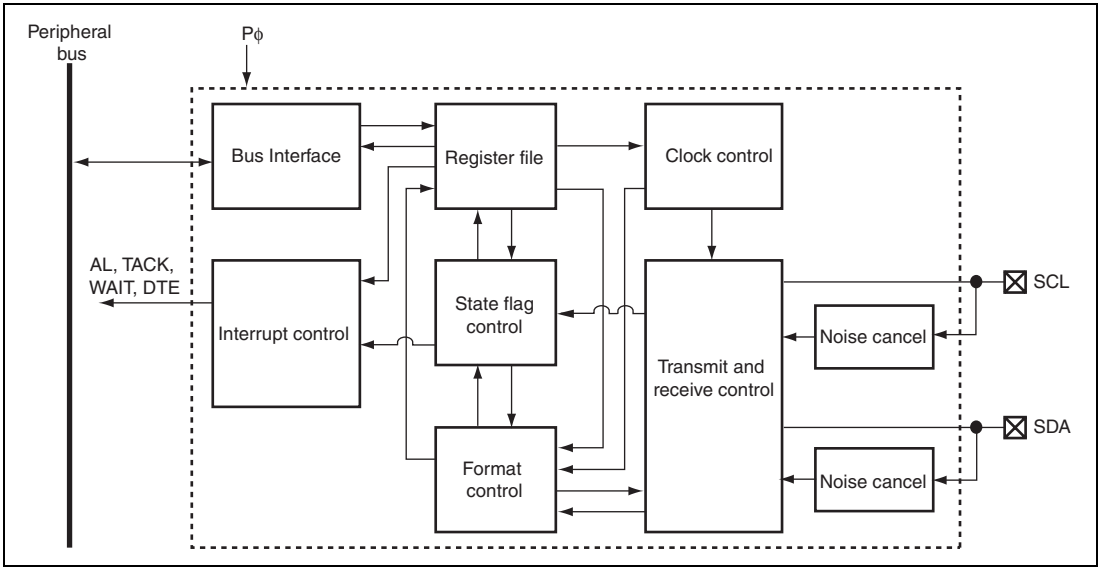


Figure 32.1 Block Diagram of I²C Bus Interface

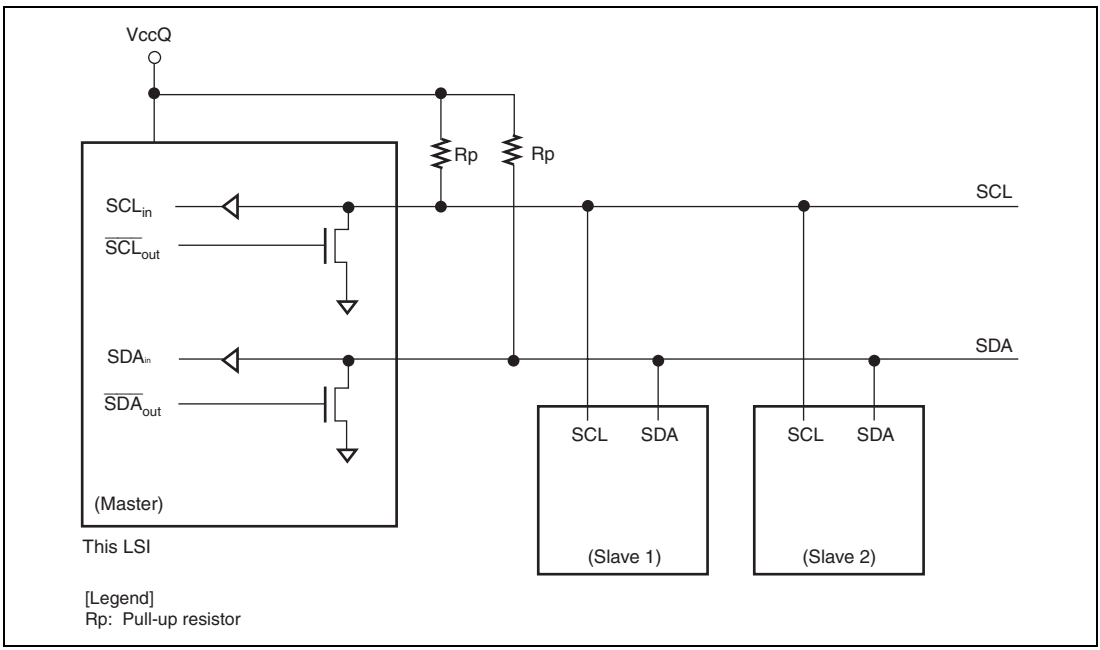


Figure 32.2 I²C Bus Interface Connections

32.2 Input/Output Pins

Table 32.1 summarizes the input/output pins used by the I²C bus interface.

Table 32.1 I²C Bus Interface Pins

Abbreviation	Function	I/O	Description
SCL0 (O/D)	I²C clock input/ output 0	I/O	I²C bus clock input/output pin of I²C0 Equipped with the bus drive function (NMOS open-drain).
SDA0 (O/D)	I²C data input/ output 0	I/O	I²C bus data input/output pin of I²C0 Equipped with the bus drive function (NMOS open-drain).
SCL1 (O/D)	I²C clock input/ output 1	I/O	I²C bus clock input/output pin of I²C1 Equipped with the bus drive function (NMOS open-drain).
SDA1 (O/D)	I²C data input/ output 1	I/O	I²C bus data input/output pin of I²C1 Equipped with the bus drive function (NMOS open-drain).

32.3 Register Descriptions

Table 32.2 shows the register configuration of the bus interface. Table 32.3 shows the register state in each processing mode.

Table 32.2 Register Configuration of Bus Interface

Name	Abbreviation	R/W	Address	Access Size
I²C bus data register	ICDR0	R/W	H'A4470000	8
I²C bus control register	ICCR0	R/W	H'A4470004	8
I²C bus status register	ICSR0	R/W	H'A4470008	8
I²C interrupt control register	ICIC0	R/W	H'A447000C	8
I²C clock control register low	ICCL0	R/W	H'A4470010	8
I²C clock control register high	ICCH0	R/W	H'A4470014	8
I²C bus data register	ICDR1	R/W	H'A4750000	8
I²C bus control register	ICCR1	R/W	H'A4750004	8
I²C bus status register	ICSR1	R/W	H'A4750008	8
I²C interrupt control register	ICIC1	R/W	H'A475000C	8
I²C clock control register low	ICCL1	R/W	H'A4750010	8
I²C clock control register high	ICCH1	R/W	H'A4750014	8

Table 32.3 Register State in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
ICDR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICCR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICSR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICIC0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICCL0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICCH0	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICDR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICCR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICSR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICIC1	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICCL1	Initialized	Initialized	Retained	Retained	Initialized	Retained
ICCH1	Initialized	Initialized	Retained	Retained	Initialized	Retained

32.3.1 I²C Bus Data Register (ICDR)

ICDR is an 8-bit readable/writable register that is used as a transmit data register when transmitting data and as a receive data register when receiving data.

This register can be written to when the DTE bit in ICSR is 1 in transmission.

Receive data of ICDR is valid when the DTE bit in ICSR is 1 in reception.

The read value of ICDR is H'FF after a reset.

Bit:	7	6	5	4	3	2	1	0
	ICDR							
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

32.3.2 I²C Bus Control Register (ICCR)

ICCR is an 8-bit readable/writable register that enables or disables the I²C bus interface, and transmission or reception, confirms the I²C bus interface bus status, and issues start/stop conditions.

Table 32.4 ICCR Register Settings

Issue Condition	When Transmitting	When Receiving
Start condition	H'94	—
Retransmission condition	H'94	H'D4 Changes from reception to transmission
Stop condition	H'90	H'C0
Changes from transmission to reception	H'81	—

Bit:	7	6	5	4	3	2	1	0
	ICE	RACK	—	TRS	—	BBSY	—	SCP
Initial value:	0	0	0	0	0	0	0	1
R/W:	R/W	R/W	R	R/W	R	R/W	R	(R/W)*

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I²C Bus Interface Enable Selects whether or not the I²C bus interface is to be used. When ICE is set to 1, transfer operations and access to each register are enabled. When ICE is cleared to 0 during data transmission, ICE releases buses and is forcibly terminated, then all registers in the I²C module are initialized. (See section 27.6, Usage Notes.) When ICE is cleared to 0 after data transmission, all registers in the I²C module are initialized. When the ICCR is updated while this module is being operated, the ICE bit must be set to 1. 0: I²C module disabled and forcibly terminated when 0 is written in data transmission. I²C module internal registers are all initialized. 1: I²C bus interface module enabled for transfer operations

Bit	Bit Name	Initial Value	R/W	Description
6	RACK	0	R/W	Receive Acknowledge The RACK bit is for storing acknowledge data that is output to the transmission device when the I²C module is receiving data. After data is received from the transmitting device, acknowledge data that has been set to this bit is sent. 0: In receive mode, 0 is output to SDA at acknowledge output timing. 1: In receive mode, 1 is output to SDA at acknowledge output timing.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	TRS	0	R/W	Transmit/Receive Select TRS selects whether the I²C bus interface operates in master transmit mode or master receive mode. Modification of the TRS bit during transfer is possible, however, modification of the operating mode is deferred until transfer of the frame containing the acknowledge bit is completed, and the changeover is made after completion of the transfer. 0: Master receive mode When 0 is written (during data transfer, operating mode is changed after a frame is transferred) 1: Master transmit mode When 1 is written (Retransmission/halt conditions are written. At this time, operating mode is changed when retransmission/halt conditions are generated after data transfer.)
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	BBSY	0	R/W	Bus Busy The BBSY flag is used to issue start and stop conditions. To issue a start condition, write 1 in BBSY and 0 in SCP. A retransmit start condition is issued in the same way. To issue a stop condition, write 0 in BBSY and 0 in SCP. Since this module is always in master mode, a bus state is not detected by the BBSY bit. When a start condition is issued from other masters, it is detected that arbitration is lost and operation is stopped. 0: A stop condition is issued (the SCP bit is also used). 1: Start and retransmit conditions are issued (the SCP bit is also used).
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	SCP	1	(R/W)*	Start Condition/Stop Condition Prohibit Controls the issuing of start and stop conditions. To issue a start condition, write 1 in BBSY and 0 in SCP. A retransmit start condition is issued in the same way. To issue a stop condition, write 0 in BBSY and 0 in SCP. This bit is always read as 1. If 1 is written, the data is not stored. 0: Writing 0 issues a start or stop condition, in combination with the BBSY flag 1: Reading always returns a value of 1. Writing is ignored.

Note: * For the SCP bit, only writing 0 is valid. Always read as 1.

32.3.3 I²C Bus Status Register (ICSR)

ICSR is an 8-bit readable/writable register that performs interrupt request.

Bit:	7	6	5	4	3	2	1	0
	SCLM	SDAM	—	BUSY	AL	TACK	WAIT	DTE
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R

Bit	Bit Name	Initial Value	R/W	Description
7	SCLM	1	R	SCL Monitor Stores data in which the SCL state is sampled at Pϕ. 0: Indicates SCL is 0. 1: Indicates SCL is 1.
6	SDAM	1	R	SDA Monitor Stores data in which the SDA state is sampled at Pϕ. 0: Indicates SDA is 0. 1: Indicates SDA is 1.
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4	BUSY	0	R	I²C Transmit State Bit Indicates the I²C state. After the start condition is generated, I²C is in the transmit state and this bit is set to 1. After the stop condition is generated, I²C is in the non-transmit state and this bit is cleared to 0. When arbitration is lost, the SDA line is open. The SCL clock line outputs clocks until the acknowledge end of the frame and releases buses. The BUSY flag is set to 1 until acknowledge end, and is cleared to 0 at acknowledge end. 0: In I²C non-transmit state after stop condition is generated. At acknowledge end after arbitration is lost 1: In I²C transmit state after start condition is generated

Bit	Bit Name	Initial Value	R/W	Description
3	AL	0	R/(W)*	Arbitration Lost This flag indicates that arbitration was lost. The I²C bus interface monitors the SDA. If the I²C bus interface detects data differing from the data it sent, it sets AL to 1 to indicate that the bus has been conflicted. AL is reset by writing 0 to this bit, write access to ICDR in transmit mode, or read access to ICDR in receive mode when DTE is 1. When arbitration lost is detected in data transfer, the SDA line is open. The SCL clock line outputs clocks until the acknowledge end of the frame and releases buses. The AL bit does not depend on the ALE bit in ICIC, and is always updated. When the ALE bit in ICIC is 0, an arbitration lost interrupt request is not issued to the interrupt controller. When other masters issue the start condition, arbitration lost is detected and operation is stopped. 0: Bus arbitration won When ICDR data is written (transmit mode) or read (receive mode) When 0 is written in AL after reading AL = 1 1: Arbitration lost If receive data changes when SCL clocks count at a high level in master receive mode If the SCL line is in a low level when SCL clocks count at a high level If a start condition is detected before this module issues a start condition If the monitored external SDA and data output from this module disagree

Bit	Bit Name	Initial Value	R/W	Description
2	TACK	0	R/(W)*	Transmit Acknowledge Bit Stores acknowledge data issued from the receiving device in transmit mode. The non-acknowledge interrupt state can be cleared by clearing the TACK bit to 0. In transmit mode, after the receiving device receives data, it returns acknowledge data, and this data is loaded into TACK. This bit is always 0 when the TACKE bit in ICIC is 0. 0: Transmit mode: Indicates that the receiving device has acknowledged the data (signal is 0) When 0 is written, indicates that the receiving device has acknowledged the data (signal is 0) 1: Transmit mode: Indicates that the receiving device has not acknowledged the data (signal is 1)
1	WAIT	0	R/(W)*	Wait Indicates the wait state after transmitting data other than the acknowledge bit. The wait state is entered by making SCL low after transmitting data other than acknowledge when WAITE in ICIC is 1. At this time, the WAIT bit is automatically set to 1 and a WAIT interrupt occurs. The wait state is canceled by clearing the WAIT bit to 0. WAIT is always 0 when WAITE in ICIC is 0. 0: Normal state 1: At wait

Bit	Bit Name	Initial Value	R/W	Description
0	DTE	0	R	Data Transmit Enable Indicates the transmit state between ICDR and internal transmit/receive buffer. This bit is read only, and automatically set or cleared. 0: When reset When data is written to ICDR in transmit mode (TRS = 1) When data is read from ICDR in receive mode (TRS = 0) When the TRS bit is changed When the retransmit/stop conditions are written to the BBSY and SCP bits in ICCR 1: When the start/retransmit conditions are generated When transmit data is sent to the transmit buffer from ICDR in transmit mode (TRS = 1) When receive data is sent to ICDR in receive mode (TRS = 0)

Note: * Only 0 can be written, for flag clearing.

32.3.4 I²C Interrupt Control Register (ICIC)

ICIC is an 8-bit register that enables/disables an interrupt request.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	ALE	TACKE	WAITE	DTEE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	ALE	0	R/W	Arbitration Lost Interrupt Enables/disables an arbitration lost interrupt request. An arbitration lost interrupt occurs if arbitration lost occurs when ALE is 1. At this time, the AL bit in ICSR is set to 1. 0: Interrupt disabled 1: Arbitration lost interrupt
2	TACKE	0	R/W	Non-acknowledge Detection Interrupt Enables/disables a non-acknowledge detection interrupt request. A non-acknowledge interrupt occurs if non-acknowledge is received when TACKE is 1. At this time, the TACK bit in ICSR is set to 1. 0: Interrupt disabled 1: Non-acknowledge detection interrupt
1	WAITE	0	R/W	Wait Interrupt Enables/disables a wait interrupt request. The wait state is entered by making SCL low after transmitting data other than acknowledge when WAITE is 1. At this time, a WAIT interrupt occurs, and the WAIT bit in ICSR is set to 1. 0: Interrupt disabled 1: Wait interrupt
0	DTEE	0	R/W	Data Transmit Enable Interrupt Enables/disables a data transmit enable interrupt request. An interrupt request is issued if the DTE bit in ICSR is set to 1 when DTEE is 1. 0: Interrupt disabled 1: Data transmit enable interrupt

32.3.5 I²C Clock Control Register Low (ICCL)

ICCL is an 8-bit readable/writable register that can set low-level time of SCL.

ICCL is initialized to H'00 by a reset.

Bit:	7	6	5	4	3	2	1	0
	ICCL							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: Settings of H'01 to H'09 are prohibited. No SCL clock is generated for ICCL = H'00 and H'FF or ICCH = H'00 and H'FF.

32.3.6 I²C Clock Control Register High (ICCH)

ICCH is an 8-bit readable/writable register that can set high-level-time of SCL.

ICCH is initialized to H'00 by a reset.

Bit:	7	6	5	4	3	2	1	0
	ICCH							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: Settings of H'01 to H'09 are prohibited. No SCL clock is generated for ICCL = H'00 and H'FF or ICCH = H'00 and H'FF.

32.3.7 Transfer Rate

An I²C transfer rate can be set by ICCL, ICCH, and Pϕ. Table 32.5 lists transfer rate settings, and Figure 32.3 shows SCL waveforms.

Table 32.5 Transfer Rate Settings

	Pϕ	ICCL	ICCH	Transfer Rate*[2]
Standard	12 MHz	H'42	H'36	100 kHz
	13.5 MHz	H'4A	H'3D	100 kHz
	24 MHz	H'82	H'6E	100 kHz
	27 MHz	H'96	H'78	100 kHz
	33 MHz	H'AF	H'9B	100 kHz
High speed*[1]	12 MHz	H'12	H'0C	400 kHz
	13.5 MHz	H'16	H'0C	397.1 kHz
	24 MHz	H'26	H'16	400 kHz
	27 MHz	H'2C	H'18	397.1 kHz
	33 MHz	H'33	H'20	397.6 kHz

Notes: 1. When the set value of ICCL/ICCH is H'0A (minimum), Pϕ requires 8 MHz if a transfer rate is 400 kHz. If Pϕ is less than 8 MHz, a transfer rate of 400 kHz is not ensured.

2. The transfer rates in above settings are theoretical values. The actual transfer rates are generally smaller than those theoretical values due to the pull-up resistor (Rp) and the capacitance (Cp). However, there is no harm in communication because the actual transfer rates are held within the I²C bus specifications. (Smaller ICCL and ICCH can increase the actual transfer rates. In this case, though, set ICCL and ICCH to ensure t_{LOW} and t_{HIGH} within the specifications.)

Notes on the transfer rate:

The transfer rates in above settings are theoretical values. The actual transfer rates are generally smaller than the theoretical values due to the pull-up resistor (Rp) and the capacitance (Cp). However, there is no harm in communication because the actual transfer rates are held within the I²C bus specifications. (Smaller ICCL and ICCH can increase the actual transfer rates. In this case, though, set ICCL and ICCH to ensure t_{LOW} and t_{HIGH} within the specifications.)

A calculation example of ICCL and ICCH:*[1]

- Conditions
  - Pϕ: 33 MHz
  - I²C transfer rate: 100 kHz
  - L/H ratio in SCL*[2]: L/H = 5/4

- Result
  - ICCL = (Pϕ ÷ I²C transfer rate) × (L ÷ (L + H))
  - = $((33 \times 10^6) \div (100 \times 10^3)) \times (5 \div (5 + 4) = 183 = (B7)_{16}$
  - ICCH = (Pϕ ÷ I²C transfer rate) × (H ÷ (L + H))
  - = $((33 \times 10^6) \div (100 \times 10^3)) \times (4 \div (5 + 4) = 146 = (92)_{16}$

Notes: 1. If the result is indivisible, some modifications are introduced to make the total count number closer to the desirable transfer rate.
2. SCL L/H ratio is arbitrary as long as it satisfies t_{LOW} and t_{HIGH}. In above case, L/H is set as 5/4.

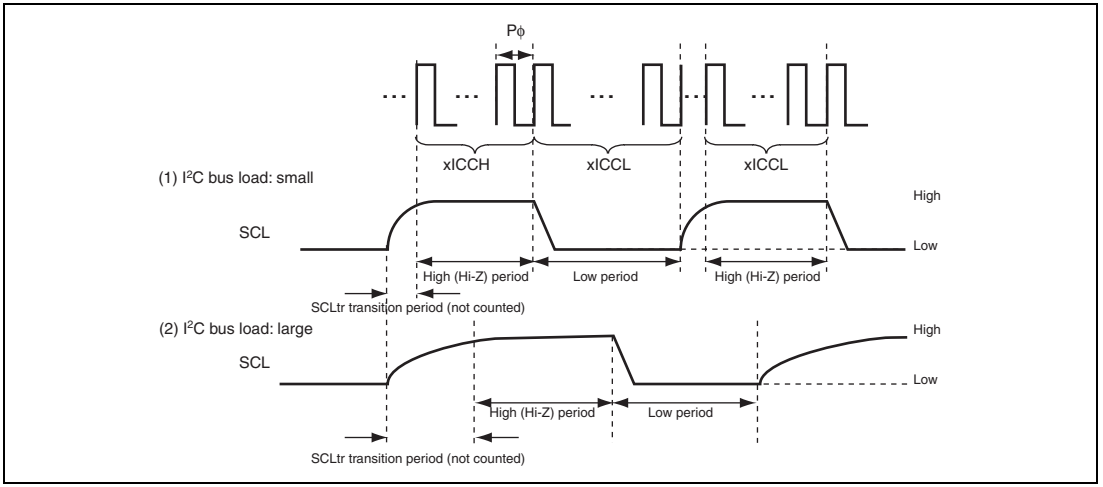


Figure 32.3 SCL Waveforms

32.4 Operation

32.4.1 I²C Bus Data Format

The I²C bus formats are addressing formats with an acknowledge bit. These are shown in figures 32.4 and 32.5. The first frame following a start condition always consists of 8 bits.

The symbols used in figures 32.4 and 32.5 are explained in Table 32.6.

Table 32.6 I²C Bus Data Format Symbols

Symbol	Description
S	Start condition. The master device drives SDA from high to low while SCL is high
Sr	Retransmit condition. The master device drives SDA from high to low while SCL is high
SLA	Slave address, by which the master device selects a slave device
R/$\overline{W}$	Indicates the direction of data transfer: from the slave device to the master device when R/$\overline{W}$ is 1, or from the master device to the slave device when R/$\overline{W}$ is 0
A	Acknowledge. The receiving device (the receiving device in master transmit mode, or the master in master receive mode) drives SDA low to acknowledge a transfer
DATA	Transmitted or received data. The bit length is 8 bits
P	Stop condition. The master device drives SDA from low to high while SCL is high

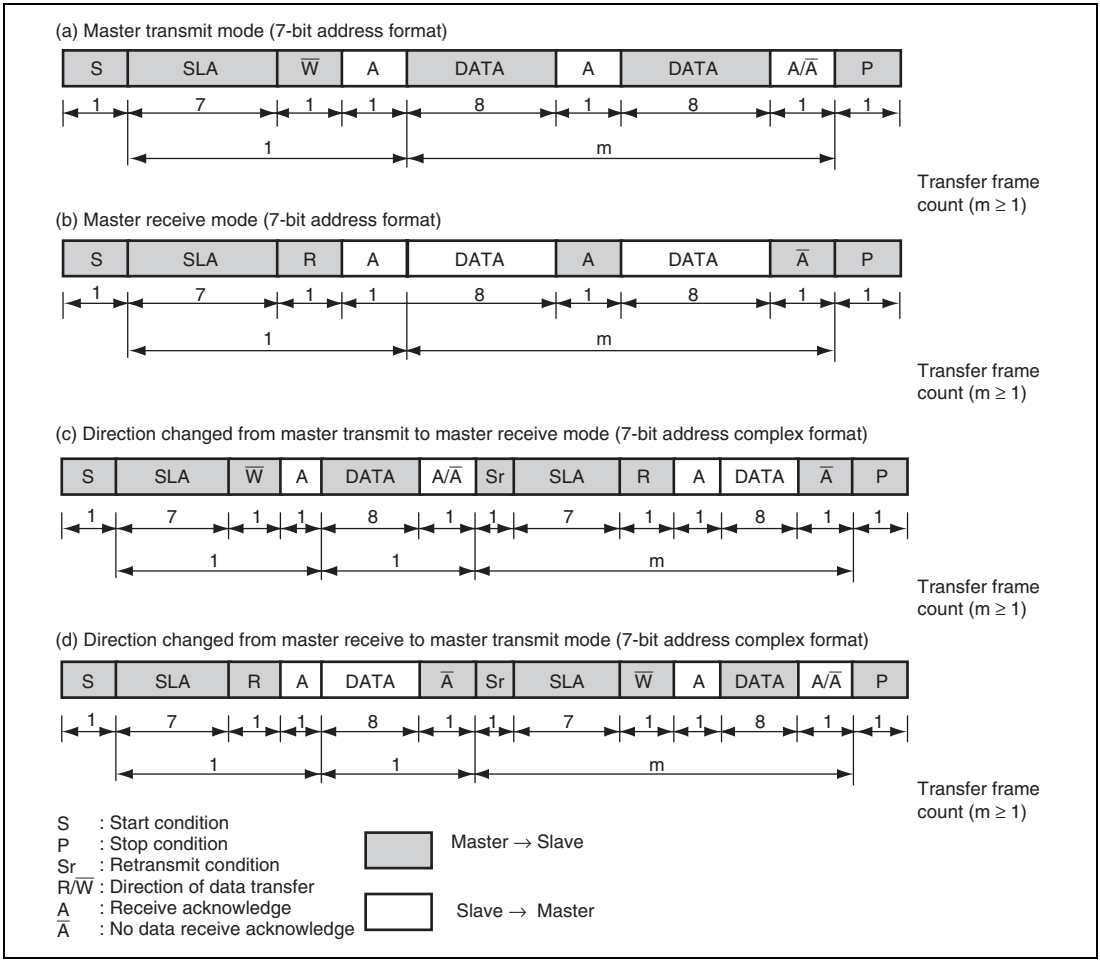


Figure 32.4 I²C Bus Data Formats (7-Bit Address Format)

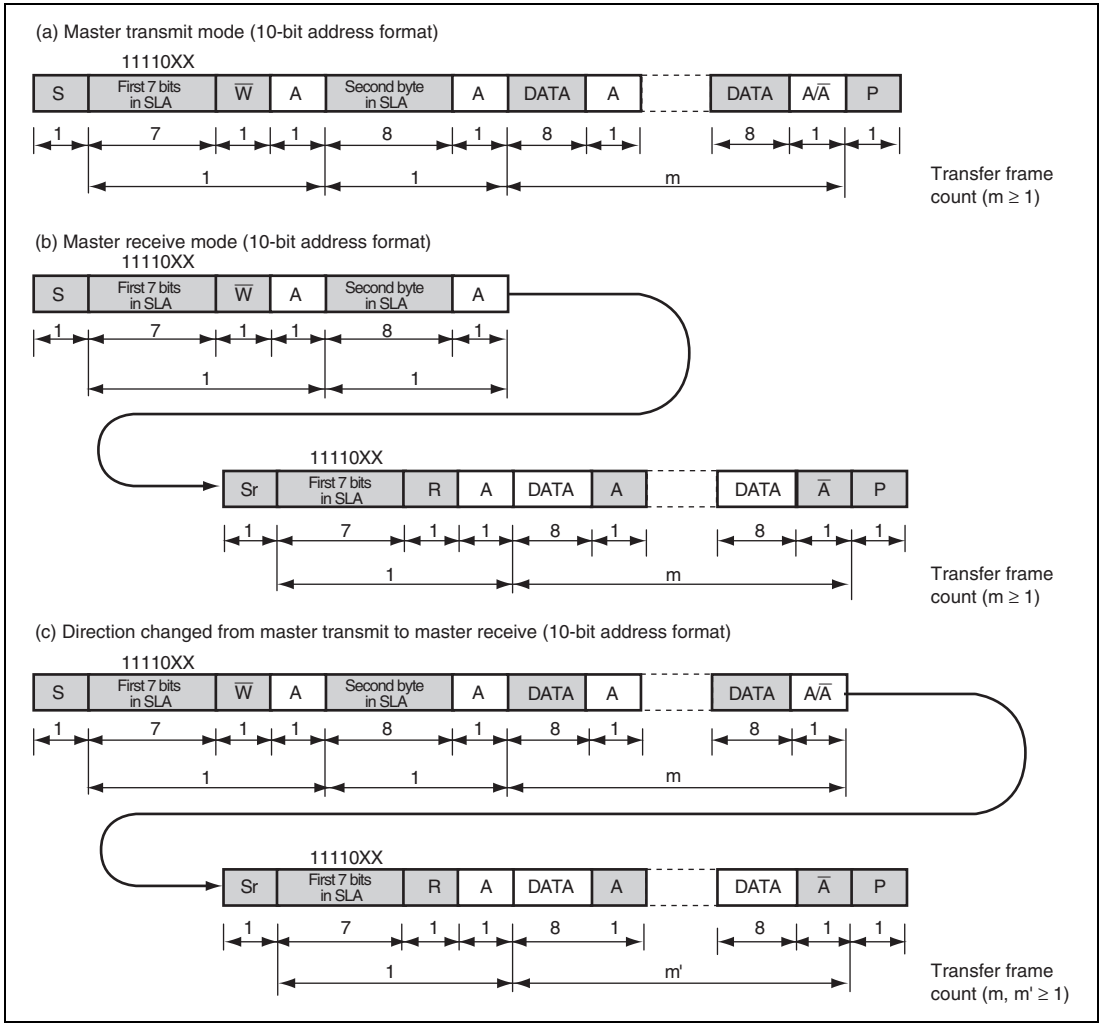


Figure 32.5 I²C Bus Data Formats (10-Bit Address Format)

32.4.2 Master Transmit Operation

In I²C bus format master transmit mode, the master device outputs the transmit clock and transmit data, and after the receiving device receives data, it returns an acknowledge signal. The transmission procedure and operations are described below.

1. Set bits in ICCL, ICCH, and ICIC according to the operating mode.
2. Write H'94 to registers to issue the start condition. The DTE bit is then set to 1.
3. To transmit data, write data to ICDR. The DTE bit is then automatically cleared.

   Data transmission starts when data is sent to the internal transmit buffer. The DTE bit is then set to 1 again.
4. To issue the stop condition, write the final data to ICDR, then write H'90 to ICCR.

   DTE is not set to 1 from issuing the stop condition to generating the stop condition. After the final data is transmitted, the stop condition is generated and operation is stopped.

   If writing to ICDR and ICCR are delayed and ACK is received before completion of the writing, a DTE interrupt is generated before generation of stop condition. For this reason, either disable the DTE interrupt after writing H'90 to ICCR register or write data to the ICDR and ICCR registers before ACK bit is generated.

   Note that data can be written to the ICDR and ICCR before generation of the ACK bit by writing to ICDR and ICCR during the period from the point at which a WAIT interrupt is generated until its source is cleared.
5. To issue the retransmit condition, write the final data to ICDR, then write H'94 to ICCR.

   DTE is not set to 1 from issuing the retransmit condition to generating the retransmit condition. After the final data is transmitted, the retransmit condition is generated.

   If writing to ICDR and ICCR are delayed and ACK is received before completion of the writing, a DTE interrupt is generated before generation of retransmit condition and software cannot control for correct communication. For this reason, be sure to write data to the ICDR and ICCR before ACK bit is generated.

   Note that data can be written to the ICDR and ICCR before generation of the ACK bit by writing to ICDR and ICCR during the period from the point at which a WAIT interrupt is generated until its source is cleared.
6. To change from transmission to reception, write the final data to ICDR, then write H'81 to ICCR. After transmitting the final data, transmission is automatically changed to reception and receiving operation is started.

By repeating step 3, data can be consequently transmitted. A non-acknowledge interrupt occurs if the received acknowledge bit is 1 (TACK = 1) when non-acknowledge is detected. To continuously transmit data without detecting non-acknowledge, set the TACKE bit in ICIC to 0, and perform steps 3 and 4 to 6. At this time, since non-acknowledge is not detected, data can be continuously transmitted.

To change transmission to reception, write H'81 to ICCR until the completion of transmission of the final data. At this time, if writing to ICCR is delayed, use a WAIT interrupt to write to ICCR until the completion of transmission of the final data.

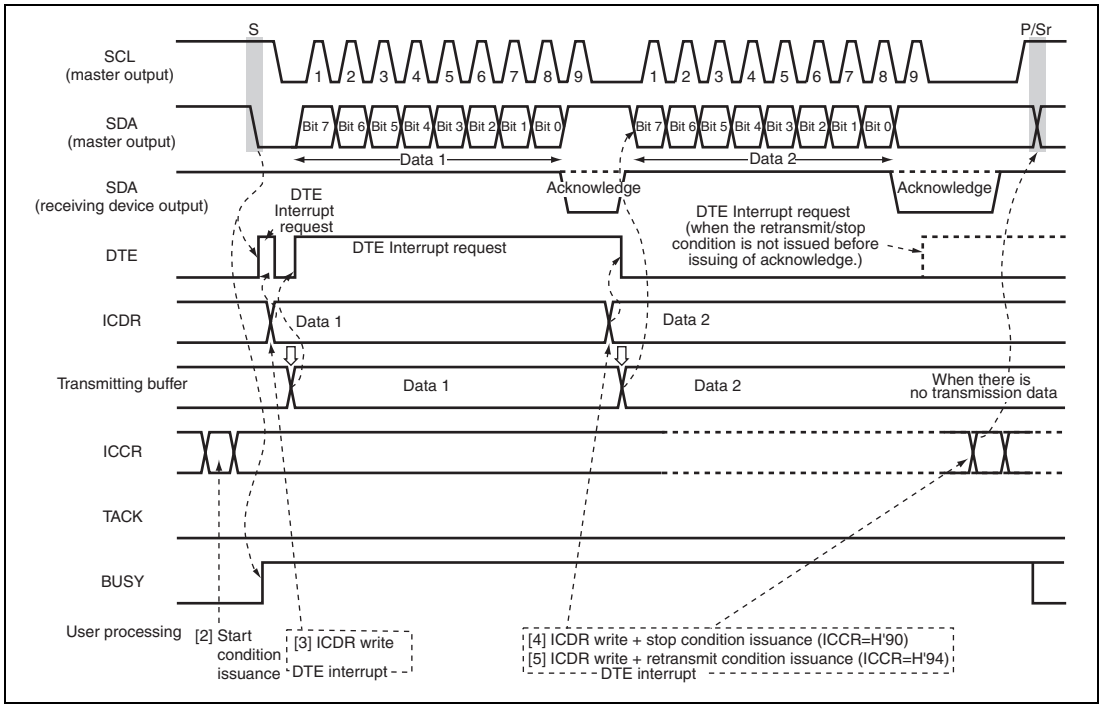


Figure 32.6 Example of Master Transmit Mode Operation Timing (WAIT = 0)

32.4.3 Master Receive Operation

In master receive mode, the master device outputs the receive clock, and returns an acknowledge signal after the transmit device transmits data. The reception procedure and operations in master receive mode are described below.

When Two or More Bytes of Data are Received Continuously:

1. Switch from transmit mode to receive mode automatically to start reception.
2. After data is received to the receive buffer, data is transmitted to ICDR and the DTE bit is set to 1. At this time, the next data reception starts.
3. To continuously receive data, read data from ICDR. When data is read from ICDR, the DTE bit is automatically cleared.
4. To issue/generate the stop condition, read data from ICDR, then write H'C0 to ICCR. After the stop condition is issued, the final data is received. After data is read from ICDR, the stop condition is generated to stop the operation.
5. To issue/generate the retransmit condition, read data from ICDR, then write H'D4 to ICCR. After the retransmit condition is issued, the final data is received. After data is read from ICDR, the retransmit condition is generated. Then, reception is automatically switched to transmission, and transmission starts.

Two or more bytes of data can be received continuously by repeating steps 2 and 3.

When continuously receiving two or more bytes of data, in order to generate a retransmit condition or stop condition during data reception, reading of the second last data from the ICDR register and writing of H'D4 or H'C0 to the ICCR register should be done. If reading of the second last data from ICDR and writing to ICCR are delayed and not completed until a NACK bit is generated at the final data reception, an ACK bit is generated instead at the final data reception. This means that the I²C protocol cannot be observed. For this reason, be sure to read from ICDR and write to ICCR until the NACK bit is generated at the final data reception.

When One Byte of Data only is Received:

When one byte of data only is received, use a WAIT interrupt to issue the stop/retransmit condition to ICCR.

1. Switch from transmit mode to receive mode automatically to start reception.
2. The WAIT bit is set to 1. At this time, write H'C0 to ICCR to issue/generate the stop condition. Write H'D4 to ICCR to issue/generate the retransmit condition.
3. The WAIT bit is cleared to 0.
4. After data is received to the receive buffer, data is transmitted to ICDR and the DTE bit is set to 1.
5. When the stop condition has been issued to ICCR, the stop condition is generated to stop the operation after data is read from ICDR. When the retransmit condition has been issued to ICCR, the retransmit condition is generated, reception is then automatically switched to transmission, and transmission starts.

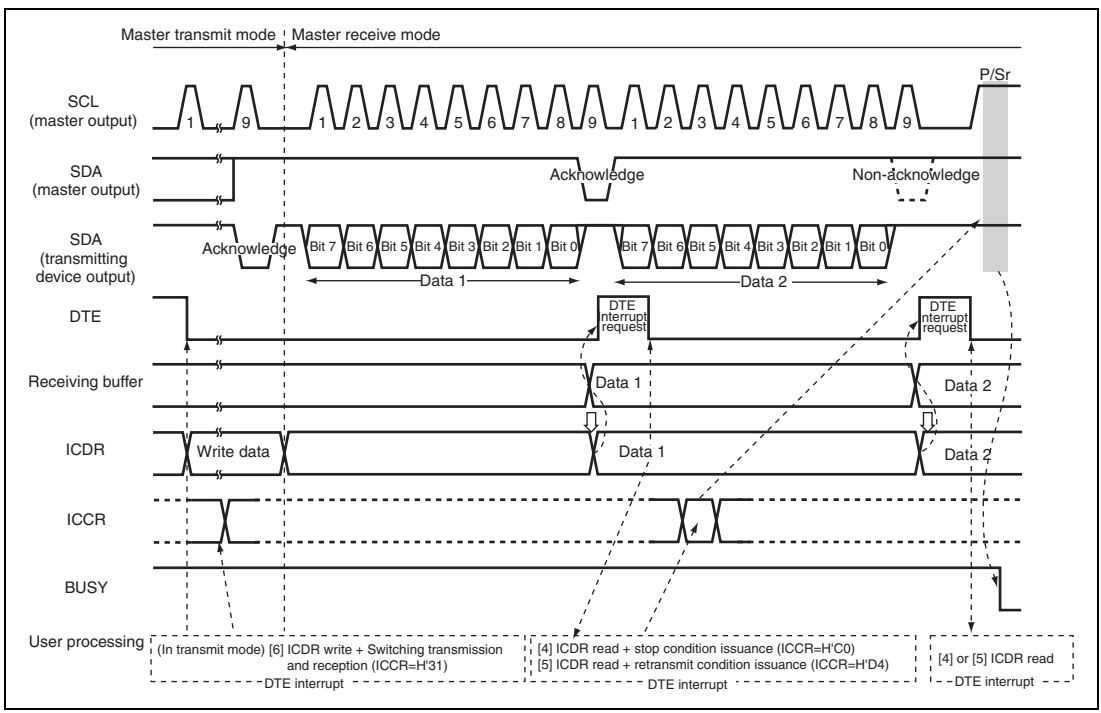


Figure 32.7 Example of Master Receive Mode Operation Timing (WAIT = 0)

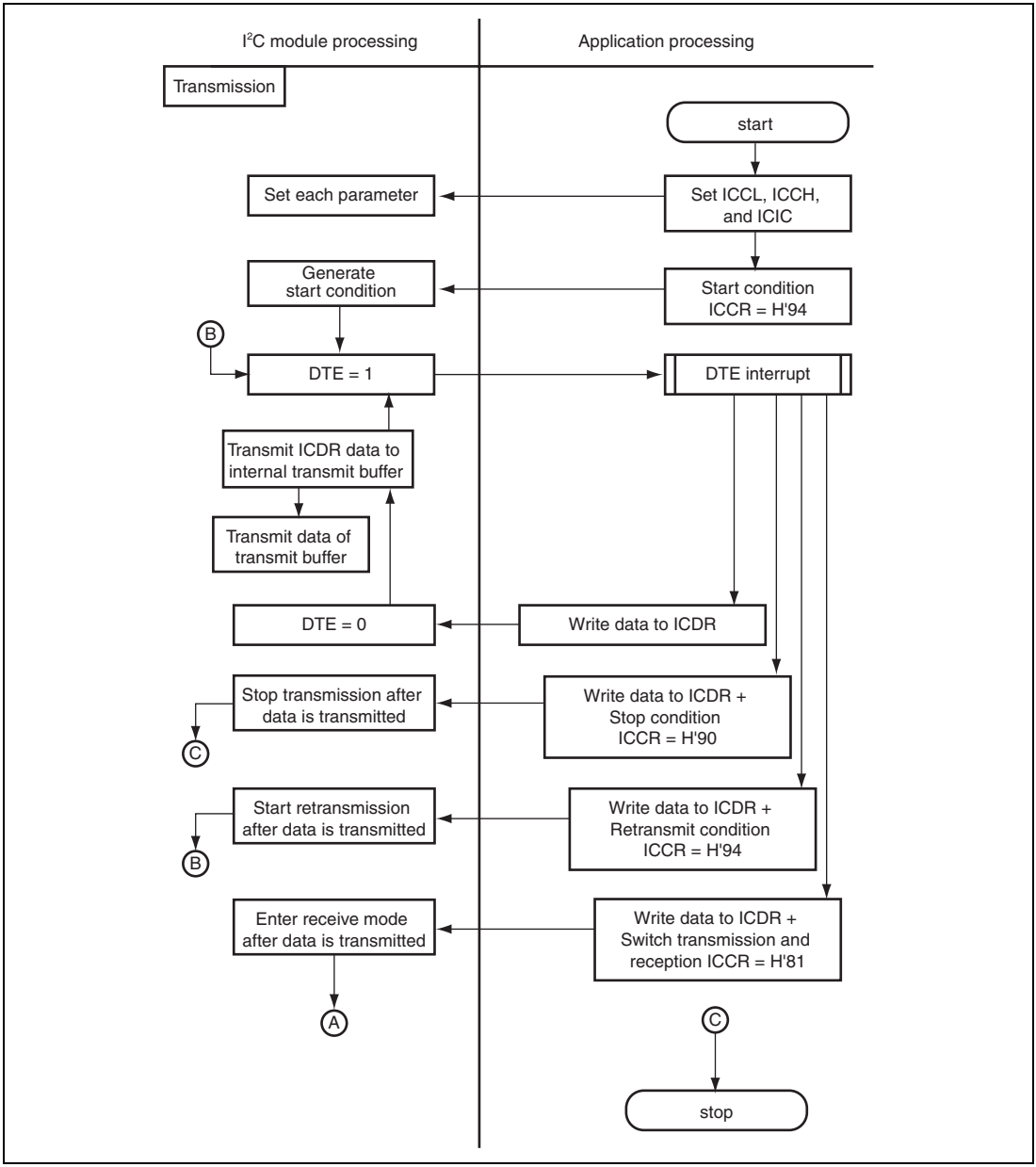


Figure 32.8 I²C Transmission Sequence (in Transmit Mode)

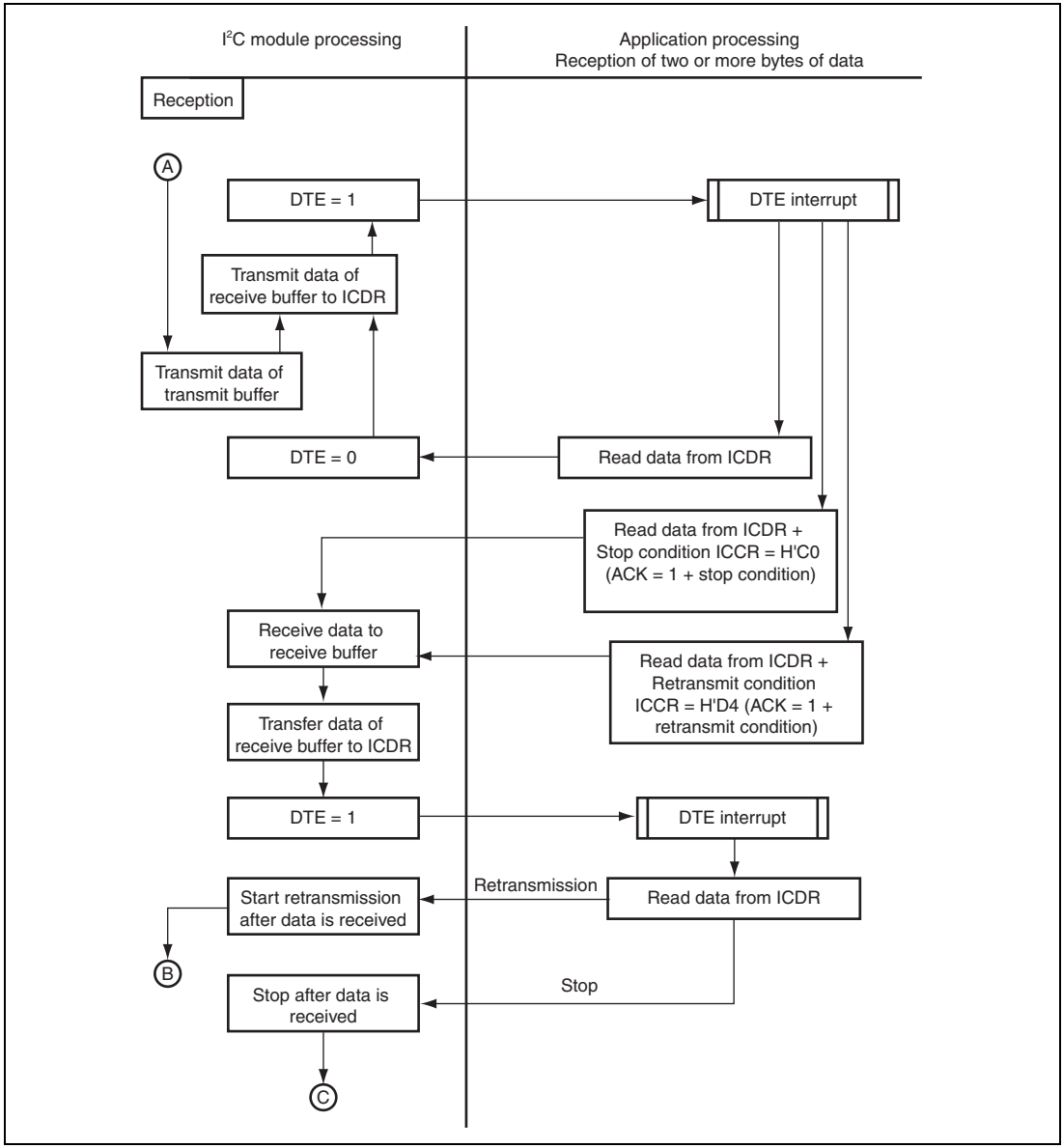


Figure 32.9 I²C Transmission Sequence (in Receive Mode of Two or More Bytes of Data)

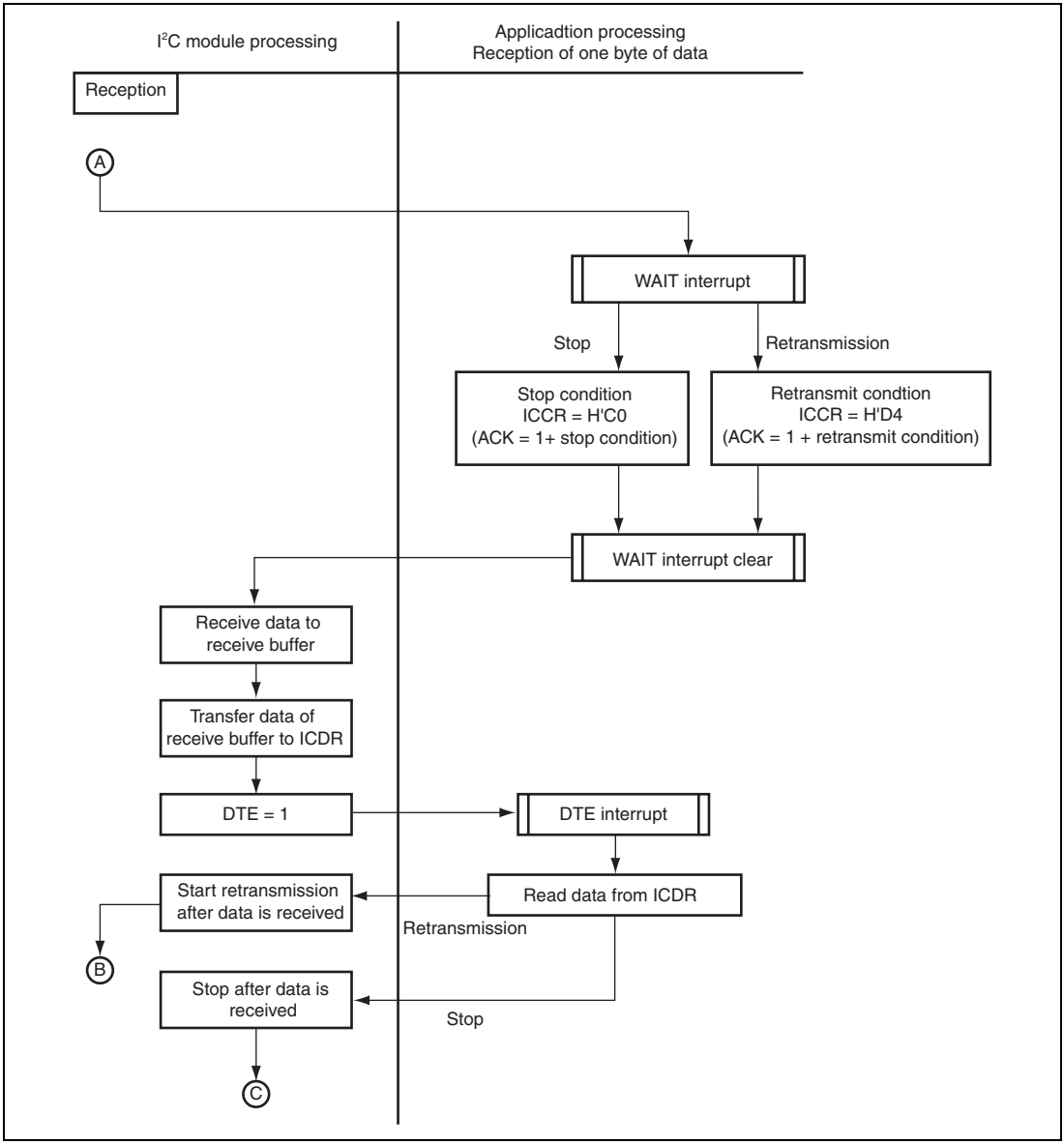


Figure 32.10 I²C Transmission Sequence (in Receive Mode of One Byte of Data)

32.4.4 Synchronizing SCL Line

When the SCL line state is changed from high to low, related devices start counting in the low period. When the clock of a device becomes low, the device holds the SCL line low until its clock becomes high (see Figure 32.11). However, even if the clock of the device is changed from low to high, the SCL line state does not change when clocks of other devices are in the low period. Therefore, the low period of the SCL line is determined by the device with the longest low period. At this time, a device with the short low period waits in the high state.

When all devices complete the low period, the clock line is open and in the high state. Therefore, the device clock and SCL line are in the same state, and start counting the high period. The SCL line enters the low state again by the device that has first completed the high period.

The low period is determined by the device with the longest low period, and the high period is determined by the device with the shortest high period. The SCL line is then synchronized.

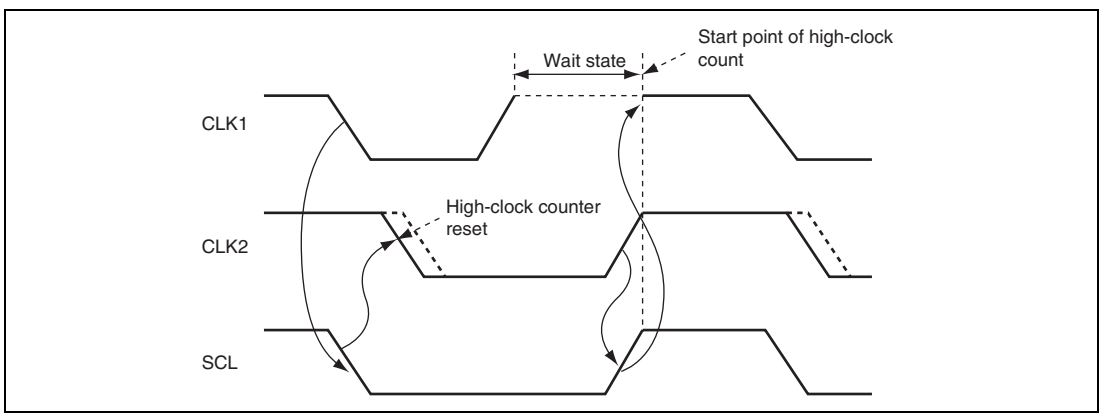


Figure 32.11 Synchronizing SCL Line

32.4.5 Noise Canceler

The logic levels at the SCL and SDA pins are routed through noise cancelers before being latched internally. Figure 32.12 shows a block diagram of the noise canceler circuit.

The noise canceler consists of three cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on Pϕ, but is not passed forward to the next circuit unless the outputs of both latches agree. If they do not agree, the previous value is held.

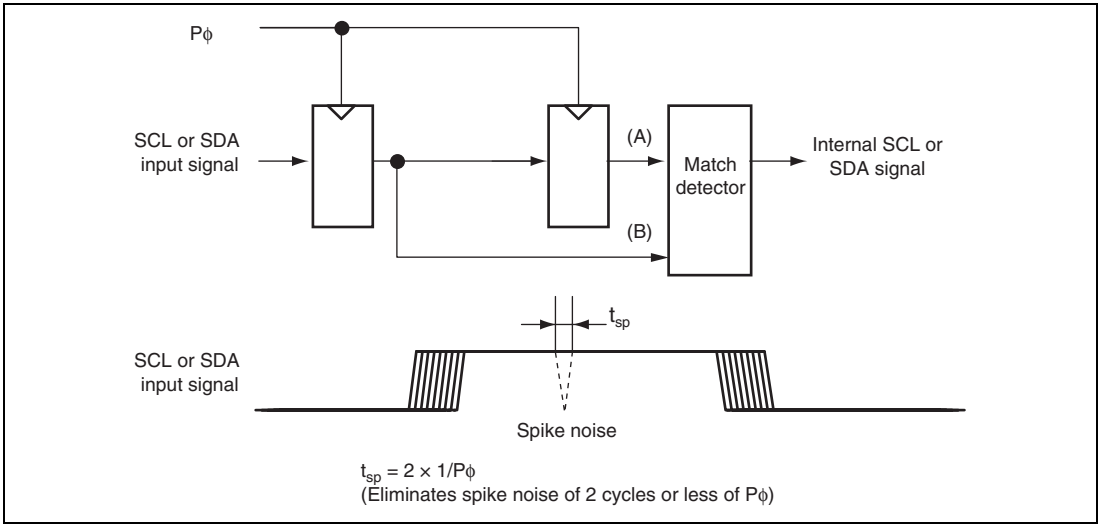


Figure 32.12 Block Diagram of Noise Canceler

32.4.6 Arbitration Lost Operation

This module can issue an arbitration lost interrupt request when bus conflict defeat occurs.

Interrupt occurrence conditions are described below.

- When received data changes while high SCL clocks are counted in master transmit/receive mode.
- When the SCL line becomes low while high SCL clocks are counted.
- When this module detects the start condition before issuing the start condition.
- When the monitored SDA (external) and data output from this module disagree.

Under the above conditions, the AL bit in ICSR is set to 1, and an arbitration lost interrupt occurs.

When arbitration lost is detected in data transmission, the SDA line is open. The SCL clock line outputs clocks until acknowledge end of the frame and opens buses. Figure 32.13 shows an example of arbitration lost interrupt operation timing.

To cancel an interrupt, clear the AL bit in ICSR to 0. Conditions for clearing to 0 are described below.

- When data is written to ICDR (in transmission), and data is read (in reception), when DTE = 1.
- When 0 is written.

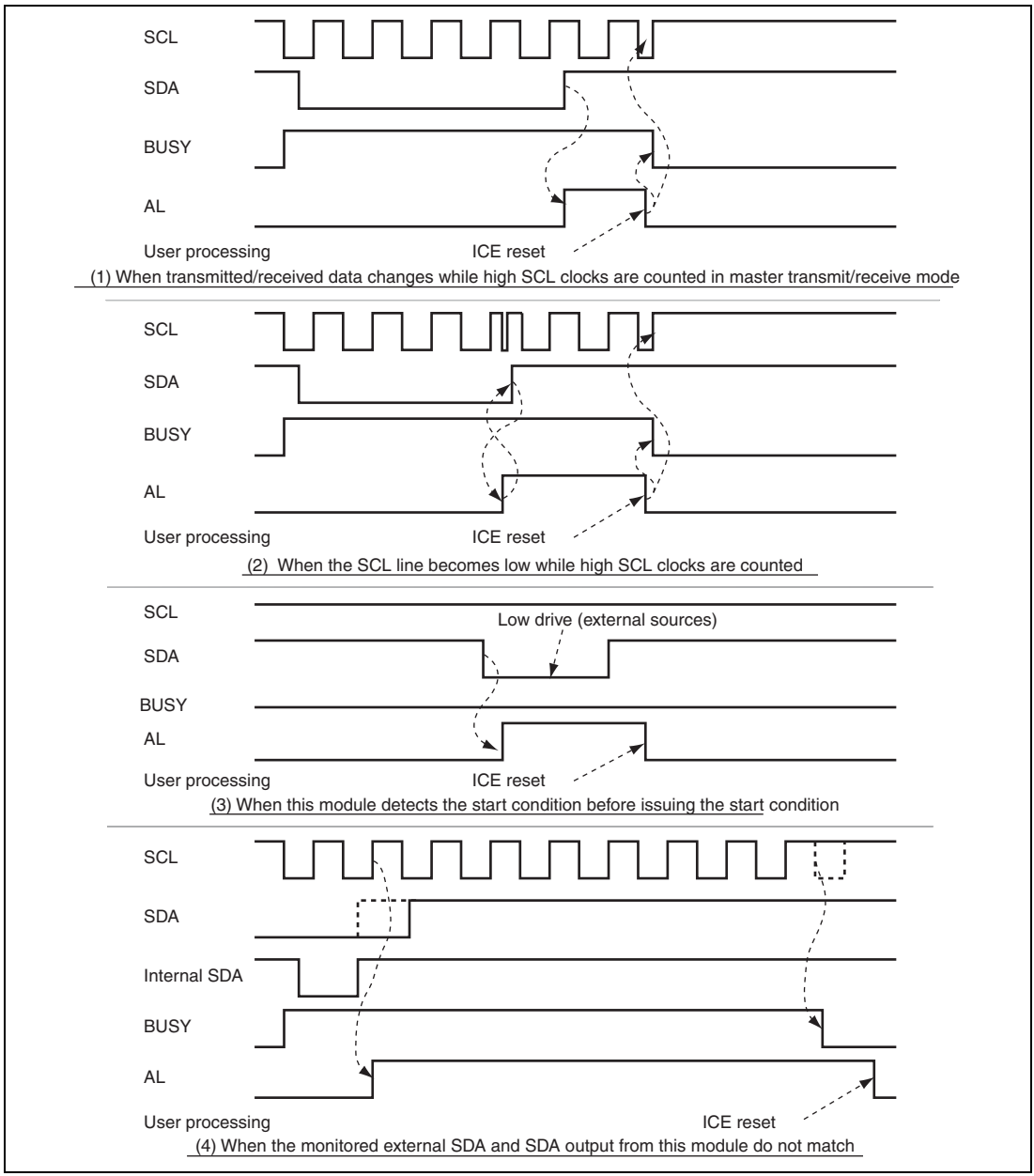


Figure 32.13 Example of Arbitration Lost Interrupt Operation Timing

32.4.7 Non-Acknowledge Operation

This module can issue a non-acknowledge interrupt when there is no acknowledge from the receiving device in transmission.

An acknowledge bit from the receiving device is stored in the TACK bit in ICSR. A non-acknowledge interrupt is issued if the TACK bit is set to 1 when TACKE in ICIC is 1.

To cancel the non-acknowledge interrupt, clear the TACK bit to 0. Figure 32.14 shows an example of non-acknowledge operation timing.

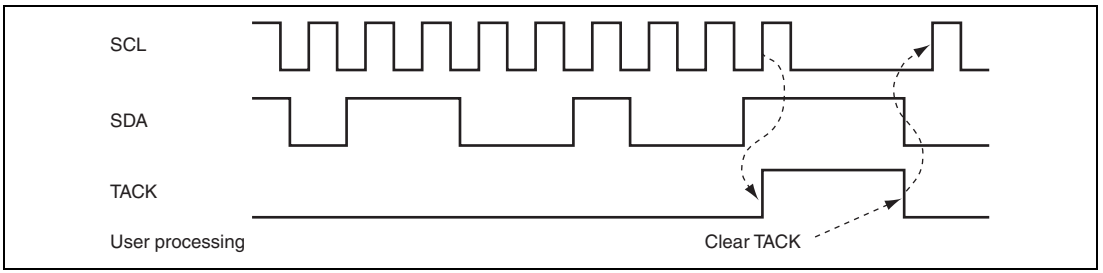


Figure 32.14 Example of Non-Acknowledge Operation Timing

32.4.8 Wait Operation

This module can make operation mode the wait state by WAIT in ICSR. WAIT is set to 1 at a falling edge of the eighth transmit clock for the master device.

After that, the transmit clock is fixed to low until WAIT is cleared to 0.

When WAIT is cleared to 0, ninth transmit clock is generated, and the state is returned from the wait state.

When communication is performed with use of the DTE interrupt alone, the I²C signals will not be held in the WAIT state. That is, use of the WAIT interrupt produces delay in communication by the length of time during which the I²C signals are held in the WAIT state.

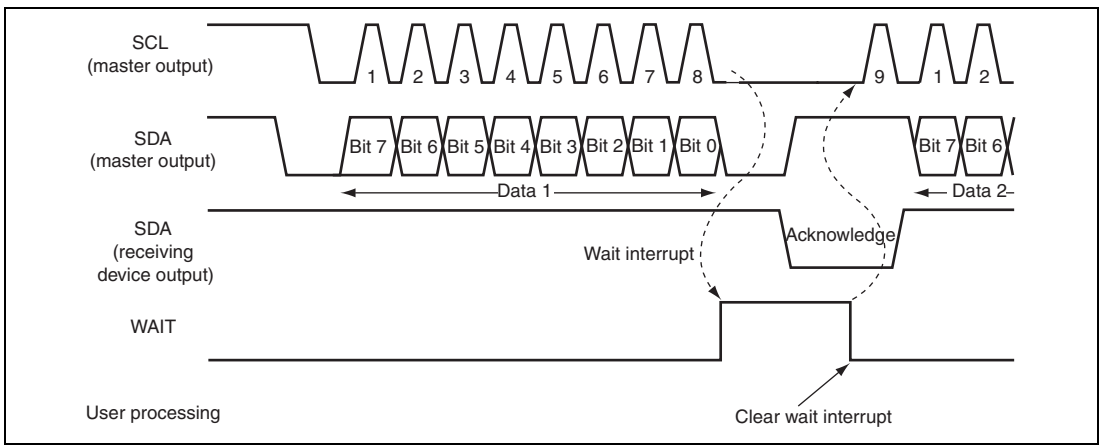


Figure 32.15 Example of Wait Operation Timing

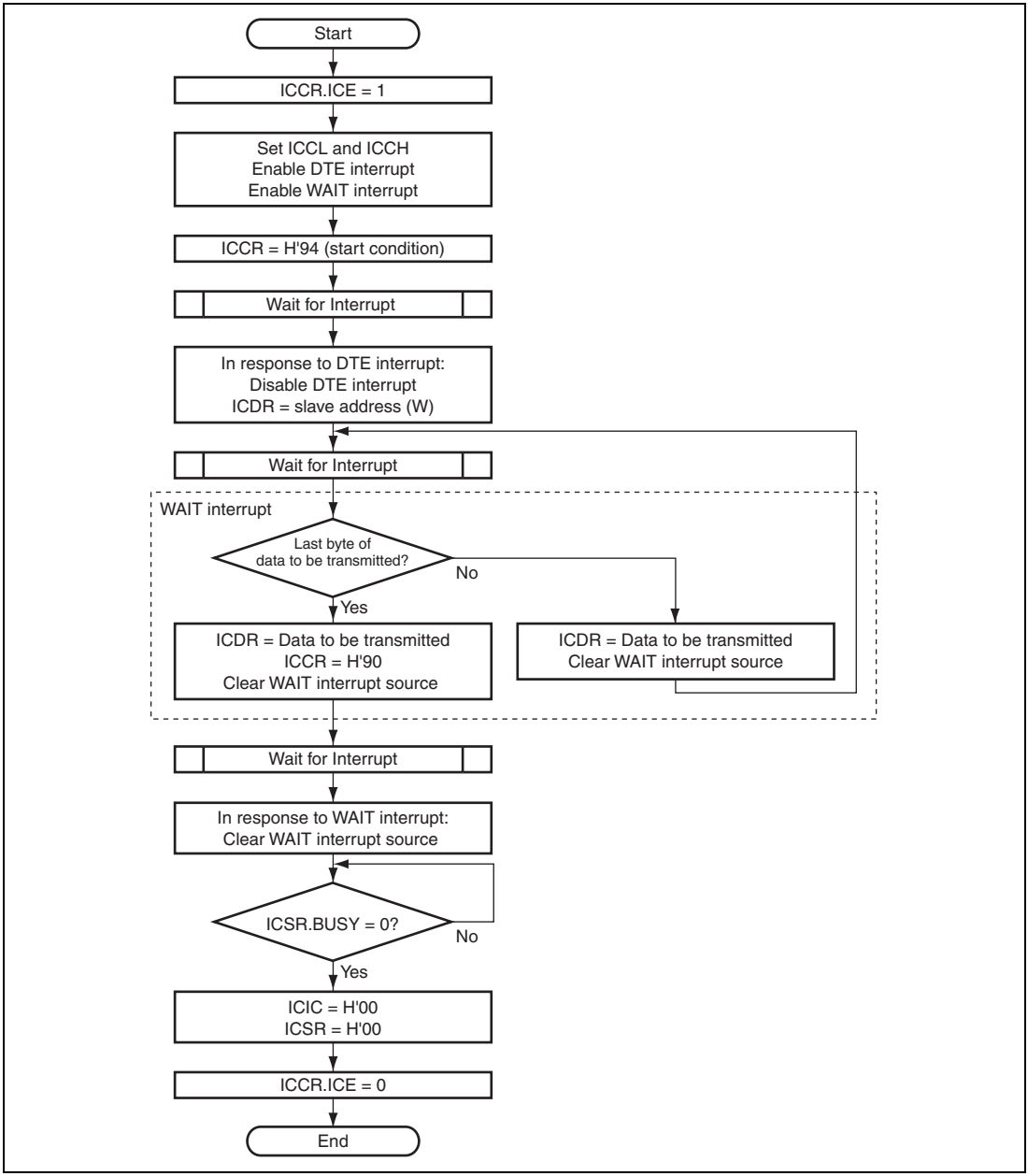


Figure 32.16 Example Flow of Software for Transmission with Use of Wait Interrupt

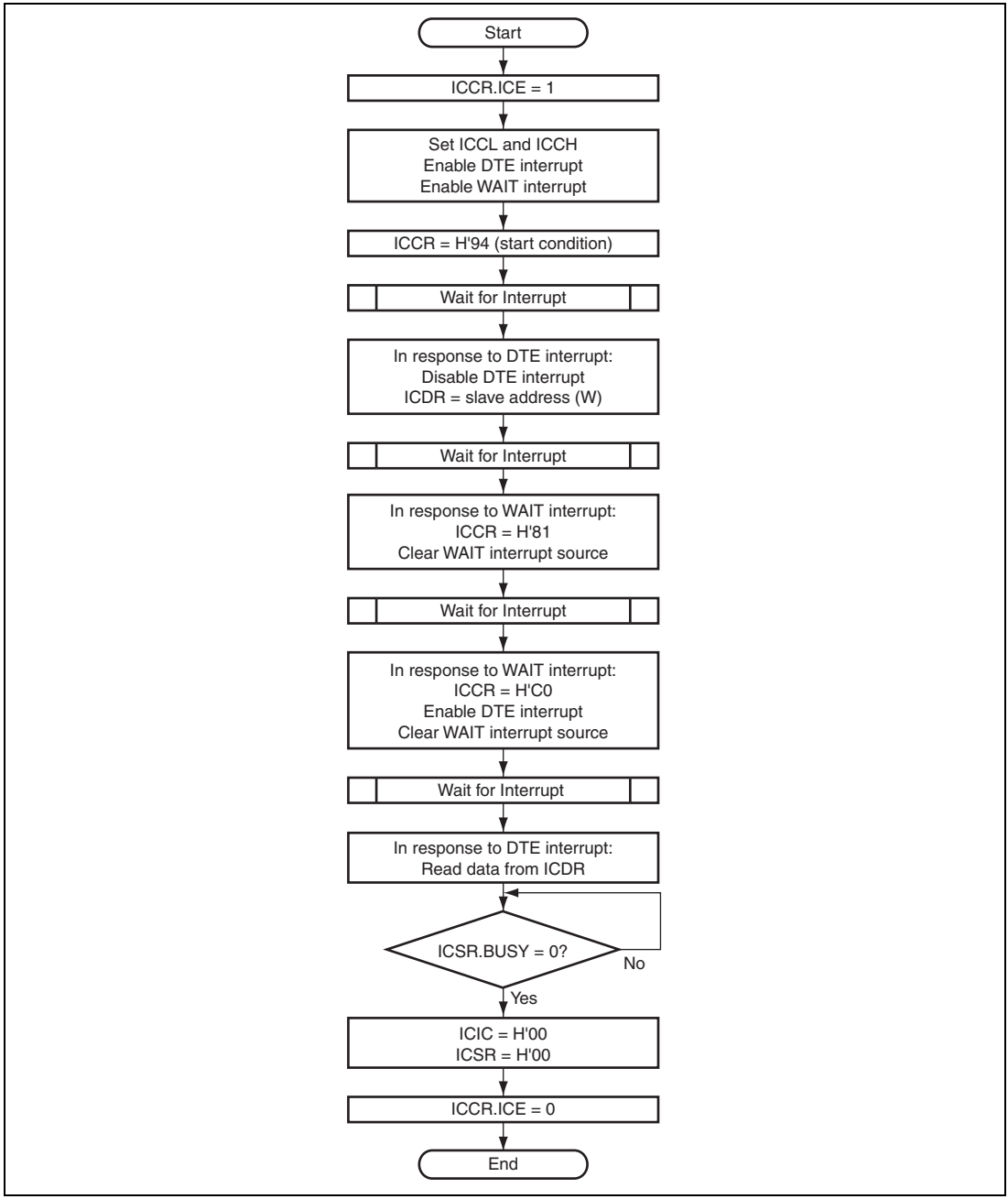


Figure 32.17 Example Flow of Software for 1-Byte Write/Read with Use of Wait Interrupt

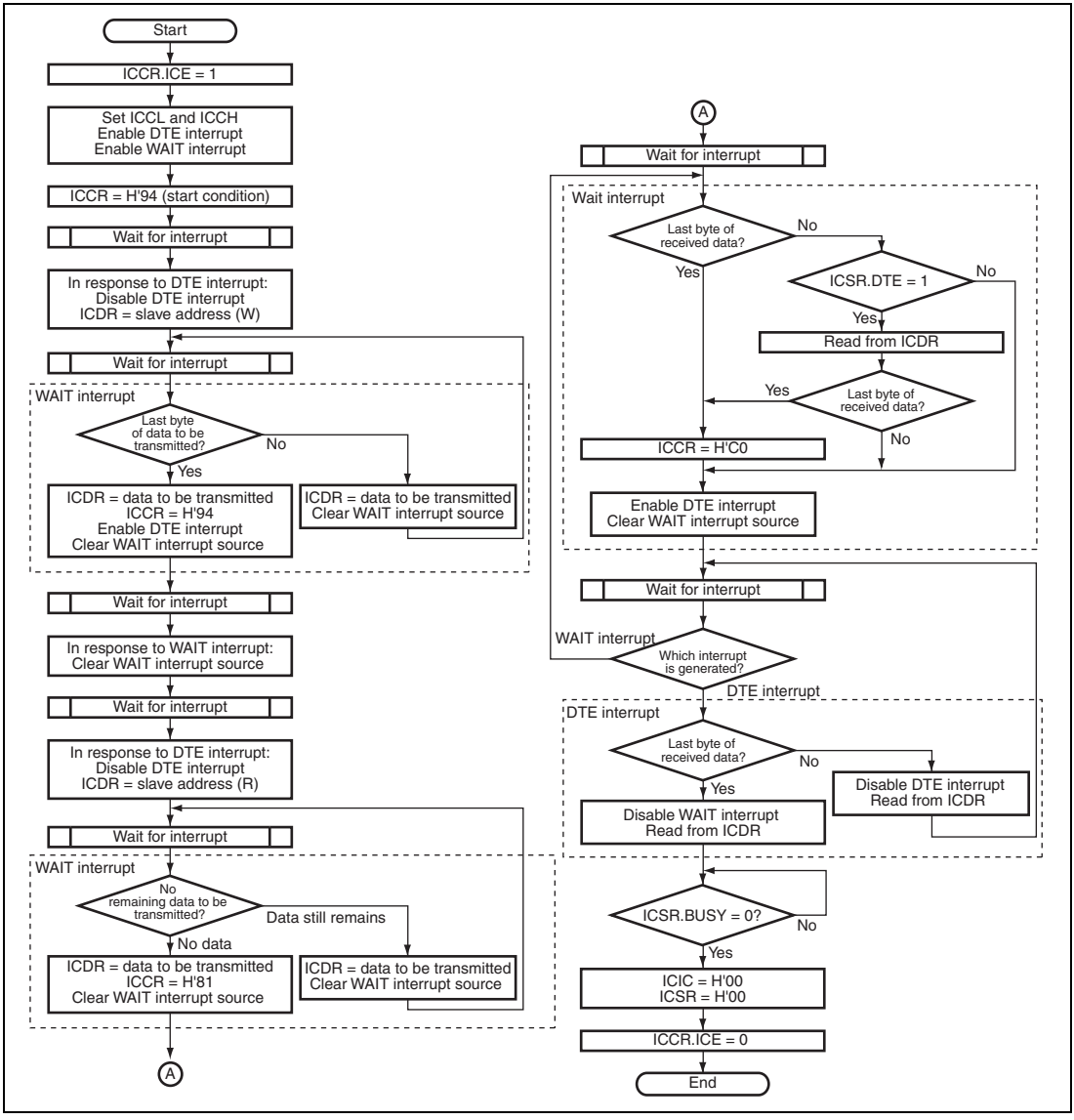


Figure 32.18 Example Flow of Software for n-Byte Write/Read with Use of Wait Interrupt

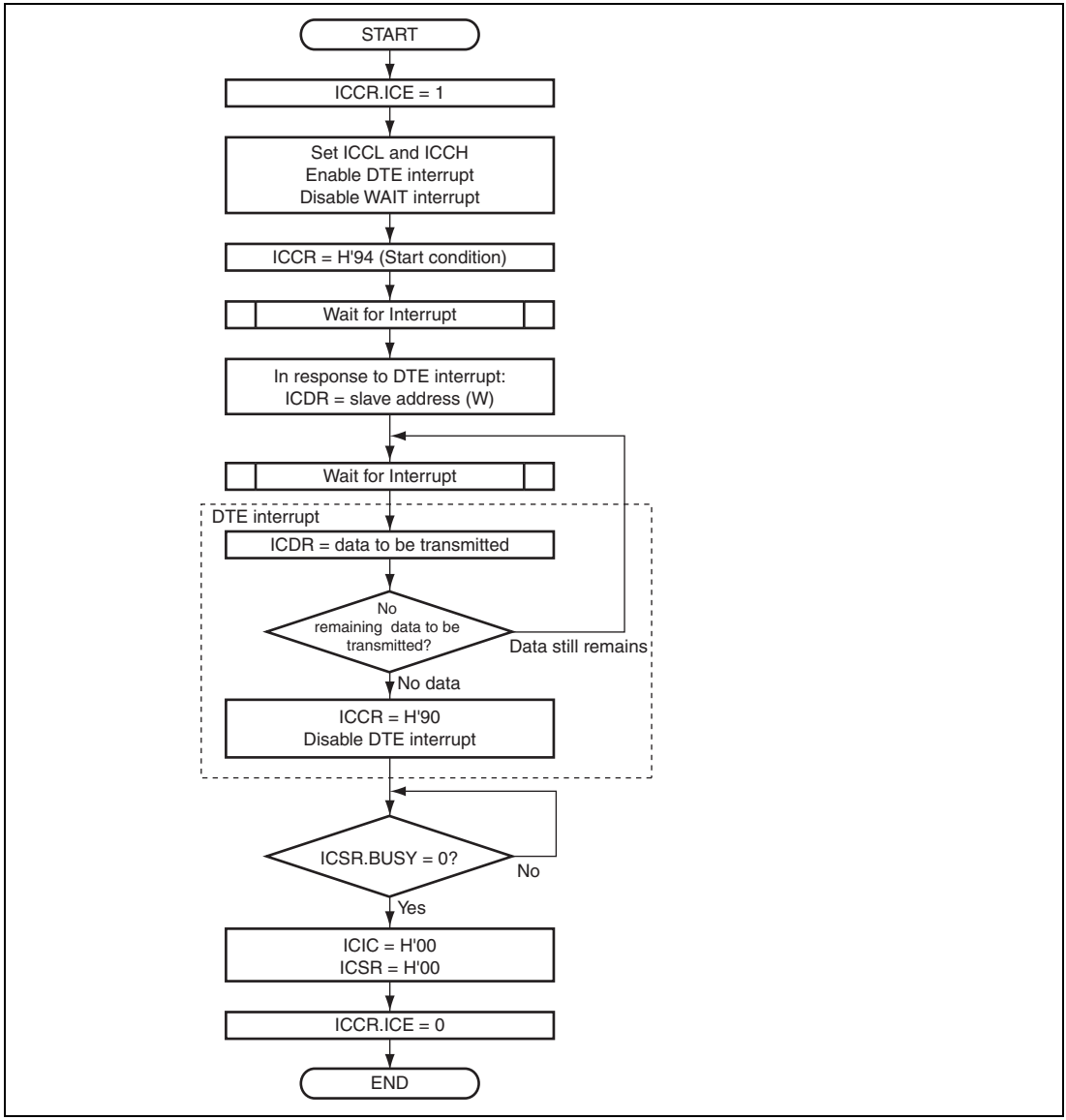


Figure 32.19 Example Flow of Software for Transmission (WAIT = 0)

32.5 Usage Notes

- Arbitration Lost

  A bus may not be released depending on the state of the slave device when arbitration lost is detected.

  When resuming the communication after an arbitration lost, the initialization by the ICE bit in ICCR is required to initialize the entire internal data.

- ICE Reset

  The SCL clock stops immediately after ICE is reset during data transmission. Therefore, a bus may not be released depending on the state of the slave device.

- Processing Concerning Stop Condition Generation

  When data is transmitted or received after a stop condition has been issued and generated, it may not be possible to generate a stop condition hereafter. Be sure to initialize all registers in the I²C module by clearing the ICE bit in ICCR to 0 after a stop condition has been generated or before making settings to trigger data transmission or reception after a stop condition has been generated.

- What to do when the I²C bus interface is not used

  Set the ICE bit in ICCR to 0 to disable the unused channels while the I²C bus interface is not used.

Section 33 Video Processing Unit (VPU)

This section is covered by a non-disclosure agreement. Please contact a Renesas Electronics sales representative for details.

33.1 Function Overview

33.1.1 Functions

The VPU (video processing unit) is a module that can:
(1) Decode bit streams that conform to VC-1 (SMPTE 412M-2006), and
(2) Encode and decode bit streams that conform to MPEG-4 (ISO/IEC 14496).

The VPU supports the following MPEG-4 video standards: Simple Profile, Advanced Simple Profile, and Short Header (hereafter collectively called MPEG-4); AVC Baseline Profile (hereafter called AVC); and VC-1 Simple Profile and Main Profile (hereafter collectively called VC-1).

The VPU has the following excellent features:

- Dynamic Timeslot Method (DTME)

  The length of slots used in pipelining can be changed dynamically according to the bus state. This makes it possible to maintain the optimal processing time for slots when the volume of bus traffic is large.
- VOP encoding for MPEG-4 and AVC

  The VPU encodes MPEG-4 images in memory in VOP units and AVC images in memory in slice units to generate bit streams. For MPEG-4, B-VOP encoding (bidirectional search) is supported. For AVC, multi-reference encoding (two-plane) is supported. In addition, Quarter-pel-precision searches in units of 8 × 8 blocks at minimum are available in AVC encoding.

  ARME (adaptive realtime motion estimation): The quality of motion estimation can be enhanced by expanding the search range and increasing the search count (common to MPEG-4 and AVC).

  POI (predict from original image): An intra prediction mode allowing searches in realtime is available (for AVC).

  ASP (active skip prediction): Controlling searches in such a way that the number of skipped macroblocks can be increased improves the image quality at a low bit rate (for AVC).

  CWQ (custom weighted quantization): Code amount control is available for each macroblock (MB) in a plane to be encoded. When encoding a portrait, for example, assigning more codes to the center of the image can enable more detailed expression.

- Video decoding for MPEG-4, AVC, and VC-1

  The VPU reads bit streams from memory and decodes them in VOP units for MPEG-4, in slice units for AVC, and in picture units for VC-1.

  Multiple concealment modes are supported so that, if an error occurs, the error area and error block boundary can be concealed.
- Deblocking filter

  A deblocking filtered image can be output additionally for both a decoded image and a local decode image when encoding.
- Video header search for MPEG-4

  A bit stream is read from memory to detect the next start code.

Section 34 Capture Engine Unit (CEU)

This LSI has two capture engine units (CEU), which is a capture module that fetches image data externally input and transfers it to the memory. The CEU is connected to the system bus via bus bridge modules.

34.1 Features

Lists the features of CEU as follows.

(1) Image data fetch

- Captures an image output from an external module and writes YCbCr data to the memory with it separated into Y data and CbCr data.
- Fetches image data other than YCbCr data, e.g. JPEG data, RGB565, from an externally connected module, such as a camera, and sequentially writes the image data to the memory.
- Fetches an interlace source image in both-field units or one-field units and writes it to the memory. In both-field capture, an image can be stored in the memory as a frame image.

(2) Filter processing

- Performs scale-down and removal of high-frequency components (only in the horizontal direction) for an image using internal filters. Note that the scaled-down image must not exceed VGA. The filter processing can be applied to only YCbCr input data.

(3) Display information acquisition

- Acquires the complexity level of the captured display and writes it to the memory. This information output at MPEG-4 encoding is useful for determining the scene change.

(4) Format conversion

- Converts image data input in the YCbCr 4:2:2 format into the YCbCr 4:2:0 format and writes it to the memory. Note that the conversion algorithm is simple skipping in which the chrominance component (CbCr) of the even-numbered lines is skipped.

34.2 Functional Overview of CEU

The functional overview of the CEU is shown in table 34.1, and the main functions and their details are shown in table 34.1.

Table 34.1 Functional Overview of CEU

Classification	Item	Function	Description	Note
Connectable camera	Size	5M pixels	2560 pixels × 1920 lines	Horizontal: 4-pixel units Vertical: 4-line units
		3M pixels	2048 pixels × 1536 lines	
		2M pixels	1632 pixels × 1224 lines	
		UXGA	1600 pixels × 1200 lines	
		SXGA (1)	1280 pixels × 1024 lines	
		SXGA (2)	1280 pixels × 960 lines	
		XGA	1024 pixels × 768 lines	
		SVGA	800 pixels × 600 lines	
		VGA	640 pixels × 480 lines	
		CIF	352 pixels × 288 lines	
		QVGA	320 pixels × 240 lines	
		QCIF	176 pixels × 144 lines	
		QQVGA	160 pixels × 120 lines	
		Sub-QCIF	128 pixels × 96 lines	
	Input format	YCbCr 4:2:2 8 bits	Cb_0, Y_0, Cr_0, $Y_{1...}$	Supports clock ratio of 1:1
			Cr_0, Y_0, Cb_0, $Y_{1...}$	
			Y_0, Cb_0, Y_1, $Cr_{0...}$	
			Y_0, Cr_0, Y_1, $Cb_{0...}$	
		YCbCr 4:2:2 16 bits	{Y0, Cb0}, {Y1, Cr0}, …	
			{Y0, Cr0}, {Y1, Cb0}, …	

Classification	Item	Function	Description	Note
Connectable camera	Input format	Binary data	Specified amount to be fetched on edges of the sync signal	Written sequentially
			Data is fetched with the horizontal sync signal as an enable signal.	
	Horizontal and vertical sync signal polarities	Arbitrary	High-active and low-active	
	Capture start location	Arbitrary	Can be specified in camera input clock units	Horizontal: 1-cycle units Vertical: 1-HD (horizontal sync signal) units
	Number of captured pixels	Arbitrary	Can be specified in 4-pixel units horizontally and in 4-line units vertically	
	Interlace	Both-field capture	Stored as a field image	Capture: 2-VD (vertical sync signal) units
			Stored as a frame image	
		One-field capture	Top field or bottom field can be specified	Capture: 1-VD units
Memory write	Output format	YCbCr 4:2:2 YCbCr 4:2:0	YCbCr 4:2:0 is realized by simple skipping	
Filter function	No scaling or scale-down	Scale-down of captured display	Desired scaling factor from 1/16 to 1 (scaled-down display must not exceed VGA)	
	Low-pass filter		Removal of high-frequency components	Only in the horizontal direction
Display information acquisition	Complexity level	Acquisition of complexity level of captured display	Variation of pixel values is indicated	Used for MPEG-4 16-line units, 8-line units, or 1-display units can be selected

Table 34.2 Main Functions of CEU and Their Details

Main Function	Detailed Description
Image data fetch	• Captures an image output from an external module and writes YCbCr data to the memory with it separated into Y data and CbCr data. • Fetches image data other than YCbCr data, e.g. JPEG data, from an externally connected module, such as a camera, and sequentially writes the image data to the memory. • Fetches an interlace source image in both-field units or one-field units and writes it to the memory. In both-field capture, an image can be stored in the memory as a frame image.
Filter processing	Performs scale-down and removal of high-frequency components (only in the horizontal direction) for an image using internal filters. Note that the scaled-down image must not exceed VGA. The filter processing can be applied to only YCbCr input data.
Display information acquisition	Acquires the complexity level of the captured display and writes it to the memory. This information output at MPEG-4 encoding is useful for determining the scene change.
Format conversion	Converts image data input in the YCbCr 4:2:2 format into the YCbCr 4:2:0 format and writes it to the memory. Note that the conversion algorithm is simple skipping in which the chrominance component (CbCr) of the even-numbered lines is skipped.

Figure 34.1 shows a block diagram of the CEU.

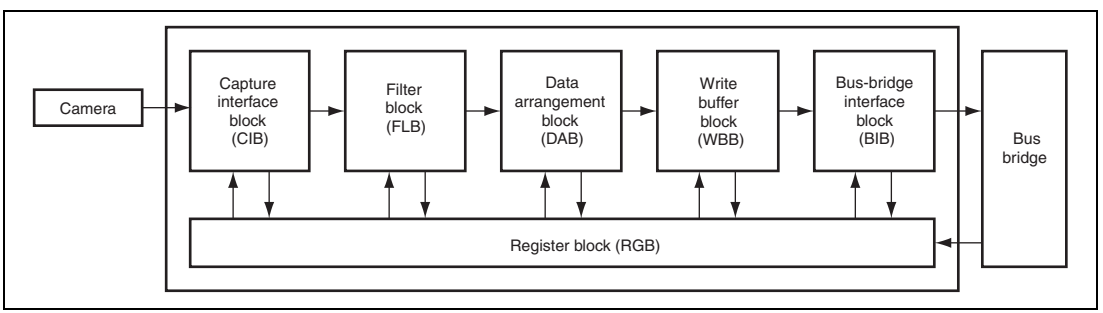


Figure 34.1 Block Diagram of CEU

34.3 Pin Configuration of CEU

The pin configuration of the CEU is shown in table 34.3.

Table 34.3 Pin Configuration of CEU

Pin Name	Function	I/O	Description
VIO0_D7 to VIO0_D0	CEU0 data bus	Input	Camera image data input to the CEU0
VIO1_D7 to VIO1_D0/ VIO0_D16 to VIO0_D8*	CEU1 data bus/ CEU0 upper data bus	Input	Camera image data input to the CEU1 or CEU0 upper byte
VIO0_CLK/VIO1_CLK	CEU clock	Input	Camera clock input to the CEU
VIO0_VD/VIO1_VD	CEU vertical sync	Input	Camera vertical sync signal input to the CEU
VIO0_HD/VIO1_HD	CEU horizontal sync	Input	Camera horizontal sync signal input to the CEU
VIO0_FLD/VIO1_FLD	Field signal	Input	Field identification signal to the CEU
VIO_CKO	Camera clock output	Output	Clock output to the camera

Note: * For VIO0_D15 to VIO0_D8/VIO1_D7 to VIO1_D0, either of the functions can be used. Only CEU0 support the 16-bit interface. In the 16-bit interface, VIO0_D15 to VIO0_D8/VIO1_D7 to VIO1_D0 are not switched since the data path is 16-bit. For the switching method, see section 48, Pin Function Controller (PFC). When the distinction according to the bus width for the data bus is not needed, VIO_D is used in this manual. Otherwise, VIO_CLK, VIO_VD, and VIO_HD are used.

34.4 Register Descriptions of CEU

The register configuration of the CEU is shown in table 34.4. The register states in each processing mode are shown in table 34.5.

Most CEU registers have a 2-plane configuration (plane A and plane B). The CEU switches the planes when using these 2-plane registers. A mirror address, which is an address that can always access the register on the unused plane, is provided for each 2-plane register. Figure 34.2 shows the timing to switch the register planes. The CEU switches the register planes at the same time a VD interrupt is asserted.

In the following register descriptions, "during operation" indicates the period that begins when the CEU is activated by the CE bit in the capture start register (CAPSR) and ends when a capture end interrupt (CPE) of the capture event flag clear register (CETCR) occurs. In the read-only bits in each register, the write value should always be 0. If a value other than 0 is written to any of these bits, correct operation cannot be guaranteed.

Table 34.4 Register Configuration of CEU

			Addresses			
Register Name	**Abbr.**	**R/W**	**Address (Plane A)**	**Address (Plane B)**	**Mirror Address**	**Access Size**
CEU0 Capture start register	CAPSR_0	R/W	H'FE91 0000	—	—	32
CEU0 Capture control register	CAPCR_0	R/W	H'FE91 0004	—	—	32
CEU0 Capture interface control register*	CAMCR_0	R/W	H'FE91 0008	—	—	32
CEU0 Capture interface cycle register*	CMCYR_0	R/W	H'FE91 000C	—	—	32
CEU0 Capture interface offset register	CAMOR_0	R/W	H'FE91 0010	H'FE91 1010	H'FE91 2010	32
CEU0 Capture interface width register	CAPWR_0	R/W	H'FE91 0014	H'FE91 1014	H'FE91 2014	32
CEU0 Capture interface input format register	CAIFR_0	R/W	H'FE91 0018	—	—	32
CEU0 register control register	CRCNTR_0	R/W	H'FE91 0028	—	—	32
CEU0 register forcible control register	CRCMPR_0	R/W	H'FE91 002C	—	—	32
CEU0 Capture filter control register	CFLCR_0	R/W	H'FE91 0030	H'FE91 1030	H'FE91 2030	32
CEU0 Capture filter size clip register	CFSZR_0	R/W	H'FE91 0034	H'FE91 1034	H'FE91 2034	32
CEU0 Capture destination width register	CDWDR_0	R/W	H'FE91 0038	H'FE91 1038	H'FE91 2038	32
CEU0 Capture data address Y register	CDAYR_0	R/W	H'FE91 003C	H'FE91 103C	H'FE91 203C	32
CEU0 Capture data address C register	CDACR_0	R/W	H'FE91 0040	H'FE91 1040	H'FE91 2040	32
CEU0 Capture data bottom-field address Y register	CDBYR_0	R/W	H'FE91 0044	H'FE91 1044	H'FE91 2044	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
CEU0 Capture data bottom-field address C register	CDBCR_0	R/W	H'FE91 0048	H'FE91 1048	H'FE91 2048	32
CEU0 Capture bundle destination size register	CBDSR_0	R/W	H'FE91 004C	H'FE91 104C	H'FE91 204C	32
CEU0 Firewall operation control register	CFWCR_0	R/W	H'FE91 005C	—	—	32
CEU0 Capture low-pass filter control register	CLFCR_0	R/W	H'FE91 0060	H'FE91 1060	H'FE91 2060	32
CEU0 Capture data output control register	CDOCR_0	R/W	H'FE91 0064	H'FE91 1064	H'FE91 2064	32
CEU0 Capture data complexity level register	CDDCR_0	R/W	H'FE91 0068	H'FE91 1068	H'FE91 2068	32
CEU0 Capture data complexity level address register	CDDAR_0	R/W	H'FE91 006C	H'FE91 106C	H'FE91 206C	32
CEU0 Capture event interrupt enable register	CEIER_0	R/W	H'FE91 0070	—	—	32
CEU0 Capture event flag clear register	CETCR_0	R/W	H'FE91 0074	—	—	32
CEU0 Capture status register	CSTSR_0	R	H'FE91 007C	—	—	32
CEU0 Capture software reset register	CSRTR_0	R/W	H'FE91 0080	—	—	32
CEU0 Capture data size register	CDSSR_0	R/W	H'FE91 0084	—	—	32
CEU0 Capture data address Y register 2	CDAYR2_0	R/W	H'FE91 0090	H'FE91 1090	H'FE91 2090	32
CEU0 Capture data address C register 2	CDACR2_0	R/W	H'FE91 0094	H'FE91 1094	H'FE91 2094	32
CEU0 Capture data bottom-field address Y register 2	CDBYR2_0	R/W	H'FE91 0098	H'FE91 1098	H'FE91 2098	32
CEU0 Capture data bottom-field address C register 2	CDBCR2_0	R/W	H'FE91 009C	H'FE91 109C	H'FE91 209C	32
CEU1 Capture start register	CAPSR_1	R/W	H'FE91 4000	—	—	32
CEU1 Capture control register	CAPCR_1	R/W	H'FE91 4004	—	—	32
CEU1 Capture interface control register*	CAMCR_1	R/W	H'FE91 4008	—	—	32
CEU1 Capture interface cycle register*	CMCYR_1	R/W	H'FE91 400C	—	—	32
CEU1 Capture interface offset register	CAMOR_1	R/W	H'FE91 4010	H'FE91 5010	H'FE91 6010	32
CEU1 Capture interface width register	CAPWR_1	R/W	H'FE91 4014	H'FE91 5014	H'FE91 6014	32
CEU1 Capture interface input format register	CAIFR_1	R/W	H'FE91 4018	—	—	32
CEU1 register control register	CRCNTR_1	R/W	H'FE91 4028	—	—	32
CEU1 register forcible control register	CRCMPR_1	R/W	H'FE91 402C	—	—	32
CEU1 Capture filter control register	CFLCR_1	R/W	H'FE91 4030	H'FE91 5030	H'FE91 6030	32
CEU1 Capture filter size clip register	CFSZR_1	R/W	H'FE91 4034	H'FE91 5034	H'FE91 6034	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
CEU1 Capture destination width register	CDWDR_1	R/W	H'FE91 4038	H'FE91 5038	H'FE91 6038	32
CEU1 Capture data address Y register	CDAYR_1	R/W	H'FE91 403C	H'FE91 503C	H'FE91 603C	32
CEU1 Capture data address C register	CDACR_1	R/W	H'FE91 4040	H'FE91 5040	H'FE91 6040	32
CEU1 Capture data bottom-field address Y register	CDBYR_1	R/W	H'FE91 4044	H'FE91 5044	H'FE91 6044	32
CEU1 Capture data bottom-field address C register	CDBCR_1	R/W	H'FE91 4048	H'FE91 5048	H'FE91 6048	32
CEU1 Capture bundle destination size register	CBDSR_1	R/W	H'FE91 404C	H'FE91 504C	H'FE91 604C	32
CEU1 Firewall operation control register	CFWCR_1	R/W	H'FE91 405C	—	—	32
CEU1 Capture low-pass filter control register	CLFCR_1	R/W	H'FE91 4060	H'FE91 5060	H'FE91 6060	32
CEU1 Capture data output control register	CDOCR_1	R/W	H'FE91 4064	H'FE91 5064	H'FE91 6064	32
CEU1 Capture data complexity level register	CDDCR_1	R/W	H'FE91 4068	H'FE91 5068	H'FE91 6068	32
CEU1 Capture data complexity level address register	CDDAR_1	R/W	H'FE91 406C	H'FE91 506C	H'FE91 606C	32
CEU1 Capture event interrupt enable register	CEIER_1	R/W	H'FE91 4070	—	—	32
CEU1 Capture event flag clear register	CETCR_1	R/W	H'FE91 4074	—	—	32
CEU1 Capture status register	CSTSR_1	R	H'FE91 407C	—	—	32
CEU1 Capture software reset register	CSRTR_1	R/W	H'FE91 4080	—	—	32
CEU1 Capture data size register	CDSSR_1	R/W	H'FE91 4084	—	—	32
CEU1 Capture data address Y register 2	CDAYR2_1	R/W	H'FE91 4090	H'FE91 5090	H'FE91 6090	32
CEU1 Capture data address C register 2	CDACR2_1	R/W	H'FE91 4094	H'FE91 5094	H'FE91 6094	32
CEU1 Capture data bottom-field address Y register 2	CDBYR2_1	R/W	H'FE91 4098	H'FE91 5098	H'FE91 6098	32
CEU1 Capture data bottom-field address C register 2	CDBCR2_1	R/W	H'FE91 409C	H'FE91 509C	H'FE91 609C	32

Note: * After changing the setting of a register (CAMCR or CMCYR) that is determined by the external module characteristics, do not start capture until at least 10 external input clock cycles have elapsed.

Table 34.5 CEU Register States in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
CAPSR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CAPCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CAMCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CMCYR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CAMOR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CAPWR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CAIFR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CRCNTR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CRCMPR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CFLCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CFSZR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDWDR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDACR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDBYR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDBCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CBDSR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CFWCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CLFCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDOCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDDCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDDAR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CEIER_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CETCR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CSTSR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CSRTR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDSSR_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDAYR2_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDACR2_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDBYR2_n	Initialized	Initialized	Retained	Retained	Initialized	Retained
CDBCR2_n	Initialized	Initialized	Retained	Retained	Initialized	Retained

Note: n = 0, 1

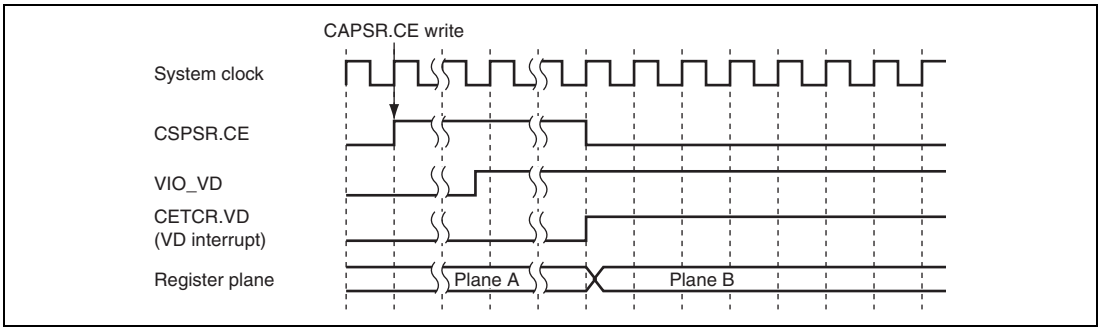


Figure 34.2 Register Plane Switching Timing (VD Polarity is High-Active)

34.4.1 Capture Start Register (CAPSR)

CAPSR captures data input to the CEU from an external module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CPKIL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CPKIL	0	R/W	Write 1 to this bit to perform a software reset of capturing. At a software reset, capturing ends immediately without completing capture operation until the end of a frame. Clear the CE bit to 0 when writing 1 to this bit. Processing of the capture software reset is indicated by this bit being set to 1. When this bit is 1, do not start capturing since reset processing is in progress. When restarting capture operations, after referring to the CPTON bit in CSTSR to ensure that the CEU is halted (in the idle state), wait until this bit is cleared to 0. The timing of restarting capture operations is shown in Figure 34.6. When a software reset is generated by this bit, a capture end interrupt (CPE bit in CETCR) may be output immediately after the software reset. However, such kind of interrupt should be ignored. Also, even if the capture end interrupt is not output, the interrupt source (CPE bit) must be cleared before capturing of the next frame. 0: Normal state 1: Software reset of capturing
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	CE	0	R/W	• Single-capture This bit reserves capturing of the next frame. When 1 is written to this bit, the capture of one frame starts from the next VD input, and stops when the one-frame capture end interrupt (CPE bit in CETCR) is asserted (Figure 34.7). To perform capture again, write 1 to this bit. After the VD or HD polarity is changed, do not write 1 to this bit until the next VD interrupt is asserted. As this bit indicates the capture reserve state, this bit is read as 1 after it is set to 1 and until VD is input. When VD is input, this bit returns to 0 and so is read as 0. The capture end is determined by the one-frame capture end interrupt (CPE bit). This is similar in data fetch mode. Registers should be set before the VD interrupt of the frame where capture starts next. The new register settings take effect at the next VD input. When registers are modified during capturing, the register settings take effect from the capture operations of the next VD input. If a setting register to which writing during capturing is prohibited is modified during capturing, an interrupt source (IGRW bit in CETCR) is generated. For details on the interrupt source, see the description on CETCR. • Continuous capture When this bit is set to 1 while the CTNCP bit in CAPCR is set to 1, continuous capture starts from the next frame (Figure 34.8). Note that this bit is not cleared to 0 but remains as 1. To stop capturing, clear this bit to 0; capturing stops after the current frame is completed. Continuous capture operations are possible in only image capture mode. The start address of the memory to which the captured data is written to must be set for each frame. 0: Stops capturing 1: Starts capturing

When both the VD (vertical sync signal) and HD (horizontal sync signal) polarities are high-active, one frame is defined as a period from a VD rising edge to the next VD rising edge, and one line as a period from an HD rising edge to the next HD rising edge. Figure 34.3 shows the timing of one frame (when both the VD and HD polarities are high-active).

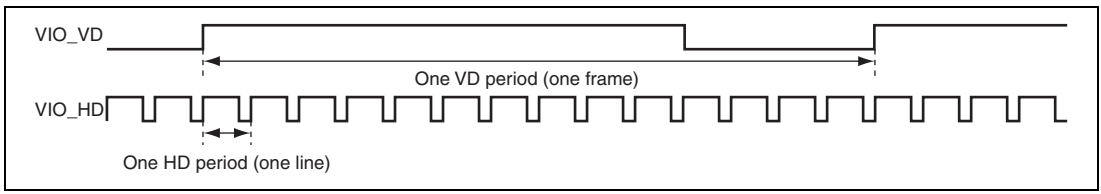


Figure 34.3 Frame Timing

When both the VD and HD polarities are high-active, similar to one frame, one field is defined as follows:

- Period from a VD rising edge to the next VD rising edge
- One line is a period from an HD rising edge to the next HD rising edge

The field identification signal FLD should be fixed for at least 1-HD period from a VD input. Figure 34.4 shows the timing of one field (when both the VD and HD polarities are high-active).

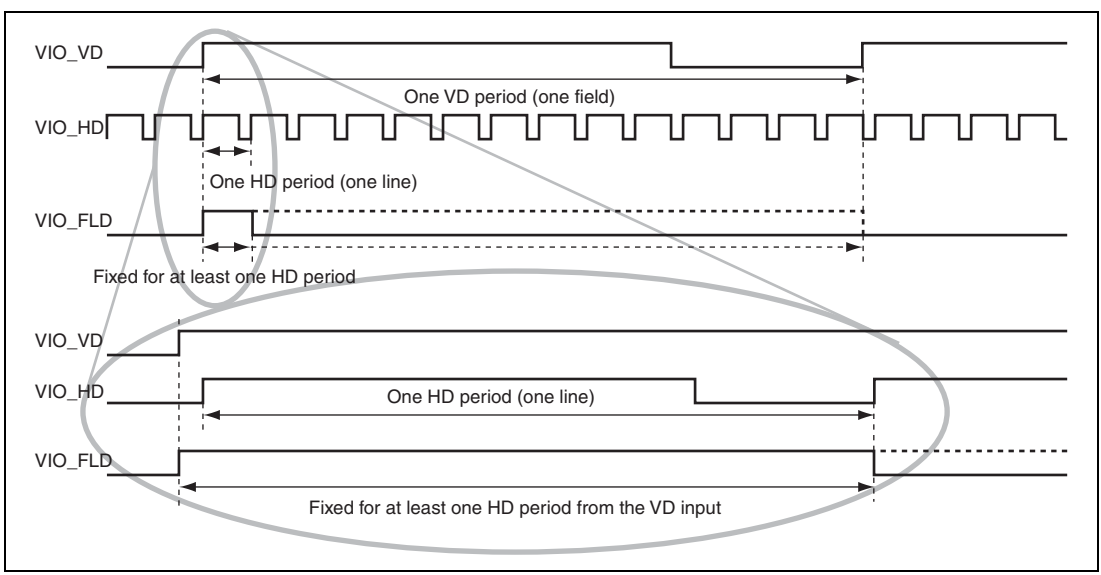


Figure 34.4 One Field Timing

In data enable fetch, one frame is defined as a period from a VD rising edge to the VD falling edge. With the HD as an enable signal (positive polarity), data of a cycle in which the HD is asserted is fetched while the VD is high. Figure 34.5 shows the timing of one frame for data enable fetch.

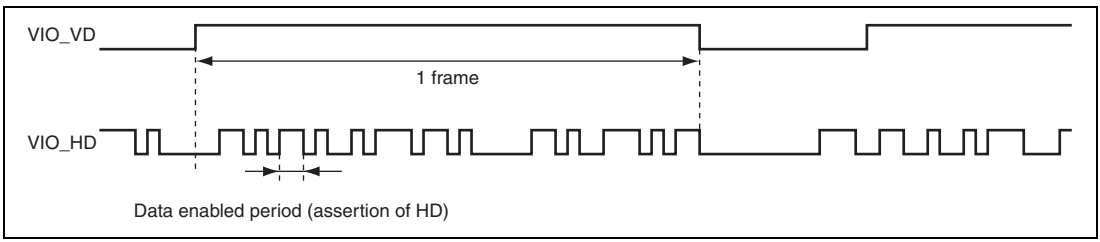


Figure 34.5 Frame Timing (Data Enable Fetch)

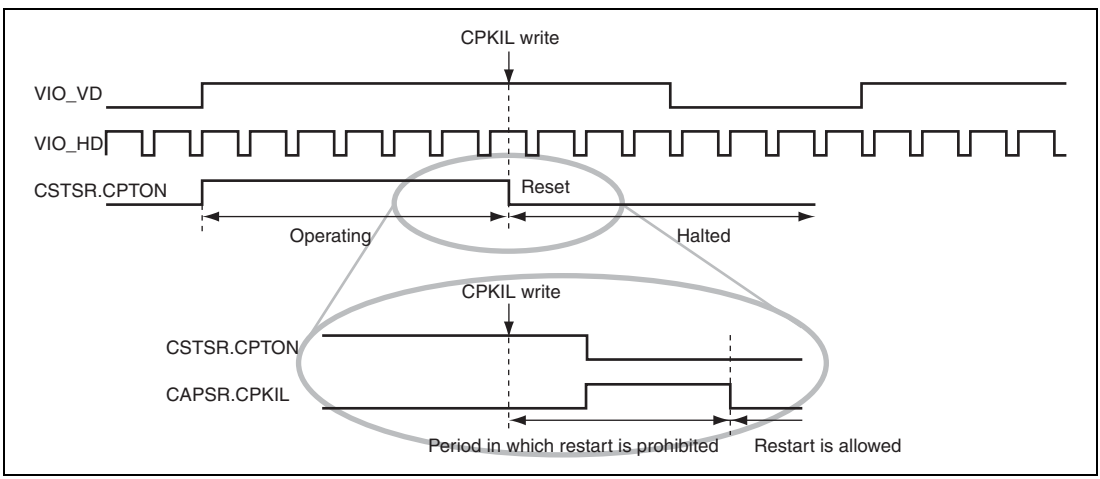


Figure 34.6 Timing of Software Reset and Restart of Capturing

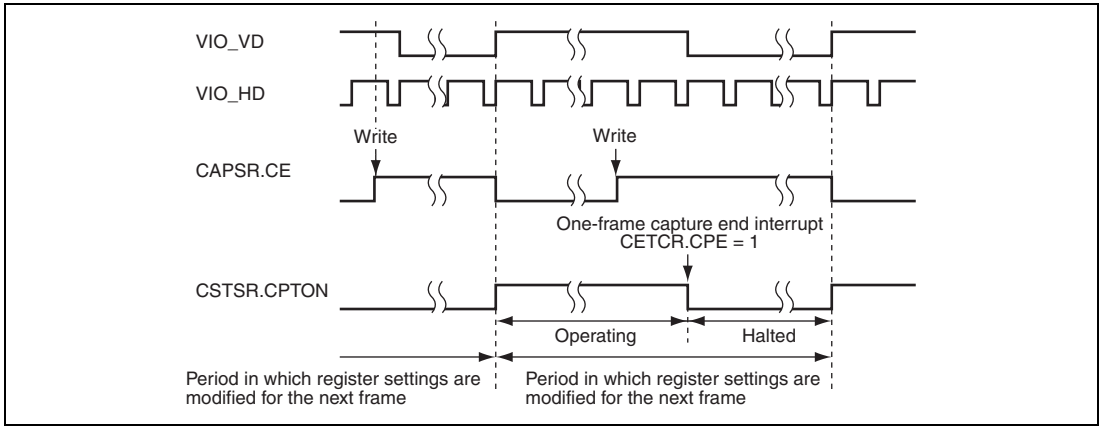


Figure 34.7 Timing of Modifying CE Bit and Register Setting in One Frame Capture

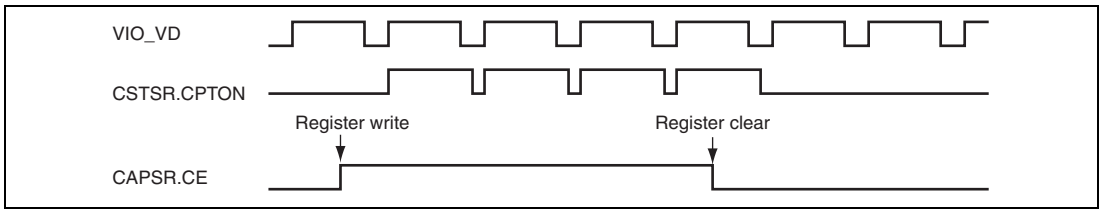


Figure 34.8 Continuous-Frame Capture

34.4.2 Capture Control Register (CAPCR)

CAPCR sets continuous-frame capture and the frame drop intervals.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FDRP[7:0]								—	—	MTCM[1:0]		—	—	—	CTNCP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	FDRP[7:0]	H'00	R/W	These bits set the frame drop interval in continuous-frame capture. When these bits are cleared to 0, frame drop is not performed, and all frames are captured. Figure 34.9 shows the value set in these bits and the timing of captured frames. The frame drop interval unit differs according to the capture setting. Table 34.6 shows the relationship between the capture setting and frame drop interval unit. The image of the frame drop timing for each capture setting when these bits are set to 2 is shown in Figure 34.10. In both-field capture, capturing is performed continuously for 2-VD periods, regardless of whether the second field is the top field or bottom field. In addition, in both-field capture, the frame drop counter is incremented when the first field has been identified as the top field or bottom field, regardless of whether the second field is the top field or bottom field. When 0 is written to the CE bit in CAPSR, capturing terminates after the current frame has been captured for a capture frame. However, for a drop frame, capturing is forcibly terminated in the CEU so no capture end interrupt (CPE bit in CETCR) is output. While CE bit is 1, do not change the setting of these bits. Note: Do not change the setting of these bits during continuous capture operations. To change the setting of these bits, stop continuous capture (CE bit = 0), clear the CTNCP bit (continuous capture) in CAPCR to 0, and then restart continuous capture. Continuous capture is performed during the period of CAPSR.CE = 1 shown in Figure 34.9.
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
21, 20	MTCM[1:0]	00	R/W	These bits specify the unit for transferring data to a bus bridge module. The access efficiency of SDRAM where the image data is to be stored can be improved by continuously accessing the addresses. To improve the write efficiency, set these bits to 11. The setting of these bits appear to be unchanged from the outside. 00: Transferred to the bus in 4-burst 1-transfer (4QW × 1) units 01: Transferred to the bus in 4-burst 2-transfer (4QW × 2) units 10: Transferred to the bus in 4-burst 4-transfer (4QW × 4) units 11: Transferred to the bus in 4-burst 8-transfer (4QW × 8) units (1) Image capture 00: Y data and C data are transferred in 4QW units 01: Y data and C data are transferred in 4QW × 2 units 10: Y data and C data are transferred in 4QW × 4 units 11: Y data and C data are transferred in 4QW × 8 units (2) Data fetch 00: Data is transferred in 4QW units 01: Data is transferred in 4QW × 2 units 10: Data is transferred in 4QW × 4 units 11: Data is transferred in 4QW × 8 units
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CTNCP	0	R/W	When capturing is started with this bit set to 1, capturing continues until the CE bit in CAPSR is cleared to 0 or a software reset is initiated by the CPKIL bit in CAPSR (see Figure 34.8). Continuous capture must be set before capturing is started. This bit is modified only after 0 is written to the CE bit to stop capturing. If this bit is modified during capturing, correct operation cannot be guaranteed. In data fetch mode, clear this bit to 0. 0: One-frame capture when the CE bit is 1 1: Continuous capture until the CE bit is cleared to 0

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

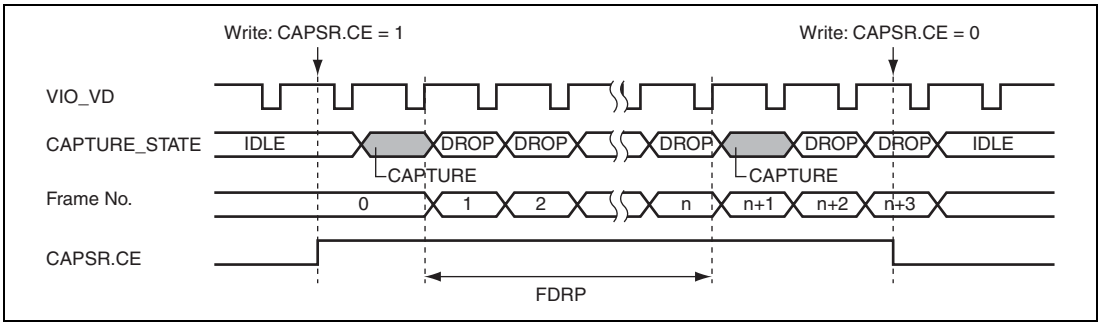


Figure 34.9 Setting of FDRP Bits and Frame Drop Timing

Table 34.6 Relationship between Capture Setting and Frame Drop Interval Unit

Input Mode	Captured Image	First Captured Image	Frame Drop Interval Unit	Capture Setting
Progressive	Frame	Frame immediately after capture start	Frame	A
Interlace	Both-field (2-VD capture)	Field immediately after capture start	2 fields (first capture field count)	B
				C
		Top field	2 fields (top-field count)	D
		Bottom field	2 fields (bottom-field count)	E
	One-field (1-VD capture)	Field immediately after capture start	First capture field	F
				G
		Top field	Top field	H
		Bottom field	Bottom field	I

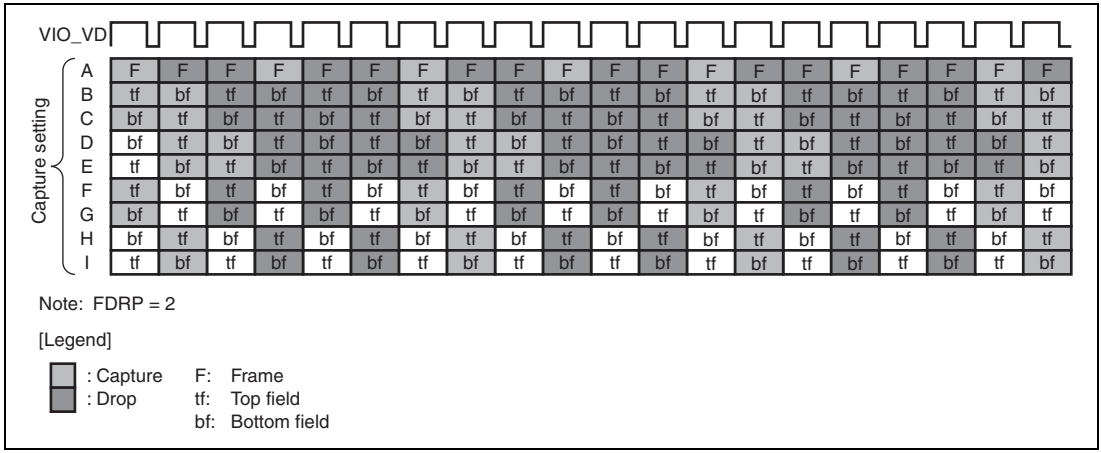


Figure 34.10 Image of Frame Drop Timing for Capture Settings (FDRP Bits = 2)

34.4.3 Capture Interface Control Register (CAMCR)

CAMCR sets the capture interface.

The following items are set by CAMCR.

- Selection between image capture operation or data fetch operation
- Polarities of the vertical and horizontal sync signals
- Input order of image data components (Y, Cb, and Cr) (only for image capture mode)
- Selection of digital image input pins (8 bits or 16 bits)
- Polarity of the field identification signal

CAMCR must be set according to the module connected. In data fetch mode, set the DTARY bits to B'0. Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Note: After changing the setting of this register, do not start capture until at least 10 external input clock cycles have elapsed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FLDPOL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DTIF	—	—	DTARY[1:0]		—	—	JPG[1:0]		—	—	VDPOL	HDPOL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	FLDPOL	0	R/W	Sets the polarity of the field identification signal (FLD) from an external module. 0: When the FLD signal is high-active, the field is detected as the top field and when low-active, the field is detected as the bottom field. 1: When the FLD signal is low-active, the field is detected as the top field and when high-active, the field is detected as the bottom field.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	DTIF	0	R/W	Sets the digital image input pins from which data is to be captured. 0: Data input to 8-bit digital image input pins is captured 1: Data input to 16-bit digital image input pins is captured
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	DTARY[1:0]	00	R/W	These bits set the input order of the luminance component and chrominance component. The order in which the luminance component (Y) and chrominance component (Cb and Cr) are input from an external module differs among modules. The CEU supports the input orders shown in Figure 34.12. Set the corresponding value in these bits. In data fetch mode, set these bits to 00. (1) 8-bit interface 00: Image input data is fetched in the order of Cb_0, Y_0, Cr_0, and Y_1 01: Image input data is fetched in the order of Cr_0, Y_0, Cb_0, and Y_1 10: Image input data is fetched in the order of Y_0, Cb_0, Y_1, and Cr_0 11: Image input data is fetched in the order of Y_0, Cr_0, Y_1, and Cb_0 (2) 16-bit interface 00: Image input data is fetched in the order of {Cb0, Y0} and {Cr0, Y1} 01: Image input data is fetched in the order of {Cr0, Y0} and {Cb0, Y1} 10: Image input data is fetched in the order of {Y0, Cb0} and {Y1, Cr0} 11: Image input data is fetched in the order of {Y0, Cr0} and {Y1, Cb0}
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	JPG[1:0]	00	R/W	These bits select the fetched data type. 00: Image capture mode (input data are separated into Y data and CbCr data for output to the memory) 01: Data synchronous fetch mode (specified size of input data are output to the specified memory addresses in order of input and in synchronization with the sync signal) 01: Data enable fetch mode (input data are fetched with HD as an enable signal and output to the specified addresses in memory in order of input)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	VDPOL	0	R/W	Sets the polarity for detection of the vertical sync signal input from an external module. Figure 34.15 shows the relationship between the VD and VD interrupt when high-active is selected. Since a VD interrupt may occur when this bit value is modified, the VD bit in CETCR must always be cleared to 0 when this bit value is changed. 0: Vertical sync signal (VD) from an external module is detected as high-active 1: Vertical sync signal (VD) from an external module is detected as low-active
0	HDPOL	0	R/W	Sets the polarity for detection of the horizontal sync signal input from an external module. Figure 34.16 shows the relationship between the HD and HD interrupt when high-active is selected. Since an HD interrupt may occur when this bit value is modified, the HD bit in CETCR must always be cleared to 0 when this bit value is changed. 0: Horizontal sync signal (HD) from an external module is detected as high-active 1: Horizontal sync signal (HD) from an external module is detected as low-active

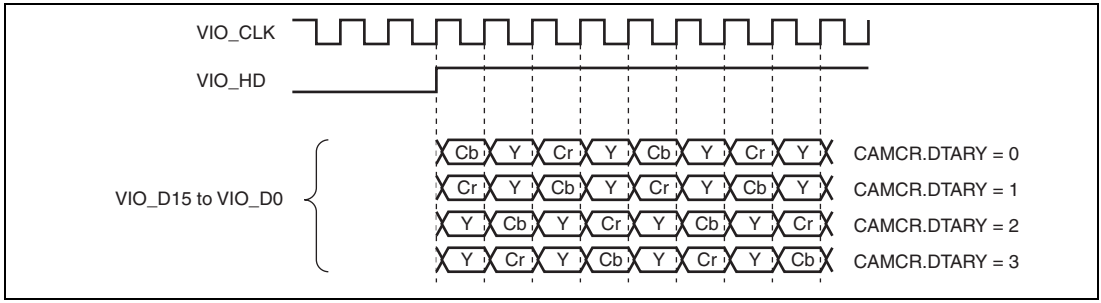


Figure 34.11 Input Order of Image Data

The JPG bit in CAMCR selects whether digital image data is fetched or data such as JPEG is fetched. In addition, when data such as JPEG is fetched, select whether the specified amount of data is continuously fetched in synchronization with the sync signal or data is fetched while the horizontal sync signal is enabled.

In data enable fetch mode, one frame is defined as a period from the rising edge to the falling edge of the vertical sync signal (VD) for data fetching. The horizontal sync signal (HD) is enabled only when the VD is high and treated as an enable signal. Data input in the cycle in which the HD is asserted (high) is fetched and output to the memory continuously.

Figures 34.12 and 34.13 show the interface timing in data enable fetch mode.

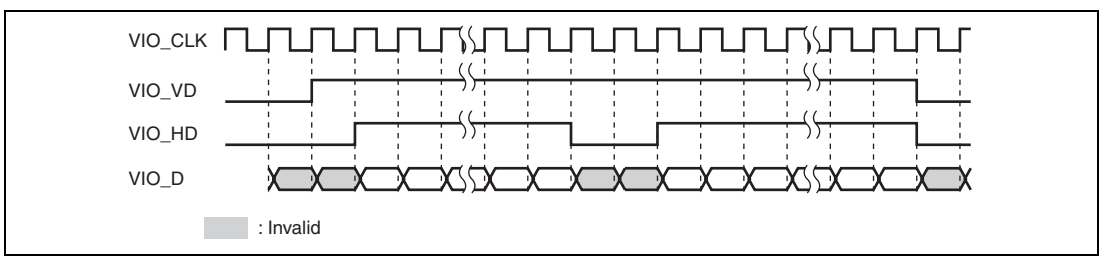


Figure 34.12 Data Enable Fetch Timing (HD Asserted (High) While VD is High)

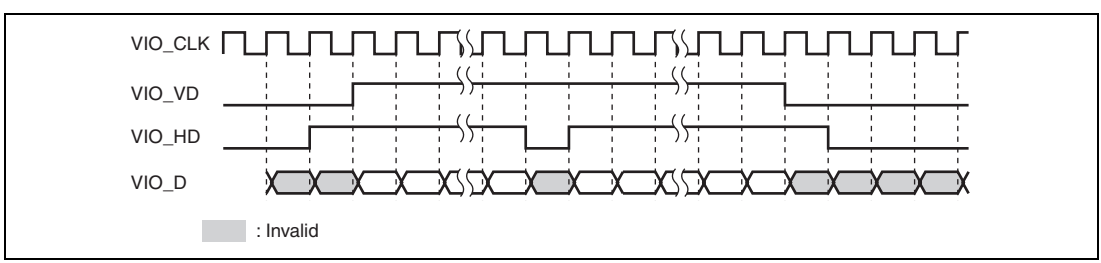


Figure 34.13 Data Enable Fetch Timing (HD Asserted (High) When VD is Not High)

This module starts fetching data at the rising edge of the VD and stops fetching data at the falling edge of the VD in data enable fetch mode. Thus, if the VD remains high and does not go low, end processing does not start. In addition, if the VD remains high and the HD also remains asserted, data continues to be fetched.

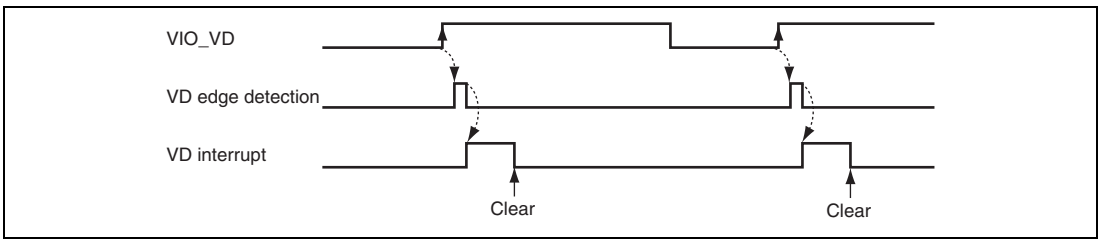


Figure 34.14 Relationship between VIO_VD and VD Interrupt when VD is High-Active

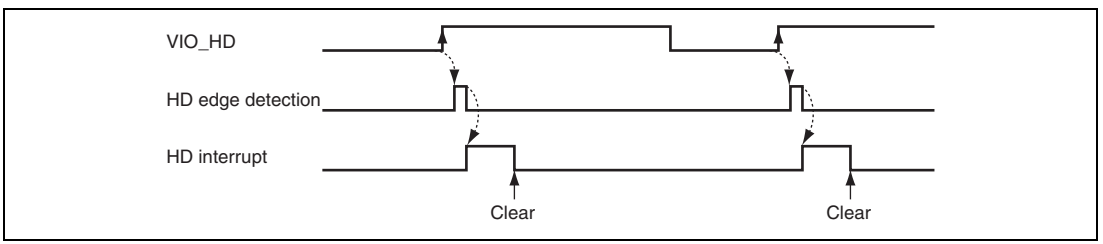


Figure 34.15 Relationship between VIO_HD and HD Interrupt when HD is High-Active

34.4.4 Capture Interface Cycle Register (CMCYR)

CMCYR is used to detect an illegal VD and an illegal HD. For HD, the number of cycles from a rising edge of HD to the next rising edge is set (falling edges when low-active is selected for HD). For VD, the number of HD inputs from a rising edge of VD to the next rising edge is set (falling edges when low-active is selected for VD).

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Set 0 in all bits of this register, during data enable fetch mode.

Note: After changing the setting of this register, do not start capture until at least 10 external input clock cycles have elapsed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VCYL[13:0]													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	HCYL[13:0]													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	VCYL[13:0]	H'0000	R/W	Vertical HD Count of External Module These bits set the number of VD cycles of an external module with the number of HD inputs. The interrupt source bit IGVS in CETCR is set to 1 when the actual number of VD cycles input from the external module differs from this setting. Set these bits for detecting an illegal VD. When these bits are all cleared to 0, the interrupt source bit IGVS in CETCR is not set to 1. Though the interrupt source bit IGVS in CETCR may be set to 1 after the VDPOL bit (VD polarity) in CAMCR is changed, this interrupt should be ignored.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	HCYL[13:0]	H'0000	R/W	Horizontal Cycle Count of External Module These bits set the number of HD cycles of an external module. The interrupt source bit IGHS in CETCR is set to 1 when the actual number of HD cycles input from the external module differs from this setting. Set these bits for detecting an illegal HD. When these bits are all cleared to 0, the interrupt source bit IGHS in CETCR is not set to 1. Though the interrupt source bit IGHS in CETCR may be set to 1 after the HDPOL bit (HD polarity) in CAMCR is changed, this interrupt should be ignored.

34.4.5 Capture Interface Offset Register (CAMOR)

CAMOR sets the location to start capturing when capturing images.

Since the number of HD (horizontal sync signal) inputs from a VD (vertical sync signal) input to the start of a valid image period, and the number of clock cycles from an HD input to the start of a valid image period differ among external modules, these must be set in CAMOR. By setting a value greater than the valid image area, part of the image can be clipped for capture. When fetching data, the setting of this register becomes the number of cycles (HD count) up to the start of a valid data period.

This register is not used, during data enable fetch mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VOFST[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HOFST[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	VOFST[11:0]	H'000	R/W	These bits specify the capture start location in terms of the HD count from a vertical sync signal (1-HD units). The blanking period from a vertical sync signal differs among external modules. Therefore, the vertical capture start location must be specified by these bits in terms of the HD count from a vertical sync signal so that an image can be captured from the valid image area (see Figure 34.16). Some external modules output a vertical sync signal as a data enable signal. In this case, there is no blanking period so these bits must be cleared to 0 (see Figure 34.17).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	HOFST[12:0]	H'0000	R/W	These bits specify the capture start location in terms of the number of clock cycles from a horizontal sync signal (1-cycle units). The blanking period from a horizontal sync signal differs among external modules. Therefore, the horizontal capture start location must be specified by these bits in terms of external input clock cycles from a horizontal sync signal so that an image can be captured from the valid image area. This is similar in data synchronous fetch mode (see Figure 34.18). Some external modules output a horizontal sync signal as a data enable signal. In this case, there is no blanking period so these bits must be cleared to 0 (see Figure 34.19). Note: The first HD (horizontal sync signal) being input simultaneously or after the first VD (vertical sync signal) is the operating condition of the CEU. These inputs are affected by the polarities (set by the VDPOL and HDPOL bits in CAMCR).

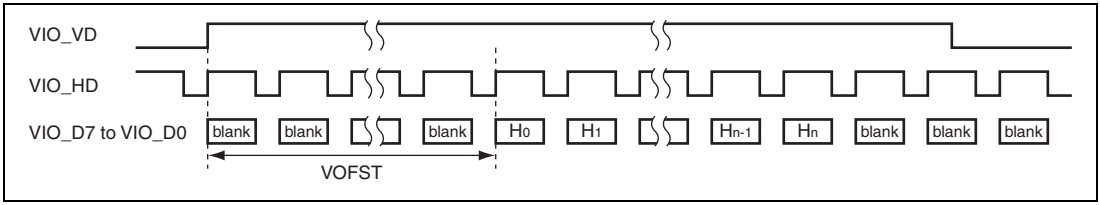


Figure 34.16 Vertical Offset

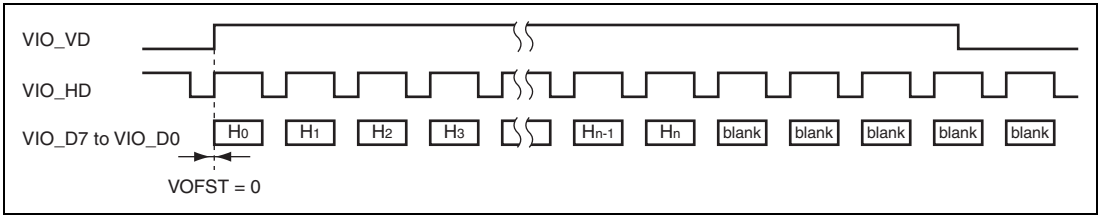


Figure 34.17 Timing when VD is Data Enable Signal

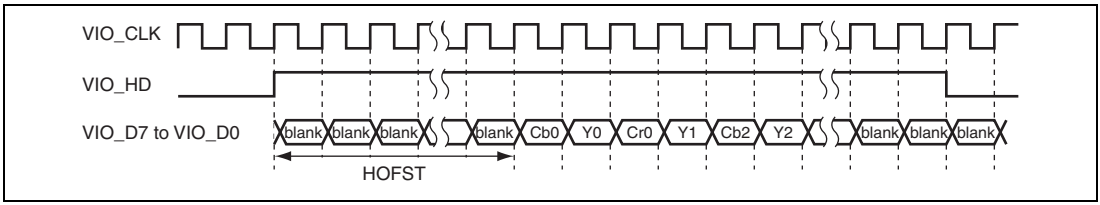


Figure 34.18 Horizontal Offset

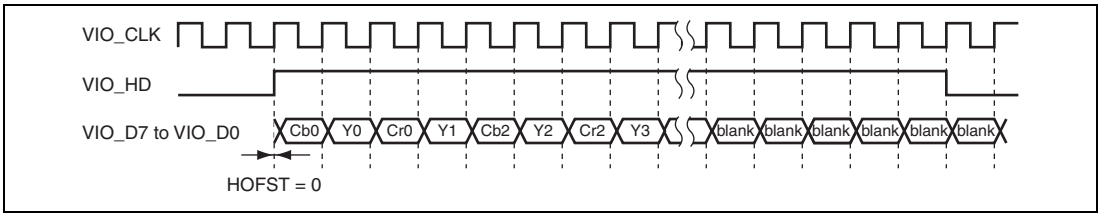


Figure 34.19 Horizontal Capture Timing (Image Capture with 8-bit Interface)

34.4.6 Capture Interface Width Register (CAPWR)

CAPWR sets the fetch (capture) cycle width when capturing images.

The cycle width unit differs according to the interface and the data type to be captured. For each setting unit, see table 34.7.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VWDTH[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HWDTH[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 18	VWDTH[11:2]	H'000	R/W	These bits specify the vertical capture period (4-HD units).
17, 16	VWDTH[1:0]		R	These bits specify the number of lines (HD count) to be captured from the location specified by the VOFST bits in CAMOR. Figure 34.20 shows the timing when the vertical blanking period is 0. The CEU captures only the number of lines (HD count) specified by these bits in the vertical direction. Make a setting in the same way to obtain data synchronization. The maximum value to be set is 1920 HD (5M pixels).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
12 to 1	HWDTH[12:1]	H'0000	R/W	These bits specify the horizontal capture period.
0	HWDTH[0]		R	These bits specify the number of cycles to be captured from the location specified by the HOFST bits. Figure 34.21 shows the timing when the horizontal blanking period is 0. The CEU captures for only the number of cycles specified by these bits in the horizontal direction. Make a similar setting for data synchronous fetch. The maximum value to be set is as follows: • 8-bit interface Image capture (8-cycle units): 5120 cycles (2560 pixels) Data synchronous fetch (4-cycle units): 2560 cycles (2560 bytes) • 16-bit interface Image capture (4-cycle units): 2560 cycles (2560 pixels) Data synchronous fetch (2-cycle units): 1280 cycles (2560 bytes) Note: In data synchronous fetch mode, set CFSZR and CDWDR according to the values set in this register. For details, see the descriptions on CFSZR and CDWDR.

Table 34.7 Unit for Setting Fetch (Capture) Cycle Width

	Vertical Direction		Horizontal Direction	
Interface	Image Capture	Data Synchronous Fetch	Image Capture	Data Synchronous Fetch
8-bit interface	4 HD	4 HD	8 cycles	4 cycles
16-bit interface	4 HD	4 HD	4 cycles	2 cycles

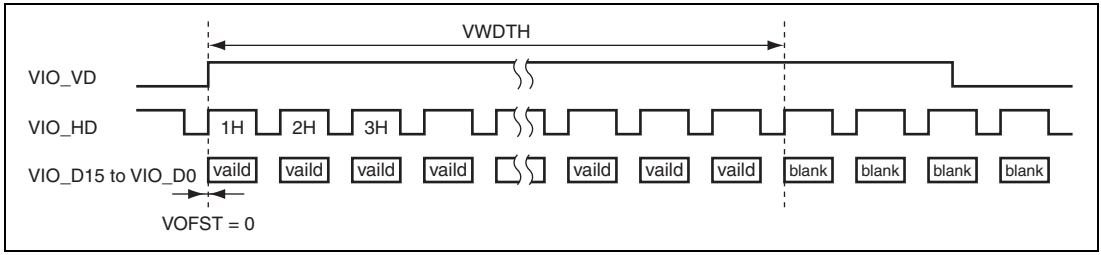


Figure 34.20 Vertical Capture Timing

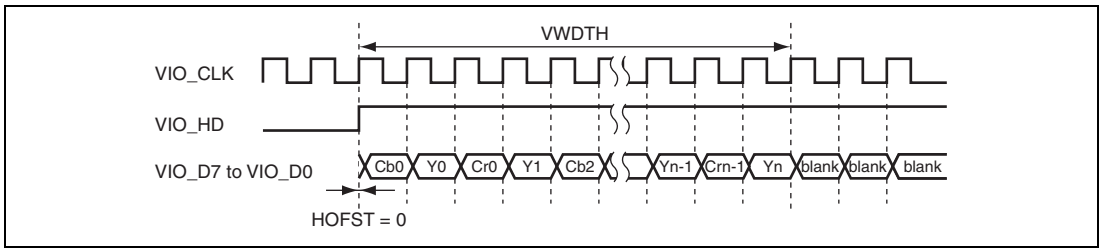


Figure 34.21 Timing when HD is Data Enable Signal (8-bit interface)

34.4.7 Capture Interface Input Format Register (CAIFR)

CAIFR sets the input mode (progressive or interlace) for capturing images, the images to be captured (frame, both-field, or one-field), and the image from which capturing starts (top field, bottom field, etc.). CAIFR is not used in data fetch mode.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

The items set by CAIFR are listed in table 34.8.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IFS	—	—	—	CIM	—	—	FCI[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	IFS	0	R/W	Sets the input mode for capturing images. 0: Progressive 1: Interlace
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CIM	0	R/W	Sets the images to be captured. Clear this bit to 0 when the input mode for image capture is progressive (frame image) or when the input mode for image capture is interlace for continuous capture of both the top and bottom fields. Set this bit to 1 when the input mode for image capture is interlace for capture of only a one-field image. 0: Capture of frame image (1 VD) or both-field image (2 VD) 1: Capture of one-field image (1 VD)

Bit	Bit Name	Initial Value	R/W	Description
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	FCI[1:0]	00	R/W	These bits set the timing to start capturing. The timing to start capturing is set by specifying the image to be captured first. Set these bits to 00 when the input mode is progressive. 00: Capture starts from the VD input immediately after the CEU activation regardless of it being a top or bottom field 01: After the CEU activation, input of a top-field image is waited, and then capture starts from the top field 10: After the CEU activation, input of a bottom-field image is waited, and then capture starts from the bottom field 11: Setting prohibited

Table 34.8 CAIFR Setting Items

Input Mode	IFS Bit	Captured Image	CIM Bit	Image to Start Capture	FCI Bits
Progressive	0	Frame	0	Frame immediately after activation	00
Interlace	1	Both-field (2-VD capture)	0	Field immediately after activation	00
				Top field	01
				Bottom field	10
				Setting prohibited	11
		One-field (1-VD capture)	1	Field immediately after activation	00
				Top field	01
				Bottom field	10
				Setting prohibited	11

In frame image capture and one-field image capture, a one-frame capture end interrupt occurs when capture for 1 VD finishes. In both-field image capture, a one-field capture end interrupt occurs when capture for 1 VD finishes and a one-frame capture end interrupt occurs when capture for 2 VD finishes. At this time, a one-field capture end interrupt occurs simultaneously with a one-frame capture end interrupt. Figure 34.23 shows the timing of a one-frame capture end interrupt and one-field capture end interrupt in both-field image capture.

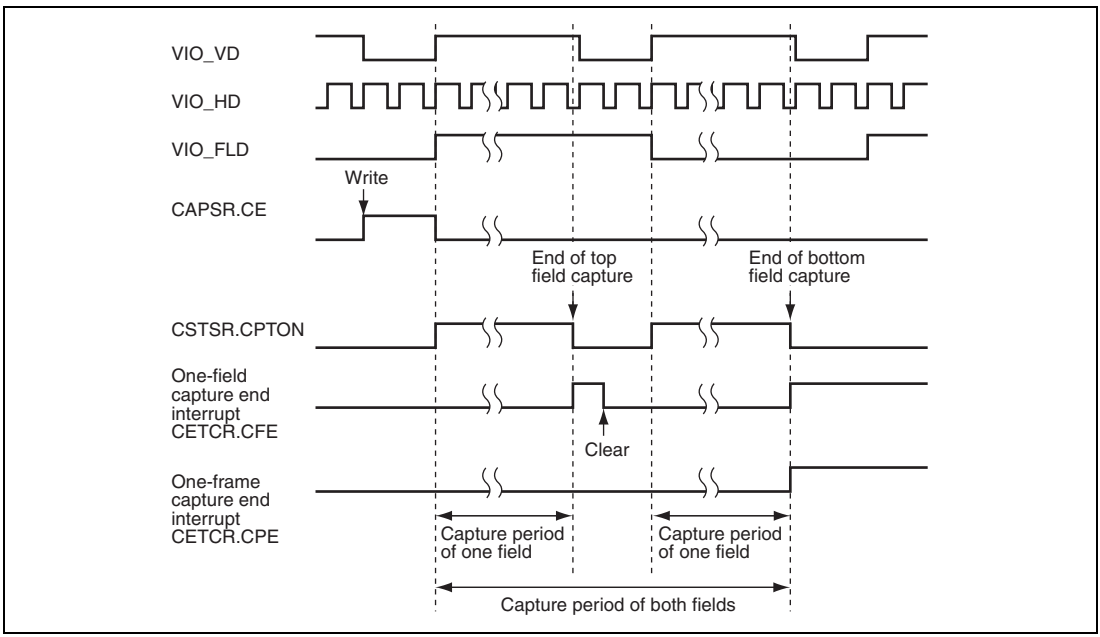


Figure 34.22 One-Frame Capture End Interrupt and One-Field Capture End Interrupt in Both-Field Image Capture

A captured frame image or captured one-field image is stored in the memory from the addresses set in CDAYR and CDACR (Figure 34.23). Captured both-field images are stored in different memory areas depending on whether it is a top-field or bottom-field image. A top-field image is stored in the memory from the addresses set in CDAYR and CDACR whereas a bottom-field image is stored in the memory from the addresses set in CDBYR and CDBCR (Figure 34.24).

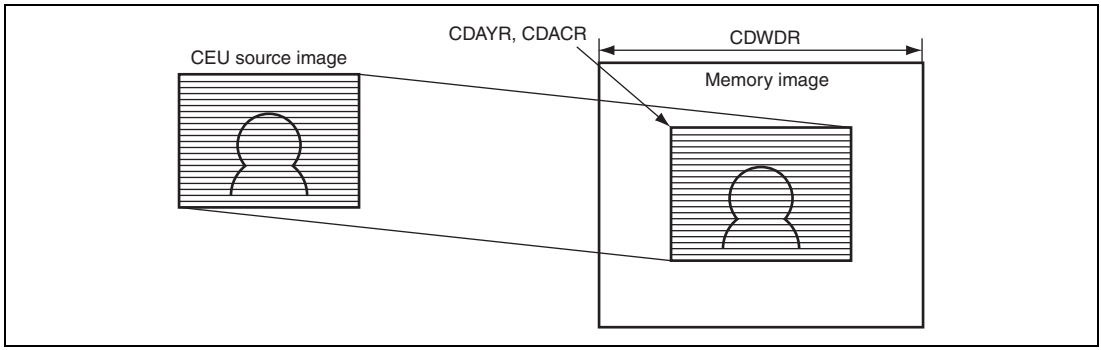


Figure 34.23 Image of Storing Captured Frame Image or Captured One-Field Image in Memory

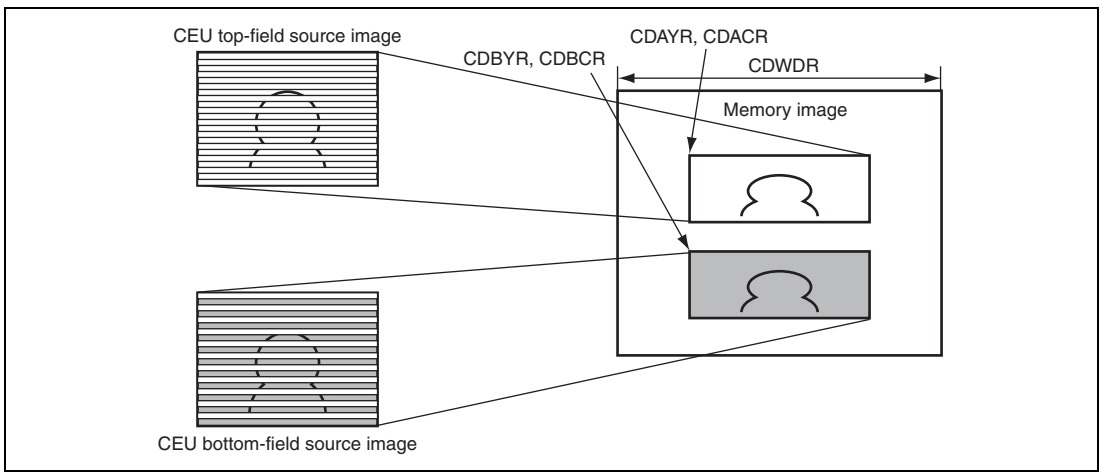


Figure 34.24 Image of Storing Captured Both-Field Images in Memory

If the FCI bits are set to B'00 for continuous capture in interlace input mode, images are continuously captured for 2 VD with the first captured field as the reference in both-field image capture (Figure 34.25). In one-field image capture, only the first captured field is continuously captured for 1 VD (Figure 34.26).

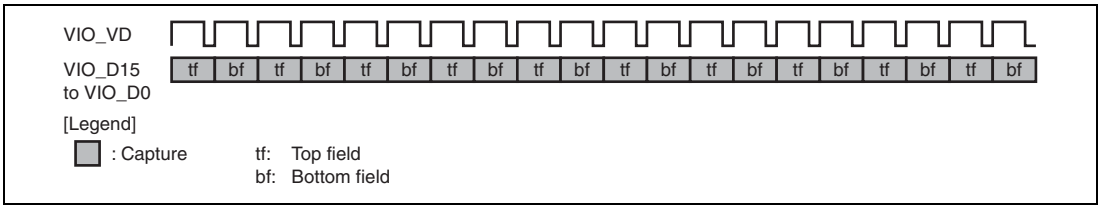


Figure 34.25 Continuous Both-Field Capture in Interlace Mode (Image Immediately after Activation is Top Field (FCI Bits = B'00))

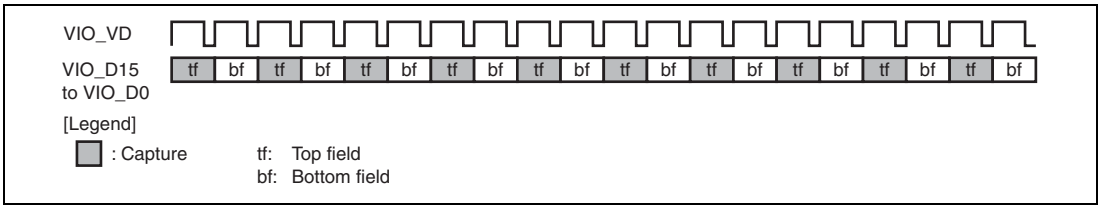


Figure 34.26 Continuous One-Field Capture in Interlace Mode (Image Immediately after Activation is Top Field (FCI Bits = B'00))

(1) Storage of Interlace Input as Frame Image

The CEU can store an interlace source image in the memory as a frame image. To store an interlace source image as a frame image, make the following register settings:

Input mode: Interlace (IFS bit = B'1)

Capture image: Both-field (CIM bit = B'0)

Image to start capture: Any setting other than the prohibited setting (FCI bits = as desired)

Figure 34.27 shows a memory image of capturing both fields of an interlace input and storing it as a frame image in the memory. Set the start addresses of the memory destination for the captured top-field image in CDAYR and CDACR, and the start addresses of the memory destination for the captured bottom-field image in CDBYR and CDBCR. When storing an interlace image as a frame image in the memory, set the horizontal image size of the memory area in CDWDR with the top-field image and bottom-field image placed next to each other as shown in Figure 34.27. In addition, set the number of captured lines of the field image in the VWDTH bits in CAPWR.

A memory image of folding the horizontal image size of the memory area in Figure 34.27 at CDWDR/2 is shown in Figure 34.28. Setting the registers to form the image in Figure 34.27 enables an interlace image to be stored as a frame image in the memory as shown in Figure 34.28.

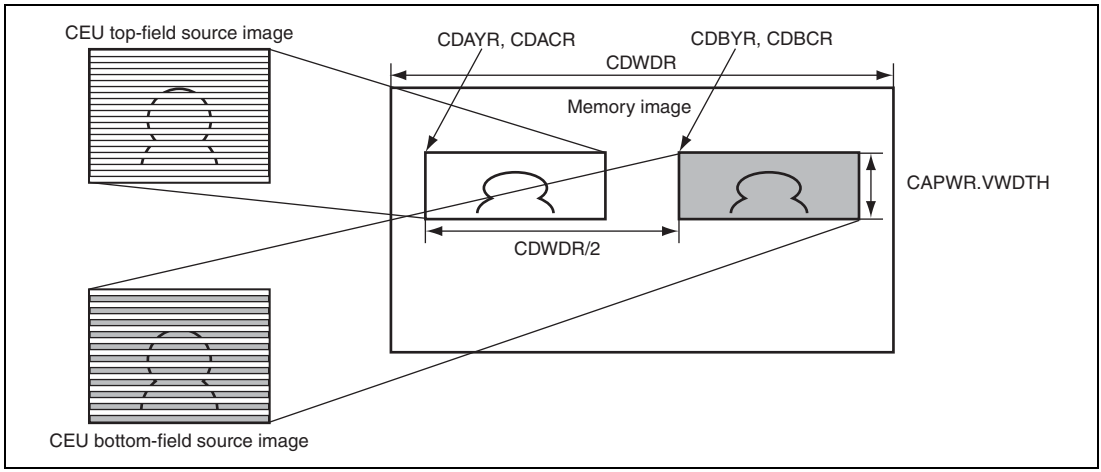


Figure 34.27 Image of Storing Captured Both-Fields of Interlace Input in Memory

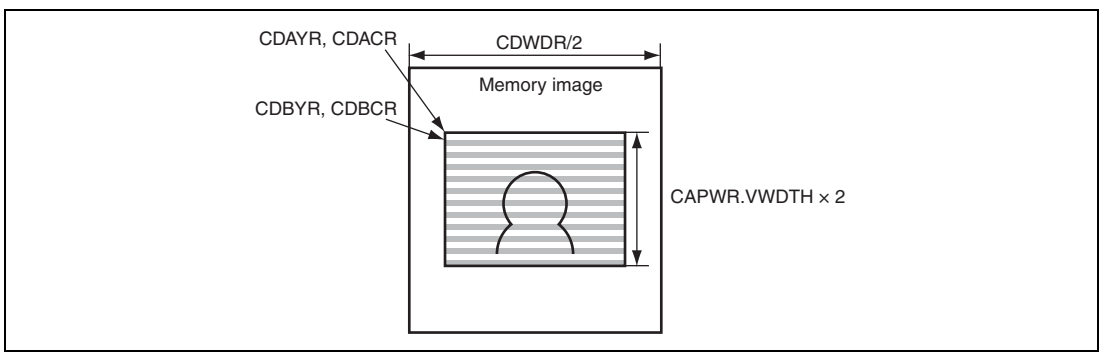


Figure 34.28 Image of Storing Interlace Input as Frame Image in Memory

34.4.8 CEU Register Control Register (CRCNTR)

CRCNTR controls switching of the planes of registers with a 2-plane configuration.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RVS	—	—	RS	RC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RVS	0	R/W	Sets the timing to switch the register plane in both-field capture. The setting of this bit is valid only when the RC bit is 1 in both-field capture. 0: Switches the register plane every 2 VD 1: Switches the register plane every 1 VD
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	RS	0	R/W	Specifies which register plane is used by the CEU in synchronization with VD. The setting of this bit is valid only when the RC bit is 0. 0: Uses plane A of the register 1: Uses plane B of the register
0	RC	0	R/W	Specifies switching of the register plane used by the CEU in synchronization with VD. If the register plane is not switched, the register plane specified by the RS bit is used. 0: Uses the specified register plane in synchronization with VD 1: Switches the register plane in synchronization with VD

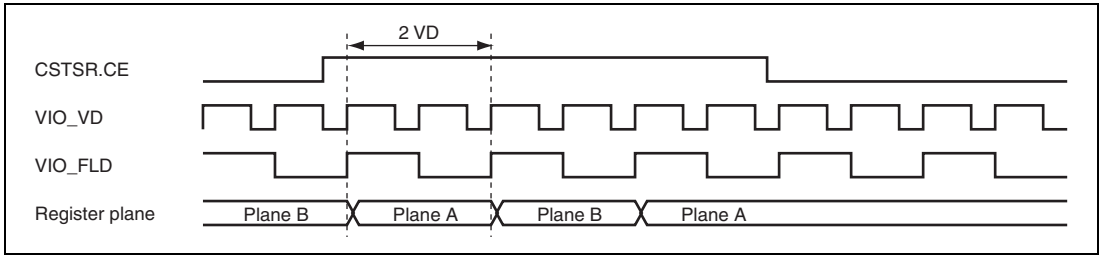


Figure 34.29 Timing for Register Plane Switching when RVS Bit is B'0

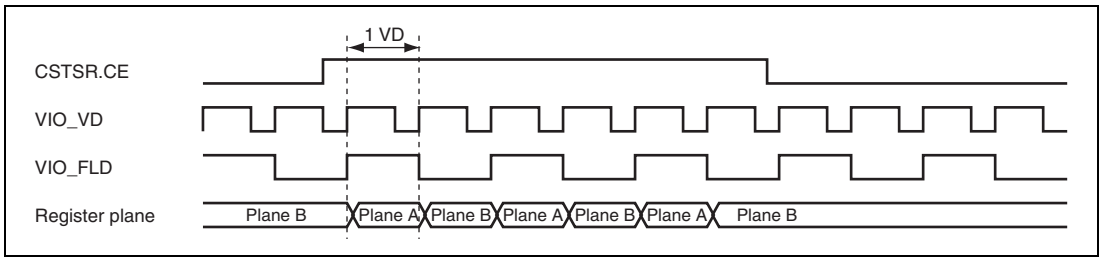


Figure 34.30 Timing for Register Plane Switching when RVS Bit is B'1

34.4.9 CEU Register Forcible Control Register (CRCMPR)

CRCMPR forcibly controls switching of the planes of registers with a 2-plane configuration. Setting this register enables direct control of register plane switching.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RA	0	R/W	Indicates the register plane currently specified. This register value automatically changes in synchronization with VD for starting capture. To start capture with plane A of the register when a setting to switch the register plane in synchronization with VD has been made (RC bit in CRCNTR is 1), specify plane B of the register using this bit. 0: Specifies plane A of the register 1: Specifies plane B of the register

34.4.10 Capture Filter Control Register (CFLCR)

CFLCR sets the scale-down factor for the filter to scale images down.

The CEU has an image scale-down filter which can be used to scale down the captured images before storing them in the memory. Set CFLCR to 0 when not performing scale-down (same size output). If a value other than 0 is set in CFLCR, scale-down is performed. In data fetch mode, set CFLCR to 0.

When handling an interlace source image as a frame image, set CFLCR to 0 not to use the filter.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VMANT[3:0]				VFRAC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HMANT[3:0]				HFRAC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	VMANT[3:0]	H'0	R/W	Mantissa Part of Vertical Scale-Down Factor The specifiable range is H'0 to H'F. When H'0 is set for the VMANT bits and H'000 is set for the VFRAC bits, the scale-down filter is not used.
27 to 19	VFRAC[11:3]	H'000	R/W	Fraction Part of Vertical Scale-Down Factor
18 to 16	VFRAC[2:0]		R	The specifiable range is H'000 to H'FF8. The fraction of the scale-down factor that cannot be set with only the VMANT bits must be set with these bits.
15 to 12	HMANT[3:0]	H'0	R/W	Mantissa Part of Horizontal Scale-Down Factor The specifiable range is H'0 to H'F. When H'0 is set for the HMANT bits and H'000 is set for the HFRAC bits, the scale-down filter is not used.
11 to 3	HFRAC[11:3]	H'000	R/W	Fraction Part of Horizontal Scale-Down Factor
2 to 0	HFRAC[2:0]		R	The specifiable range is H'000 to H'FF8. The fraction of the scale-down factor that cannot be set with only the HMANT bits must be specified with these bits.

An image scale-down filter is installed in the CEU, and the captured images can be scaled down and stored in the memory.

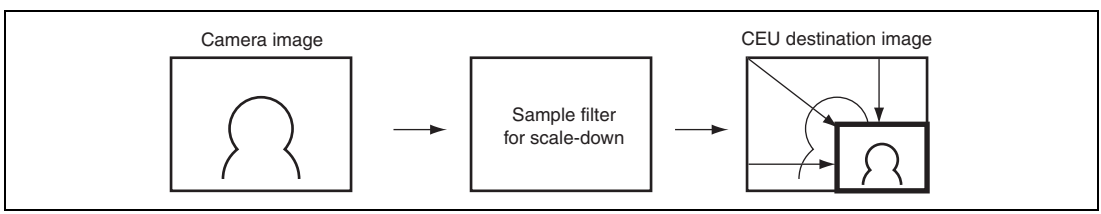


Figure 34.31 Scale-Down of Captured Image

The formulas for obtaining the MANT (VMANT or HMANT) and FRAC (VFRAC or HFRAC) values from the input pixel count and output pixel count of the filter are shown below. Set the MANT and FRAC bits in order to obtain the desired output pixel count from the number of pixels input to the CEU.

First, calculate preliminary MANT and FRAC values. The parameters needed for calculation are defined as follows:

$$\alpha = MANT \times 4096 + FRAC \quad \text{. . . Formula 1}$$

$$SCL \text{ (scaling factor)} = \frac{4096}{\alpha} \quad \text{. . . Formula 2}$$

Assuming an operator $\lfloor x \rfloor$ which discards fractions of an integer x, the MANT and FRAC values can be temporarily set as follows, according to formula 1 and formula 2.

$$MANT = \left\lfloor \frac{1}{\mathrm{SCL}} \right\rfloor, \quad FRAC = \left\lfloor 512 \times \left(\frac{1}{SCL} - MANT \right) \right\rfloor \times 8$$

Here, the scaled-down filter output size ($SIZE_D$) can be calculated using the input image size S_{in} (8-bit interface: half of the CAPWR setting, 16-bit interface: CAPWR setting) in the following formula.

$$SIZE_D = \left\lfloor 1 + \left(\left\lfloor \frac{1}{2} + \frac{S_{in} - 1}{MANT_{pre}} \right\rfloor - 1 \right) \times \frac{MANT_{pre} \times 4096}{\alpha} \right\rfloor \quad \text{. . . Formula 3}$$

$$\begin{bmatrix} MANT_{pre} = 1 \longrightarrow (0 \le MANT < 2) \\ MANT_{pre} = 2 \longrightarrow (2 \le MANT < 4) \\ MANT_{pre} = 4 \longrightarrow (4 \le MANT < 8) \\ MANT_{pre} = 8 \longrightarrow (8 \le MANT\ ) \end{bmatrix}$$

The number of output pixels can be obtained by substituting the temporarily calculated MANT, FRAC, and input image size into these formulas. If the calculated number of output pixels is smaller than the number of output pixels used to obtain the preliminary MANT and FRAC values, recalculate with a smaller FRAC (α) value, and set the MANT and FRAC values in this register so that a pixel value greater than the desired number of output pixels can be obtained.

Example: Scale down 640 pixels to 480 pixels

SCL = 480/640 = 3/4, and the preliminary settings of MANT = 1, $MANT_{pre}$ = 1, and FRAC = H'550 are made. Substituting these in the following formula results in an output pixel count of 479.

$$SIZE_D = \left\lfloor 1 + \left(\left\lfloor \frac{1}{2} + \frac{S_{in} - 1}{MANT_{pre}} \right\rfloor - 1 \right) \times \frac{MANT_{pre} \times 4096}{\alpha} \right\rfloor \quad \text{. . . Formula 3}$$

Since this output pixel count is smaller than the desired output pixel count of 480, the formula is recalculated with a FRAC value of H'548, a value eight less than the previous time. The obtained result of output pixel count = 480 is equal to the desired output pixel count of 480, so this register is set as MANT = 1 and FRAC = H'548.

Table 34.9 Setting Examples for Each Scale-Down Filter Factor

Scale-Down Factor	FRAC		MANT	Input Pixel Count	Output Pixel Count	Clipping Size (CFSZR)
	Decimal	Hexadecimal				
7/8	576	H'240	1	640	560	560
3/4	1352	H'548	1	640	480	480
5/8	2448	H'990	1	640	400	400
1/2	0	H'0	2	640	320	320
3/8	2728	H'AA8	2	640	240	240
1/3	0.0	H'0	3	640	213	212
1/4	0.0	H'0	4	640	160	160
1/5	0.0	H'0	5	640	128	128
1/6	0.0	H'0	6	640	107	104
1/7	0.0	H'0	7	640	91	88
1/8	0.0	H'0	8	640	80	80
1/16	4088	H'FF8	15	640	40	40

Note: This scale-down filter uses a VGA-size line memory for scale-down. Therefore, when an image larger than the VGA size is input for scale-down, settings must be made so that the output image size is equal to or larger than the SubQCIF size and equal to or smaller than the VGA size. When an image is not scaled down (same size output), this restriction does not apply.

34.4.11 Capture Filter Size Clip Register (CFSZR)

CFSZR sets the clipping size for fine adjustment of the image size output from the filter, and must be set in combination with CFLCR. When clipping the output size of the filter, set the clipping size as a number of pixels, and the setting unit should be four pixels. CFSZR must be set even when scale-down is not performed (same size output).

In data synchronous fetch mode, set CFSZR according to the setting of CAPWR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VFCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	HFCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 18	VFCLP[11:2]	H'000	R/W	These bits set the vertical clipping value of the filter output size (4-pixel units).
17, 16	VFCLP[1:0]		R	
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 1	HFCLP[11:1]	H'000	R/W	These bits specify the horizontal clipping value of the filter output size (4-pixel units).
0	HFCLP[0]		R	

The scale-down filter in the CEU may output an odd number of pixels or lines depending on the settings. To adjust the output size of the filter, the CEU clips the destination image by using the number of pixels specified in CFSZR, as shown in Figure 34.32. The clipping size must be specified vertically and horizontally in 4-pixel units.

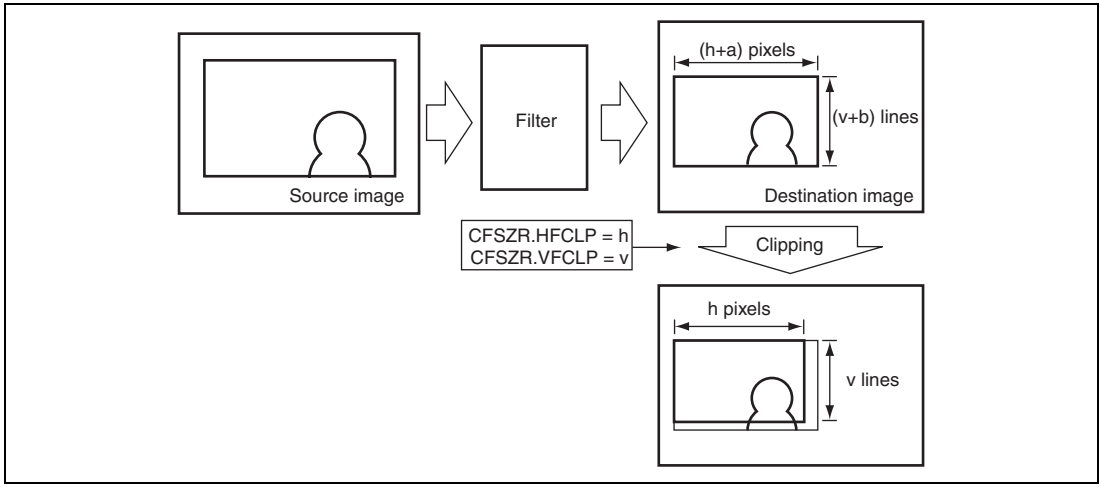


Figure 34.32 Clipping of Image Output from Filter

The pixels to be clipped are counted from the top-left corner of a display. The pixels located to the right of the specified number of pixels or below the specified number of lines are discarded by the clipping function. If the number of pixels specified in CFSZR is larger than that output from the filter, correct operation cannot be guaranteed. To avoid this, the clipping size specified in CFSZR must be equal to or smaller than the number of pixels output from the filter.

Note: In data synchronous fetch mode, the following settings are required. Data cannot be fetched correctly unless the following settings are made.
8-bit interface: VFCLP = CAPWR.VWDTH
HFCLP = CAPWR.HWDTH/2
16-bit interface: VFCLP = CAPWR.VWDTH
HFCLP = CAPWR.HWDTH

34.4.12 Capture Destination Width Register (CDWDR)

CDWDR sets the horizontal image size in the memory area where the captured image is to be output in 4-byte units (4-pixel units).

In data synchronous fetch mode, set CDWDR according to the setting of CAPWR.

This register is not used, during data enable fetch mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CHDW[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 2 1, 0	CHDW[12:2] CHDW[1:0]	H'0000	R/W R	These bits specify the horizontal image size in the memory area where the captured image is to be stored (4-byte units). The image data captured by the CEU is stored in the memory. If the right end of the captured image does not match the horizontal image size in the memory area as shown in Figure 34.33, some addresses must be skipped at the right end of the image when storing the captured image. Therefore, the horizontal image size in the memory area where the captured image is to be stored must be set in these bits. The maximum value to be set is 8188 bytes (8188 pixels). In data synchronous fetch mode, set as follows: 8-bit interface: CHDW = CAPWR.HWDTH 16-bit interface: CHDW = CAPWR.HWDTH × 2

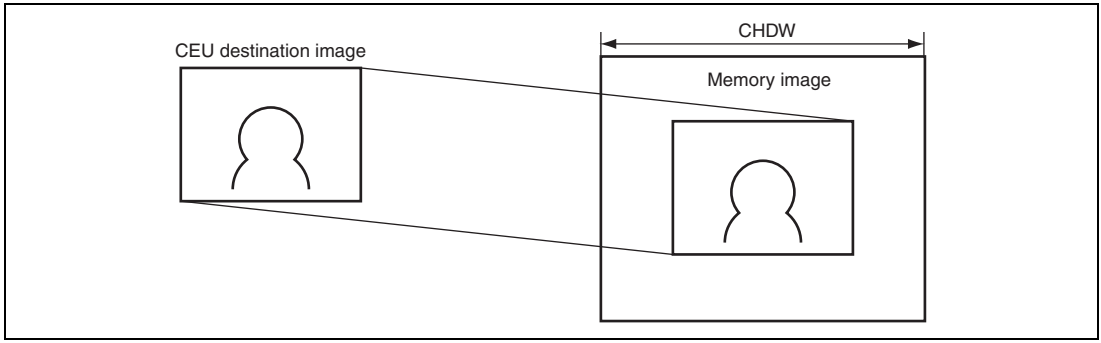


Figure 34.33 Captured Image and Memory Area Image

34.4.13 Capture Data Address Y Register (CDAYR)

CDAYR specifies the address where the luminance (Y) component of the captured data is to be stored in frame image capture or one-field image capture, the address where the luminance (Y) component of the captured top field is to be stored in both-field image capture, and the address where the fetched data is to be stored in data fetch. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the Y (luminance) component of the captured data is to be stored by CDAYR. In both-field image capture, set the start address of the memory area where the Y (luminance) component of the captured top-field image is to be stored by CDAYR. In data fetch, set the start address of the memory area where data is to be stored by CDAYR.

Because the address must be specified in 32 bits, the address set by CDAYR must be in longword units. As the setting is in 4-pixel units for image capture and 4-byte units for data fetch, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAYR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAYR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	CAYR[31:2] CAYR[1:0]	H'0000 0000	R/W R	• Frame image capture: These bits set the address for storing the Y (luminance) component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the Y (luminance) component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the Y (luminance) component data of the captured top-field data (4-pixel units). • Data fetch: These bits set the address for storing data (4-byte units). • Data enable fetch bundle write: These bits set the address for storing data (32-byte units).

Set the address of the starting point of the memory area where the fetched data is to be stored in this register, as shown in Figure 34.34.

- Frame image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the Y component of the captured top-field image is to be stored.
- Data fetch: Set the address of the starting point of the memory area where the fetched data is to be stored. In data fetch mode, the data is simply stuffed in order from the start address so the end address becomes as follows:

  End address = CDAYR + number of fetched bytes

- Data enable fetch bundle write: Set the address in 32-byte units.

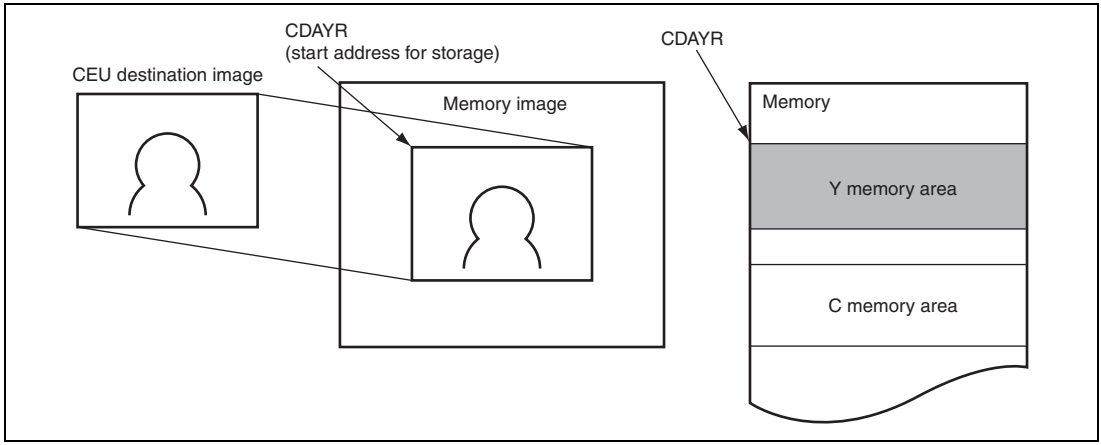


Figure 34.34 Relationship between Captured Image and Y Component Memory Area

34.4.14 Capture Data Address C Register (CDACR)

CDACR specifies the address where the chrominance (C) component of the captured data is to be stored in frame image capture or one-field image capture, and the address where the chrominance (C) component of the captured top field is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the C (chrominance) component of the captured data is to be stored by CDACR. In both-field image capture, set the start address of the memory area where the C (chrominance) component of the captured top-field image is to be stored by CDACR. CDACR is not used in data fetch.

Because the address must be specified in 32 bits, the address set by CDACR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CACR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CACR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	CACR[31:2] CACR[1:0]	H'0000 0000	R/W R	• Frame image capture: These bits set the address for storing the C (chrominance) component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the C (chrominance) component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the C (chrominance) component data of the captured top-field data (4-pixel units).

Set the address of the starting point of the memory area where the C component of the captured image is to be stored in this register, as shown in Figure 34.35. The C component has an output data format like that in Figure 34.36, and is saved in the memory in this format.

- Frame image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the C component of the captured top-field image is to be stored.

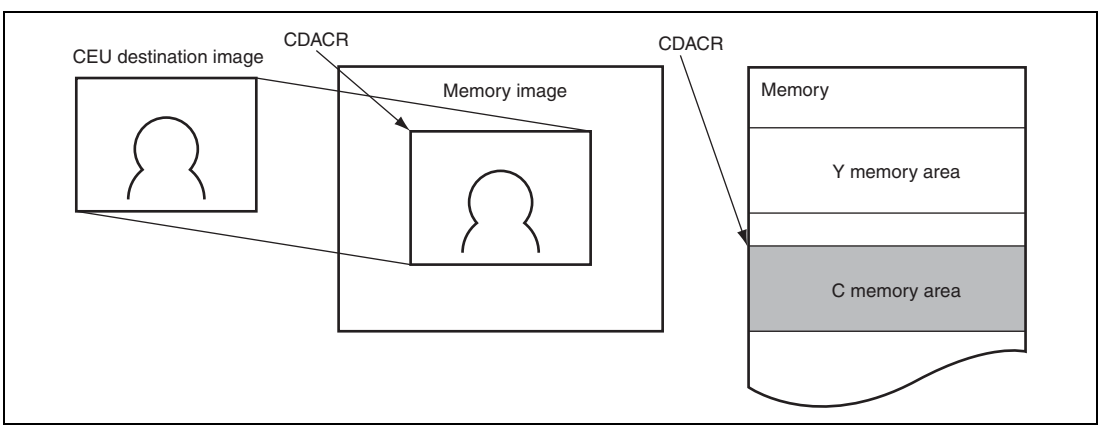


Figure 34.35 Relationship between Captured Image and C Component Memory Area

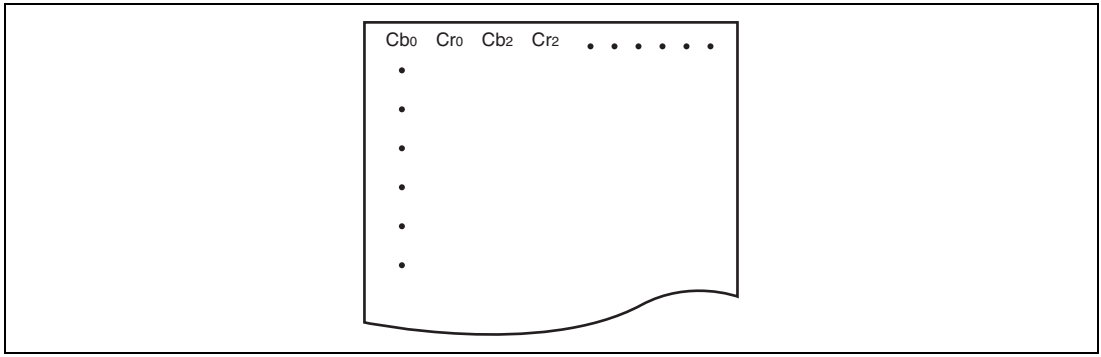


Figure 34.36 Image of Storing C Components in Memory

34.4.15 Capture Data Bottom-Field Address Y Register (CDBYR)

CDBYR specifies the address where the luminance (Y) component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the Y (luminance) component of the captured bottom-field image is to be stored by CDBYR. CDBYR is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBYR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBYR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBYR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBYR[31:2]	H'0000 0000	R/W	These bits set the address for storing the Y (luminance) component data of the captured bottom-field data (4-pixel units).
1, 0	CBYR[1:0]		R	

Set the address of the starting point of the memory area where the Y component of the captured bottom-field image is to be stored in this register, as shown in Figure 34.37.

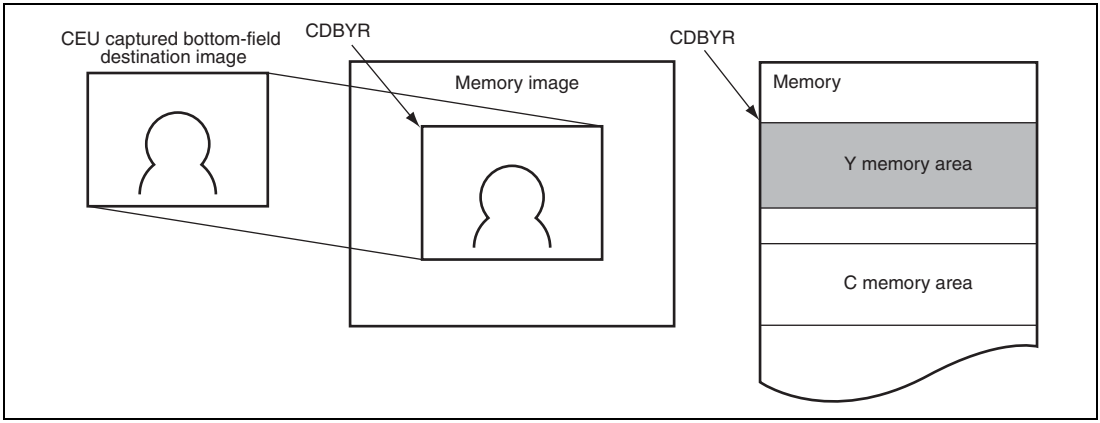


Figure 34.37 Relationship between Captured Bottom-Field Image and Y Component Memory Area

34.4.16 Capture Data Bottom-Field Address C Register (CDBCR)

CDBCR specifies the address where the chrominance (C) component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the C (chrominance) component of the captured bottom-field image is to be stored by CDBCR. CDBCR is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBCR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBCR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBCR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBCR[31:2]	H'0000 0000	R/W	These bits set the address for storing the C (chrominance) component data of the captured bottom-field data (4-pixel units).
1, 0	CBCR[1:0]		R	

Set the address of the starting point of the memory area where the C component of the captured bottom-field image is to be stored in this register, as shown in Figure 34.38. The C component has an output data format like that in Figure 34.39, and is saved in the memory in this format.

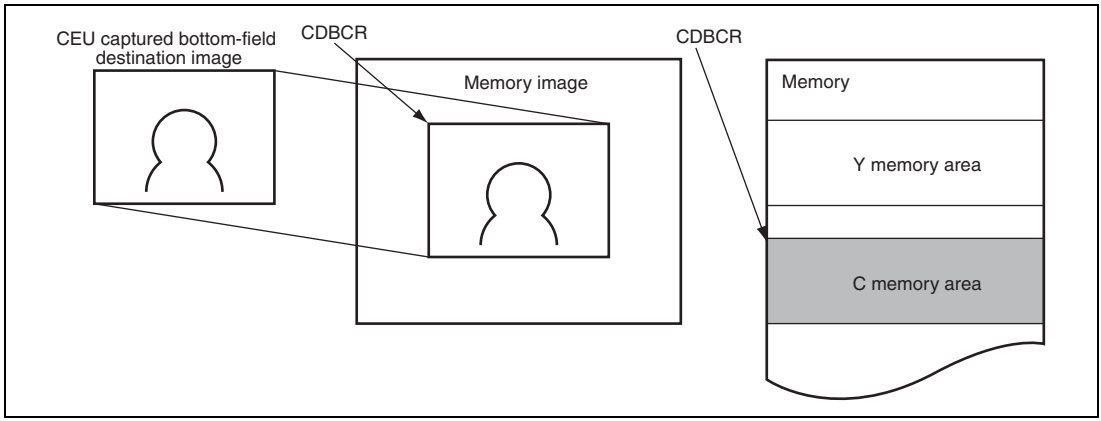


Figure 34.38 Relationship between Captured Bottom-Field Image and C Component Memory Area

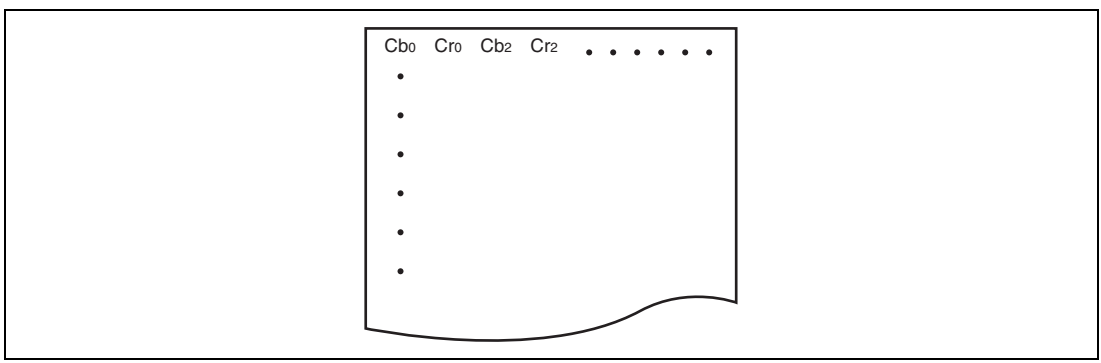


Figure 34.39 Image of Storing C Components in Memory

34.4.17 Capture Bundle Destination Size Register (CBDSR)

CBDSR sets the size of output to memory in a bundle write. The number of output lines should be specified for image capture or data synchronous fetch. The number of bytes should be specified for data enable fetch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CBVS[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBVS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	CBVS[22:3] CBVS[2:0]	H'000	R/W R	These bits select the number of lines or number of bytes for output to the memory in a bundle write. Image capture and data synchronous fetch: Number of lines for output to the memory in a bundle write. Unit: 8 lines, min.: 8 lines, max.: 4088 lines (H'FF8) Data enable fetch: Number of bytes for output to the memory in a bundle write. Unit: 32 bytes, min.: 512 bytes, max.: 6291456 lines (H'600000)

(a) Image capture and data synchronous fetch

Set the number of lines of captured data to be written to the memory by a bundle write as a multiple of eight. This register is valid only when the CBE bit in CDOCR is 1. When the CBE bit in CDOCR is 1 and this register cleared to H'0, this module operates with the number of lines of captured data to be written to the memory as eight. The maximum number of lines that can be set is 4088 (H'FF8). Only bits CBVS[11:3] are valid.

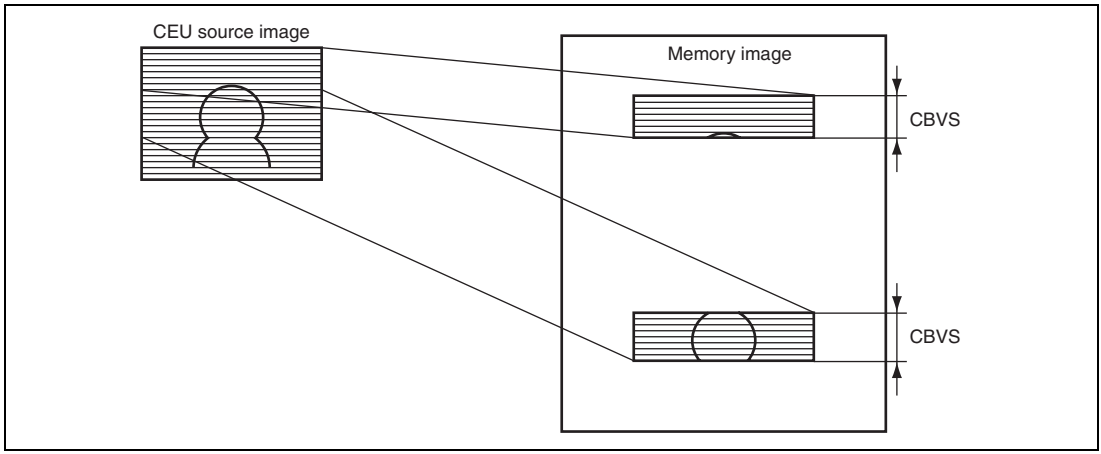


Figure 34.40 Image of Storing Captured Image in Memory by Bundle Write

(b) Data enable fetch

Set the number of bytes of captured data to be written to the memory by a bundle write as a multiple of 32. This register is valid only when the CBE bit in CDOCR is 1. The minimum settable size is 512 bytes. When a number smaller than 512 bytes is specified, operation is not guaranteed.

34.4.18 Capture Low-Pass Filter Control Register (CLFCR)

CLFCR specifies whether or not to operate the low-pass filter. In data fetch mode, clear the LPF bit to B'0.

The characteristic of the low-pass filter installed in the CEU causes the phase location of the image processed by the low-pass filter to be shifted right by one pixel compared to the raw image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LPF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LPF	0	R/W	Enables or disables operation of the low-pass filter. The low-pass filter removes high-frequency components from the destination image in the horizontal direction. Clear this bit to 0 in data fetch mode. 0: Low-pass filter not used 1: Low-pass filter used (only in the horizontal direction)

34.4.19 Firewall Operation Control Register (CFWCR)

CFWCR specifies the upper limit of the write addresses in data enable fetch. When the VD input from an external module dose not go low and end notification is not given, this register can prevent writing to memory from being out of control.

This register is enabled only in data enable fetch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FWV[26:11]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FWV[10:0]											—	—	—	—	FWE
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	FWV[26:0]	H'0000008	R/W	These bits specify the upper limit of a write address. Specify the upper 27 bits of the 32-bit address. The upper limit of an address is FWV[26:0]<<5 + H'1F.
4 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FWE	0	R/W	With the setting of FWE = 1, when an address exceeds the value set with FWV, the address is retained and an interrupt source FWF is set. After this, the address is not incremented and data is overwritten on the upper limit address. 0: Firewall is not activated. 1: Firewall is activated.

34.4.20 Capture Data Output Control Register (CDOCR)

CDOCR sets the format for outputting captured data to the memory. In data fetch mode, set the CDS bit to B'1 and the CBE bit to B'0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CBE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CFXAD	—	—	—	CDS	—	COLS	COWS	COBS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CBE	0	R/W	Controls the number of lines of captured data to be written to the memory. • Image capture This bit controls the number of lines of captured data to be written to the memory. When bundle write is set by this register, captured data is written in line units specified by CBDSR to the addresses specified by CDAYR and CDACR, and CDAYR2 and CDACR2 (CDBYR and CDBCR, and CDBYR2 and CDBCR2 for the bottom field in both-field capture) alternately (Figure 34.41). When captured data has been written for the number of lines set by CBDSR, a write end interrupt corresponding to each address setting register occurs. However, after write for one-frame (one-field) capture ends, a bundle write end interrupt does not occur even when bundle write has finished.

Bit	Bit Name	Initial Value	R/W	Description
16	CBE	0	R/W	• Data synchronous fetch • This bit controls the number of lines of captured data to be written to the memory. When bundle write is set by this register, captured data is written in line units specified by CBDSR to the addresses specified by CDAYR and CDAYR2 alternately. When captured data has been written for the number of lines set by CBDSR, a write end interrupt corresponding to each address setting register occurs. However, after write for one-frame capture ends, a bundle write end interrupt does not occur even when bundle write has finished. • Data enable fetch This bit controls the number of bytes of captured data to be written to the memory. When bundle write is set by this register, captured data is written in byte units specified by CBDSR to the addresses specified by CDAYR and CDAYR2 alternately. When captured data has been written for the number of bytes set by CBDSR, a write end interrupt corresponding to each address setting register occurs. Also, only in data enable fetch, a bundle write end interrupt occurs when bundle write has finished after write for one-frame capture ends. Table 34.10 shows the correspondence between address setting registers and write end interrupt sources. Figure 34.42 shows the timing of write end interrupts in image capture and data synchronous fetch. Figure 34.43 shows the timing of write end interrupts in data enable fetch. 0: Normal write 1: Bundle write
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	CFXAD	0	R/W	Sets the fixed address mode for outputting images or data processed by the CEU to the memory. If fixed address mode is selected, the addresses to which the CEU outputs images are not incremented and are always the same. In fixed address mode, addresses whose lower five bits are 0 must be set to the output address registers, CDAYR, CDACR, CDBYR, and CDBCR. 0: Output addresses for the output section are incremented (normal operation) 1: Output addresses for the output section are fixed (fixed address mode)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CDS	0	R/W	Sets the image format when outputting the image data captured in the YCbCr 4:2:2 format to the memory. When 0 is written to this bit, only the luminance component (Y) is output and no chrominance components (Cb and Cr) are output for the odd-numbered lines. With an interlace source image, similarly only the luminance component (Y) is output and no chrominance components (Cb and Cr) are output for the odd-numbered lines of the field. In data fetch mode, set this bit to 1. 0: Converts the YCbCr 4:2:2 format to the YCbCr 4:2:0 format before outputting data to the memory 1: Outputs data in the YCbCr 4:2:2 format to the memory without conversion
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	COLS	0	R/W	Controls swapping in longword units for data output from the CEU. 0: Data is not swapped in longword units 1: Data is swapped in longword units

Bit	Bit Name	Initial Value	R/W	Description
1	COWS	0	R/W	Controls swapping in word units for data output from the CEU. 0: Data is not swapped in word units 1: Data is swapped in word units
0	COBS	0	R/W	Controls swapping in byte units for data output from the CEU. 0: Data is not swapped in byte units 1: Data is swapped in byte units

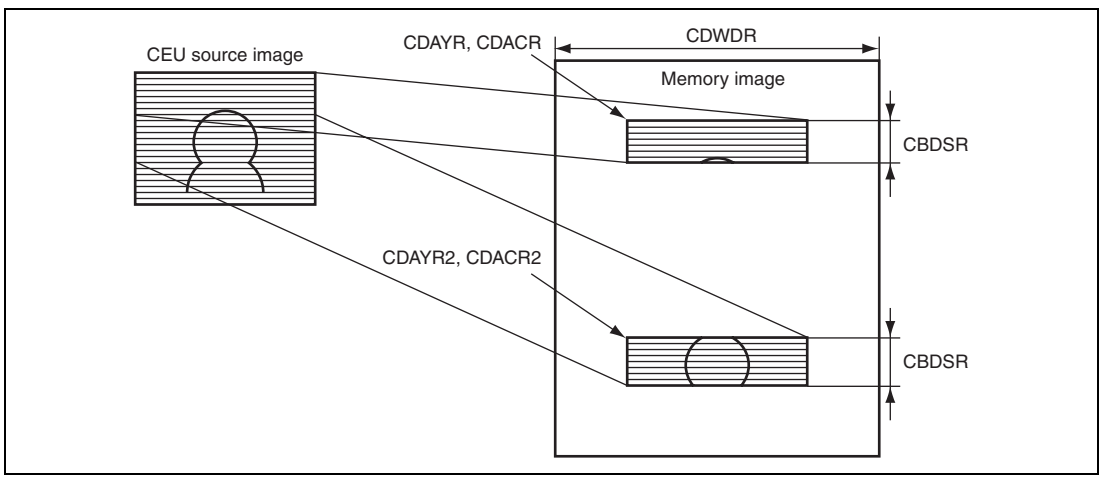


Figure 34.41 Image of Bundle Write to Memory

Table 34.10 Correspondence between Address Setting Registers and Write End Interrupt Sources

Address Setting Registers	Bundle Write End Interrupt Source
CDAYR, CDACR	CPBE1 bit in CETCR
CDAYR2, CDACR2	CPBE2 bit in CETCR
CDBYR, CDBCR	CPBE3 bit in CETCR
CDBYR2, CDBCR2	CPBE4 bit in CETCR

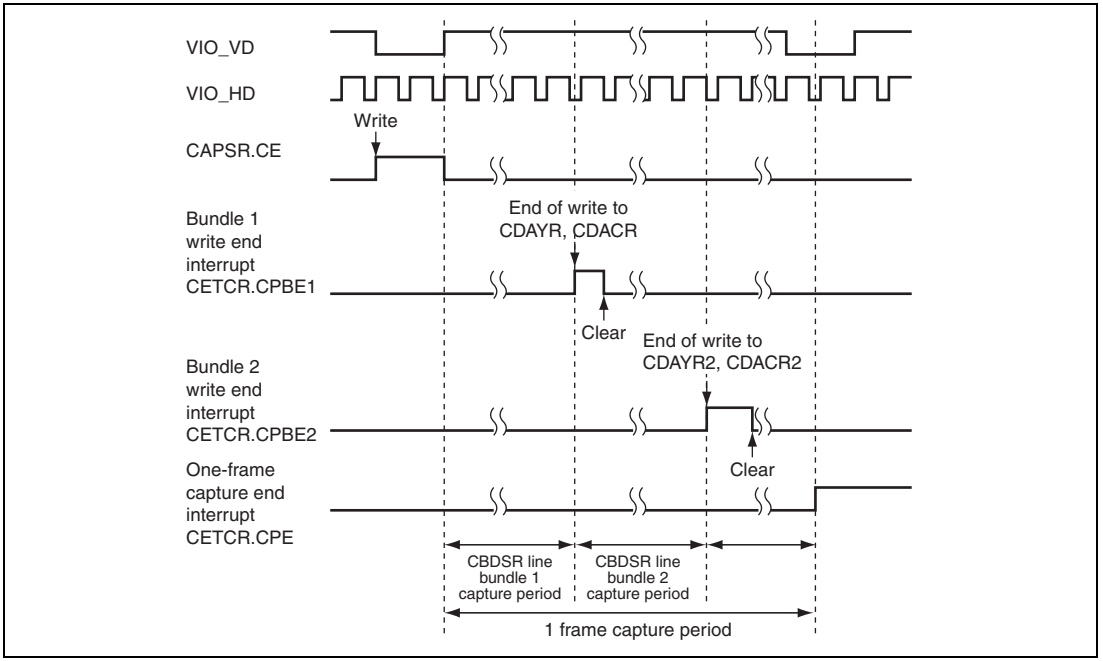


Figure 34.42 Timing of Write End Interrupts (Image Capture, Data Synchronous Fetch)

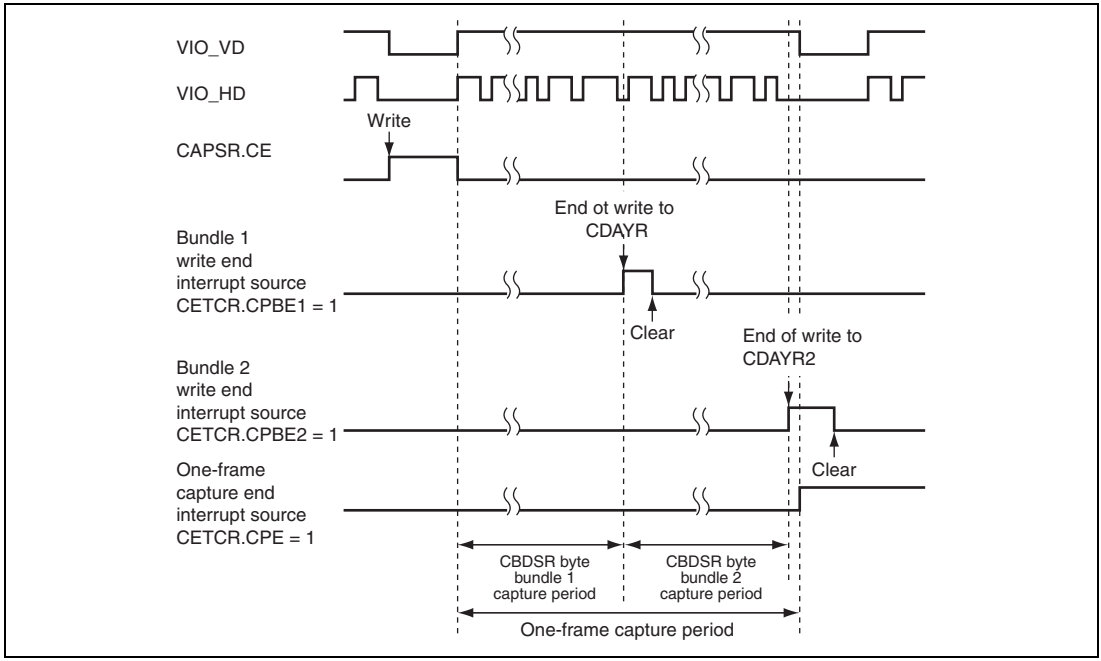


Figure 34.43 Timing of Write End Interrupts (Data Enable Fetch)

For data output from the CEU, the COLS, COWS, and COBS bits control swapping in longword, word, and byte units, respectively. Set these bits when data is misaligned because of a difference in endian. The data swapping bits are shown below. These bits can be set similarly in data fetch mode.

Data can be swapped in byte, word, longword, and byte-word-longword units, as shown in Figure 34.44. To enable data swapping, set the corresponding control bit to B'1.

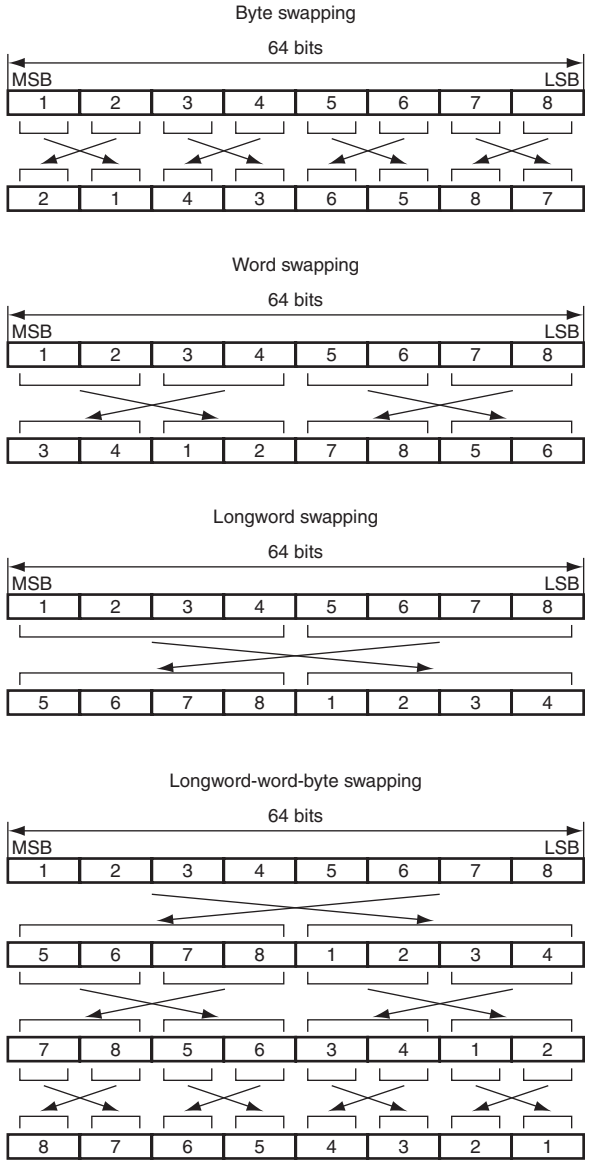


Figure 34.44 Data Swapping by Data Aligner

34.4.21 Capture Data Complexity Level Register (CDDCR)

CDDCR sets how to acquire the complexity level of the captured image. The CEU calculates the complexity level of the image from the luminance component value of the captured image. The result of comparing the complexity levels of continuous images can be useful information in detecting a scene change. CDDCR is not used in data fetch mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	DCOE	—	—	DCSL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DCOE	0	R/W	Enables or disables output of the complexity level of the captured image to the memory. When writing 1 to this bit, always set a memory address to write the complexity level. If capturing is started with no address set, data in the memory may be overwritten. In data fetch mode, clear this bit to 0. 0: Complexity level of the captured image is not output to the memory 1: Complexity level of the captured image is output to the memory
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	DCSL[1:0]	00	R/W	These bits select the operation pixel range when calculating the complexity level of an image. 00: Calculates the complexity level of the luminance component for 16 lines of the captured image 01: Calculates the complexity level of the luminance component for eight lines of the captured image 10: Calculates the complexity level of the luminance component for one display of the captured image 11: Setting prohibited

34.4.22 Capture Data Complexity Level Address Register (CDDAR)

CDDAR specifies the address where the complexity level of the captured image is to be stored. The CEU writes the complexity level of the captured image to the memory when the DCOE bit in CDDCR is B'1. Set the start address of the memory area where the complexity level of the captured image is to be stored by CDDAR. CDDAR is not used in data fetch mode.

Note that the destination address should be set for each frame (field) also in continuous capture.

Whether or not to acquire the complexity level of the captured image is set by the DCOE bit in CDDCR, and the acquisition range (16 lines, eight lines, or one display) is set by the DCSL bit in CDDCR.

Because the address must be specified in 32 bits, the address set by CDDAR must be in longword units. As the setting is in longword units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCAR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCAR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	DCAR[31:2]	H'0000 0000	R/W	These bits set the address for storing the complexity level of the captured image (4-byte units).
1, 0	DCAR[1:0]		R	

Set the address of the starting point of the memory area where the complexity level of the captured image is to be stored in this register, as shown in Figure 34.45. The complexity level has an output data format in longword units like that in Figure 34.46, and is saved in the memory in this format.

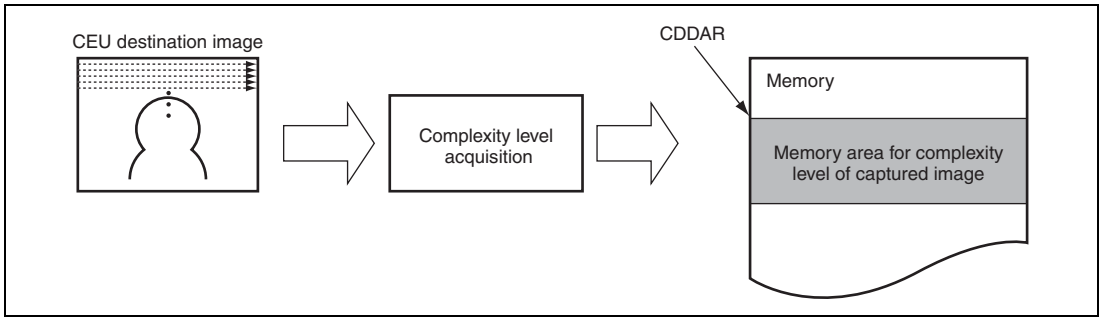


Figure 34.45 Memory Area for Complexity Level of Captured Image

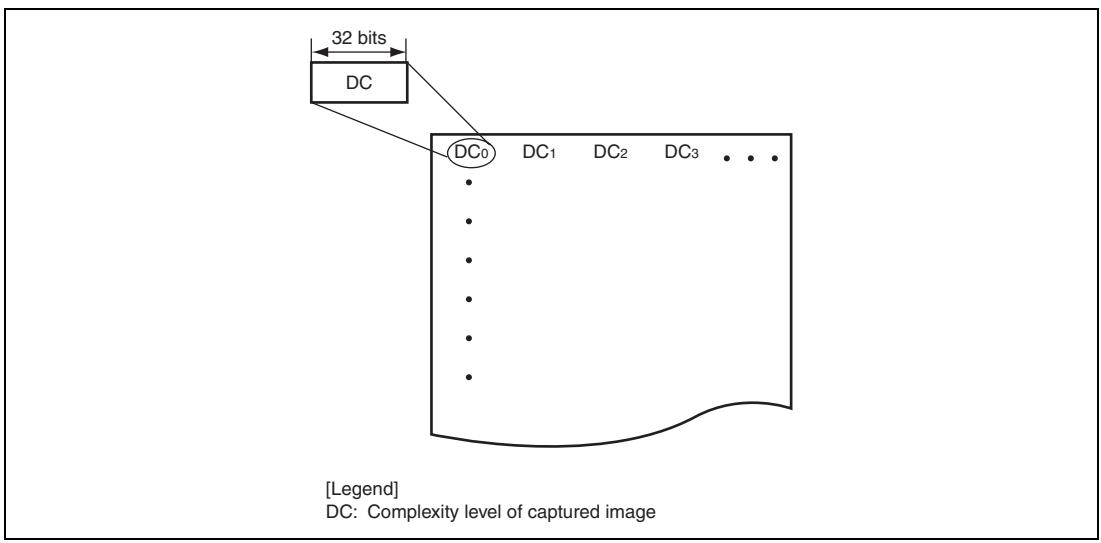


Figure 34.46 Image of Storing Complexity Level of Captured Image in Memory

34.4.23 Capture Event Interrupt Enable Register (CEIER)

CEIER enables or disables interrupts of the event flag register that generates CEU interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NVDIE	NHDIE	FWFIE	DCOFIE	—	VBPIE	—	IGVSIE	IGHSIE	CDTOFIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPBE4IE	CPBE3IE	CPBE2IE	CPBE1IE	—	—	VDIE	HDIE	—	—	—	IGRWIE	—	—	CFEIE	CPEIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NVDIE	0	R/W	Non-VD Interrupt Enable Disable this interrupt (NVDIE = 0) for data enable fetch. 0: Disables a non-VD interrupt 1: Enables a non-VD interrupt
24	NHDIE	0	R/W	Non-HD Interrupt Enable Disable this interrupt (NHDIE = 0) for data enable fetch. 0: Disables a non-HD interrupt 1: Enables a non-HD interrupt
23	FWFIE	0	R/W	FWF Interrupt Enable 0: Disables a FWF interrupt 1: Enables a FWF interrupt
22	DCOFIE	0	R/W	DCOF Interrupt Enable 0: Disables a DCOF interrupt 1: Enables a DCOF interrupt
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	VBPIE	0	R/W	VBP Interrupt Enable 0: Disables a VBP interrupt 1: Enables a VBP interrupt

Bit	Bit Name	Initial Value	R/W	Description
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	IGVSIE	0	R/W	IGVS Interrupt Enable 0: Disables an IGVS interrupt 1: Enables an IGVS interrupt
17	IGHSIE	0	R/W	IGHS Interrupt Enable 0: Disables an IGHS interrupt 1: Enables an IGHS interrupt
16	CDTOFIE	0	R/W	CDTOF Interrupt Enable 0: Disables a CDTOF interrupt 1: Enables a CDTOF interrupt
15	CPBE4IE	0	R/W	CPBE4 Interrupt Enable 0: Disables a CPBE4 interrupt 1: Enables a CPBE4 interrupt
14	CPBE3IE	0	R/W	CPBE3 Interrupt Enable 0: Disables a CPBE3 interrupt 1: Enables a CPBE3 interrupt
13	CPBE2IE	0	R/W	CPBE2 Interrupt Enable 0: Disables a CPBE2 interrupt 1: Enables a CPBE2 interrupt
12	CPBE1IE	0	R/W	CPBE1 Interrupt Enable 0: Disables a CPBE1 interrupt 1: Enables a CPBE1 interrupt
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	VDIE	0	R/W	VD Interrupt Enable 0: Disables a VD interrupt 1: Enables a VD interrupt
8	HDIE	0	R/W	HD Interrupt Enable 0: Disables an HD interrupt 1: Enables an HD interrupt

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	IGRWIE	0	R/W	Register-Access-During-Capture Interrupt Enable 0: Disables a register-access-during-capture interrupt 1: Enables a register-access-during-capture interrupt
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CFEIE	0	R/W	CFE Interrupt Enable 0: Disables a CFE interrupt 1: Enables a CFE interrupt
0	CPEIE	0	R/W	One-Frame Capture End Interrupt Enable 0: Disables a one-frame capture end interrupt 1: Enables a one-frame capture end interrupt

34.4.24 Capture Event Flag Clear Register (CETCR)

CETCR notifies the CPU of the source of an interrupt that is generated in the CEU. The flags of CETCR can be used as interrupt signals. When the corresponding interrupt is enabled, an interrupt is generated. To clear the interrupt, clear the bit corresponding to the interrupt source to 0. After several cycles have passed after modifying the bit, the interrupt is cleared.

To clear the bit corresponding to the interrupt source to be cleared to 0 and retain that state, write 1 to that bit. For example, to clear only the CPE bit to 0, write H'FFFF FFFE to CETCR.

In CETCR, only bits to which 0 is written are cleared. Bits to which 1 is written retain their current values. To clear an interrupt source, write 0 only to the bit corresponding to the interrupt source to be cleared, and 1 to the other bits.

Note: Since the CETCR value becomes undefined in the following cases, clear all bits in CETCR to 0.

- VD and HD bits immediately after reset
- All bits after the clock is stopped
- VD and HD bits after the polarities of the capture interface sync signals are changed

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NVD	NHD	FWF	DCOF	—	VBP	—	IGVS	IGHS	CDTOF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPBE4	CPBE3	CPBE2	CPBE1	—	—	VD	HD	—	—	—	IGRW	—	—	CFE	CPE
Initial value:	0	0	0	0	0	0	—	—	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NVD	0	R/W	This bit is used for an interrupt indicating that no VD was input. A non-VD interrupt occurs when the 14-bit internal counter becomes full. Accordingly, this bit is set to 1 when no VD has been input for at least 16,384 lines since the last VD was input.

Bit	Bit Name	Initial Value	R/W	Description
24	NHD	0	R/W	This bit is used for an interrupt indicating that no HD was input. The timing for a non-HD interrupt to occur differs depending on the bit width of the digital image input pins. 8-bit digital image input pins: Occurs when the 11-bit internal counter that is incremented every eight cycles becomes full. Accordingly, this bit is set to 1 when no HD has been input for at least 16,376 cycles since the last HD was input. 16-bit digital image input pins: Occurs when the 12-bit internal counter that is incremented every four cycles becomes full. Accordingly, this bit is set to 1 when no HD has been input for at least 16,380 cycles since the last HD was input. When connecting a camera whose HD is fixed low when VD is low, this bit may be set to 1. Ignore this interrupt during data enable fetch.
23	FWF	0	R/W	The interrupt is generated when data is written to the address that exceeds the value specified with CFWCR.FMV. This bit is set to 1 when data is written to the address that exceeds the value specified with CFWCR.FMV while CFWCR.FWE = 1.
22	DCOF	0	R/W	This bit is used for an interrupt indicating that the data requested to be transferred during acquisition of the complexity level of an image has been overwritten with subsequent data. Since there is no buffer to store data while acquiring the complexity level of an image, if data transfer to the bus is not completed until the next complexity level is calculated, data waiting to be transferred to the bus will be overwritten by the next data. This bit is set to 1 when the complexity level of the next image overwrites the complexity level of the image waiting to be transferred to the bus.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
20	VBP	0	R/W	This bit is used for an interrupt indicating that VD has been input while the CEU holds data (insufficient V.B.P.). The conditions for a VBP interrupt to occur are as follows: • Condition 1 VD is input when there is captured data within the CEU • Condition 2 The last transfer data cannot be internally detected due to a write buffer overflow or an illegal HD so that the end timing is unclear until the next VD (By generating a VBP interrupt at the VD input timing, capture fail can be announced.) When a VBP interrupt occurs, a capture end interrupt (CPE bit in CETCR) does not occur and the image of that frame is not captured correctly. Though a capture end interrupt (CPE bit) will occur on rare occasions, it should be ignored in this case. Capturing cannot be performed until the next VD (even if the CE bit (capture reservation signal) in CAPSR is 1, capturing does not start). In the case of condition 2, instead of waiting for a VBP interrupt to occur, execute a software reset (CPKIL bit in CAPSR) to stop capturing and then restart capturing. In this case, since capture operation is terminated without waiting for the next VD, a VBP interrupt does not occur and capturing can be performed from the next VD.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	IGVS	0	R/W	This bit is used for an interrupt generated when the number of VD cycles set in CMCYR differ from the number of VD cycles input from an external module. This bit is set to 1 when there is an illegal VD input from an external module. This bit is set to 1 when the number of HD cycles for the VD input to the CEU differs from the value set in the VCYL bits in CMCYR. Note however that when the VCYL bits are cleared to 0, this interrupt does not occur.

Bit	Bit Name	Initial Value	R/W	Description
17	IGHS	0	R/W	This bit is used for an interrupt generated when the number of HD cycles set in CMCYR differ from the number of HD cycles input from an external module. This bit is set to 1 when there is an illegal HD input from an external module. This bit is set to 1 when the number of clock cycles for the HD input to the CEU differs from the value set in the HCYL bits in CMCYR. Note however that when the HCYL bits are cleared to 0, this interrupt does not occur.
16	CDTOF	0	R/W	This bit is used for an interrupt indicating that data overflowed in the CRAM of the write buffer. Since data is input at realtime from an external module in capture operations, the frame image is overwritten unless the captured data is transferred from the CEU internal buffer to the memory at a certain or higher transfer rate. This bit is set to 1 when writing the data in the CRAM of the CEU internal write buffer to the bus is not performed within time and data has overflowed.
15	CPBE4	0	R/W	This bit is used for an interrupt indicating that writing to CDBYR2 and CDBCR2 in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines set in CBDSR has been captured and the last data transfer to the bus has completed. However, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).
14	CPBE3	0	R/W	This bit is used for an interrupt indicating that writing to CDBYR and CDBCR in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines set in CBDSR has been captured and the last data transfer to the bus has completed. However, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).

Bit	Bit Name	Initial Value	R/W	Description
13	CPBE2	0	R/W	This bit is used for an interrupt indicating that writing to CDAYR2 and CDACR2 in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines (number of bytes in data enable fetch) set in CBDSR has been captured and the last data transfer to the bus has completed. However, in image capture or data synchronous fetch, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).
12	CPBE1	0	R/W	This bit is used for an interrupt indicating that writing to CDAYR and CDACR in a bundle write has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input. This bit is set to 1 when data for the number of lines (number of bytes in data enable fetch) set in CBDSR has been captured and the last data transfer to the bus has completed. However, in image capture or data synchronous fetch, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	VD	Undefined	R/W	This bit is used for an interrupt indicating that VD (vertical sync signal) was input from an external module. This bit is set to 1 when a VD input from an external module is detected. Immediately after the VDPOL bit in CAMCR is modified, a pseudo VD is input and this bit is set to 1. The VD interrupt after the VDPOL bit is modified should be ignored.

Bit	Bit Name	Initial Value	R/W	Description
8	HD	Undefined	R/W	This bit is used for an interrupt indicating that HD (horizontal sync signal) was input from an external module. This bit is set to 1 when an HD input from an external module is detected. Immediately after the HDPOL bit in CAMCR is modified, a pseudo HD is input and this bit is set to 1. The HD interrupt after the HDPOL bit is modified should be ignored.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	IGRW	0	R/W	This bit is used for an interrupt indicating that during capturing, access was attempted to a register to which writing during operation is prohibited. Among the CEU registers, writing during capturing is prohibited for some registers. Table 34.11 shows which registers can/cannot be written to during capturing. This bit is set to 1 when a register to which writing during capturing is prohibited has been written to.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	CFE	0	R/W	This bit is used for an interrupt indicating that capturing of one field from an external module has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next VD input (see Figure 34.47). This interrupt occurs only in both-field capture mode.
0	CPE	0	R/W	This bit is used for an interrupt indicating that capturing of one frame from an external module has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next VD input. This interrupt indicates that capturing of one frame has finished. This bit is set to 1 when the image of the size set in CAPWR is captured and the last data transfer to the bus finished (see Figure 34.48).

Table 34.11 Registers that Can/Cannot Be Modified during Capturing

Register Name	Modification during Capturing	Register Name	Modification during Capturing
CAPSR	Possible	CDBCR	Possible
CAPCR	Prohibited	CBDSR	Possible
CAMCR	Prohibited	CFWCR	Possible
CMCYR	Prohibited	CLFCR	Possible
CAMOR	Possible	CDOCR	Possible
CAPWR	Possible	CDDCR	Possible
CAIFR	Prohibited	CDDAR	Possible
CRCNTR	Possible	CEIER	Possible
CRCMPR	Prohibited	CETCR	Possible
CFLCR	Possible	CSTSR	Prohibited
CFSZR	Possible	CSRTR	Possible
CDWDR	Possible	CDSSR	Prohibited
CDAYR	Possible	CDAYR2	Possible
CDACR	Possible	CDACR2	Possible
CDBYR	Possible	CDBYR2	Possible
		CDBCR2	Possible

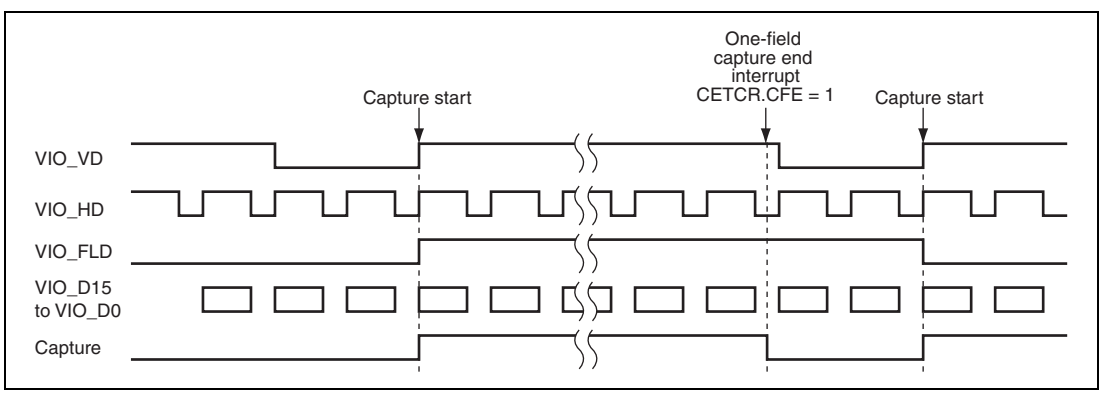


Figure 34.47 CFE Generation Timing

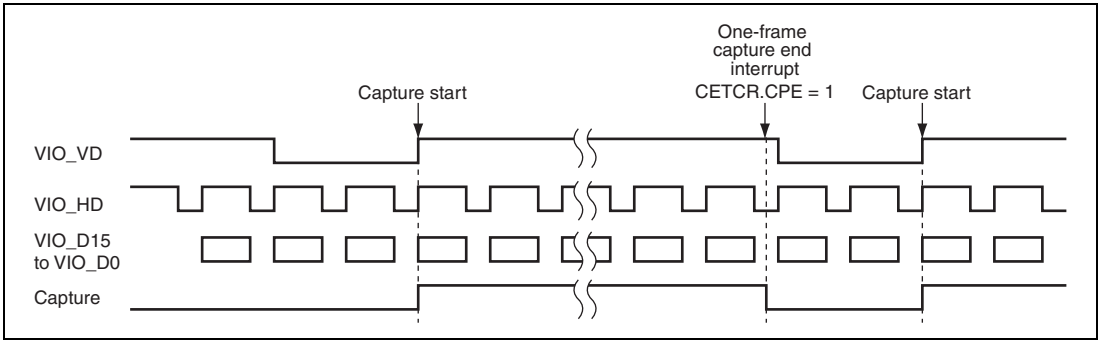


Figure 34.48 CPE Generation Timing

34.4.25 Capture Status Register (CSTSR)

CSTSR indicates the internal status of the CEU. CSTSR differs from CETCR in that no interrupt is generated for the events indicated in CSTSR.

The CEU operating/halt state can be confirmed using CSTSR. To confirm the halt state of the CEU, make sure that all status bits (bits 9, 8, and 0) indicating that the CEU is operating are cleared to 0 for sure.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CRST	—	—	—	—	—	—	—	CP FLD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CP REQ	CP ACK	—	—	—	—	—	—	—	CP TON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	CRST	0	R	Indicates which register plane is currently used. 0: Plane A of the register is being used 1: Plane B of the register is being used

Bit	Bit Name	Initial Value	R/W	Description
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CPFLD	0	R	Indicates which field is being captured. 0: Bottom field is being captured 1: Top field is being captured
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	CPREQ	0	R	Indicates the request output port level of the bridge interface.
8	CPACK	0	R	Indicates the acknowledge input port level of the bridge interface.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CPTON	0	R	Indicates that the CEU is operating. This bit retains 1 during the period that starts from the internal VD at capture start and ends when a one-frame capture end interrupt occurs. Figure 34.49 shows the CEU operating period.

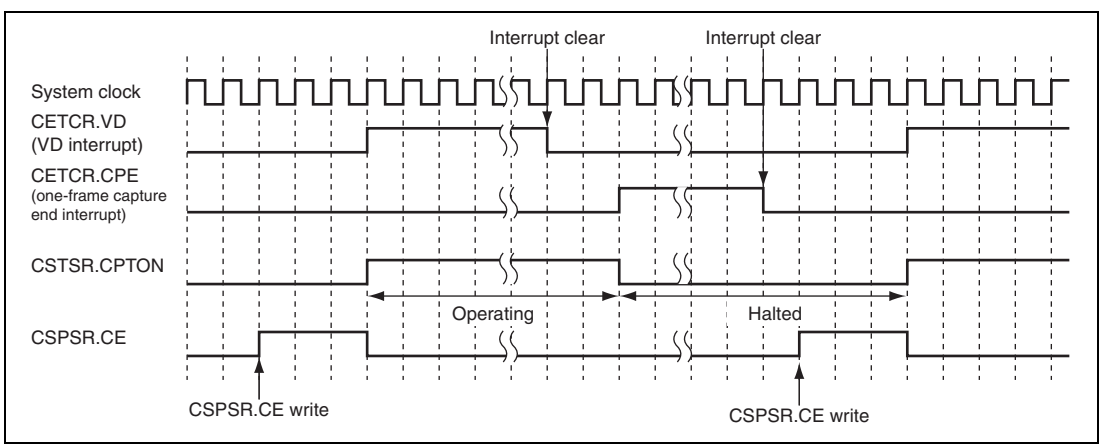


Figure 34.49 Operating Status during Capturing

34.4.26 Capture Software Reset Register (CSRTR)

CSRTR executes a module reset of the CEU internal circuit. This reset processing that does not take the bus state in account should not be used in the normal state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ALL RST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ALLRST	0	R/W	Resets the entire CEU and request signals to the bus. 0: Normal state 1: Performs a module reset of the entire sync circuit of the CEU

Note: Do not modify the ALLRST bit except for in special cases, such as when the entire system hangs. If the ALLRST bit is written to while the CEU is operating, a fatal error may occur; the system may get out of control, or memory areas other than the specified area may be overwritten.

34.4.27 Capture Data Size Register (CDSSR)

CDSSR indicates the size of data written to the memory in data enable fetch. As this register indicates a correct value at the end of capture, confirm this register when capture is completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDSS[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDSS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDSS[31:0]	H'0000 0000	R	Indicate the size of data written to the memory in data enable fetch. In a bundle write, size of data written to the selected address at the end of one-frame capture is indicated. In a bundle write, as soon as the number of bytes specified by CBDSR is transferred to the bus, address to which data is written is switched. Therefore, if one-frame capture is completed at the same time as a bundle write is completed, this register indicates H'0000 0000. Figures 34.50 and 34.51 show the overall timing of the CDSSR operation in a bundle write.

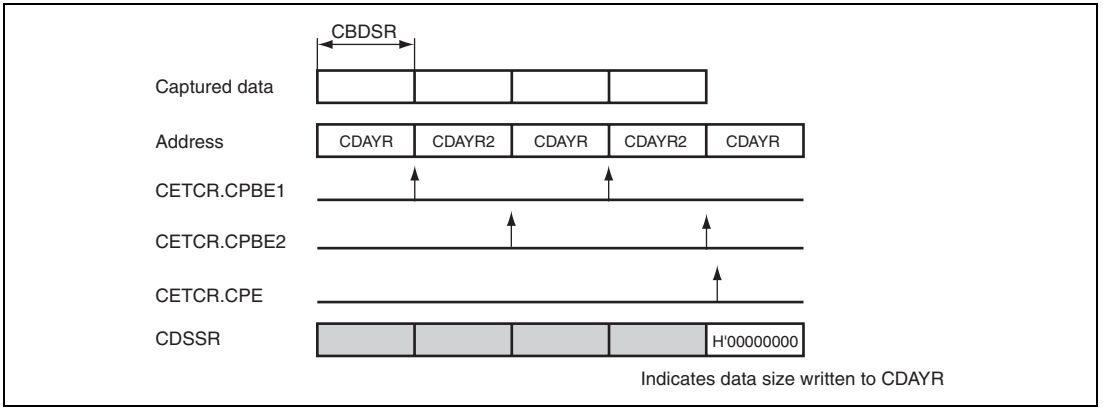


Figure 34.50 Overall Timing of CDSSR Operation in Bundle Write (When Bundle Write End and Capture End Coincide)

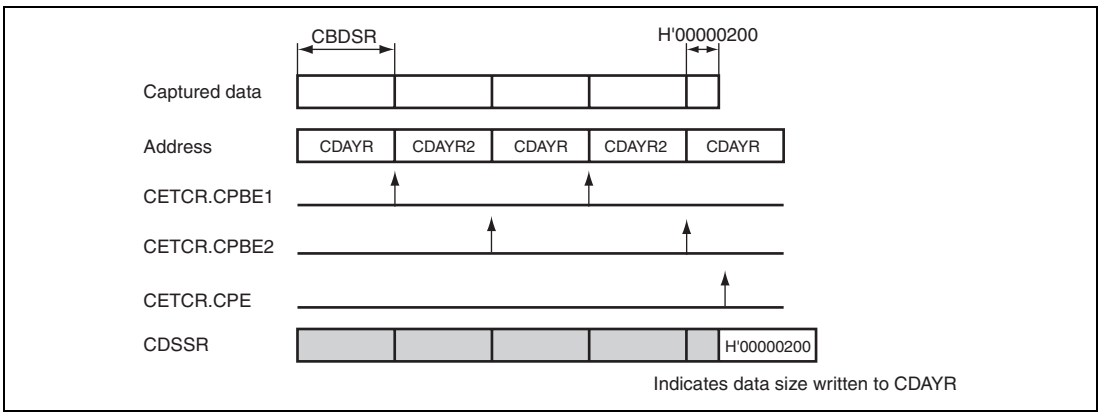


Figure 34.51 Overall Timing of CDSSR Operation in Bundle Write (When Bundle Write End and Capture End Does Not Coincide)

34.4.28 Capture Data Address Y Register 2 (CDAYR2)

CDAYR2 specifies the address for the luminance (Y) component used in a bundle write and the address for data storage in a bundle write in data fetch. CDAYR2 is used only in a bundle write.

CDAYR2 specifies the address where the Y component of the captured data is to be stored in frame image capture or one-field image capture, the address where the Y component of the captured top field is to be stored in both-field image capture, and the address where data is to be stored in data fetch. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the Y component of the captured data is to be stored by CDAYR2. In both-field image capture, set the start address of the memory area where the Y component of the captured top-field image is to be stored by CDAYR2. In data fetch, set the start address of the memory area to be used for data storage.

Because the address must be specified in 32 bits, the address set by CDAYR2 must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAYR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAYR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CAYR2[31:2]	H'0000 0000	R/W	• Frame image capture: These bits set the address for storing the Y component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the Y component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the Y component data of the captured top-field data (4-pixel units). • Data synchronous fetch: These bits set the address for storing data (4-byte units). • Dana enable fetch: These bits set the address for storing data (32-byte units).
1, 0	CAYR2[1:0]		R	

Set the address of the starting point of the memory area where the Y component of the captured image is to be stored in this register, as shown in Figure 34.52.

- Frame image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the Y component of the captured top-field image is to be stored.

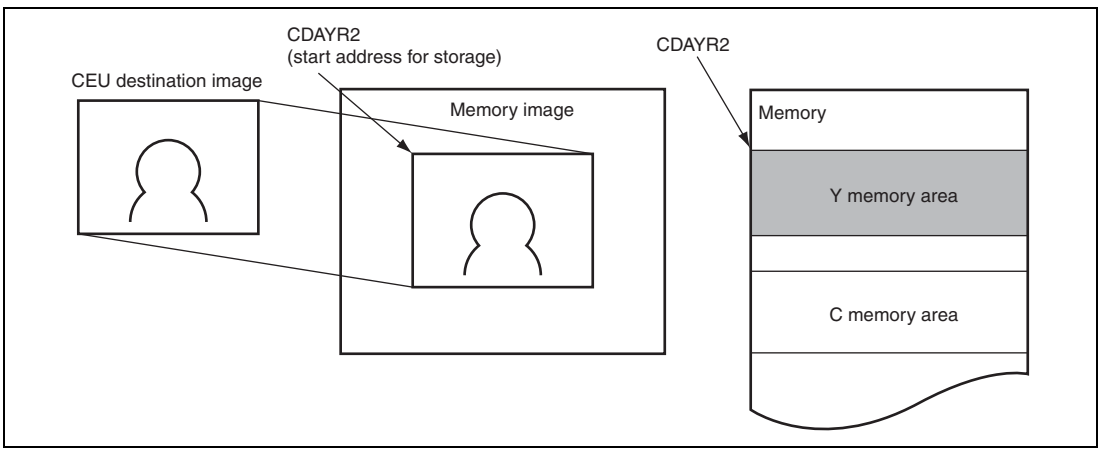


Figure 34.52 Relationship between Captured Image and Y Component Memory Area

34.4.29 Capture Data Address C Register 2 (CDACR2)

CDACR2 specifies the address for the chrominance (C) component used in a bundle write. CDACR2 is used only in a bundle write.

CDACR2 specifies the address where the C component of the captured data is to be stored in frame image capture or one-field image capture, and the address where the C component of the captured top field is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the C component of the captured data is to be stored by CDACR2. In both-field image capture, set the start address of the memory area where the C component of the captured top-field image is to be stored by CDACR2. CDACR2 is not used in data fetch.

Because the address must be specified in 32 bits, the address set by CDACR2 must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CACR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CACR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	CACR2[31:2] CACR2[1:0]	H'0000 0000	R/W R	• Frame image capture: These bits set the address for storing the C component data of the captured data (4-pixel units). • One-field image capture: These bits set the address for storing the C component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the C component data of the captured top-field data (4-pixel units).

In this register, set the address of the starting point of the memory area where the captured data is to be stored in a bundle write, as shown in Figure 34.53. The C component has an output data format as shown in Figure 34.54, and is saved in the memory in this format.

- Frame image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the C component of the captured top-field image is to be stored.

The C component has an output data format as shown in Figure 34.53, and is saved in the memory in this format.

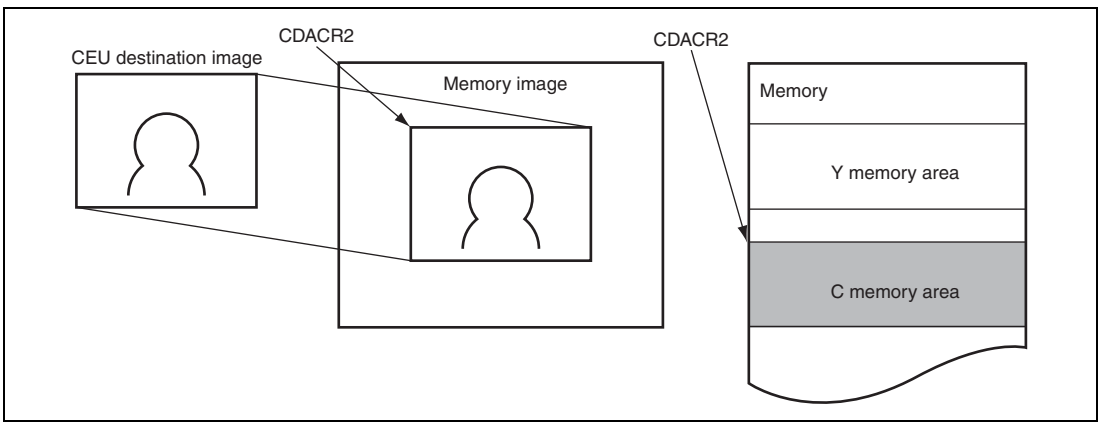


Figure 34.53 Relationship between Captured Image and C Component Memory Area

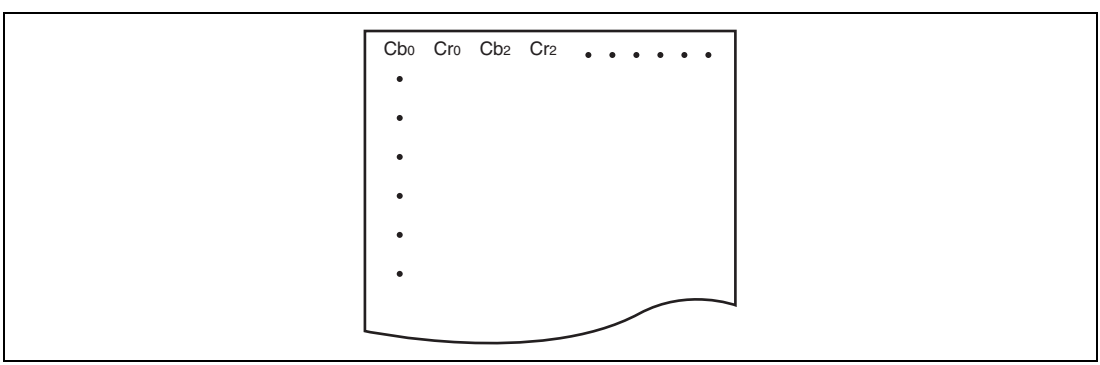


Figure 34.54 Image of Storing C Components in Memory

34.4.30 Capture Data Bottom-Field Address Y Register 2 (CDBYR2)

CDBYR2 specifies the address for the luminance (Y) component of the bottom field used in a bundle write. CDBYR2 is used only in a bundle write.

CDBYR2 specifies the address where the Y component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the Y component of the bottom-field image captured in both-field image capture is to be stored by CDBYR2. CDBYR2 is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBYR2 must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBYR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBYR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBYR2[31:2]	H'0000 0000	R/W	These bits set the address for storing the Y component data of the captured bottom-field data (4-pixel units).
1, 0	CBYR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the Y component of the captured bottom-field image is to be stored in a bundle write, as shown in Figure 34.55.

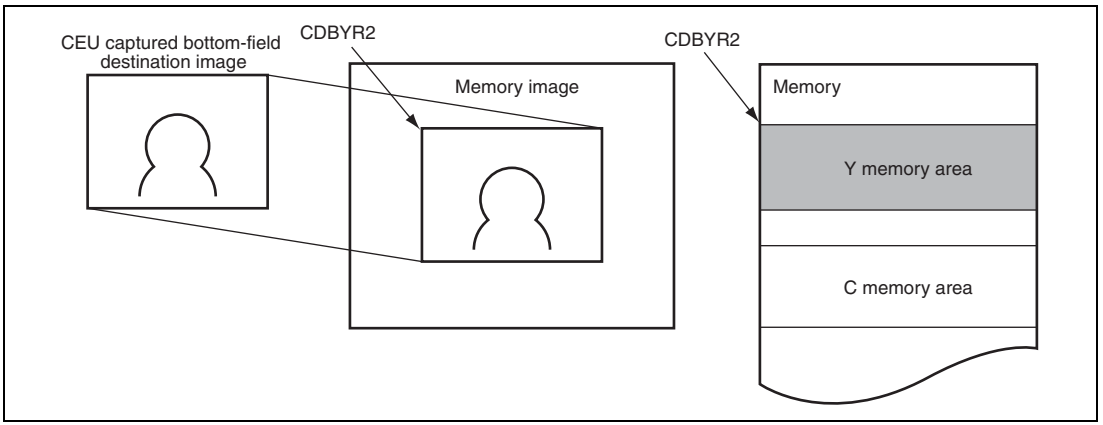


Figure 34.55 Relationship between Captured Bottom-Field Image and Y Component Memory Area

34.4.31 Capture Data Bottom-Field Address C Register 2 (CDBCR2)

CDBCR2 specifies the address for the chrominance (C) component of the bottom field used in a bundle write. CDBCR2 is used only in a bundle write.

CDBCR2 specifies the address where the C component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the C component of the bottom-field image captured in both-field image capture is to be stored by CDBCR2. CDBCR2 is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBCR2 must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CBCR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBCR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBCR2[31:2]	H'0000 0000	R/W	These bits set the address for storing the C component data of the captured bottom-field data (4-pixel units).
1, 0	CBCR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the C component of the captured bottom-field image is to be stored in a bundle write, as shown in Figure 34.56. The C component has an output data format as shown in Figure 34.57, and is saved in the memory in this format.

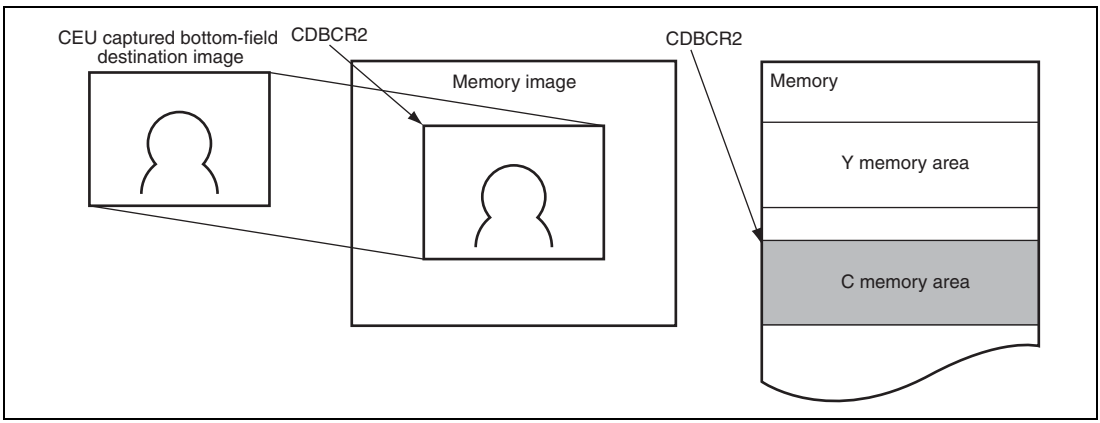


Figure 34.56 Relationship between Captured Bottom-Field Image and C Component Memory Area

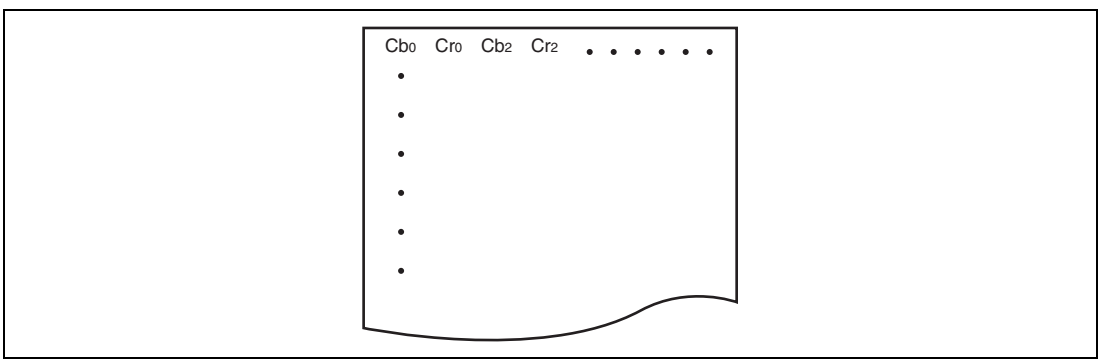


Figure 34.57 Image of Storing C Components in Memory

34.5 Usage Notes for CEU

34.5.1 Conditions for Connection to an External Module

(1) Clock Frequency

The external input clock should have a frequency at least the same as the CEU operating clock frequency (Bϕ), with jitter on both sides included.

CEU operating clock frequency ≥ External input clock frequency

(2) Blanking Period

The period from the last valid pixel in each line to the next horizontal sync signal HD must be at least 20 cycles.

(3) Fixed Period of Field Identification Signal

The field identification signal FLD should be fixed for at least 1-HD period since a VD input

34.5.2 Restrictions on Input/Output Functions

Table 34.12 lists the restrictions regarding the CEU input/output functions.

Table 34.12 Restrictions on CEU Input/Output Functions

Item	Restrictions
External module interface	The operating clock of the external module should always have a frequency at most the same as that of the CEU operating clock, with jitter on both sides included.
	Selecting the interface, or modifying the frequency of the external module operating clock or HD/VD polarity must be done when capture operations are halted for sure.
	The capture horizontal size in image capture must be specified as follows: 8-bit interface: 8-cycle units 16-bit interface: 4-cycle units
	The capture horizontal size in data fetch must be specified as follows: 8-bit interface: 4-cycle units 16-bit interface: 2-cycle units

Item	Restrictions
External module interface	The capture vertical size must be specified in 4-line units.
	The maximum number of cycles in the horizontal sync signal period should be 16,383 external input clock cycles.
	The maximum number of lines (HD count) in the vertical sync signal should be 16,383 lines.
	The minimum number of captured pixels should be sub-QCIF (128 × 96).
	The maximum number of captured pixels should be 5M (2560 × 1920) pixels.
	Captured size in data enable fetch Maximum: 6 Mbytes (2048 × 1536 × 2) Minimum: 16 bytes (must be specified in 4-byte units)
Memory output	The output address must be specified in longword units.
	The horizontal size of the destination image (memory) must be specified in 4-pixel units.
	The number of horizontal output pixels (= horizontal clipping size) must be specified in 4-pixel units.
	The number of vertical output lines (HD) (= vertical clipping size) must be specified in 4-line (HD) units.
	In fixed address mode, the lower five bits in an output address must be 0, and the output size must be specified in 8-pixel (8-byte) units.
	In data enable fetch bundle write, the output address must be specified in 32-byte units.
Internal processing	The filter clipping size must be specified as a value equal to or lower than the actual output size of the filter.

Section 35 Video Engine Unit (VEU)

The video engine unit (VEU) is a module that processes an image read from memory. The image processing functions include YCbCr and RGB format conversion, scaling, and tone reduction. This LSI has two VEU modules (VEU0 and VEU1).

35.1 Features

- Format conversion using the RGB ⇔ YCbCr conversion function
- Scaling of an image using the filter function
- Tone reduction (quantization) to pack RGB data in 32-bit units
- Dithering for tone reduction of RGB data
- Removal of high-frequency components using the low-pass filter function
- Low-pass filter is applied to only the boundary of the blocks using the deblocking filter function
- Median filter function
- Edge enhancement of an image (enhancer function)

35.2 Functional Overview of VEU

The functional overview of the VEU is shown in Table 35.1. Some functions of the VEU cannot be used at the same time unless the VEU is re-activated. Table 35.2 shows which functions can/cannot be used simultaneously during one VEU activation.

Note that when the instructions at other places in this manual differ from instructions in the following table, the instructions in section 35.4, Usage Notes for VEU, are given priority.

Table 35.1 Functional Overview of VEU

Item	Function	Description	Note
Input format	YCbCr (4:4:4/4:2:2/4:2:0) RGB pack		
Output format	YCbCr (4:4:4/4:2:2/4:2:0) RGB pack		
Read mode	Normal read Bundle read	8 to 960 lines can be set as the number of lines in bundle read	
Low-pass filter	Removal of high-frequency components		
Deblocking filter	Removal of high-frequency components only at the boundary of blocks		
Enhancer	Enhancement of image		
Median filter	Removal of shot noise		
Rotation/ inversion of image	Vertical or horizontal inversion	Can be specified independently	A combination of both functions can realize rotation by 180 degrees
	Rotation by 90 or 270 degrees	Rotated clockwise	
Scale-up, scale-down, or no scaling	Scale-up or scale-down of memory display	Any scaling factor from × 1/16 to × 16	
Format conversion	YCbCr ⇔ RGB conversion	Bidirectional conversion between YCbCr format and RGB format	

Item	Function	Description	Note
Dithering (tone reduction)	24 bpp	Full colors (16,700,000 colors)	Dithering not possible
	18 bpp	260,000 colors	Dithering not possible
	16 bpp	High colors (65,536 colors)	
	12 bpp	4,096 colors	
	8 bpp	256 colors	
Maximum image size	16M pixels	4092 pixels × 4092 lines	
Minimum image size	16 × 16 pixels	16 pixels × 16 lines	

Note: The scaling factor of the filter can be set between 1/16 and 16. For details, see section 35.3.11, VEU Resize Filter Control Register (VRFCR), and section 35.3.12, VEU Resize Filter Size Clip Register (VRFSR).

Table 35.2 Simultaneous Usage of Functions in One VEU Activation

	Bundle Mode	Color Conversion	Low-Pass Filter	Deblocking Filter	Median Filter	Enhancer	Vertical/ Horizontal Inversion	90°/270° Rotation	Scale-Up/ Scale-Down
Bundle Mode	—	O	x	x	x	O	O	x	O
Color Conversion	O	—	O	O	O	O	O	O	O
Low-Pass Filter	x	O	—	x	x	x	O	O	x
Deblocking Filter	x	O	x	—	O*	x	O	O	x
Median Filter	x	O	x	x*	—	x	O	O	x
Enhancer	O	O	x	x	x	—	O	x	O
Vertical/ Horizontal Inversion	O	O	O	O	O	O	—	O	O
90°/270° Rotation	x	O	O	O	O	x	O	—	x
Scale-Up/ Scale-Down	O	O	x	x	x	O	O	x	—

[Legend]

O: Possible

x: Not possible

Notes: The register values corresponding to the items in this table are shown below.

Bundle mode:	VESTR.VBE
Color conversion:	VTRCR.TE
Low-pass filter:	VFMCR.LPHV
Deblocking filter:	VFMCR.LPHV && VFMCR.DBLK
Median filter:	VFMCR.MED
Enhancer:	VENHR.ENHH || VENHR.ENHV
Vertical/horizontal inversion:	VFMCR.VMRR/VFMCR.HMRR
90°/270° rotation:	VFMCR.POTR/VFMCR.POTL
Scale-up/scale-down:	VRFCR ! = 0

* The order of deblocking filter → median filter is allowed. However, a reversed order is not allowed.

Figure 35.1 shows a block diagram of the VEU.

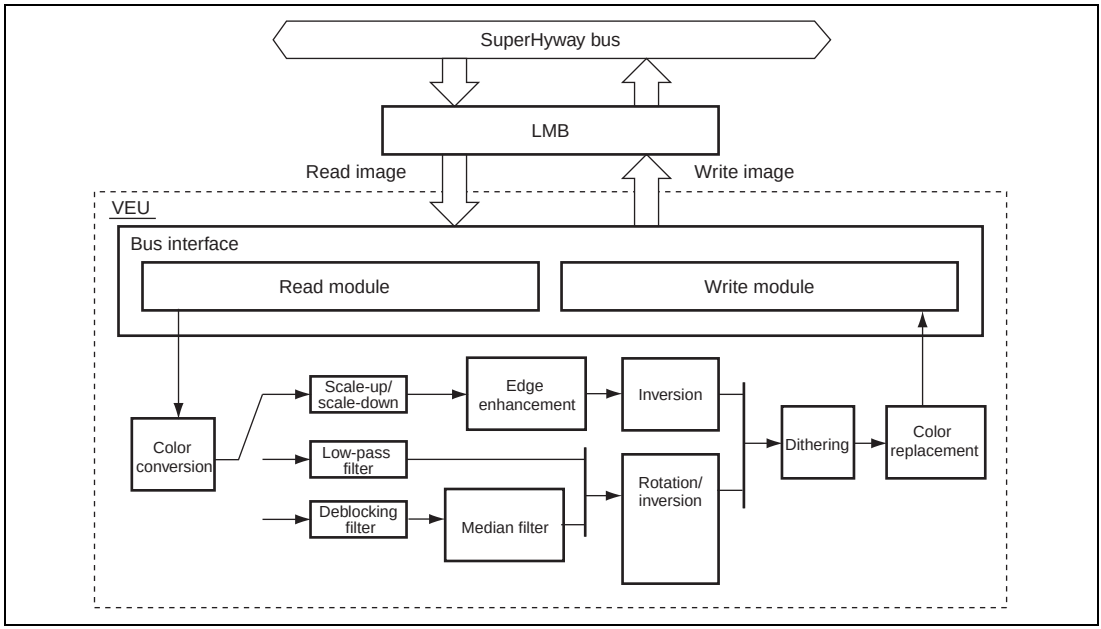


Figure 35.1 Block Diagram of VEU

Note: In Figure 35.1, inversion indicates horizontal or vertical inversion, and rotation indicates 90° or 270° rotation.
In the inversion circuit after scale-up/scale-down, horizontal inversion, vertical inversion, or 180° rotation, which is a combination of the two types of inversion, can be performed.
In the rotation/inversion circuit after the median filter, in addition to the possible operations in the inversion circuit after scale-up/scale-down, 90° rotation, 270° rotation, a combination of 90° rotation and horizontal inversion, or a combination of 90° rotation and vertical inversion can be performed.

- Operation

  Figure 35.2 shows the sequence of the VEU operation processing. Though the VEU has various functions, some functions cannot be used simultaneously during one VEU activation, as shown in Table 35.2. Either flow 1 or flow 2 shown in Figure 35.2 is possible. Accordingly, processing in different flows cannot be performed together.

  1. Flow 2 is selected when the following condition is satisfied.

     VESTR.VBE || (VENHR.ENHH || VENHR.ENHV) || (VRFCR != 0) and VFMCE.FLTPI = 0

  2. Flow 1 is selected in cases other than above.

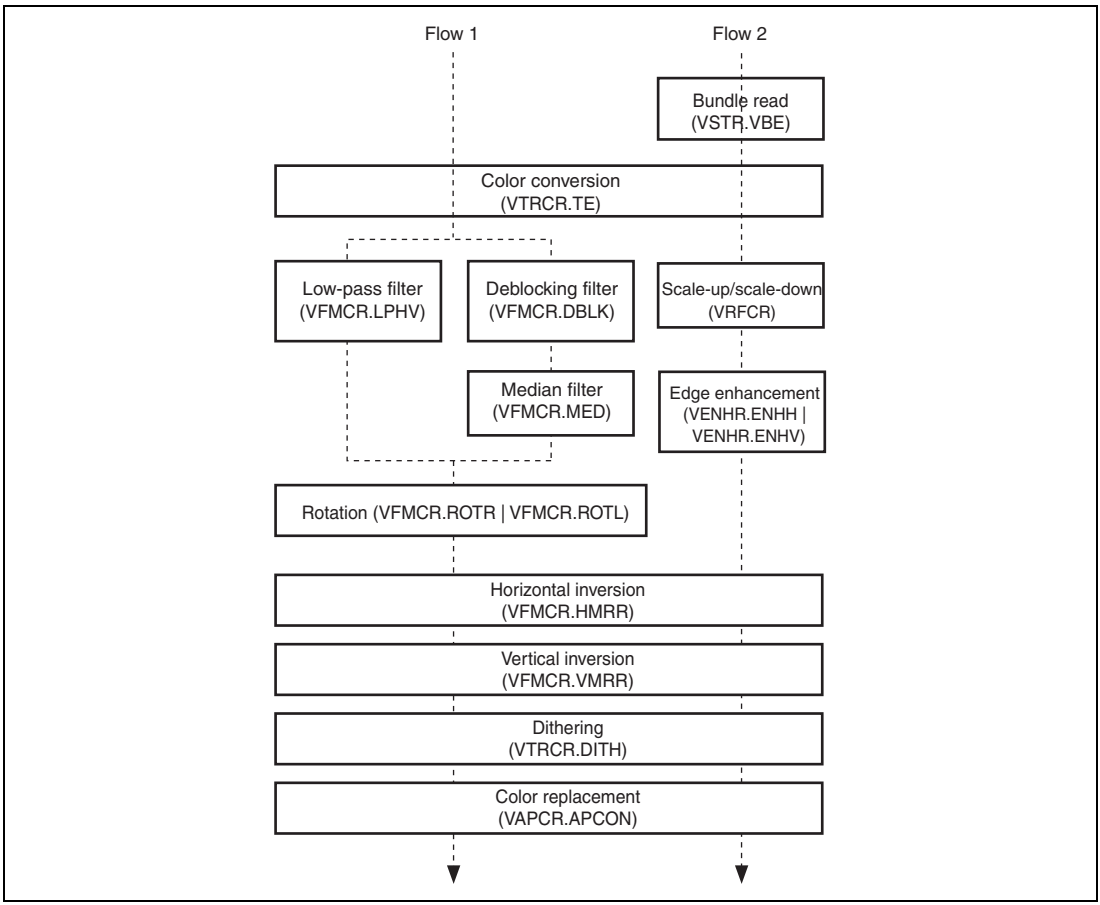


Figure 35.2 Register Settings, Processing Flows, and Simultaneously Executable Functions of VEU

35.3 Register Descriptions of VEU

(1) R/W Restrictions on VEU Registers

The read/write restrictions on the VEU registers are listed below. If the following register handling is not guaranteed, a malfunction may occur.

1. For the read-only bits in all VEU registers and the reserved areas, writing 1 is prohibited. Do not specify an unspecified value.
2. Though a value other than 0 may be read from a read-only bit, do not modify this bit.
3. For registers (bits) to which writing during operation is prohibited, do not modify them during operation (reading is possible). The VEU operating state can be confirmed by reading the VE bit in the VEU start register (VESTR). Modify the above registers when this bit is 0.

(2) Terms and Abbreviations Used in This Section

The terms used in this section are described below.

1. "software reset" indicates that the VEU processing was halted and the current processing is stopped. The image processing result of the frame in which a software reset occurs is not guaranteed.
2. "module reset" indicates the VEU internal circuit was forcibly reset. In a module reset, reset processing is performed with no consideration to the hardware state. Therefore, if a module reset is generated when the VEU is operating normally, the VEU peripheral hardware may become unable to operate.
3. "during operation" in this section indicates a state in which the VE bit in the VEU start register (VESTR) is set to 1.
4. When a bit name in a register is referred to in this section, it is indicated in the format of (register name).(bit name).

   Example: VESTR.VE

(3) List of Registers

The register configuration of the VEU is shown in Table 35.3. The register states in each processing mode are shown in Table 35.4.

Table 35.3 Register Configuration of VEU

Register Name	Abbr.	R/W	Address	Access Size
VEU0 start register	VESTR_0	R/W	H'FE92 0000	32
VEU0 source memory width register	VESWR_0	R/W	H'FE92 0010	32
VEU0 source size register	VESSR_0	R/W	H'FE92 0014	32
VEU0 source address Y register	VSAYR_0	R/W	H'FE92 0018	32
VEU0 source address C register	VSACR_0	R/W	H'FE92 001C	32
VEU0 bundle source size register	VBSSR_0	R/W	H'FE92 0020	32
VEU0 destination memory width register	VEDWR_0	R/W	H'FE92 0030	32
VEU0 destination address Y register	VDAYR_0	R/W	H'FE92 0034	32
VEU0 destination address C register	VDACR_0	R/W	H'FE92 0038	32
VEU0 transform control register	VTRCR_0	R/W	H'FE92 0050	32
VEU0 resize filter control register	VRFCR_0	R/W	H'FE92 0054	32
VEU0 resize filter size clip register	VRFSR_0	R/W	H'FE92 0058	32
VEU0 enhance register	VENHR_0	R/W	H'FE92 005C	32
VEU0 resize filter sub control register	VRSCR_0	R/W	H'FE92 0064	32
VEU0 resize filter size clip offset register	VRSOR_0	R/W	H'FE92 0068	32
VEU0 filter mode control register	VFMCR_0	R/W	H'FE92 0070	32
VEU0 vertical tap coefficient register	VVTCR_0	R/W	H'FE92 0074	32
VEU0 horizontal tap coefficient register	VHTCR_0	R/W	H'FE92 0078	32
VEU0 designated color register	VAPCR_0	R/W	H'FE92 0080	32
VEU0 conversion color register	VECCR_0	R/W	H'FE92 0084	32
VEU0 fill color specification register	VFLCR_0	RW	H'FE92 0088	32
VEU0 address fixed register	VAFXR_0	R/W	H'FE92 0090	32
VEU0 swapping register	VSWPR_0	R/W	H'FE92 0094	32
VEU0 event interrupt enable register	VEIER_0	R/W	H'FE92 00A0	32
VEU0 event register	VEVTR_0	R/W	H'FE92 00A4	32
VEU0 status register	VSTAR_0	R	H'FE92 00B0	32
VEU0 module reset register	VBSRR_0	R/W	H'FE92 00B4	32
VEU0 resize passband register	VRPBR_0	R/W	H'FE92 00C8	32

Register Name	Abbr.	R/W	Address	Access Size
VEU1 start register	VESTR_1	R/W	H'FE92 4000	32
VEU1 source memory width register	VESWR_1	R/W	H'FE92 4010	32
VEU1 source size register	VESSR_1	R/W	H'FE92 4014	32
VEU1 source address Y register	VSAYR_1	R/W	H'FE92 4018	32
VEU1 source address C register	VSACR_1	R/W	H'FE92 401C	32
VEU1 bundle source size register	VBSSR_1	R/W	H'FE92 4020	32
VEU1 destination memory width register	VEDWR_1	R/W	H'FE92 4030	32
VEU1 destination address Y register	VDAYR_1	R/W	H'FE92 4034	32
VEU1 destination address C register	VDACR_1	R/W	H'FE92 4038	32
VEU1 transform control register	VTRCR_1	R/W	H'FE92 4050	32
VEU1 resize filter control register	VRFCR_1	R/W	H'FE92 4054	32
VEU1 resize filter size clip register	VRFSR_1	R/W	H'FE92 4058	32
VEU1 enhance register	VENHR_1	R/W	H'FE92 405C	32
VEU1 resize filter sub control register	VRSCR_1	R/W	H'FE92 4064	32
VEU1 resize filter size clip offset register	VRSOR_1	R/W	H'FE92 4068	32
VEU1 filter mode control register	VFMCR_1	R/W	H'FE92 4070	32
VEU1 vertical tap coefficient register	VVTCR_1	R/W	H'FE92 4074	32
VEU1 horizontal tap coefficient register	VHTCR_1	R/W	H'FE92 4078	32
VEU1 designated color register	VAPCR_1	R/W	H'FE92 4080	32
VEU1 conversion color register	VECCR_1	R/W	H'FE92 4084	32
VEU1 fill color specification register	VFLCR_1	RW	H'FE92 4088	32
VEU1 address fixed register	VAFXR_1	R/W	H'FE92 4090	32
VEU1 swapping register	VSWPR_1	R/W	H'FE92 4094	32
VEU1 event interrupt enable register	VEIER_1	R/W	H'FE92 40A0	32
VEU1 event register	VEVTR_1	R/W	H'FE92 40A4	32
VEU1 status register	VSTAR_1	R	H'FE92 40B0	32
VEU1 module reset register	VBSRR_1	R/W	H'FE92 40B4	32
VEU1 resize passband register	VRPBR_1	R/W	H'FE92 40C8	32

Table 35.4 VEU Register States in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
VESTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VESWR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VESSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VSAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VSACR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VBSSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VEDWR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VDAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VDACR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VTRCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VRFCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VRFSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VENHR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VFMCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VVTCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VHTCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VAPCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VECCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VFLCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VAFXR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VSWPR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VEIER_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VEVTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VSTAR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VBSRR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
VRPBR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Note: n = 0, 1

35.3.1 VEU Start Register (VESTR)

VESTR controls activation of the VEU and software reset of VEU processing. Before starting VEU processing by VESTR, all registers related to the VEU must be set.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VBE	—	—	—	—	—	—	—	VE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBE	0	R/W	Sets the bundle read mode (N-line read mode). In bundle read mode (N-line read mode), the VEU divides a one-frame image at the number of lines set by VBSSR and reads data for that number of lines. When the VEU is activated in bundle read mode (N-line read mode), rotation processing, deblocking filter processing, low-pass filter processing, and median filter processing cannot be executed (horizontal inversion, vertical inversion, and horizontal-vertical inversion are possible). 0: Normal operation 1: Bundle read (N-line read) mode
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	VE	0	R/W	Controls the VEU processing start and software reset of processing (processing halt). If this bit is set to 1 when the VEU is halted, the VEU starts processing images in the memory according to the register settings, and this bit is read as 1. In addition, if 0 is written to this bit when it is 1, the current VEU processing is immediately terminated. The end of software reset processing can be confirmed when this bit becomes 0 after a software reset has been issued to the VEU. Confirm that this bit is 0 before restarting the VEU after a software reset. 0: NOP (software reset processing when 0 is written to this bit when it is 1) 1: VEU processing starts

If B'1 is written to the VE bit when the VBE bit is B'1 at VEU activation, bundle read mode (N-line read mode) is selected. In bundle read mode, data is read from two memory areas alternately for the number of lines set by the VBSS bits in VBSSR. If reading for the entire frame is not finished when the VEU has completed reading for the number of lines set by VBSSR, a bundle end interrupt (VEVTR.VBEND) occurs and the VEU waits for reading to be restarted. To restart reading, first clear the bundle end interrupt source, and then write B'1 to the VE bit again (write with the VBE bit still set to B'1). After repeating this until reading for the entire frame finishes, a one-frame end interrupt (VEVTR.VEEND) occurs. The image of processing in bundle read mode is shown in Figure 35.5. The status of normal image processing, read processing in bundle read mode, and waiting of read to be restarted in bundle read mode can be confirmed by reading VSTAR.

Notes on software reset:

1. Though the VEU end interrupt flag (VEVTR.VEEND) may become B'1 depending on the timing to issue a VEU software reset, the processed image of the frame in which a software reset was issued is not guaranteed.
2. Even when the VEEND flag does not become B'1 after a VEU software reset, the VEEND flag must always be cleared to B'0.
3. When issuing a software reset in bundle read mode, issue it before reading is restarted subsequent to a bundle end interrupt.

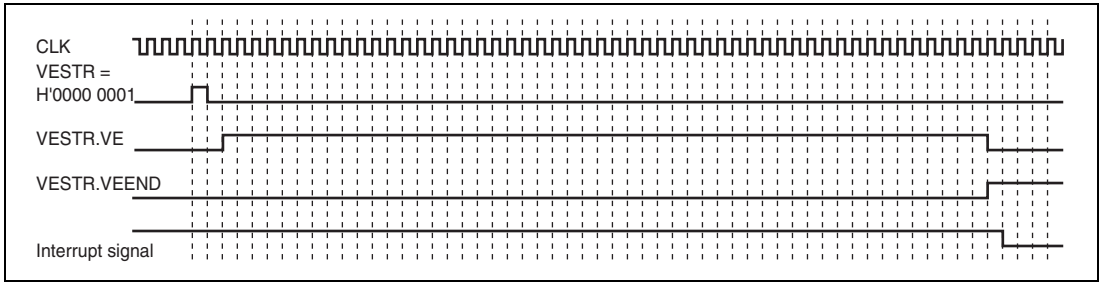


Figure 35.3 Operation Timing of VE Bit and Each Status

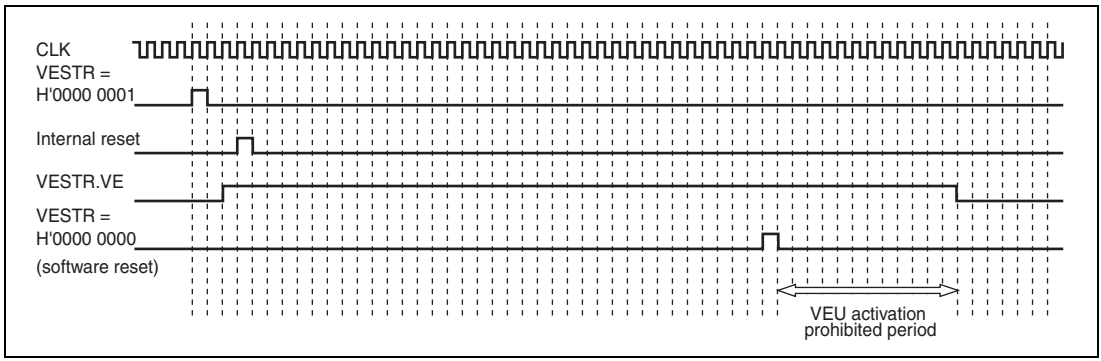


Figure 35.4 Operation Timing of Software Reset and Each Status

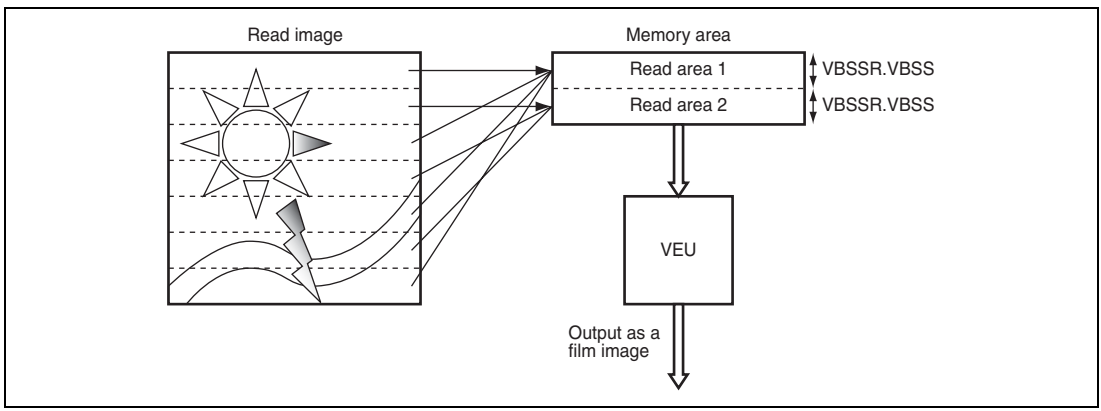
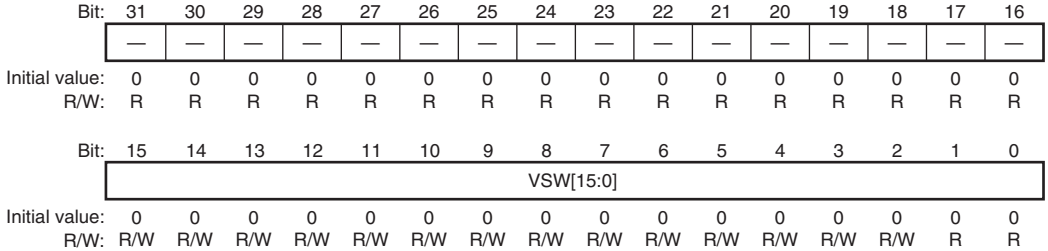


Figure 35.5 Image of Bundle Read Mode Processing

35.3.2 VEU Source Memory Width Register (VESWR)

VESWR sets the memory width of the source memory area of the VEU.

Modifying this register during operation is prohibited.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2 1, 0	VSW[15:2] VSW[1:0]	H'0000	R/W R	These bits set the width of the source memory area of the VEU (4-pixel units). This setting should be made in 2-pixel units when the RPKF bits in VTRCR are set to 0,1,3,7, or 13.

In the VSW bits, set the horizontal width of the source memory area where the source image is located in the VEU processing as a number of bytes. This setting must be made in byte units that correspond to four pixels of the source image. If the read image is in the YCbCr format, make a setting matching the Y component. For reading data in the YCbCr 4:2:0, YCbCr 4:2:2, and YCbCr 4:4:4 formats, the VSW setting, VSW setting, and (VSW setting × 2) are applied respectively to the horizontal width of the source memory area for the C component. When the RPKF bits in VTRCR are set to 0,1,3,7, or 13 (RGB 2 bytes/pixel or RGB 4 bytes/pixel), a setting in byte units that corresponds to two pixels of the source image is possible. As shown in (A) in Figure 35.6, when an image is clipped from the memory area, the right-end address of a line's image is discontinuous from the left-end address of the next line's image. Compared to this, (B) in Figure 35.6 shows an example where the horizontal size of an image input to the VEU matches the horizontal length of the memory. In this case, the right-end address of a line's image and the left-end address of the next line's image have continuous values.

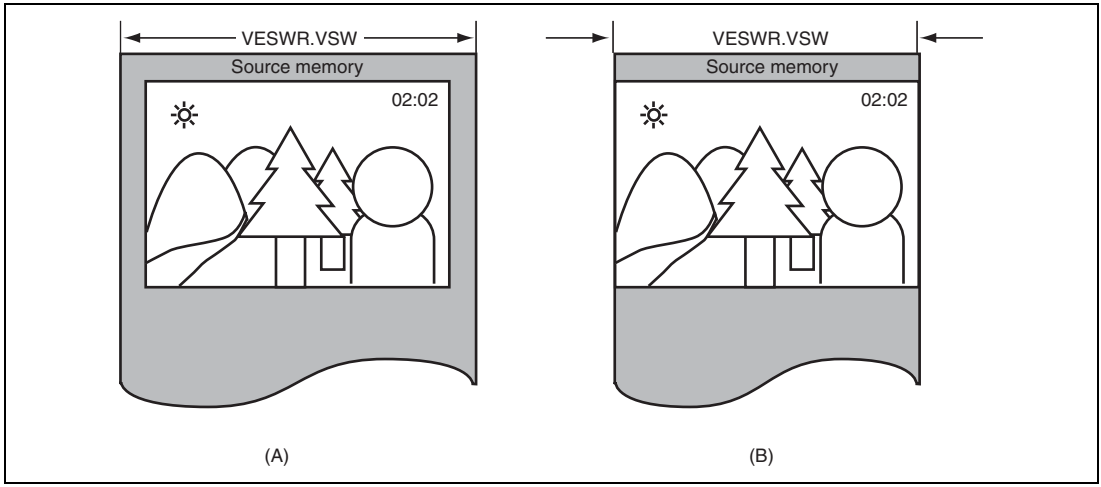


Figure 35.6 Relationship between Image Clipping Size and VESWR

35.3.3 VEU Source Size Register (VESSR)

VESSR sets the vertical and horizontal sizes (number of pixels) of the image read by the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VVSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VHSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	VVSS[11:0]	H'000	R/W	These bits set the number of vertical pixels read by the VEU. Set the number of vertical read pixels of the source image read by the VEU in these bits (see Figure 35.7). • Scale-up/down, enhancer, or bundle read is not performed (see Figure 35.7): A setting in 2-pixel units is possible for the following cases: Reading and writing are both in RGB format. RPKF bit in VTRCR is either 3 (RGB 2 bytes/pixel), 0, 1, 7, 13 (RGB 4 bytes/pixel) and WPKF bit in VTRCR is either 1, 2, 6 (RGB 2 bytes/pixel), 14, 19, 20, 22, 23 (RGB 4 bytes/pixel). Other than those settings, specify a setting in 4-pixel units. • Scale-up/down, enhancer, or bundle read is performed; a setting in 1-pixel unit is possible (see Figure 35.7). However, the data should be allocated in the memory to specify 2-pixel units when the input format is YCbCr4:2:0. • The maximum value to be set is 4092 pixels.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	VHSS[11:0]	H'000	R/W	These bits set the number of horizontal pixels read by the VEU. Set the number of horizontal read pixels of the source image read by the VEU in these bits (see Figure 35.7). • Scale-up/down, enhancer, or bundle read is not performed (see Figure 35.7): A setting in 2-pixel units is possible for the following cases: Reading and writing are both in RGB format. RPKF bit in VTRCR is either 3 (RGB 2 bytes/pixel), 0, 1, 7, 13 (RGB 4 bytes/pixel) and WPKF bit in VTRCR is either 1, 2, 6 (RGB 2 bytes/pixel), 14, 19, 20, 22, 23 (RGB 4 bytes/pixel). Other than those settings, specify a setting in 4-pixel units. • Scale-up/down, enhancer, or bundle read is performed; a setting in 1-pixel unit is possible (see Figure 35.7). However, set in the VESWR.VSW[15:0] a value (a value that VHSS[11:0] specify in 4-pixel units) × 4 / (P_density_y in use format) or more. Further, the data should be allocated in the memory to specify in 2-pixel units when the input format is YCbCr4:2:0. • The maximum value to be set is 4092 pixels.

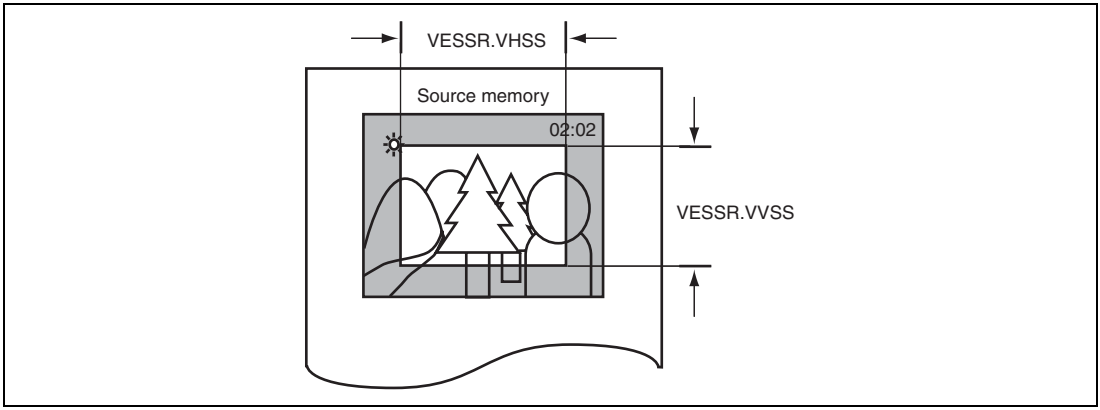


Figure 35.7 Relationship between Read Size and VESSR

35.3.4 VEU Source Address Y Register (VSAYR)

VSAYR sets the start address for the Y/RGB plane of the image read by the VEU.

Modifying this register during operation is prohibited. However, this register can be modified in the restart wait state (when VSTAR = H'0000 1001) in N-line read mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VSAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	VSAY[31:2]	H'0000 0000	R/W	These bits set the start address for reading the Y/RGB plane by the VEU (longword units).
1, 0	VSAY[1:0]		R	

Set the start address for the Y/RGB plane of the source image of the VEU in the VSAY bits, as shown in Figure 35.8. When reading a YCbCr image, specify the start address of the Y plane. When reading an RGB image, specify the start address (top-left address in the read image area) of the RGB plane. Figure 35.8 shows the input format for the Y plane of the YCbCr format. For the input format of the RGB format, refer to Table 35.8.

Note: In bundle read mode, set two addresses alternately for each VEVTR.VBEND interrupt.

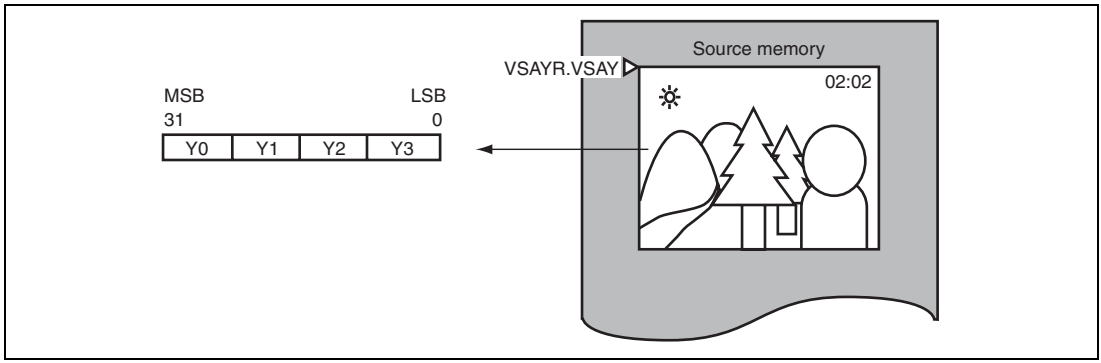


Figure 35.8 Address Set in VSAYR and Y-Plane Format

35.3.5 VEU Source Address C Register (VSACR)

VSACR sets the start address for the C plane of the image read by the VEU.

Modifying this register during operation is prohibited. However, this register can be modified in the restart wait state (when VSTAR = H'0000 1001) in N-line read mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VSAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	VSAC[31:2]	H'0000 0000	R/W	These bits set the start address for reading the C plane by the VEU. (1-longword units. 2-longword units for the YCbCr 4:4:4 format.)
1, 2	VSAC[1:0]		R	

Set the start address for the C plane of the source image of the VEU in the VSAC bits, as shown in Figure 35.9. When reading a YCbCr image, specify the start address (top-left address in the read image area) of the C plane. Figure 35.9 shows the input format for the C plane. When reading an RGB image, VSACR is not used. VSACR should be set in 1-longword units for the YCbCr 4:2:0 or YCbCr 4:2:2 format, and in 2-longword units for the YCbCr 4:4:4 format.

Note: In bundle read mode, set two addresses alternately for each VEVTR.VBEND interrupt.

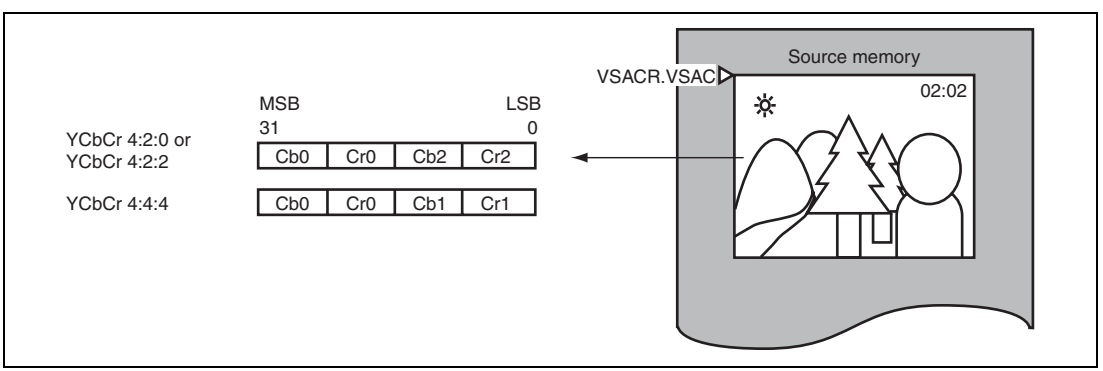


Figure 35.9 Address Set in VSACR and C-Plane Format

35.3.6 VEU Bundle Source Size Register (VBSSR)

VBSSR sets the number of lines (number of pixels) to be read when the VEU is activated once in bundle read mode (N-line read mode). VBSSR needs to be set only in bundle read mode.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VBSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 3	VBSS[11:3]	H'000	R/W	These bits set the number of lines to be read when the VEU is activated once in bundle read mode (N-line read mode) (8-line units).
2 to 0	VBSS[2:0]		R	

Set the number of lines of the source image to be read by the VEU in one processing in bundle read mode in the VBSS bits. The VBSS bits should be set in 8-line units, and the maximum value to be set is 960 lines.

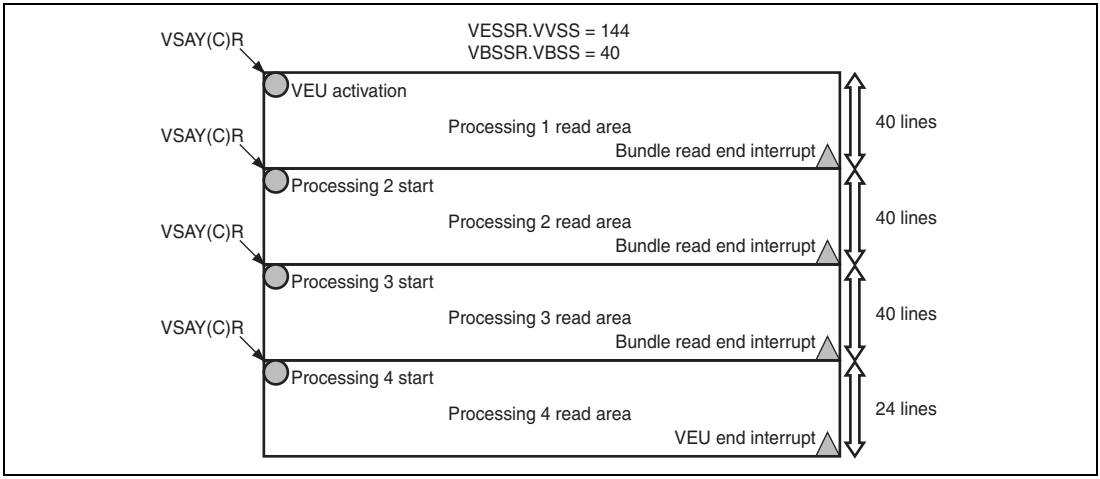


Figure 35.10 Relationship between Read Image in Bundle Read Mode and Read Image in Each Processing

35.3.7 VEU Destination Memory Width Register (VEDWR)

VEDWR sets the memory width of the destination memory area of the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2 1, 0	VDW[15:2] VDW[1:0]	H'0000	R/W R	These bits set the width of the destination memory area of the VEU (longword units).

Set the horizontal width of the destination memory area where the destination image is located as a number of bytes in the VDW bits. This setting must be made in byte units that correspond to

four pixels of the destination image. If the drawing image is in the YCbCr format, make a setting matching the Y component. For writing data in the YCbCr 4:2:0, YCbCr 4:2:2, and YCbCr 4:4:4 formats, the VDW setting, VDW setting, and (VDW setting × 2) are applied respectively to the horizontal width of the destination memory area for the C component. When the WPKF bits in VTRCR are set to 1, 2, 6 (RGB 2 bytes/pixel), 8 to 14, 19, 20, 22, 23 (RGB 4 bytes/pixel), a setting in byte units that corresponds to two pixels of the destination image is possible. As shown in (A) in Figure 35.11, when an image is pasted to the back display or the like in the memory area, the right-end address of a line's image is discontinuous from the left-end address of the next line's image. Compared to this, (B) in Figure 35.11 shows an example where the horizontal size of an image output from the VEU matches the horizontal length of the memory. In this case, the right-end address of a line's image and the left-end address of the next line's image have continuous values.

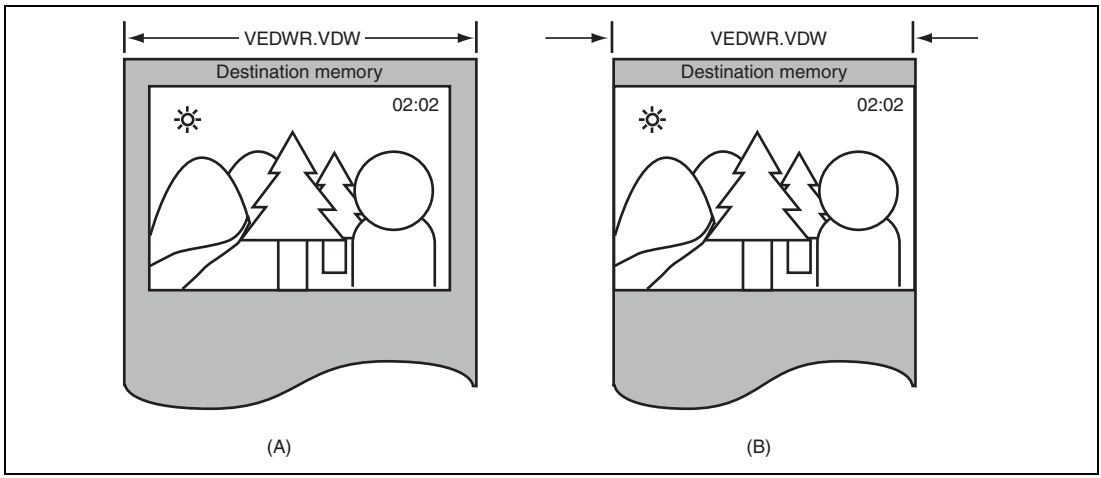


Figure 35.11 Relationship between Image Pasting Size and VEDWR

35.3.8 VEU Destination Address Y Register (VDAYR)

VDAYR sets the start address for the Y/RGB plane of the image drawn by the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	VDAY[31:2] VDAY[1:0]	H'0000 0000	R/W R	These bits set the start address for drawing the Y/RGB plane by the VEU (longword units).

Set the start address of the destination image (top-left address in the destination image area) of the VEU in the VDAY bits, as shown in Figure 35.12. When drawing a YCbCr image, specify the start address of the Y plane. When drawing an RGB image, specify the start address of the RGB plane.

When VTRCR.WPKF = 6 (RGB 16-bpp), a setting in word units is possible.

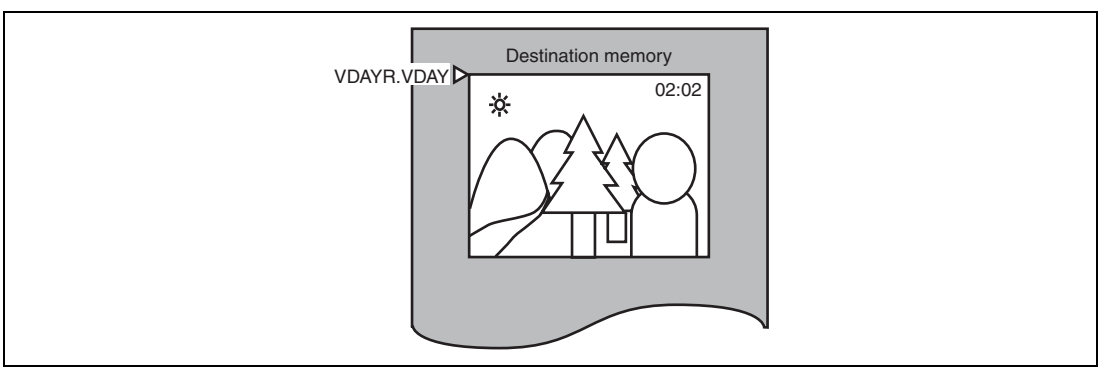


Figure 35.12 Address Set in VDAYR

The point specified as the address to be set in VDAYR changes when the image is rotated or inverted. This specification point of the address for each processing is shown in Table 35.5. Figure 35.13 illustrates the address specification point in block processing, and Figure 35.14 illustrates that in line processing. The gray quadrangle is assumed as the destination image and the ☆ marks as the address specification points.

The address specification point if neither scale-up/down, enhancer, bundle read, nor VFMCR.FLTPI = 0 is performed is the top-left address for the block located in one of the four corners with the destination image divided into blocks of 16 × 16 pixels. In a case that scale-up/down, enhance, and bundle read are not used and even FLTPI = 0, or either scale-up/down, enhance, or bundle read is used, each address specification point becomes one of the four corners in the destination image.

Table 35.5 Address Specification Point for Setting VDAYR

	No Rotation/ Inversion	90° Rotation	270° Rotation	90° Rotation + Horizontal Inversion	90° Rotation + Vertical Inversion	Horizontal Inversion	Vertical Inversion	Horizontal Inversion + Vertical Inversion
Block processing: No scale-up/down, enhancer, or bundle read and even FLTPI = 0.	—	D	E	F	G	A	B	C
Line processing: Scale-up/down, enhancer, or bundle read performed and even VFMCR.FLTPI = 1.	—	x	x	x	x	H	J	K

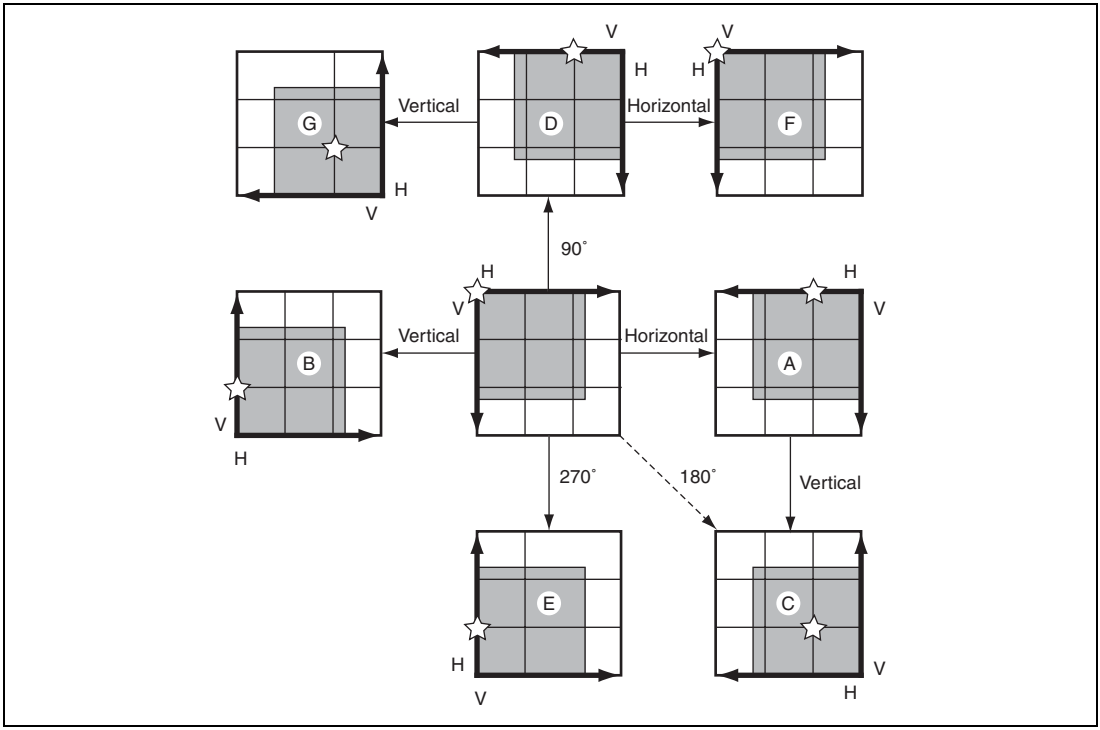


Figure 35.13 Location of Address Specification Point for VDAYR (Block Processing)

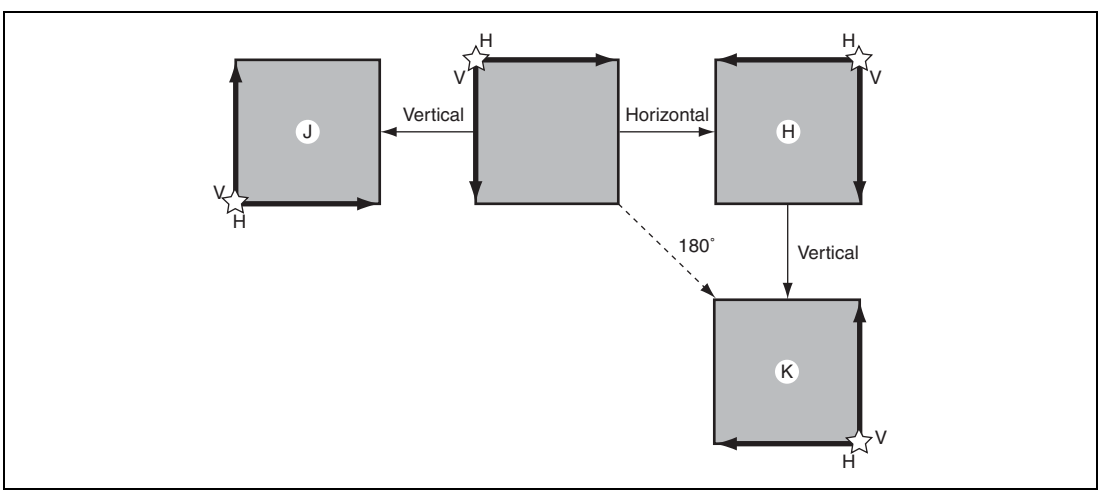


Figure 35.14 Location of Address Specification Point for VDAYR (Line Processing)

When drawing a YCbCr image, specify an address of the Y plane. When drawing an RGB image, specify an address of the RGB plane.

The formulas for setting the addresses are shown below.

Offset_ad	Becomes the address location of the top-left corner in the destination image	
yuv420out	Drawing format is YCbCr 4:2:0	1: VTRCR. CHDS = 0 0: VTRCR. CHDS = 1 or 2
clip_hsize	Horizontal size of line processing output	VRFSR.VHCLP
clip_vsize	Vertical size of line processing output	VRFSR.VVCLP
clip_vsize_c	Vertical size (C) of line processing output	(clip_vsize + 1)/2: VTRCR. CHDS = 0 (round-down) clip_vsize: VTRCR. CHDS = 1 or 2
dest_width	Destination memory width	VEDWR.VEW
dest_width_c	Destination memory width (C)	dest_width: VTRCR.CHDS = 2 dest_width × 2: VTRCR.CHDS = 2
src_hblk	Horizontal block count	VESSR.VHSS/16 (round-up)
src_vblk	Vertical block count	VESSR.VVSS/16 (round-up)
src_sideh	Horizontal size of right-end block	(VESSR.VHSS + 15)%16 + 1
src_sidev	Vertical size of bottom-end block	(VESSR.VHSS + 15)%16 + 1
< >	Operator changing a negative value to 0	0 when the value in < > is 0 or lower, otherwise the value in < >

For P_density_y of an RGB output, see table 35.6. For P_density_y and P_density_c of a YCbCr output, see table 35.7.

[Address for Y component output]

0 VDAYR = offset_ad

F VDAYR = offset_ad

D VDAYR = offset_ad + <((src_vblk – 2) × 16 + src_sidev)> × (4/P_density_y)

A VDAYR = offset_ad + <((src_hblk – 2) × 16 + src_sideh)> × (4/P_density_y)

E VDAYR = offset_ad + <((src_hblk – 2) × 16 + src_sideh)> × dest_width

B VDAYR = offset_ad + <((src_vblk – 2) × 16 + src_sidev)> × dest_width

G VDAYR = offset_ad + <((src_hblk – 2) × 16 + src_sideh)> × dest_width + <((src_vblk – 2) × 16 + src_sidev)> × (4/P_density_y)

C VDAYR = offset_ad + <((src_vblk – 2) × 16 + src_sidev)> × dest_width + <((src_hblk – 2) × 16 + src_sideh)> × (4/P_density_y)

H VDAYR = offset_ad + clip_hsize × (4/P_density_y)

J VDAYR = offset_ad + (clip_vsize – 1) × dest_width

K VDAYR = offset_ad + (clip_vsize – 1) × dest_width + clip_hsize × (4/P_density_y)

[Address for C component output]

0 VDACR = offset_ad

F VDACR = offset_ad

D VDACR = offset_ad + <((src_vblk – 2) × 16 + src_sidev)> × (4/P_density_c)

A VDACR = offset_ad + <((src_hblk – 2) × 16 + src_sideh)> × (4/P_density_c)

E VDACR = offset_ad + <((src_hblk – 2) × 16 + src_sideh)>/(1 + yuv420out) × dest_width_c

B VDACR = offset_ad + <((src_vblk – 2) × 16 + src_sidev)>/(1 + yuv420out) × dest_width_c

G VDACR = offset_ad + <((src_hblk – 2) × 16 + src_sideh)>/(1 + yuv420out) × dest_width_c + <((src_vblk – 2) × 16 + src_sidev)> × (4/P_density_c)

C VDACR = offset_ad + <((src_vblk – 2) × 16 + src_sidev)>/(1 + yuv420out) × dest_width_c + <((src_hblk – 2) × 16 + src_sideh)> × (4/P_density_c)

H VDACR = offset_ad + clip_hsize × (4/P_density_c)

J VDACR = offset_ad + (clip_vsize_c – 1) × dest_width_c

K VDACR = offset_ad + (clip_vsize_c – 1) × dest_width_c + clip_hsize × (4/P_density_c)

35.3.9 VEU Destination Address C Register (VDACR)

VDACR specifies the start address for the C plane of the image drawn by the VEU.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	VDAC[31:2] VDAC[1:0]	H'0000 0000	R/W R	These bits set the start address for drawing the C plane by the VEU. (1-longword units. 2-longword units for the YCbCr 4:4:4 format.)

Set the start address of the destination image (top-left address in the destination image area) of the VEU in the VDAC bits, as shown in Figure 35.15. When drawing a YCbCr image, specify the start address of the C plane. When reading an RGB image, VDACR is not used.

The point specified as the address to be set in VDACR changes when the image is rotated or inverted. As this specification point is similar to that for VDAYR, see Figure 35.13 and Table 35.5 for details. However, when an image is drawn in the YCbCr 4:2:0 format, the number of lines output in the vertical direction for the C component is half of the number of lines for the Y component or an RGB output. When calculating the address in the vertical direction (memory width × number of vertical lines), take notice that the number of lines will become half. (See the description in section 35.3.8, VEU Destination Address Y Register (VDAYR).)

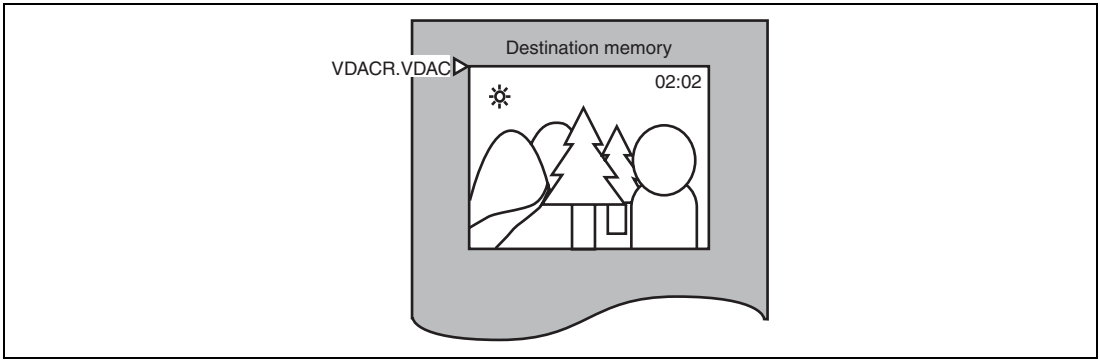


Figure 35.15 Address Set in VDACR

35.3.10 VEU Transform Control Register (VTRCR)

VTRCR mainly sets the processing contents related to color conversion by the VEU and the input/output data format.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PAD[7:0]								CHDS[1:0]		—	WPKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CHRR[1:0]		—	—	RPKF[3:0]				—	—	—	DITH	TM2	TM1	TE	RY
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PAD[7:0]	H'00	R/W	These bits set the PAD value to be stuffed into the output data when the VEU output data is in the RGB format. Some forms of RGB output data packs (see Table 35.6) include a PAD field as shown in Figure 35.16. In the PAD field, any 8-bit data can be added. Specify the value to be added to the PAD field to these bits.

Bit	Bit Name	Initial Value	R/W	Description
23, 22	CHDS[1:0]	00	R/W	These bits select the output format when the VEU output data is in the YCbCr format. For the YCbCr format, data is output in one of the pack patterns shown in Table 35.7. Conversion from YCbCr 4:4:4 to YCbCr 4:2:2 or YCbCr 4:2:0 is performed by taking the average of two pixels for the horizontal direction, and simply skipping pixels for the vertical direction. The planes for outputting the Y and C data to memory are separate. Therefore, the ratio of the Y:C memory areas is 2:1 for YCbCr 4:2:0, 2:2 for YCbCr 4:2:2, and 2:4 for YCbCr 4:4:4. 00: YCbCr 4:2:0 output for YCbCr-format output 01: YCbCr 4:2:2 output for YCbCr-format output 10: YCbCr 4:4:4 output for YCbCr-format output 11: Setting prohibited
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20 to 16	WPKF[4:0]	00000	R/W	These bits set the output data pack form when the VEU output data is in the RGB format. The VEU packs output data into a 32-bit pack. When data is output in the RGB format and YCbCr format, the data is packed into one of the patterns shown in tables 35.6 and 35.7, respectively. For RGB-format output, set a value in the WPKF column in Table 35.18 in these bits. When the RGB stuffing output pack is used (WPKF = H'10, H'11, H'12, or H'15), the start line must have the pack form of phase 0 shown in Table 35.6.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	CHRR[1:0]	00	R/W	These bits set the source image form for the VEU. When a YCbCr image is input, output is done in one of the pack patterns shown in Table 35.7. 00: Source image is read in the YCbCr 4:2:0 format 01: Source image is read in the YCbCr 4:2:2 format 10: Source image is read in the YCbCr 4:4:4 format 11: Setting prohibited
13, 12	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	RPKF[3:0]	000	R/W	These bits set the RGB input data pack form when the VEU input data is in the RGB format. Table 35.8 shows the relationship between the RGB format selected by these bits and the input form. When the VEU input data is in the RGB format, set a value in the RPKF column in Table 35.8 in these bits.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DITH	0	R/W	Selects whether dithering is applied or not when performing tone reduction of an RGB image. The VEU has a tone reduction function for RGB images. The number of tones after this processing is determined by the WPKF bit setting. Tone reduction processing for RGB images uses quantization which causes the resultant image to contain false contours. This problem becomes more serious as the number of tones is reduced, and it generally degrades the image quality. The VEU can perform dithering to lessen this degradation caused by tone reduction. 0: Does not apply dithering at tone reduction for the VEU destination image in the RGB format 1: Applies dithering at tone reduction for the VEU destination image in the RGB format

Bit	Bit Name	Initial Value	R/W	Description
3	TM2	0	R/W	Selects the color converting equation of color conversion circuit. 00: Use color conversion of full-range RGB[0,255] ⇔ Y[16,235], CbCr[16,240] conforms to ITU-R BT.601. 01: Use color conversion of full range RGB[0,255] ⇔ YCbCr[0,255] conforms to ITU-R BT.601. 10: Use color conversion of full-range RGB[0,255] ⇔ Y[16,235], CbCr[16,240] conforms to ITU-R BT.709. 11: Use color conversion of full-range RGB[16,235] ⇔ Y[16,235], CbCr[16,240] conforms to ITU-R BT.709.
2	TM1	0	R/W	
1	TE	0	R/W	The TE bit turns on or off the circuit for RGB ⇔ YCbCr conversion. The RY bit specifies whether the image input is in the YCbCr format or RGB format. Table 35.9 shows the relationship between the TE and RY bit settings and the input/output data format. 00: Reads a YCbCr source image and does not perform YCbCr → RGB conversion 01: Reads an RGB source image and does not perform RGB → YCbCr conversion 10: Reads a YCbCr source image and performs YCbCr → RGB conversion 11: Reads an RGB source image and performs RGB → YCbCr conversion
0	RY	0	R/W	

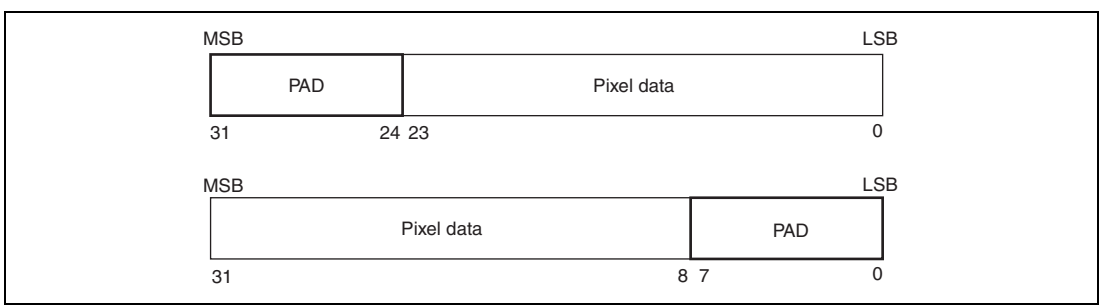


Figure 35.16 Data Pack Forms Including PAD

Table 35.6 RGB Data Output Pack Forms

No.	WPKF	Bit Rate [bpp]	P Density_y [pix/LW]	Phase	Bit																																					
					31 to 24								23 to 16								15 to 8								7 to 0											
0	B'00000	8	4	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	R3	R3	R3	B3	B3
1	B'00001	12	2	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
2	B'00010				R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
6	B'00110	16	2	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1
8	B'01000	18	1	—	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
10	B'01010				PAD								R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
13	B'01101				PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
14	B'01110				R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
16	B'10000		4/3	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
				1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
				2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
17	B'10001		4/3	0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
				1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
				2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
18	B'10010		4/3	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
				1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
				2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
19	B'10011	24	1	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
20	B'10100				R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
21	B'10101		4/3	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
				1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
				2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
22	B'10110	18	1	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0
23	B'10111				0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0	G0

Table 35.7 YCbCr Data Pack Forms

No.	Compo-nent	Bit Rate [bpp]	P_density_y, P_density_c [pix/LW]	Bit																																						
				31 to 24									23 to 16								15 to 8									7 to 0											
0	Y data	16	4	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
	C data		4	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2
1	Y data	16	4	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
	C data		4	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2
2	Y data	24	4	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
	C data		2	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb1	Cb1	Cb2	Cb1	Cb1	Cb1	Cb1	Cb1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1
			2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cb3	Cb3	Cb3	Cb3	Cb3	Cb3	Cb3	Cb3	Cr3	Cr3	Cr3	Cr3	Cr3	Cr3	Cr3	Cr3

Table 35.8 RGB Data Input Pack Forms

No.	RPKF	Bit Rate [bpp]	P Density-y [pix/LW]	Phase	Bit																																					
					31 to 24								23 to 16								15 to 8								7 to 0											
0	B'00000	24	1	—	X	X	X	X	X	X	X	X	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
1	B'00001			—	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	X	X	X	X	X	X	X	X
2	B'00010		4/3	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
				1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
				2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
3	B'00011	16	2	—	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1
7	B'00111	18	1	—	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0
13	B'01101	16	1	—	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0

[Legend] X: Don't care

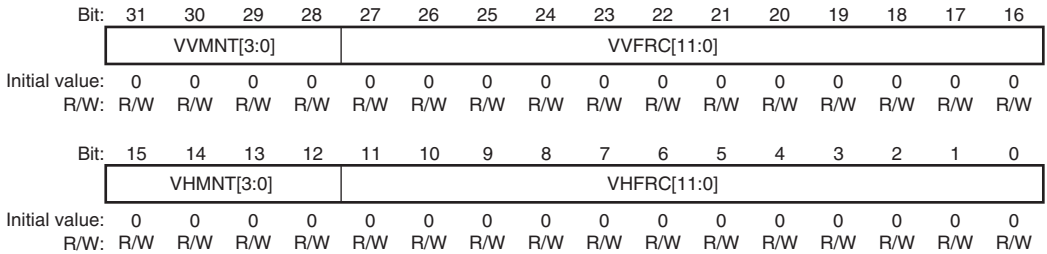
Table 35.9 Relationship between TE and RY Bits in VTRCR and Input/Output Data Format

TE Bit	RY Bit	Input	Output
0	0	YCbCr format	YCbCr format
0	1	RGB format	RGB format
1	0	YCbCr format	RGB format
1	1	RGB format	YCbCr format

35.3.11 VEU Resize Filter Control Register (VRFCR)

VRFCR sets the scale-up/down factor for the image scaling filter. When performing scale-up/down processing, some other processings cannot be performed simultaneously (see Table 35.14).

Modifying this register during operation is prohibited.



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	VVMNT[3:0]	H'0	R/W	Mantissa Part of Vertical Scaling Factor These bits set the vertical scaling factor. A value from H'0 to H'F can be set. When the VVMNT bits are set to H'0 and the VVFRC bits to H'000, the output is the same size.
27 to 16	VVFRC[11:0]	H'000	R/W	Fraction Part of Vertical Scaling Factor These bits set the vertical scaling factor. A value from H'000 to H'FFF can be set.
15 to 12	VHMNT[3:0]	H'0	R/W	Mantissa Part of Horizontal Scaling Factor These bits set the horizontal scaling factor. A value from H'0 to H'F can be set. When the VHMNT bits are set to H'0 and the VHFRC bits to H'000, the output is the same size.

Bit	Bit Name	Initial Value	R/W	Description
11 to 0	VHFRC[11:0]	H'000	R/W	Fraction Part of Horizontal Scaling Factor These bits set the horizontal scaling factor. A value from H'000 to H'FFF can be set.

The VEU has an image scaling filter that can scale the image up and down, as shown in Figure 35.17. The scaling filter cannot be operated simultaneously with the median filter, deblocking filter, or low-pass filter, and rotation is also prohibited. Accordingly, set B'0 to the bits in VFMCR, except the HMRR (horizontal inversion) and VMRR (vertical inversion) bits, when setting VRFCR.

The scaling factor for scale-up/down can be set between 1/16 to 16.

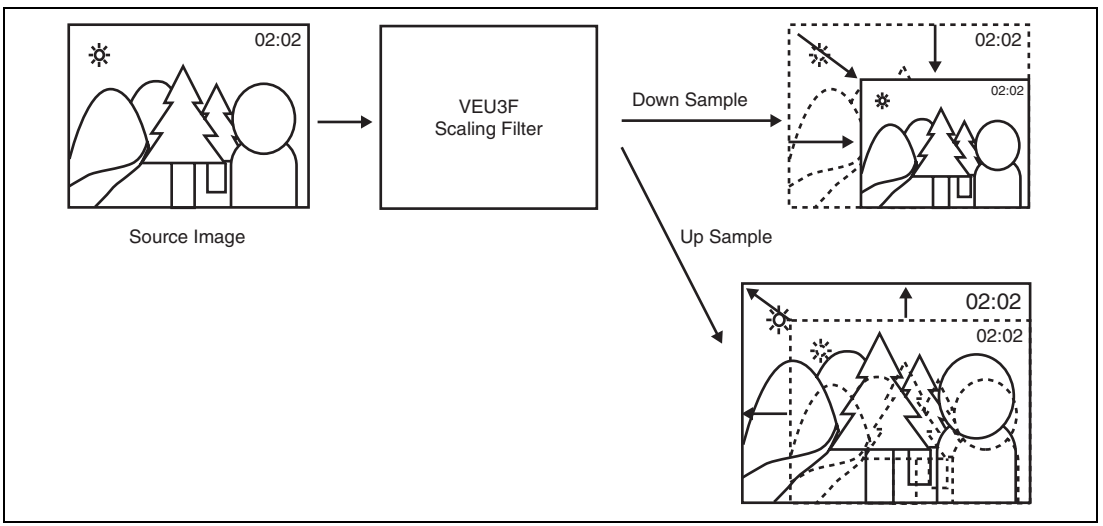


Figure 35.17 Scaling of Image by VEU

The formulas for obtaining the output pixel count of the scale-up/-down from the scaling ratio and the input pixel count are defined as follows:

$$\alpha = MANT \times 4096 + FRAC \quad \text{. . . Formula 1}$$

$$SCL \text{ (scaling factor)} = \frac{4096}{\alpha} \quad \text{. . . Formula 2}$$

In a formula listed below, decimal place of a calculation result of <A> is rounded off.

Here, an input image size is indicated as Sin, and an output image size as SIZE.

Note: Horizontal: Sin = VESSR.VHSS
Vertical: Sin = VESSR.VVSS

[Scale-down]

$$SIZE =< 1 + (< 1 + (Sin - 1) / MANTpre > - 1) \times MANTpre \times SCL > \quad [1/16 < SCL \leq 1]$$

…Formula 3

MANTpre = 1 [1 ≤ MANT > 4]
= 2 [4 ≤ MANT > 8]
= 4 [8 ≤ MANT > 16]

[Scale-up]

$$SIZE =< 1 + (Sin - 1) \times SCL > \quad [1 < SCL \leq 16, VRSCR.AMD = 0]$$ …Formula 4

$$SIZE =< Sin \times SCL > \quad [1 < SCL \leq 16, VRSCR.AMD = 1]$$ …Formula 5

Note: See section 35.3.14, VEU Resize Filter Control Register (VRSCR) details on the AMD bit

Example: Scale up 88 pixels to 352 pixels (VRSCR.AME = 0)

SCL = 352/88 = 4, suppose SCL = 4, and the preliminary settings of MANT = 0 and FRAC = 1024 are made based on formulas 1 and 2. Substituting these in formula 5 results in an output pixel count of 349. Since this calculated value is smaller than the desired output pixel count of 352, set FRAC to a value that is smaller than this value but at the same time the maximum value (MANT = 0, FRAC = H'3FF). At this point, the output pixel count is 349, which is not the desired output pixel count so that the calculation to be repeated. Setting to a value FRAC = H'3F7, the output pixel count becomes 352 which is the desired output pixel count. So set MANT = 0, FRAC = H'3F7 for the scaling up 88 pixels to 352.

Table 35.10 Settings for Each Horizontal Scaling Factor of Scaling Filter

		FRAC		
Scaling Factor	**AMD**	**Decimal**	**Hexadecimal**	**MANT**
8	0	508	H'1FC	0
4	0	1017	H'3F9	0
2	0	2039	H'7F7	0
1.5	0	2723	H'AA3	0
8	1	512	H'200	0
4	1	1024	H'400	0
2	1	2048	H'800	0
1.5	1	2730	H'AAA	0
7/8	0	585	H'249	1
3/4	0	1365	H'555	1
5/8	0	2457	H'999	1
1/2	0	0	H'0	2
1/4	0	0	H'0	4
1/8	0	0	H'0	8

35.3.12 VEU Resize Filter Size Clip Register (VRFSR)

VRFSR sets the clipping (discarded) size of fractions in pixels output from the filter, and must be set in combination with VRFCR.

Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VVCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VHCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	VVCLP[11:0]	H'000	R/W	These bits set the vertical clipping size after scale-up/down in pixel units (1-pixel units). This setting should be made in 2-pixel units when the YCbCr format is 4:2:0.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 1	VHCLP[11:1]	H'000	R/W	These bits set the horizontal clipping size after scale-up/down in pixel units (4-pixel units). This setting should be made in 2-pixel units when the WPKF bits (write pack) in VTRCR are set to 1, 2, 6, 8 to 14, 19, 20, 22, or 23.
0	VHCLP[0]		R	

The VVCLP and VHCLP bits in the VRFSR register set the output size when line processing (scale-up/down, enhancer, or bundle read) is performed. The size can be specified in 4-pixel units horizontally and in 1-pixel (2-pixel for YCbCr4:2:0 format) units vertically in VRFSR. However, a setting in 2-pixel units horizontally is possible for RGB 2 bytes/pixel (VTRCR.WPKF=1, 2, 6) and RGB 4 bytes/pixel (VTRCR.WPKF=8 to 14, 19, 20, 22, 23).

The maximum value to be set in the VVCLP and VHCLP bits in the VRFSR register is 4092 pixels. The minimum value for them is 16 pixels. Note that the values should be such that VRFSR.VHCLP + VRSOR.VHCLOFS ≤ 4092.

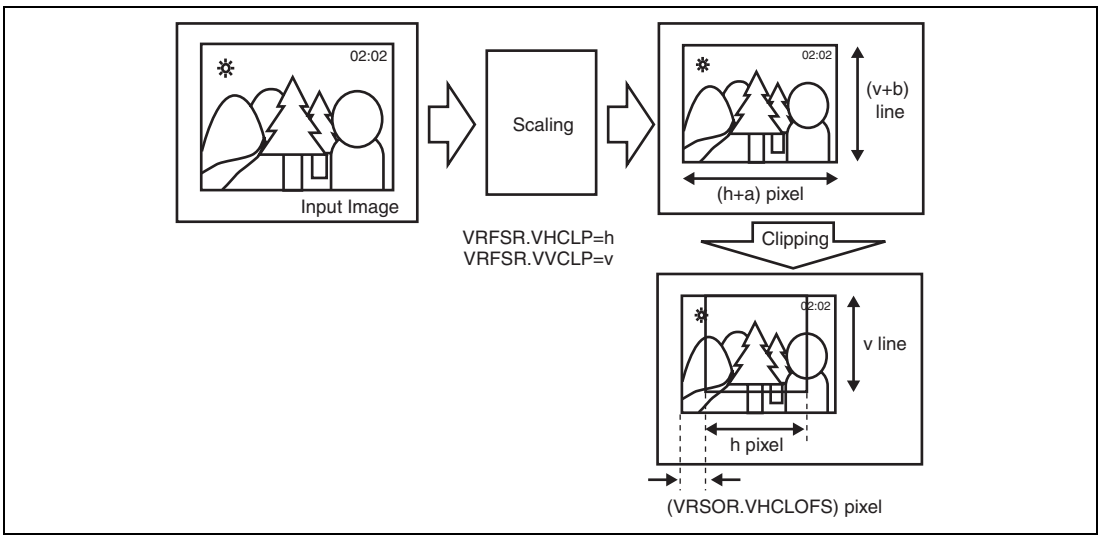


Figure 35.18 Clipping of Image Output from VEU Scaling Filter

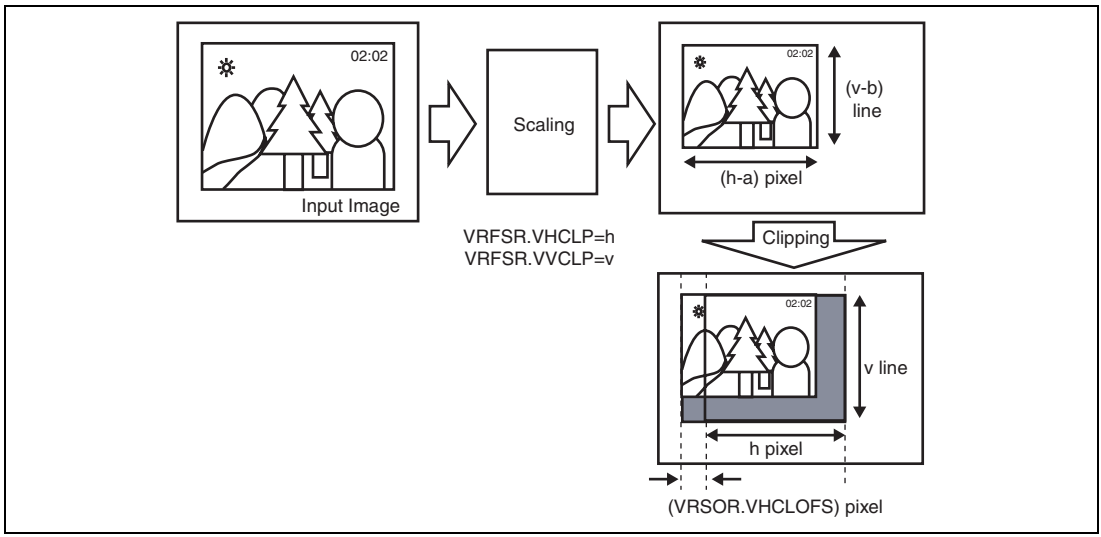


Figure 35.19 Shifting of Image Output from VEU Scaling Filter

If the values are smaller than the filter output pixel count, output images are clipped as shown in Figure 35.18. The number of output pixels is counted from the position shifted to the right by the offset specified in the VHCLOFS bit in the VRSOR register. The pixels that are located at the right of the specified pixel count or below the specified line count are discarded by the clipping function.

If the values are larger than the filter output pixel count, images are output by complementing pixels up to the specified pixel count according to the FMD bit in the VRSCR register, where the pixels are counted from the position shifted to the right by the offset specified in the VHCLOFS bit in the VRSOR register (see Figure 35.19).

However, if the horizontal number of pixels specified in VRFSR is larger than the number of the scaling filter output pixels rounded up to the nearest multiple of 16, the complementary function does not work. In this case, the VEU may be hung up, so the VRFSR value should be a value that is obtained by calculating by the formulas 1 to 6 and rounding up to a multiple of 16.

If the filter output pixel count obtained by the formulas 1 to 6 is equal to 16M +N (N = 1 to 16), VRFSR.VV(H)CLP ≤ 16M+16.

When scale-up/down is not performed, set VRFSR.VHCLP = VESSR.VHSS, VRFSR.VVCLP = VESSR.VVSS.

35.3.13 VEU Enhance Register (VENHR)

VENHR sets up the filter (enhancer) for enhancing the edges of images. Some other processings cannot be performed simultaneously with the edge enhancement processing (see Table 35.2). Modifying this register during operation is prohibited.

The enhancer cannot be operated simultaneously with the median filter, deblocking filter, or low-pass filter, and rotation is also prohibited. Accordingly, set B'0 to the bits in VFMCR, except the HMRR (horizontal inversion) and VMRR (vertical inversion) bits, when setting VENHR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENSCL[2:0]			—	—	—	—	—	—	ENHV	ENHH
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	ENSCL[2:0]	010	R/W	These bits set the factor for edge enhancement. 001: Enhancement level 1 (max) 010: Enhancement level 2 011: Enhancement level 3 100: Enhancement level 4 (min) Other than above: Setting prohibited
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ENHV	0	R/W	Enables or disables edge enhancement in the vertical direction. 0: Vertical edge enhancement is not performed 1: Vertical edge enhancement is performed
0	ENHH	0	R/W	Enables or disables edge enhancement in the horizontal direction. 0: Horizontal edge enhancement is not performed 1: Horizontal edge enhancement is performed

35.3.14 VEU Resize Filter Control Register (VRSCR)

VRSCR performs scaling filter adjustments for line processing (with scaling, enhance, and bundle read). For block processing, set all of the initial values of VRSCR to 0 (without scaling, enhance, or bundle read).

Writing 1 to the read-only bits is prohibited. If 1 is written to any of these bits a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AMD	FMD	LC[1:0]		—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	AMD	0	R/W	This bit specifies the pixel count generated when the scaling filter performs a scale-up. When AMD = 1 is specified, int (n * scale-up factor) pixels are created by complementing horizontal and vertical pixels in the destination image for AMD = 0. 0: The pixel count obtained at a scale-up is 1+int ((n1)*scale-up factor). 1: The pixel count obtained at a scale-up is int (n*scale-up factor).
30	FMD	0	R/W	This bit specifies the pixel complementing method for the scaling filter. When the scaling filter outputs an image that does not fill the clip size, the pixels are complemented up to the clip size. 0: Copy right (lower) pixels and provide complementary pixels. 1: Provide complementary right(lower) pixels of a color specified by VFLCR.

Bit	Bit Name	Initial Value	R/W	Description
29, 28	LC[1:0]	00	R/W	These bits specify the number of pixels for source image horizontal-left clipping. From 0 to 3 can be specified. Because the source address registers VSAYR and VSCAR are both in longword units, if the input format is YCbCr, the starting position for horizontal clipping from the background image is in four-pixel units. Therefore, if you perform background image clipping from pixel position at the left end 4M+LC (M; integer, LC: 0-3), 4M is adjusted by the VSAYR read start address, and the lower two bits are adjusted by this bit (LC). When the value of this bit has been set, input to the scaling filter is specified by the VHSS bits in VESSR and the LC bits in VRSCR for the horizontal direction and by the VVSS bits in VESSR for the vertical direction.
27 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

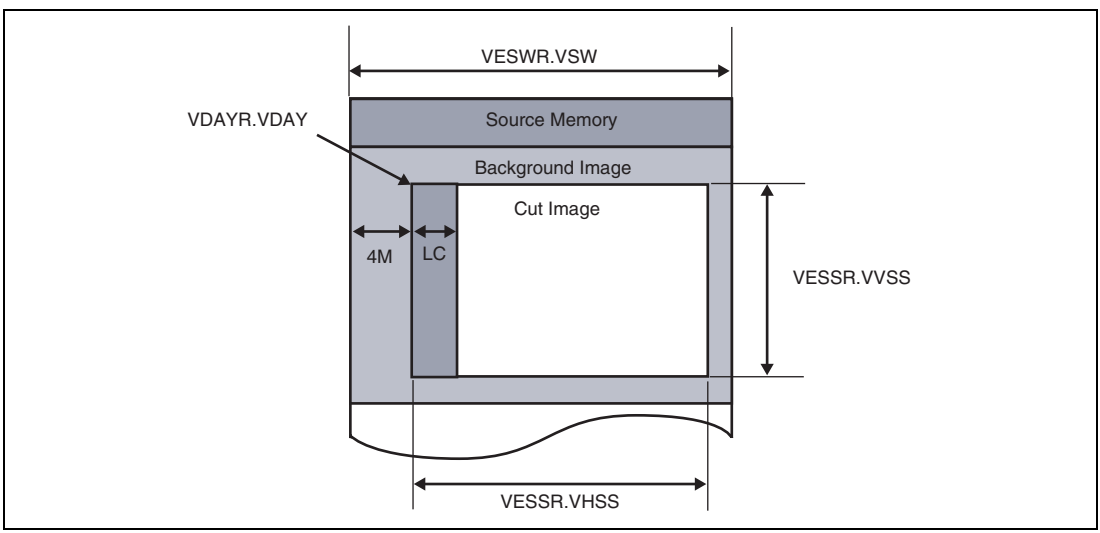


Figure 35.20 Relations between the Background Image and Clipping Pixel Location

35.3.15 VEU Resize Filter Size Clip Offset Register (VRSOR)

VRSOR sets the left clip offset when images are clipped bases on the filter output pixel count. For details on the offset positions, see figures 35.18 and 35.19. When following Flow 2 in Figure 35.2, set an offset that is smaller than the horizontal number of pixels of the output image after scaling down.

Also, make sure such that VRFSR.VHCLP + VRSOR.VHCLOF ≤ 4092.

When this register is set, the VRFCR and VRFSR registers should also be set. Since the register setting is used as the output size when Flow 2 describe in section 35.2, Functional Overview of VEU (Scale-up/down, enhancer, or bundle read is performed), this register should be set even if scale-up/down is not performed.

Writing to this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	VHCLOFS[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	VHCLOFS [7:0]	H'00	R/W	Specifies the lateral clip offset by the number of pixel (one-pixel unit) after scale-up/down.

35.3.16 VEU Filter Mode Control Register (VFMCR)

VFMCR sets the operating mode for filter processing.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MEDFST	FLTPI	—	—	—	—	—	—	—	MED
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TPN	—	—	DBLK	LPHV	—	—	VMRR	HMRR	—	—	ROTL	ROTR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	MEDFST	0	R/W	Changes the order of applying the median filter and deblocking filter when the FLTPI bit is set to 1. 0: Applies the LPF/deblocking filer and median filter in this order. 1: Applies the median filer and LPF/deblocking filter in this order.
24	FLTPI	0	R/W	Controls fast processing of LPF, deblocking filer, and median filter. 0: Disables fast processing of LPF, deblocking filer, and median filter. 1: Enables fast processing of LPF, deblocking filer, and median filter. When the FLTPI is set to 1, the possible settings are restricted to the followings: (1) VFMCR.ROTR = 0 (2) VFMCR.ROTL = 0 (3) VESTR.VBE = 0 (4) VRSOR.VHCLOFS = 0 (5) VRFSR.VVCLP = VSSR.VVSS (6) VRFSR.VHCLP = VSSR.VHSS

Bit	Bit Name	Initial Value	R/W	Description
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	MED	0	R/W	Enables or disables usage of the median filter. When the median filter is applied to an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. In bundle read mode, the median filter cannot be used. Therefore, when this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the filter operation is shown in Table 35.11. 0: Median filter is not used 1: Median filter is applied to the source image
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	TPN	0	R/W	Sets the number of taps of the low-pass filter (LPF). 0: LPF tap count is set to 3 taps 1: LPF tap count is set to 5 taps
11, 10	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
9	DBLK	0	R/W	Enables or disables usage of the deblocking filter. When the deblocking filter is applied to an image in bundle read mode, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the filter operation is shown in Table 35.11. 0: LPF is applied to the entire source image (normal mode) 1: LPF is applied to only the boundary of the 8 × 8 pixel blocks of the source image (deblocking mode)

Bit	Bit Name	Initial Value	R/W	Description
8	LPHV	0	R/W	Enables or disables usage of the low-pass filter. In bundle read mode, the low-pass filter cannot be used. When the low-pass filter is applied to an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the filter operation is shown in Table 35.11. 0: NOP 1: LPF is applied to the source image
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
5	VMRR	0	R/W	Enables or disables usage of the vertical inversion (symmetric with regard to horizontal axis) filter. The relationship between this bit and the rotate/invert operation is shown in Table 35.12. 0: NOP 1: Vertical inversion (symmetric with regard to horizontal axis) filter is applied to the source image
4	HMRR	0	R/W	Enables or disables usage of the horizontal inversion (symmetric with regard to vertical axis) filter. The relationship between this bit and the rotate/invert operation is shown in Table 35.12. 0: NOP 1: Horizontal inversion (symmetric with regard to vertical axis) filter is applied to the source image
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	ROTL	0	R/W	Enables or disables usage of the 270-degree rotation (clockwise) filter. In bundle read mode, rotation cannot be performed. When rotation processing is performed for an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the rotate/invert operation is shown in Table 35.12. 0: NOP 1: 270-degree rotation (clockwise) filter is applied to the source image
0	ROTR	0	R/W	Enables or disables usage of the 90-degree rotation (clockwise) filter. In bundle read mode, rotation cannot be performed. When rotation processing is performed for an image, scale-up/down processing and edge enhance processing cannot be performed at the same time. When this bit is set to 1, set VRFCR, VENHR, and the VBE bit in VESTR to 0. (Interoperating with the scale-up/down processing, edge enhance processing, and bundle read mode is prohibited.) The relationship between this bit and the rotate/invert operation is shown in Table 35.12. 0: NOP 1: 90-degree rotation (clockwise) filter is applied to the source image

Table 35.11 Relationship between MED, DBLK, and LPHV Bits and Filter Operation

MED Bit	DBLK Bit	LPHV Bit	Filter Operation
0	0	0	Passed through
0	0	1	Low-pass filter
0	1	1	Deblocking filter
1	1	1	Deblocking filter + median filter*
1	0	0	Median filter
Other than above			Setting prohibited

Note: * The order of deblocking filter → median filter cannot be changed.

Table 35.12 Relationship between VMRR, HMRR, ROTL, and ROTR Bits and Rotate/Invert Operation

VMRR Bit	HMRR Bit	ROTL Bit	ROTR Bit	Rotate/Invert Operation
0	0	0	0	No rotation/inversion
0	0	0	1	Rotated by 90 degrees clockwise
0	0	1	0	Rotated by 270 degrees clockwise
0	1	0	1	Rotated by 90 degrees clockwise and then inverted in the horizontal direction
1	0	0	1	Rotated by 90 degrees clockwise and then inverted in the vertical direction
0	1	0	0	Inverted in the horizontal direction
1	0	0	0	Inverted in the vertical direction
1	1	0	0	Rotated by 180 degrees
Other than above				Setting prohibited

To rotate the image 270 degrees and then invert it in the horizontal direction, rotate the image 90 degrees and then invert it in the vertical direction. Likewise, to rotate the image 270 degrees and then invert it in the vertical direction, rotate the image 90 degrees and then invert it in the horizontal direction.

Figures 35.21 and 35.22 show the relationship between the rotated/inverted images and raw image.

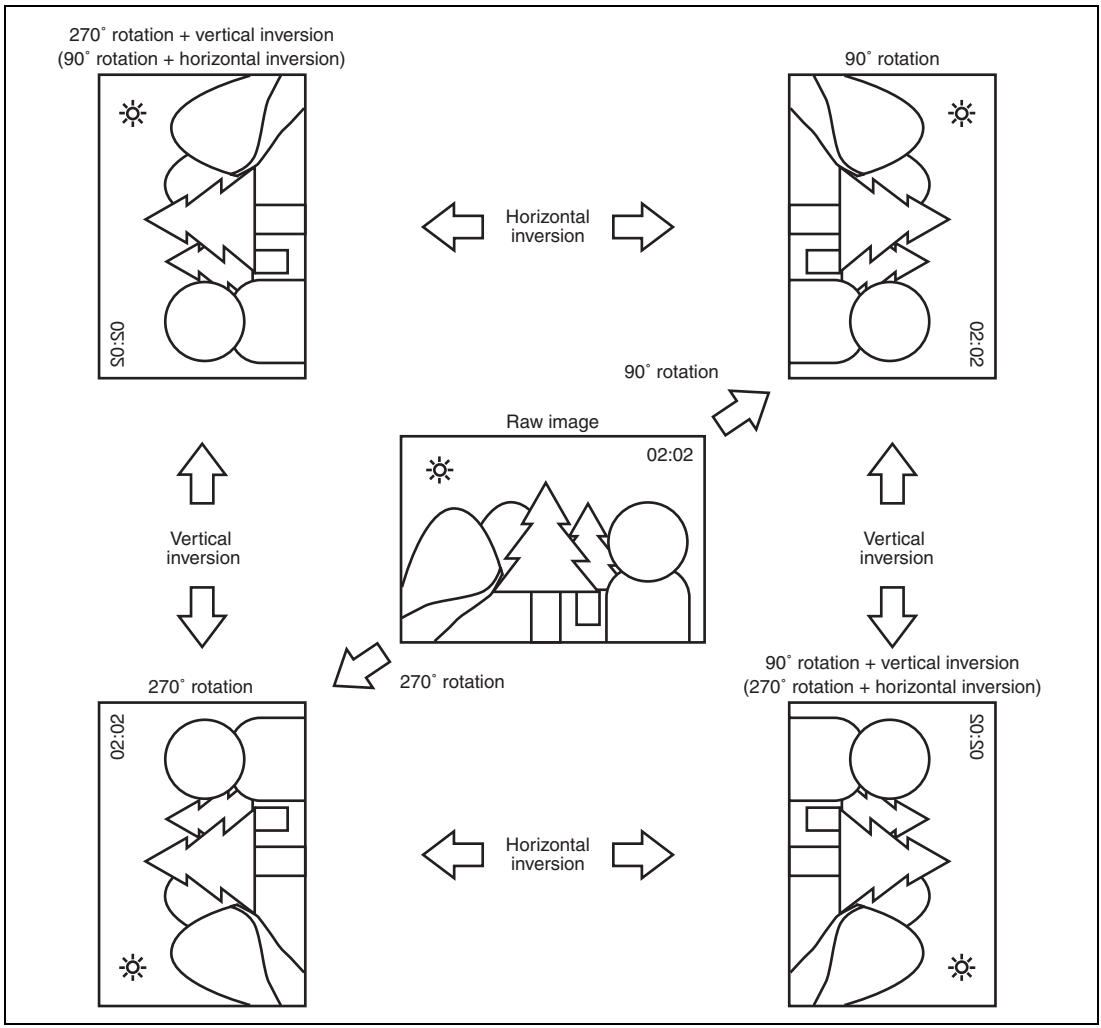


Figure 35.21 Relationship between Rotated/Inverted Images and Raw Image

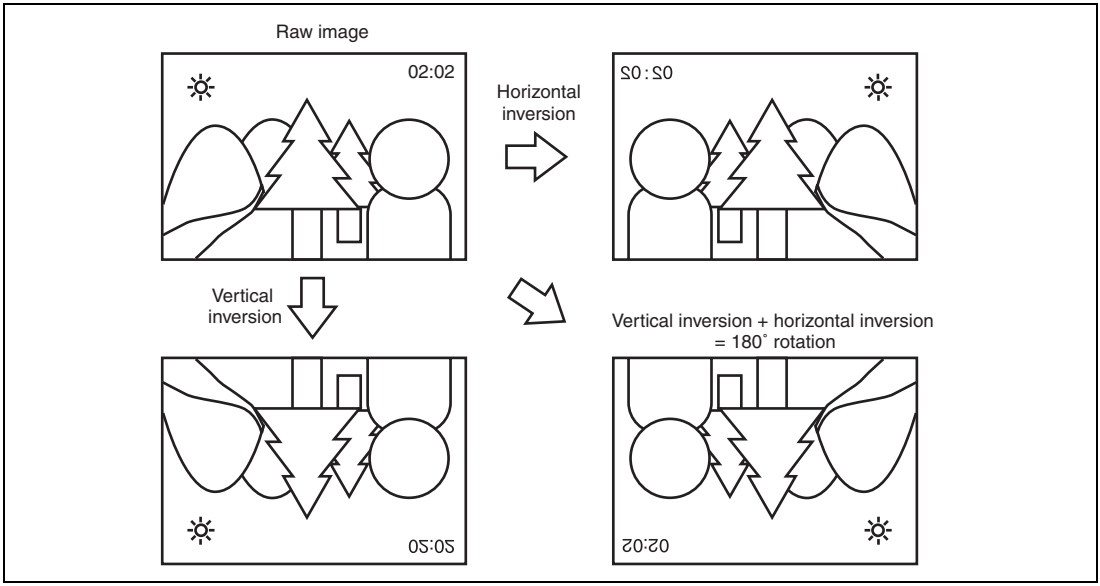


Figure 35.22 Relations between Inverted Images and Raw Image

35.3.17 VEU Vertical Tap Coefficient Register (VVTCR)

VVTCR sets the vertical tap coefficients when applying the low-pass filter.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	VSHFT[2:0]			VTPC4[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VTPC3[3:0]				VTPC2[3:0]				VTPC1[3:0]				VTPC0[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 20	VSHFT[2:0]	000	R/W	These bits specify the shift capacity after vertical tap calculation. Set how many pixels the location is to be shifted right after adding each vertical tap coefficient. This setting must correspond to the sum of the VTPC0 to VTPC4 bits. See Table 35.13 for details on the values corresponding to the total value of VVTCR.VTPC.
19 to 16	VTPC4[3:0]	H'0	R/W	These bits set vertical tap coefficient 4 (clear these bits to 0 for a 3-tap filter). Set the component of the pixel two pixels right of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.
15 to 12	VTPC3[3:0]	H'0	R/W	These bits set vertical tap coefficient 3. Set the component of the pixel on the right of that location to be given when applying a low-pass filter.

Bit	Bit Name	Initial Value	R/W	Description
11 to 8	VTPC2[3:0]	H'0	R/W	These bits set vertical tap coefficient 2. Set the component of that location to be given when applying a low-pass filter.
7 to 4	VTPC1[3:0]	H'0	R/W	These bits set vertical tap coefficient 1. Set the component of the pixel on the left of that location to be given when applying a low-pass filter
3 to 0	VTPC0[3:0]	H'0	R/W	These bits set vertical tap coefficient 0 (clear these bits to 0 for a 3-tap filter). Set the component of the pixel two pixels left of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.

Table 35.13 Shifting Value Corresponds to Sum of VTPC4 to VTPC0

ΣVVTCR.VTPC	VVTCR.VSHFT
4	2
8	3
16	4
32	5
64	6

Examples of setting VVTCR are shown below. When the VTPC4 bits are H'0 and the VTPC0 bits are H'0, set the TPN bit in VFMCR to B'0 because a 3-tap low-pass filter is to be used. In all other cases, set the TPN bit to B'1 because a 5-tap low-pass filter is to be used.

Table 35.14 Values Set in VVTCR

Bit Name	VSHFT	VTPC4	VTPC3	VTPC2	VTPC1	VTPC0
Values to be set	2	0	1	2	1	0
	3	0	1	6	1	0
	3	1	2	2	2	1
	3	1	1	4	1	1
	4	0	1	14	1	0
	4	0	3	10	3	0
	4	0	5	6	5	0
	4	1	1	12	1	1
	4	1	2	10	2	1
	4	1	3	8	3	1
	4	1	4	6	4	1
	4	2	3	6	3	2
	4	3	3	4	3	3
	5	0	9	14	9	0
	5	1	8	14	8	1
	5	1	9	12	9	1
	5	1	10	10	10	1
	5	2	7	14	7	2
	5	2	9	10	9	2
	5	3	6	14	6	3
	5	3	7	12	7	3
	5	3	8	10	8	3
	5	4	5	14	5	4
	5	4	7	10	7	4
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	6	11	14	14	14	11
	6	12	13	14	13	12

Note: Horizontal and vertical taps should be set in the same size.

35.3.18 VEU Horizontal Tap Coefficient Register (VHTCR)

VHTCR sets the horizontal tap coefficients when applying the low-pass filter.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	HSHFT[2:0]			HTPC4[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HTPC3[3:0]				HTPC2[3:0]				HTPC1[3:0]				HTPC0[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 20	HSHFT[2:0]	000	R/W	These bits specify the shift capacity after horizontal tap calculation. Set how many pixels the location is to be shifted right after adding each horizontal tap coefficient. For details on the setting, see the description of the VSHFT bits in VVTCR. See Table 35.13 for details on the values corresponding to the total value of VHTCR.HTPC.
19 to 16	HTPC4[3:0]	H'0	R/W	These bits set horizontal tap coefficient 4. Set the component of the pixel two pixels right of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.
15 to 12	HTPC3[3:0]	H'0	R/W	These bits set horizontal tap coefficient 3. Set the component of the pixel on the right of that location to be given when applying a low-pass filter.
11 to 8	HTPC2[3:0]	H'0	R/W	These bits set horizontal tap coefficient 2. Set the component of that location to be given when applying a low-pass filter.
7 to 4	HTPC1[3:0]	H'0	R/W	These bits set horizontal tap coefficient 1. Set the component of the pixel on the left of that location to be given when applying a low-pass filter

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	HTPC0[3:0]	H'0	R/W	These bits set horizontal tap coefficient 0. Set the component of the pixel two pixels left of that location to be given when applying a 5-tap low-pass filter. Clear these bits to 0 when applying a 3-tap low-pass filter.

Table 35.15 Shifting Value Corresponds to Sum of HTPC4 to HTPC0

ΣVHTCR.HTPC	VHTCR.HSHFT
4	2
8	3
16	4
32	5
64	6

Examples of setting VHTCR are shown below. When both the HTPC4 bits and HTPC0 bits are H'0, set the TPN bit in VFMCR to B'0 because a 3-tap low-pass filter is to be used. In all other cases, set the TPN bit to B'1 because a 5-tap low-pass filter is to be used.

Table 35.16 Values Set in VHTCR

Bit Name	HSHFT	HTPC4	HTPC3	HTPC2	HTPC1	HTPC0
Values to be set	2	0	1	2	1	0
	3	0	1	6	1	0
	3	1	2	2	2	1
	3	1	1	4	1	1
	4	0	1	14	1	0
	4	0	3	10	3	0
	4	0	5	6	5	0
	4	1	1	12	1	1
	4	1	2	10	2	1
	4	1	3	8	3	1
	4	1	4	6	4	1
	4	2	3	6	3	2
	4	3	3	4	3	3
	5	0	9	14	9	0
	5	1	8	14	8	1
	5	1	9	12	9	1
	5	1	10	10	10	1
	5	2	7	14	7	2
	5	2	9	10	9	2
	5	3	6	14	6	3
	5	3	7	12	7	3
	5	3	8	10	8	3
	5	4	5	14	5	4
	5	4	7	10	7	4
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	5	5	5	12	5	5
	5	5	6	10	6	5
	5	5	7	8	7	5
	6	11	14	14	14	11
	6	12	13	14	13	12

Note: Horizontal and vertical taps should be set in the same size.

35.3.19 VEU Designated Color Register (VAPCR)

VAPCR specifies the designated color.

When any pixel in an image to be processed by the VEU is the same as the designated color set by this register, that pixel is replaced with the conversion color set by VECCR. For an RGB output, clear the lower bits except the valid bits to 0. For a YCbCr output, only the YCbCr 4:4:4 format is supported.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	APCON	RAPC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAPC[7:0]								BAPC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	APCON	0	R/W	Enables or disables replacement of the designated color. 0: VEU processed image is output as it is 1: VEU processed image is output after pixels corresponding to the designated color have been replaced with the conversion color
23 to 16	RAPC[7:0]	H'00	R/W	These bits specify the R (Cb) component of the designated color.
15 to 8	GAPC[7:0]	H'00	R/W	These bits specify the G (Y) component of the designated color.
7 to 0	BAPC[7:0]	H'00	R/W	These bits specify the B (Cr) component of the designated color.

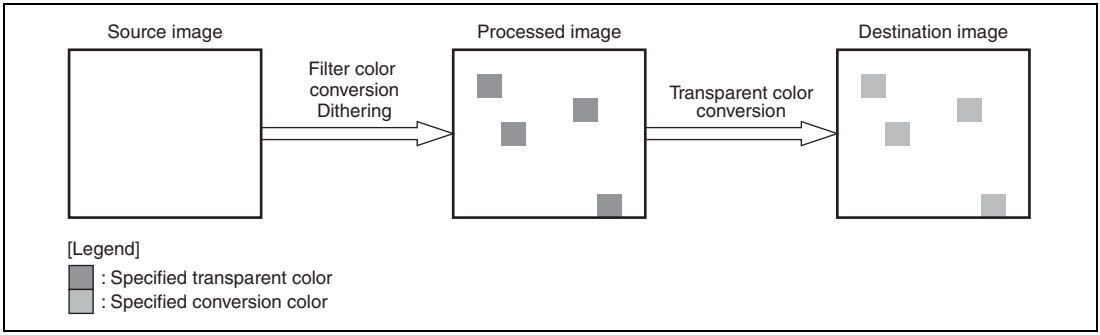


Figure 35.23 Replacement of Designated – Color Pixels in VEU Processed Image with Conversion – Color Pixels

35.3.20 VEU Conversion Color Register (VECCR)

VECCR specifies the conversion color.

When any pixel in an image to be processed by the VEU is the same as the designated color set by VAPCR, that pixel is replaced with the conversion color set by this register. For an RGB output, clear the lower bits except for the valid bits to 0. For a YCbCr output, only the YCbCr 4:4:4 format is supported.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RCHGC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GCHGC[7:0]								BCHGC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	RCHGC[7:0]	H'00	R/W	These bits specify the R (Cb) component of the conversion color.
15 to 8	GCHGC[7:0]	H'00	R/W	These bits specify the G (Y) component of the conversion color.
7 to 0	BCHGC[7:0]	H'00	R/W	These bits specify the B (Cr) component of the conversion color.

35.3.21 VEU Fill Color Specification Register (VFLCR)

VFLCR is a fill color specification register.

When the FMD bit in VRSCR is set to 1, VFLCR specifies the fill color used to complement pixels when the pixel area created by the VEU scaling filter is smaller than the clip area specified by the clip size register. For RGB output, specify 0 for the lower bits other than the valid bits. RGB

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	RFILC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GFILC[7:0]								BFILC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	RFILC[7:0]	H'00	R/W	These bits specify the R (Cb) component of the fill color.
15 to 8	GFILC[7:0]	H'00	R/W	These bits specify the G (Y) component of the fill color.
7 to 0	BFILC[7:0]	H'00	R/W	These bits specify the B (Cr) component of the fill color.

35.3.22 VEU Address Fixed Register (VAFXR)

VAFXR specifies fixed mode for the addresses of the data output from the VEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VAFIX
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VAFIX	0	R/W	Sets fixed mode for output addresses. When this bit is 1, the addresses to draw the image output from the VEU are fixed to the values set in VDAYR. When this bit is 0, the output address is incremented according to the data and VEDWR. Fixed address mode is enabled only when the VEU destination image is in the RGB format. For YCbCr destination images, set the following bits or register to 0; the bits MED, LPHV, ROTL, and ROTR in VFMCR (which prohibit median filter, low-pass filter, and rotation), VRFCR (same-size output), and the lower three bits in VDAYR and VRFSR (which disable other than burst transfer). 0: Output addresses are not in fixed address mode 1: Output addresses are in fixed address mode

35.3.23 VEU Swapping Register (VSWPR)

VSWPR sets swapping within the 64-bit data at the data input/output section of the VEU.

When dividing addresses into 64 bits, make the following setting for each case.

- When swapping data in longword units for all inputs/outputs: H'0000 0044
- When swapping data in word units for all inputs/outputs: H'0000 0022
- When swapping data in byte units for all inputs/outputs: H'0000 0011
- When swapping data in byte units from the MSB to the LSB: H'0000 0077

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VEOLS	VEOWS	VEOBS	—	VEILS	VEIWS	VEIBS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	VEOLS	0	R/W	Sets swapping in longword units for output data. In longword swapping for output data, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU output section (Figure 35.24). 0: Output data is not swapped in longword units 1: Output data is swapped in longword units
5	VEOWS	0	R/W	Sets swapping data in word units for output data. In word swapping for output data, data is swapped in word units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU output section (Figure 35.25). 0: Output data is not swapped in word units 1: Output data is swapped in word units

Bit	Bit Name	Initial Value	R/W	Description
4	VEOBS	0	R/W	Sets swapping data in byte units for output data. In byte swapping for output data, data is swapped in byte units within each 16 bits for 64-bit data in the VEU output section (Figure 35.26). 0: Output data is not swapped in byte units 1: Output data is swapped in byte units
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	VEILS	0	R/W	Sets swapping data in longword units for input data. In longword swapping for input data, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU input section (Figure 35.24). 0: Input data is not swapped in longword units 1: Input data is swapped in longword units
1	VEIWS	0	R/W	Sets swapping data in word units for input data. In word swapping for input data, data is swapped in word units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the VEU input section (Figure 35.25). 0: Input data is not swapped in word units 1: Input data is swapped in word units
0	VEIBS	0	R/W	Sets swapping data in byte units for input data. In byte swapping for input data, data is swapped in byte units within each 16 bits for 64-bit data in the VEU input section (Figure 35.26). 0: Input data is not swapped in byte units 1: Input data is swapped in byte units

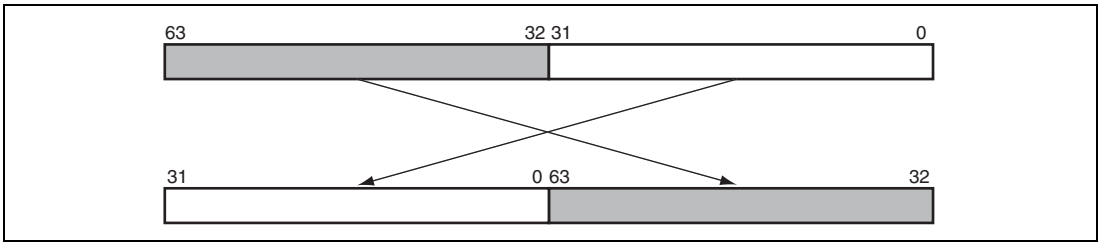


Figure 35.24 Relations before and after Data Swapping in Longword Units

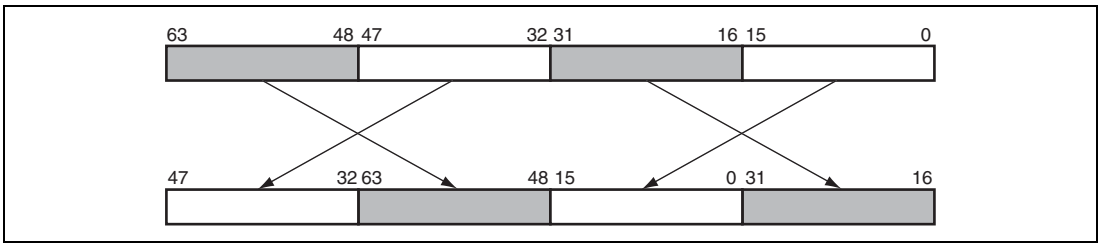


Figure 35.25 Relations before and after Data Swapping in Word Units

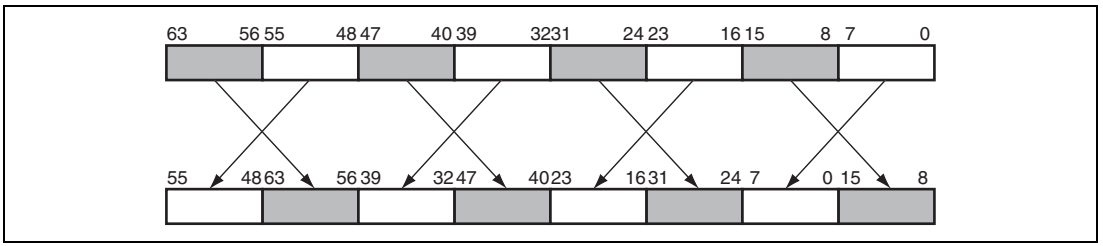


Figure 35.26 Relations before and after Data Swapping in Byte Units

35.3.24 VEU Event Interrupt Enable Register (VEIER)

VEIER enables or disables output of the interrupt signal of a VEVTR flag.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VBENDE	—	—	—	—	—	—	—	VEENDE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBENDE	0	R/W	Enables or disables the interrupt signal of VEVTR.VBEND to be output. 0: Disables output of the VEVTR.VBEND interrupt signal 1: Enables output of the VEVTR.VBEND interrupt signal
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VEENDE	0	R/W	Enables or disables the interrupt signal of VEVTR.VEEND to be output. 0: Disables output of the VEVTR.VEEND interrupt signal 1: Enables output of the VEVTR.VEEND interrupt signal

35.3.25 VEU Event Register (VEVTR)

VEVTR indicates the interrupt source when an internal interrupt occurs in the VEU. Whether output of the interrupt signal of each source in VEVTR is enabled or disabled is set by VEIER.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VB END	—	—	—	—	—	—	—	VE END
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBEND	0	R/W	This flag is set to 1 when VEU processing has finished for a single read (reading of the number of lines set in VBSSR) in bundle read mode (N-line read mode). The VEU enters the read restart wait state when this flag is set. Therefore, after clearing the interrupt source, switch the address registers, and start reading again (VESTR = H'0000 0011). [Reading] 0: Indicates in progress of reading in bundle read processing, or this flag has already been cleared 1: Indicates that restart is being waited in bundle read processing [Writing] 0: Clears this flag by writing 0 to it 1: Holds the current value
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	VEEND	0	R/W	This flag is set to 1 when VEU processing has totally finished. This flag is not cleared unless 0 is written to clear the bit after VEU processing ends. Therefore, 0 must be written to this bit to clear it before the VEU is activated again. [Reading] 0: Indicates that VEU processing has not finished, or this flag has already been cleared 1: Indicates that VEU processing has finished [Writing] 0: Clears this flag by writing 0 to it 1: Holds the current value

35.3.26 VEU Status Register (VSTAR)

VSTAR indicates the internal status of the VEU and the internal signal states.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	INTL	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VB PRC	—	—	—	VB READ	—	—	—	—	—	—	—	VE PRC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INTL	0	R	Indicates the assert state of the VEU interrupt signal to the CPU. 0: Indicates that no interrupt signal is asserted at the VEU interrupt signal port 1: Indicates that an interrupt signal is asserted at the VEU interrupt signal port

Bit	Bit Name	Initial Value	R/W	Description
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	VBPRC	0	R	Indicates the VEU status in bundle read mode. For details, see Table 35.27. 0: Indicates that the VEU is not operating in bundle read (N-line read) mode 1: Indicates that the VEU is operating in bundle read (N-line read) mode
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VBREAD	0	R	For details, see Table 35.18. 0: Indicates that the VEU is waiting to be restarted in bundle read (N-line read) mode 1: Indicates that the VEU is reading in bundle read (N-line read) mode
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VEPRC	0	R	Indicates the same value as the VE bit in VESTR. For details, see Table 35.18. 0: VEU is ready (halted) 1: VEU is busy (operating state)

The VEU operates in either normal operating mode or bundle read mode. In normal operating mode, the registers are not to be modified between activation and processing end. In bundle read mode, there are two operating states: read processing state and the state waiting for the VEU to be restarted by software. The values when reading VSTAR in each state are shown in Table 35.17. Figure 35.27 shows the VSTAR state transitions in normal operating mode, and Figure 35.27 shows the VSTAR state transitions in bundle read mode.

Table 35.17 VSTAR Value Read in Each State

Operating Mode		VBPRC Bit	VBREAD Bit	VEPRC Bit
Normal operation	Halted	0	0	0
	Operating	0	0	1
N-line read mode	Halted	0	0	0
	During read processing	1	1	1
	Waiting for restart	1	0	1

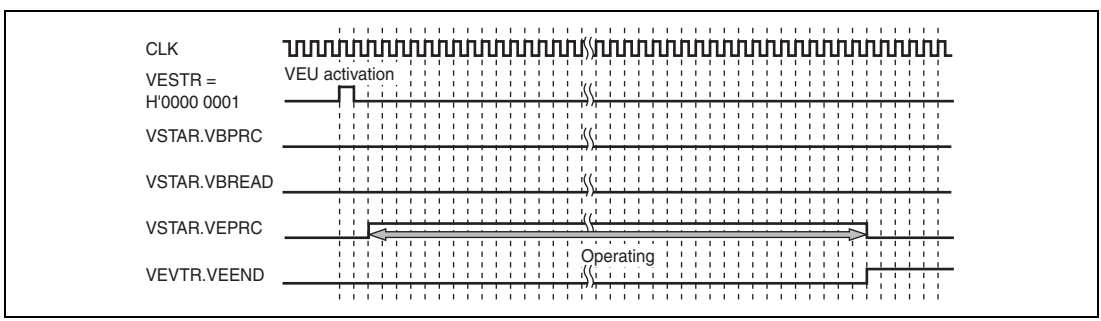


Figure 35.27 VSTAR State Transitions in Normal Operating Mode

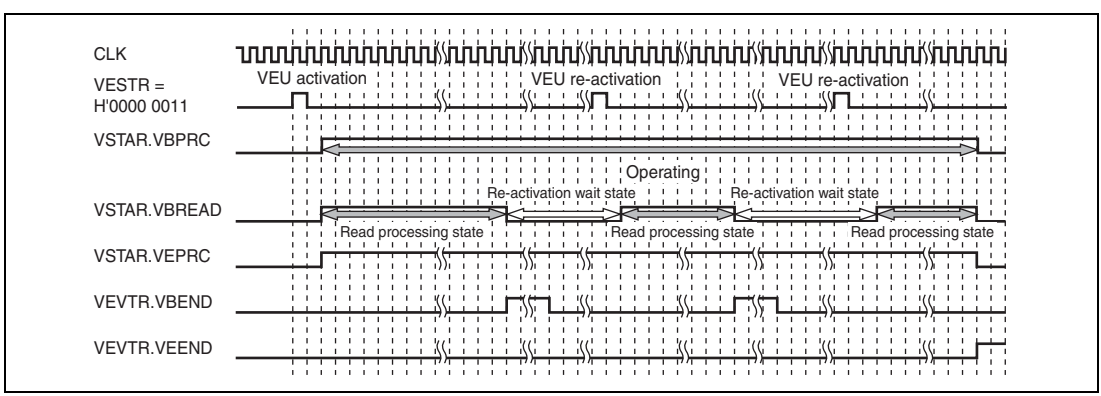


Figure 35.28 VSTAR State Transitions in Bundle Read Mode

35.3.27 VEU Module Reset Register (VBSRR)

VBSRR executes the module reset of the VEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

If a module reset is performed during operation, the logic circuits handshaking with the CPU bus are forcibly reset, and malfunction may also affect modules other than the VEU. To correctly break the handshake with the CPU bus before VEU termination, see section 35.3.1, VEU Start Register (VESTR).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ALL RST	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ALLRST	0	W	Module Reset When 1 is written to this bit, all internal control signals of the VEU are reset. 0: Setting prohibited 1: Internal reset of the VEU
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

35.3.28 VEU Resize Passband Register (VRPBR)

VRPBR sets the signal passbands when scale-up/down is performed. When this register is set, the VRFCR and VRFSR registers should be also set. Writing this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	VBW[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	HBW[6:0]						
Initial vlaue:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 16	VBW[6:0]	H'00	R/W	Sets the vertical signal passband when scaling-up/down.
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	HBW[6:0]	H'00	R/W	Sets the horizontal signal passband when scaling-up/down.

When scaling up vertically (the VMNT bit in the VRFCR register is 0), set 64 in the VBW bit in the VRPBR register.

When scaling down vertically (the VMNT bit in the VRFCR register is other than 0), set a value obtained by the following formula in the VBW bit in the VRPBR register:

$$VSPBR.VBW = \left(64 \times \frac{4096 \times VMANTpre}{4096 \times VRFCR.VMNT + VRFCR.VFRC} \right)$$

Where VMANTpre is

4(8 ≤ VRFCR.VMNT<16)
2(4 ≤ VRFCR.VMNT<8)
1(1 ≤ VRFCR.VMNT<4)

Note that <A> is an operation that rounds value A down to the nearest integer.

When scaling up horizontally (the HMNT bit in the VRFCR register is 0), set 64 in the HBW bit in the VRPBR register.

When scaling down horizontally (the HMNT bit in the VRFCR register is other than 0), set a value obtained by the following formula in the HBW bit in the VRPBR register:

$$VSPBR.HBW = \left(64 \times \frac{4096 \times HMANTpre}{4096 \times VRFCR.HMNT + VRFCR.HFRC} \right)$$

Where HMANTpre is

4(8 ≤ VRFCR.HMNT<16)
2(4 ≤ VRFCR.HMNT<8)
1(1 ≤ VRFCR.HMNT<4)

Note that <A> is an operation that rounds value A down to the nearest integer.

35.4 Usage Notes for VEU

(1) Restrictions during Operation

Do not stop the clock or specify the module standby state by setting the MSTP bit during VEU operation.

(2) Restrictions on Input/Output Functions

Note that when restrictions at other places in this manual differ from restrictions in the following table, restrictions in the following table are given priority.

Table 35.18 Restrictions on VEU Input/Output Functions

Item	Pack Type	Restrictions
Input	RGB565, 4 bytes/pixel	• The start address for input must be specified in longword units. • The horizontal size of the raw image (memory) must be specified in byte units that correspond to two pixels of the source image.
	Others	• The start address for input must be specified in longword units. However, when the RGB stuffing pack is input, the address must be specified so that reading starts from the pack of phase 0. • The horizontal size of the raw image (memory) must be specified in byte units that correspond to four pixels of the destination image.
Output	RGB565 4 bytes/pixel	• The output address must be specified in longword units. • The horizontal size of the destination image (memory) must be specified in byte units that correspond to two pixels of the destination image.
	Others	• The output address must be specified in longword units. • The horizontal size of the destination image (memory) must be specified in byte units that correspond to four pixels of the destination image.

Section 36 Blending Engine Unit (BEU)

The blending engine unit, BEU, blends three displays, and has a multiwindow function that displays four windows overlaying the blended display. This LSI has two BEU modules (BEU0 and BEU1).

36.1 Features

The BEU is used with being connected to the buses via bus bridge modules, and also connected to the VOU and LCDC. The following lists the features.

- Supports video display.
- Supports OSD (On-Screen Display).
- Supports graphic display.
- Blends the three planes of Video1, Video2, and OSD/Graphic.
- The three displays can be blended at desired positions.
- Any one of the three inputs can be used as the parent display.
- The location of a child display can overflow from the parent display, but the overflowed area is not output.
- Raster operation 2 functions
- Multiwindow function (four windows are displayed overlaying the three blended displays)
- Selection between output to the memory, output to the LCDC, and simultaneous output to the memory and LCDC

36.2 Functional Overview of BEU

This LSI has two BEU modules. Functions of each module are different in terms of whether this module operates together with the LCDC, VOU.

Table 36.1 LCDC, VOU Joint Operation

Module	LCDC Joint Operation	VOU Joint Operation
BEU (1)	Provided	Not provided
BEU (2)	Not provided	Provided

The functional overview of the BEU (Blending Engine Unit) is shown in Table 36.2.

Table 36.2 Functional Overview of BEU

Classification	Item	Function	Description	Note
Input format	YCbCr format	YCbCr 4:4:4/4:2:2/4:2:0 αYCbCr 4:4:4/4:2:2/4:2:0	α, Y, and C are input from separate planes. For YCbCr 4:2:0, the vertical line is read twice.	With input system 1, conversion is possible for RGB/YCbCr under the ITU-R BT.709 and ITU-R BT.601 standards.
	RGB format	RGB pack		
Output format	YCbCr format	YCbCr 4:4:4/4:2:2/4:2:0	Y and C are output to separate planes	With output, conversion is possible for RGB/YCbCr under the ITU-R BT.709 and ITU-R BT601 standards.
	RGB format	RGB pack	RGB pack output	
Source/ destination image size	Maximum	4092 × 4092	YCbCr420: 2-pixel units both horizontally and vertically YCbCr422: 2-pixel units horizontally In other formats 1-pixel units can be specified.	
	Minimum	1 × 1 pixels		
Dithering (tone reduction)		24 bpp	Full color (16,777,216 colors)	Dithering not possible
		18 bpp	260,000 colors	Dithering not possible
		16 bpp	High color (65,536 colors)	
		12 bpp	4,096 colors	
		8 bpp	256 colors	

Classification	Item	Function	Description	Note
Blending	PinP	Three planes (two video image planes and OSD/Graphic) are blended.	• Any of the three inputs can be used as the parent display. • The three displays can be blended at desired locations. • Overflow from the parent display area is allowed, but the overflowed area is not output. • The tile pattern can be blended. • Transparent color can be specified for input systems 1 to 3	*
OSD	Data format	8 bpp		
	CLUT size	32-bit αRGB 32-bit αYCbCr		
Raster operation 2		Three types of drawing color processing	Addition, replacement, and subtraction	
Multiwindow function		Four windows	Four windows are overlaid on the blended three planes.	• Setting image size and output position in 1-pixel units is possible. • Overlapping between windows is prohibited.
Display data output	Destination setting	Output system + write back to memory	Outputting display data to an output system is done simultaneously with writing the data back to memory.	

Note: * When the format for reading images is set to the RGB format, the source image also has to be in the RGB format. Likewise, when the format for reading images is set to the YCbCr format, the source image also has to be in the YCbCr format. However, input system 1 can use either the RGB format or YCbCr format as the source image format, regardless of the selected format for reading images. Therefore, if the source image format differs from the format for reading images, color conversion must be performed in input system 1. The pack form can be specified independently for each input system. The transparent color can be specified for RGB565, RGB666, RGB888, and YCbCr input formats.

Figure 36.1 shows a block diagram of the BEU.

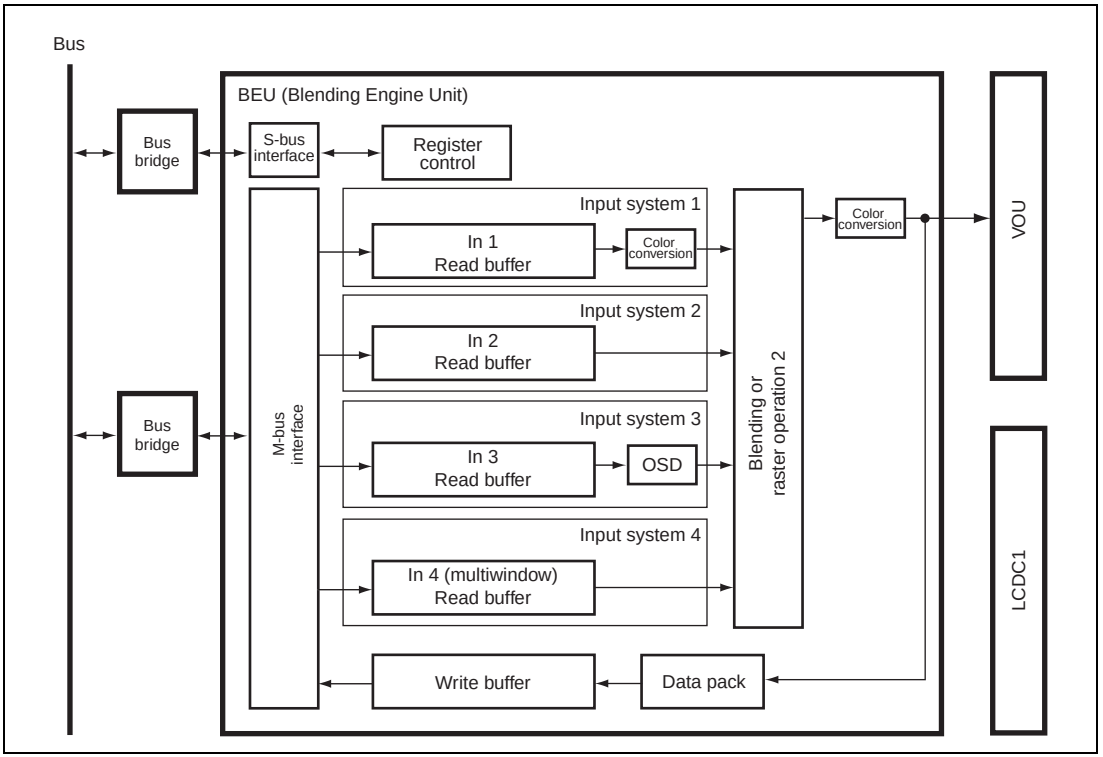


Figure 36.1 Block Diagram of BEU

36.3 Description of BEU Registers

(1) R/W Restrictions on BEU Registers

The read/write restrictions on the BEU registers are listed below. If the following register handling is not guaranteed, a malfunction may occur.

1. For the read-only bits in all BEU registers and the reserved areas, writing 1 is prohibited. Do not specify an unallowable value.
2. Most BEU registers have a 2-plane configuration (plane A and plane B). For registers (bits) to which writing during operation is prohibited, do not modify them during operation (reading is possible). The BEU operating state can be confirmed by reading the BEPRC bit in the BEU status register (BSTAR). Modify the above registers when this bit is B'0. When a mirror address is written to for a 2-plane register, the register on the unused plane is always accessed. To directly write to plane A or plane B, prohibit writing to the plane currently used.

(2) Terms and Abbreviations Used in This Section

The terms used in this section are described below.

1. "forcible termination" or "software reset" indicates that the BEU2 processing was forcibly terminated. In this case, the current processing is stopped but hardware terminates normally. The image processing result of the frame in which a software reset occurs is not guaranteed.
2. "module reset" indicates the BEU internal circuit was forcibly reset. In a module reset, reset processing is performed with no consideration to the hardware state. Therefore, if a module reset is generated when the BEU is operating normally, the BEU peripheral hardware may become unable to operate.
3. "during operation" in this section indicates a state in which the BEPRC bit in the BEU status register (BSTAR) is set to B'1.
4. When a bit name in a register is referred to in this section, it is indicated in the format of (register name).(bit name).
   Example: BESTR.BEIVK

(3) List of Registers

The register configuration of the BEU is shown in Table 36.3. The register states in each processing mode are shown in Table 36.4.

Table 36.3 Register Configuration of BEU

			Addresses			
Register Name	Abbr.	R/W	Address (Plane A)	Address (Plane B)	Mirror Address	Access Size
BEU (1) start register	BESTR	R/W	H'FE93 0000	—	—	32
BEU (1) source memory width register 1	BSMWR1	R/W	H'FE93 0010	H'FE93 1010	H'FE93 2010	32
BEU (1) source size register 1	BSSZR1	R/W	H'FE93 0014	H'FE93 1014	H'FE93 2014	32
BEU (1) source address Y register 1	BSAYR1	R/W	H'FE93 0018	H'FE93 1018	H'FE93 2018	32
BEU (1) source address C register 1	BSACR1	R/W	H'FE93 001C	H'FE93 101C	H'FE93 201C	32
BEU (1) source address α register 1	BSAAR1	R/W	H'FE93 0020	H'FE93 1020	H'FE93 2020	32
BEU (1) source image format register 1	BSIFR1	R/W	H'FE93 0024	H'FE93 1024	H'FE93 2024	32
BEU (1) source memory width register 2	BSMWR2	R/W	H'FE93 0028	H'FE93 1028	H'FE93 2028	32
BEU (1) source size register 2	BSSZR2	R/W	H'FE93 002C	H'FE93 102C	H'FE93 202C	32
BEU (1) source address Y register 2	BSAYR2	R/W	H'FE93 0030	H'FE93 1030	H'FE93 2030	32
BEU (1) source address C register 2	BSACR2	R/W	H'FE93 0034	H'FE93 1034	H'FE93 2034	32
BEU (1) source address α register 2	BSAAR2	R/W	H'FE93 0038	H'FE93 1038	H'FE93 2038	32
BEU (1) source image format register 2	BSIFR2	R/W	H'FE93 003C	H'FE93 103C	H'FE93 203C	32
BEU (1) source memory width register 3	BSMWR3	R/W	H'FE93 0040	H'FE93 1040	H'FE93 2040	32
BEU (1) source size register 3	BSSZR3	R/W	H'FE93 0044	H'FE93 1044	H'FE93 2044	32
BEU (1) source address Y register 3	BSAYR3	R/W	H'FE93 0048	H'FE93 1048	H'FE93 2048	32
BEU (1) source address C register 3	BSACR3	R/W	H'FE93 004C	H'FE93 104C	H'FE93 204C	32
BEU (1) source address α register 3	BSAAR3	R/W	H'FE93 0050	H'FE93 1050	H'FE93 2050	32
BEU (1) source image format register 3	BSIFR3	R/W	H'FE93 0054	H'FE93 1054	H'FE93 2054	32
BEU (1) tile pattern size register	BTPSR	R/W	H'FE93 0058	H'FE93 1058	H'FE93 2058	32
BEU (1) multidisplay source memory width register 1	BMSMWR1	R/W	H'FE93 0070	—	—	32
BEU (1) multidisplay source size register 1	BMSSZR1	R/W	H'FE93 0074	—	—	32
BEU (1) multidisplay source address Y register 1	BMSAYR1	R/W	H'FE93 0078	—	—	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
BEU (1) multidisplay source address C register 1	BMSACR1	R/W	H'FE93 007C	—	—	32
BEU (1) multidisplay source memory width register 2	BMSMWR2	R/W	H'FE93 0080	—	—	32
BEU (1) multidisplay source size register 2	BMSSZR2	R/W	H'FE93 0084	—	—	32
BEU (1) multidisplay source address Y register 2	BMSAYR2	R/W	H'FE93 0088	—	—	32
BEU (1) multidisplay source address C register 2	BMSACR2	R/W	H'FE93 008C	—	—	32
BEU (1) multidisplay source memory width register 3	BMSMWR3	R/W	H'FE93 0090	—	—	32
BEU (1) multidisplay source size register 3	BMSSZR3	R/W	H'FE93 0094	—	—	32
BEU (1) multidisplay source address Y register 3	BMSAYR3	R/W	H'FE93 0098	—	—	32
BEU (1) multidisplay source address C register 3	BMSACR3	R/W	H'FE93 009C	—	—	32
BEU (1) multidisplay source memory width register 4	BMSMWR4	R/W	H'FE93 00A0	—	—	32
BEU (1) multidisplay source size register 4	BMSSZR4	R/W	H'FE93 00A4	—	—	32
BEU (1) multidisplay source address Y register 4	BMSAYR4	R/W	H'FE93 00A8	—	—	32
BEU (1) multidisplay source address C register 4	BMSACR4	R/W	H'FE93 00AC	—	—	32
BEU (1) multidisplay source image format register	BMSIFR	R/W	H'FE93 00F0	—	—	32
BEU (1) blend control register 0	BBLCR0	R/W	H'FE93 0100	H'FE93 1100	H'FE93 2100	32
BEU (1) blend control register 1	BBLCR1	R/W	H'FE93 0104	—	—	32
BEU (1) process control register	BPROCR	R/W	H'FE93 0108	H'FE93 1108	H'FE93 2108	32
BEU (1) multiwindow control register 0	BMWCR0	R/W	H'FE93 010C	—	—	32
Blend location register 1	BLOCR1	R/W	H'FE93 0114	H'FE93 1114	H'FE93 2114	32
Blend location register 2	BLOCR2	R/W	H'FE93 0118	H'FE93 1118	H'FE93 2118	32
Blend location register 3	BLOCR3	R/W	H'FE93 011C	H'FE93 111C	H'FE93 211C	32
BEU (1) multidisplay location register 1	BMLOCR1	R/W	H'FE93 0120	—	—	32
BEU (1) multidisplay location register 2	BMLOCR2	R/W	H'FE93 0124	—	—	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
BEU (1) multidisplay location register 3	BMLOCR3	R/W	H'FE93 0128	—	—	32
BEU (1) multidisplay location register 4	BMLOCR4	R/W	H'FE93 012C	—	—	32
BEU (1) multidisplay transparent color control register 1	BMPCCR1	R/W	H'FE93 0130	—	—	32
BEU (1) multidisplay transparent color control register 2	BMPCCR2	R/W	H'FE93 0134	—	—	32
Blend pack form register	BPKFR	R/W	H'FE93 0140	H'FE93 1140	H'FE93 2140	32
BEU (1) transparent color control register 0	BPCCR0	R/W	H'FE93 0144	H'FE93 1144	H'FE93 2144	32
BEU (1) transparent color control register 11	BPCCR11	R/W	H'FE93 0148	H'FE93 1148	H'FE93 2148	32
BEU (1) transparent color control register 12	BPCCR12	R/W	H'FE93 014C	H'FE93 114C	H'FE93 214C	32
BEU (1) transparent color control register 21	BPCCR21	R/W	H'FE93 0150	H'FE93 1150	H'FE93 2150	32
BEU (1) transparent color control register 22	BPCCR22	R/W	H'FE93 0154	H'FE93 1154	H'FE93 2154	32
BEU (1) transparent color control register 31	BPCCR31	R/W	H'FE93 0158	H'FE93 1158	H'FE93 2158	32
BEU (1) transparent color control register 32	BPCCR32	R/W	H'FE93 015C	H'FE93 115C	H'FE93 215C	32
BEU (1) destination memory width register	BDMWR	R/W	H'FE93 0160	H'FE93 1160	H'FE93 2160	32
BEU (1) destination address Y register	BDAYR	R/W	H'FE93 0164	H'FE93 1164	H'FE93 2164	32
BEU (1) destination address C register	BDACR	R/W	H'FE93 0168	H'FE93 1168	H'FE93 2168	32
BEU (1) address fixed register	BAFXR	R/W	H'FE93 0180	H'FE93 1180	H'FE93 2180	32
BEU (1) swapping register	BSWPR	R/W	H'FE93 0184	H'FE93 1184	H'FE93 2184	32
BEU (1) event interrupt enable register	BEIER	R/W	H'FE93 0188	—	—	32
BEU (1) event register	BEVTR	R/W	H'FE93 018C	—	—	32
BEU (1) register control register	BRCNTR	R/W	H'FE93 0194	—	—	32
BEU (1) status register	BSTAR	R	H'FE93 0198	—	—	32
BEU (1) module reset register	BBRSTR	R/W	H'FE93 019C	—	—	32
BEU (1) register-plane forcible setting register	BRCHR	R/W	H'FE93 01A0	—	—	32
Color Lookup Table	CLUT	R/W	H'FE93 3000 to H'FE93 33FF	—	—	32
BEU (2) start register	BESTR	R/W	H'FE93 4000	—	—	32
BEU (2) source memory width register 1	BSMWR1	R/W	H'FE93 4010	H'FE93 5010	H'FE93 6010	32
BEU (2) source size register 1	BSSZR1	R/W	H'FE93 4014	H'FE93 5014	H'FE93 6014	32
BEU (2) source address Y register 1	BSAYR1	R/W	H'FE93 4018	H'FE93 5018	H'FE93 6018	32
BEU (2) source address C register 1	BSACR1	R/W	H'FE93 401C	H'FE93 501C	H'FE93 601C	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
BEU (2) source address α register 1	BSAAR1	R/W	H'FE93 4020	H'FE93 5020	H'FE93 6020	32
BEU (2) source image format register 1	BSIFR1	R/W	H'FE93 4024	H'FE93 5024	H'FE93 6024	32
BEU (2) source memory width register 2	BSMWR2	R/W	H'FE93 4028	H'FE93 5028	H'FE93 6028	32
BEU (2) source size register 2	BSSZR2	R/W	H'FE93 402C	H'FE93 502C	H'FE93 602C	32
BEU (2) source address Y register 2	BSAYR2	R/W	H'FE93 4030	H'FE93 5030	H'FE93 6030	32
BEU (2) source address C register 2	BSACR2	R/W	H'FE93 4034	H'FE93 5034	H'FE93 6034	32
BEU (2) source address α register 2	BSAAR2	R/W	H'FE93 4038	H'FE93 5038	H'FE93 6038	32
BEU (2) source image format register 2	BSIFR2	R/W	H'FE93 403C	H'FE93 503C	H'FE93 603C	32
BEU (2) source memory width register 3	BSMWR3	R/W	H'FE93 4040	H'FE93 5040	H'FE93 6040	32
BEU (2) source size register 3	BSSZR3	R/W	H'FE93 4044	H'FE93 5044	H'FE93 6044	32
BEU (2) source address Y register 3	BSAYR3	R/W	H'FE93 4048	H'FE93 5048	H'FE93 6048	32
BEU (2) source address C register 3	BSACR3	R/W	H'FE93 404C	H'FE93 504C	H'FE93 604C	32
BEU (2) source address α register 3	BSAAR3	R/W	H'FE93 4050	H'FE93 5050	H'FE93 6050	32
BEU (2) source image format register 3	BSIFR3	R/W	H'FE93 4054	H'FE93 5054	H'FE93 6054	32
BEU (2) tile pattern size register	BTPSR	R/W	H'FE93 4058	H'FE93 5058	H'FE93 6058	32
BEU (2) multidisplay source memory width register 1	BMSMWR1	R/W	H'FE93 4070	—	—	32
BEU (2) multidisplay source size register 1	BMSSZR1	R/W	H'FE93 4074	—	—	32
BEU (2) multidisplay source address Y register 1	BMSAYR1	R/W	H'FE93 4078	—	—	32
BEU (2) multidisplay source address C register 1	BMSACR1	R/W	H'FE93 407C	—	—	32
BEU (2) multidisplay source memory width register 2	BMSMWR2	R/W	H'FE93 4080	—	—	32
BEU (2) multidisplay source size register 2	BMSSZR2	R/W	H'FE93 4084	—	—	32
BEU (2) multidisplay source address Y register 2	BMSAYR2	R/W	H'FE93 4088	—	—	32
BEU (2) multidisplay source address C register 2	BMSACR2	R/W	H'FE93 408C	—	—	32
BEU (2) multidisplay source memory width register 3	BMSMWR3	R/W	H'FE93 4090	—	—	32
BEU (2) multidisplay source size register 3	BMSSZR3	R/W	H'FE93 4094	—	—	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
BEU (2) multidisplay source address Y register 3	BMSAYR3	R/W	H'FE93 4098	—	—	32
BEU (2) multidisplay source address C register 3	BMSACR3	R/W	H'FE93 409C	—	—	32
BEU (2) multidisplay source memory width register 4	BMSMWR4	R/W	H'FE93 40A0	—	—	32
BEU (2) multidisplay source size register 4	BMSSZR4	R/W	H'FE93 40A4	—	—	32
BEU (2) multidisplay source address Y register 4	BMSAYR4	R/W	H'FE93 40A8	—	—	32
BEU (2) multidisplay source address C register 4	BMSACR4	R/W	H'FE93 40AC	—	—	32
BEU (2) multidisplay source image format register	BMSIFR	R/W	H'FE93 40F0	—	—	32
BEU (2) blend control register 0	BBLCR0	R/W	H'FE93 4100	H'FE93 5100	H'FE93 6100	32
BEU (2) blend control register 1	BBLCR1	R/W	H'FE93 4104	—	—	32
BEU (2) process control register	BPROCR	R/W	H'FE93 4108	H'FE93 5108	H'FE93 6108	32
BEU (2) multiwindow control register 0	BMWCR0	R/W	H'FE93 410C	—	—	32
Blend location register 1	BLOCR1	R/W	H'FE93 4114	H'FE93 5114	H'FE93 6114	32
Blend location register 2	BLOCR2	R/W	H'FE93 4118	H'FE93 5118	H'FE93 6118	32
Blend location register 3	BLOCR3	R/W	H'FE93 411C	H'FE93 511C	H'FE93 611C	32
BEU (2) multidisplay location register 1	BMLOCR1	R/W	H'FE93 4120	—	—	32
BEU (2) multidisplay location register 2	BMLOCR2	R/W	H'FE93 4124	—	—	32
BEU (2) multidisplay location register 3	BMLOCR3	R/W	H'FE93 4128	—	—	32
BEU (2) multidisplay location register 4	BMLOCR4	R/W	H'FE93 412C	—	—	32
BEU (2) multidisplay transparent color control register 1	BMPCCR1	R/W	H'FE93 4130	—	—	32
BEU (2) multidisplay transparent color control register 2	BMPCCR2	R/W	H'FE93 4134	—	—	32
Blend pack form register	BPKFR	R/W	H'FE93 4140	H'FE93 5140	H'FE93 6140	32
BEU (2) transparent color control register 0	BPCCR0	R/W	H'FE93 4144	H'FE93 5144	H'FE93 6144	32
BEU (2) transparent color control register 11	BPCCR11	R/W	H'FE93 4148	H'FE93 5148	H'FE93 6148	32
BEU (2) transparent color control register 12	BPCCR12	R/W	H'FE93 414C	H'FE93 514C	H'FE93 614C	32
BEU (2) transparent color control register 21	BPCCR21	R/W	H'FE93 4150	H'FE93 5150	H'FE93 6150	32

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
BEU (2) transparent color control register 22	BPCCR22	R/W	H'FE93 4154	H'FE93 5154	H'FE93 6154	32
BEU (2) transparent color control register 31	BPCCR31	R/W	H'FE93 4158	H'FE93 5158	H'FE93 6158	32
BEU (2) transparent color control register 32	BPCCR32	R/W	H'FE93 415C	H'FE93 515C	H'FE93 615C	32
BEU (2) destination memory width register	BDMWR	R/W	H'FE93 4160	H'FE93 5160	H'FE93 6160	32
BEU (2) destination address Y register	BDAYR	R/W	H'FE93 4164	H'FE93 5164	H'FE93 6164	32
BEU (2) destination address C register	BDACR	R/W	H'FE93 4168	H'FE93 5168	H'FE93 6168	32
BEU (2) address fixed register	BAFXR	R/W	H'FE93 4180	H'FE93 5180	H'FE93 6180	32
BEU (2) swapping register	BSWPR	R/W	H'FE93 4184	H'FE93 5184	H'FE93 6184	32
BEU (2) event interrupt enable register	BEIER	R/W	H'FE93 4188	—	—	32
BEU (2) event register	BEVTR	R/W	H'FE93 418C	—	—	32
BEU (2) register control register	BRCNTR	R/W	H'FE93 4194	—	—	32
BEU (2) status register	BSTAR	R	H'FE93 4198	—	—	32
BEU (2) module reset register	BBRSTR	R/W	H'FE93 419C	—	—	32
BEU (2) register-plane forcible setting register	BRCHR	R/W	H'FE93 41A0	—	—	32
Color Lookup Table	CLUT	R/W	H'FE93 7000 to H'FE93 73FF	—	—	32

Table 36.4 BEU Register States in Each Processing Mode

Register Abbreviation	Power-on Reset	Manual-Reset	Software-Standby	Module-Standby	R-Standby	U Standby	Sleep
BESTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSMWR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSSZR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSAYR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSACR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSAAR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSIFR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSMWR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSSZR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSAYR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSACR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSAAR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSIFR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSMWR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSSZR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSAYR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSACR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSAAR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSIFR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BTPSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSMWR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSSZR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSAYR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSACR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSMWR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSSZR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSAYR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSACR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSMWR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Register Abbreviation	Power-on Reset	Manual-Reset	Software-Standby	Module-Standby	R-Standby	U Standby	Sleep
BMSSZR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSAYR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSACR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSMWR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSSZR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSAYR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSACR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMSIFR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BBLCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BBLCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPROCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMWCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BLOCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BLOCR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BLOCR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMLOCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMLOCR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMLOCR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMLOCR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMPCCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BMPCCR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPKFR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR11	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR21	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR22	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR31	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BPCCR32	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BDMWR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

Register Abbreviation	Power-on Reset	Manual-Reset	Software-Standby	Module-Standby	R-Standby	U Standby	Sleep
BDAYR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BDACR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BAFXR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSWPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BEIER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BEVTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BRCNTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BSTAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BBRSTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
BRCHR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CLUT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

36.3.1 BEU Start Register (BESTR)

BESTR activates and halts the BEU, and enables or disables input displays. Before activating the BEU, all registers that specify the operating parameters, except for this register, must be set.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CHON3	CHON2	CHON1	—	—	—	—	—	—	—	BEIVK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	CHON3	0	R/W	Each bit specifies whether the corresponding input system is used or not. 1 must be written to the bit corresponding to the input system selected as the parent display by the PWD bits in BBLCR1. If the parent display is set not to be used, the BEVIO bit in BEVTR is set to 1 immediately after the BEU is activated, and processing is halted. CHONn (n = 1 to 3) 0: Input system n is not used. 1: Input system n is used.
9	CHON2	0	R/W	
8	CHON1	0	R/W	
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	BEIVK	0	R/W	Controls the start and stop of BEU processing. If 1 is written to this bit when the BEU is halted, the BEPRC bit in BSTAR is set to 1 and blending starts. In addition, if 0 is written to this bit when the BEU is in the operating state (BEPRC bit = 1), BEU processing is halted after the software reset processing is carried out. Note that if 0 is written to this bit when the BEU is halted (BEPRC bit = 0), no operation takes place. 0: NOP (software reset processing if 0 is written to this bit when BEPRC bit = 1). 1: BEU processing starts.

36.3.2 BEU Source Memory Width Registers 1 to 3 (BSMWR1 to BSMWR3)

BSMWR are registers that set the memory width of the source memory areas of BEU input systems 1 to 3.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSMW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2	BSMW[15:2]	H'0000	R/W	These bits set the source memory area width of input systems 1 to 3 (2-longword or longword units). These bits set the horizontal width of the source memory area where the source image is located. These bits are set in 2-longword units for input data in the YCbCr or RGB24bpp (stuffing) format, and in longword units for input data in other formats. For a YCbCr 4:4:4 image, the setting is doubled in the BEU because the data size of the C plane is twice as large as that of the Y plane.
1, 0	BSMW[1:0]		R	

36.3.3 BEU Source Size Registers 1 to 3 (BSSZR1 to BSSZR3)

BSSZR are registers that set the vertical and horizontal sizes of the images read by BEU input systems 1 to 3.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	BVSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	RW

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BHSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	BVSS[11:0]	H'000	R/W	These bits set the number of vertical pixels read by input systems 1 to 3. These bits set the number of vertical read pixels of the source image read by input systems 1 to 3. When input system n is YCbCr 4:2:0, these bits can be set in 2-pixel units. For other formats, these bits can be set in 1-pixel units. When input system n is the parent display, the size set in these bits becomes the destination image size. Therefore, when the BEU operates together with the LCDC, make this setting match the destination image size of the interacting block. When input system n is the child display and YCbCr4:2:0, the data up to the 2-pixel units should be ready in the memory. Specify 4-pixel units for BHSS when input system 3 is in OSD mode.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
11 to 0	BHSS[11:0]	H'000	R/W	These bits set the number of horizontal pixels read by input systems 1 to 3. These bits set the number of horizontal read pixels of the source image read by input systems 1 to 3. When input system n is YCbCr4:2:2 or YCbCr4:2:0, these bits can be set in 2-pixel units, for other formats these bits can be set in 1-pixel units. The setting should be in 8-pixel units in LCDC mode (only for the parent display size). When input system n is the parent display, the size set in these bits becomes the destination image size. Therefore, when the BEU operates together with the LCDC, make this setting match the destination image size of the interacting block. When input system n is the child display and YCbCr4:2:0, the data up to the 2-pixel units should be ready in the memory. Specify 4-pixel units for BHSS when input system 3 is in OSD mode.

36.3.4 BEU Source Address Y Registers 1 to 3 (BSAYR1 to BSAYR3)

BSAYR are registers that set the start address for the Y/RGB plane of the images read by BEU input systems 1 to 3.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BSAY[31:2]	H'0000 0000	R/W	These bits set the start address for reading the Y/RGB plane of input systems 1 to 3 (2-longword or longword units). When reading a YCbCr image, set the start address of the Y plane. When reading an RGB image, set the start address of the RGB plane. These bits are set in 2-longword units for input data in the YCbCr or RGB24bpp (stuffing) format, and in longword units for input data in other formats.
1, 0	BSAY[1:0]		R	

36.3.5 BEU Source Address C Registers 1 to 3 (BSACR1 to BSACR3)

BSACR are registers that set the start address for the C plane of the images read by BEU input systems 1 to 3.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BSAC[31:2]	H'0000 0000	R/W	These bits set the start address for reading the C plane of input systems 1 to 3 (longword units). When reading a YCbCr image, set the start address of the C plane. Note that when reading an RGB image, these registers are not used.
1, 0	BSAC[1:0]		R	

36.3.6 BEU Source Address α Registers 1 to 3 (BSAAR1 to BSAAR3)

BSAAR are registers that set the start address for the α plane of the images read by BEU input systems 1 to 3.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSAA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSAA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BSAA[31:2]	H'0000 0000	R/W	These bits set the start address for reading the α plane of input systems 1 to 3 (2-longword units). When reading a YCbCr image, set the start address of the α plane. Note that when reading an RGB image, these registers are not used.
1, 0	BSAA[1:0]		R	

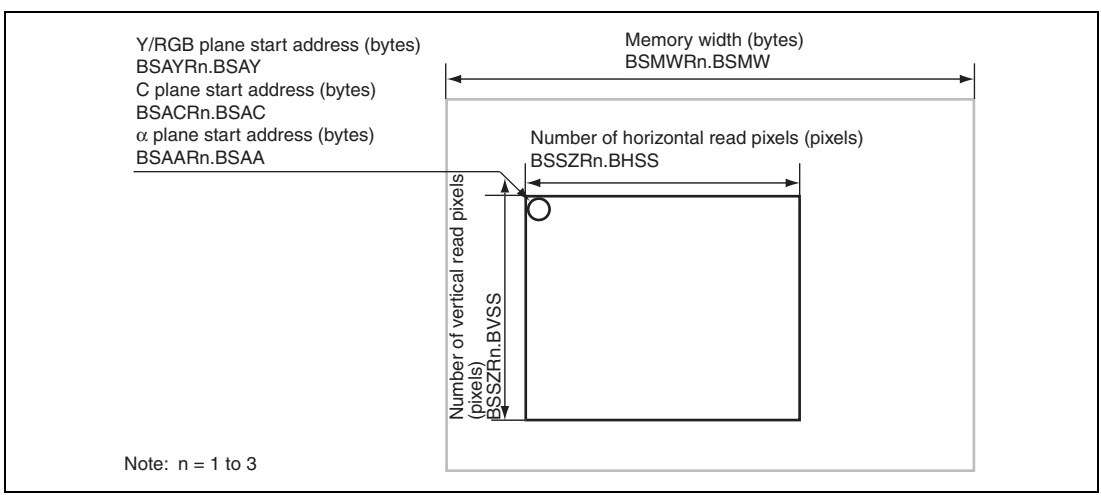


Figure 36.2 Number of Read Pixels of Input System n and Start Addresses

36.3.7 BEU Source Image Format Registers 1 to 3 (BSIFR1 to BSIFR3)

BSIFR are registers that specify the format of the images read by BEU input systems 1 to 3.

In the read-only bits, the write value should always be 0. If an attempt is made to write 1 to any read-only bit, malfunction may occur.

- Input system 1 (BSIFR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	IN1TM2	IN1TM	IN1TE	—	CHRR[2:0]			—	—	—	RPKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	IN1TM2	0	R/W	This bit selects whether the ITU-R BT.601 standard or the ITU-R BT.709 standard for YCbCr are used with 8-bit full-scale RGB color conversion. 0: The ITU-R BT.601 standard for YCbCr with 8-bit full-scale RGB color conversion are used. 1: The ITU-R BT.709 standard for YCbCr with 8-bit full-scale RGB color conversion are used.
13	IN1TM	0	R/W	Selects whether to use color conversion of ITU-R BT.601 defined YCbCr and 8-bit full-scale RGB or color conversion of 8-bit full-scale YCbCr and 8-bit full-scale RGB. 0: Uses color conversion of full-range RGB[0,255] and compressed-range YCbCr[16,235/240]. 1: Uses color conversion of full-range RGB[0,255] and full-range YCbCr[0,255].

Bit	Bit Name	Initial Value	R/W	Description
12	IN1TE	0	R/W	Turns on or off the RGB ⇔ YCbCr conversion circuit. 0: No RGB ⇔ YCbCr conversion 1: RGB ⇔ YCbCr conversion performed Note: When performing RGB ⇔ YCbCr conversion, read the source image so that the conversion result has the same format as the other input systems.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	CHRR[2:0]	000	R/W	The format of the image input to input system 1 is determined by the settings of the RY bit in BPKFR and the IN1TE bit. When a YCbCr image is input to input system 1, the data pack form can be selected by the setting of these bits. The data pack forms are shown in Table 36.6. 000: Image read in the YCbCr 4:4:4 format 001: Image read in the YCbCr 4:2:2 format 010: Image read in the YCbCr 4:2:0 format 011: Image read in the αYCbCr 4:4:4 format 100: Image read in the αYCbCr 4:2:2 format 101: Image read in the αYCbCr 4:2:0 format Others: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	RPKF[4:0]	00000	R/W	These bits set the RGB input data pack form when the image input to input system 1 is in the RGB format. The format of the image input to input system 1 is determined by the settings of the RY bit in BPKFR and the IN1TE bit. When an RGB image is input to input system 1, the data pack form can be selected from the data pack forms in Table 36.5 by the setting of these bits.

- Input system 2 (BSIFR2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CHRR[2:0]			—	—	—	RPKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	CHRR[2:0]	000	R/W	When the RY bit in BPKFR is 0, the image input to input system 2 is in the YCbCr format, and the data pack form can be selected by the setting of these bits. The data pack forms for a YCbCr image input in input system 2 are shown in Table 36.6. 000: Image read in the YCbCr 4:4:4 format 001: Image read in the YCbCr 4:2:2 format 010: Image read in the YCbCr 4:2:0 format 011: Image read in the αYCbCr 4:4:4 format 100: Image read in the αYCbCr 4:2:2 format 101: Image read in the αYCbCr 4:2:0 format Others: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	RPKF[4:0]	00000	R/W	These bits set the RGB input data pack form when the image input to input system 2 is in the RGB format. When the RY bit in BPKFR is 1, the image input to input system 2 is in the RGB format, and the data pack form can be selected from the data pack forms in Table 36.5 by the setting of these bits.

- Input system 3 (BSIFR3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MOD1	MOD0	—	CHRR[2:0]			—	—	—	RPKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	MOD1	0	R/W	Sets the method for fetching source image data when input system 3 is in OSD mode. This bit should be set to 1 (CLUT) only when input system 3 is in OSD mode. Access to read from or write to the CLUT during operation is prohibited. When input system 3 is an OSD and this bit is 1, the color information and alpha values are obtained by referencing the CLUT. The CLUT contents can be modified by longword access from H'FE93 3000 to H'FE93 33FF. The color information and alpha values of the CLUT are stored in the order shown in Figure 36.3. Note that the initial values of the CLUT are undefined. 0: Same graphic image or video input as input systems 1 and 2 1: Color information and alpha values are referenced from the CLUT.

Bit	Bit Name	Initial Value	R/W	Description
12	MOD0	0	R/W	Sets whether input system 3 is operated in normal mode or OSD mode. When this bit is 0, input system 3 handles graphic images and video inputs, and the operation is the same as that of input system 1 or 2. In this case, the number of read pixels of the source image is set by BSSZR3. When this bit is 1, input system 3 handles OSD images, and supports repeated pattern output. Repeated pattern output indicates output of tiles with the same picture in rows. The size of one tile is determined by the BHSS and BVSS bits in BSSZR3, and the size in which to paste it to the destination display is determined by the TPHS and TPVS bits in BTPSR. The output appears different according to the magnitude relation of the BSSZR3 and BTPSR settings, as shown in Figure 36.4. 0: Input system 3 is used as the same Graphic or Video as input systems 1 and 2 1: Input system 3 is used as OSD.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	CHRR[2:0]	000	R/W	When the RY bit in BPKFR is 0, the image input to input system 3 is in the YCbCr format, and the data pack form can be selected by the setting of these bits. The data pack forms for a YCbCr image input in input system 3 are shown in Table 36.6. 000: Image read in the YCbCr 4:4:4 format 001: Image read in the YCbCr 4:2:2 format 010: Image read in the YCbCr 4:2:0 format 011: Image read in the αYCbCr 4:4:4 format 100: Image read in the αYCbCr 4:2:2 format 101: Image read in the αYCbCr 4:2:0 format Others: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	RPKF[4:0]	00000	R/W	These bits set the RGB input data pack form when the image input to input system 3 is in the RGB format. When the RY bit in BPKFR is 1, the image input to input system 3 is in the RGB format, and the data pack form can be selected from the data pack forms in Table 36.5 by the setting of these bits.

Table 36.5 RGB Data Input Pack Forms

No	RPKF [4:0]	Bit Rate [bpp]	Phase	Bit																																					
				31 to 24								23 to 16								15 to 8								7 to 0													
0	B'00000	24	—	α	α	α	α	α	α	α	α	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0								
1	B'00001		—	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	α	α	α	α	α	α	α	α								
2	B'00010		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1								
			1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2								
			2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3								
3	B'00011	16		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1								
7	B'00111	18		—	—	—	—	—	—	—	—	—	—	—	—	—	—	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0								
11	B'01011	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1								
			1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2								
			2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3								
12	B'01100	24		α	α	α	α	α	α	α	α	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0								
13	B'01101	16		—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0								

Table 36.6 YCbCr Data Pack Forms

YCbCr 4:4:4 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1

YCbCr 4:2:2/4:2:0 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2

αYCbCr 4:4:4 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
α data	α0	α0	α0	α0	α0	α0	α0	α0	α1	α1	α1	α1	α1	α1	α1	α1	α2	α2	α2	α2	α2	α2	α2	α2	α3	α3	α3	α3	α3	α3	α3	α3
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1

αYCbCr 4:2:2/4:2:0 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
α data	α0	α0	α0	α0	α0	α0	α0	α0	α1	α1	α1	α1	α1	α1	α1	α1	α2	α2	α2	α2	α2	α2	α2	α2	α3	α3	α3	α3	α3	α3	α3	α3
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2

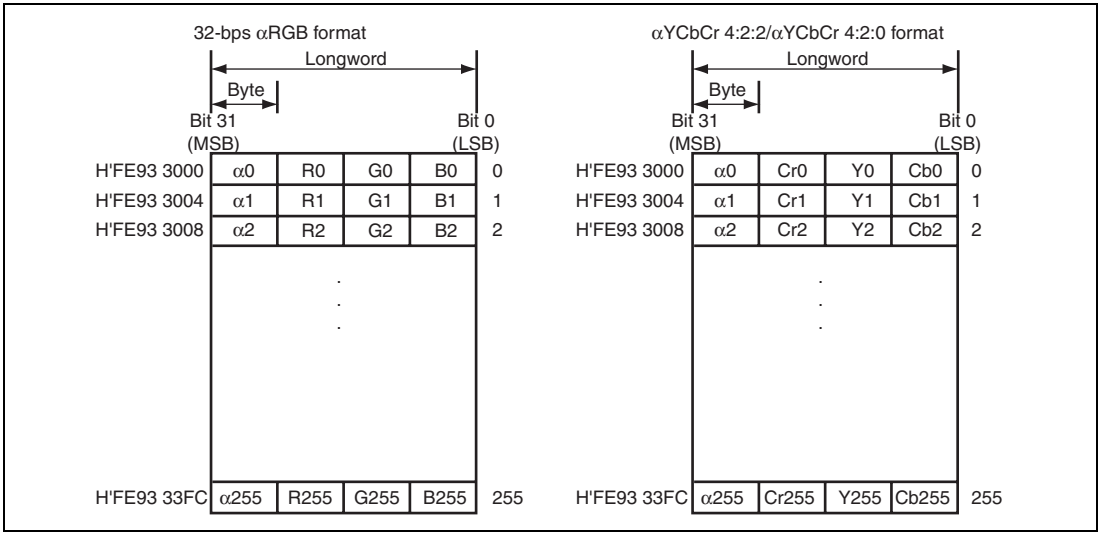


Figure 36.3 Order of Storing CLUT Color Information

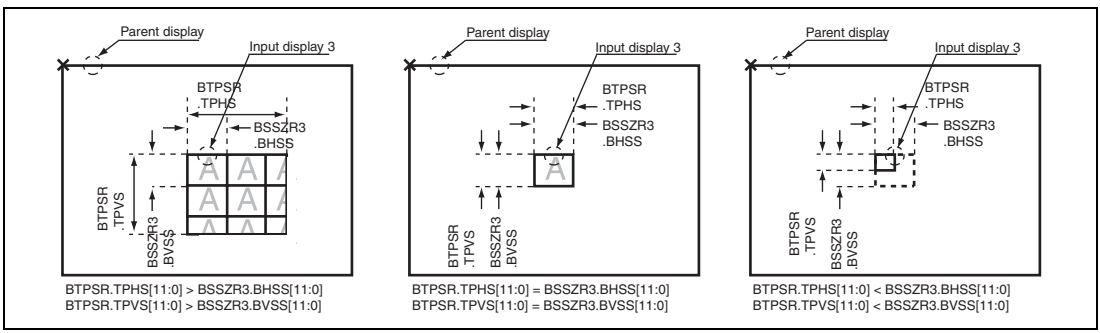


Figure 36.4 Repeated Pattern Output

36.3.8 BEU Tile Pattern Size Register (BTPSR)

BTPSR specifies the size for pasting images to the destination display when BEU input system 3 is in OSD mode.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	TPVS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TPHS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	TPVS[11:0]	H'000	R/W	These bits set the vertical output size of a tile pattern. Set the vertical size for pasting to the destination display when OSD mode is selected for input system 3 in these bits. For These bits can be set in 1-pixel units. This setting is ignored when input system 3 is not in OSD mode.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	TPHS[11:0]	H'000	R/W	These bits set the horizontal output size of a tile pattern. Set the horizontal size for pasting to the destination display when OSD mode is selected for input system 3 in these bits. These bits can be set in 1-pixel units. This setting is ignored when input system 3 is not in OSD mode.

36.3.9 BEU Multidisplay Source Memory Width Registers 1 to 4 (BMSMWR1 to BMSMWR4)

BMSMWR are registers that set the memory width of the source memory areas of multidisplays 1 to 4.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BSMW[17:16]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSMW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 to 2	BSMW[17:2]	H'00000	R/W	These bits set the source memory area width of multidisplays (2-longword or longword units). These bits are set in 2-longword units for input data in the YCbCr or RGB24bpp (stuffing) format, and in longword units for input data in other formats. As the setting is in 2-longword or longword units, the lower two bits are for read-only.
1, 0	BSMW[1:0]		R	

36.3.10 BEU Multidisplay Source Size Registers 1 to 4 (BMSSZR1 to BMSSZR4)

BMSSZR are registers that set the vertical and horizontal sizes of the images read by multidisplays 1 to 4.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	BVSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	RW

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	BHSS[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	BVSS[11:0]	H'000	R/W	These bits set the number of vertical read pixels of the source image read by multidisplays. For YCbCr 4:2:0 format, these bits can be set in 2-pixel units. For other formats, these bits can be set in 1-pixel units.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	BHSS[11:0]	H'000	R/W	These bits set the number of horizontal read pixels of the source image read by multidisplays. For YCbCr 4:2:2 and YCbCr 4:2:0 formats, these bits can be set in 2-pixel units. For other formats, these bits can be set in 1-pixel units.

Note: For each multidisplay, in the horizontal direction the lower 2 bits of the CHLC bits in BMLOCRn are truncated. From that location, the value specified by bits CHLC1 and CHLC0 in BMLOCRn is added to the value specified by bits BHSS in BMSSZRn. The resulting value is rounded up to 4-pixel units and an area of the resulting number of pixels is treated as an image manipulation area. A setting causing multidisplays 1 to 4 to overlap each other and a setting in which the image manipulation area runs into the parent screen are prohibited.

36.3.11 BEU Multidisplay Source Address Y Registers 1 to 4 (BMSAYR1 to BMSAYR4)

BMSAYR are registers that set the start address for the Y/RGB plane of the images read by multidisplays 1 to 4.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BSAY[31:2]	H'0000 0000	R/W	When reading a YCbCr image, set the start address of the Y plane. When reading an RGB image, set the start address of the RGB plane. These bits are set in 2-longword units for input data in the YCbCr or RGB24bpp (stuffing) format, and in longword units for input data in other formats. As the setting is in 2-longword or longword units, the lower two bits are for read-only.
1, 0	BSAY[1:0]		R	

36.3.12 BEU Multidisplay Source Address C Registers 1 to 4 (BMSACR1 to BMSACR4)

BMSACR are registers that set the start address for the C plane of the images read by multidisplays 1 to 4.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BSAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BSAC[31:2]	H'0000 0000	R/W	When reading a YCbCr image, set the start address of the C plane. These bits are set in 2-longword units. As the setting is in 2-longword units, the lower two bits are for read-only. Note that when reading an RGB image, these registers are not used.
1, 0	BSAC[1:0]		R	

36.3.13 BEU Multidisplay Source Image Format Register (BMSIFR)

BMSIFR specifies the format of the images read by multidisplays 1 to 4.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CHRR[1:0]		—	—	—	RPKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	CHRR[1:0]	00	R/W	When the RY bit in BPKFR is 0, the image input to the multiwindow is in the YCbCr format, and the data pack form can be selected by the setting of these bits. The data pack forms for a YCbCr image input in the multiwindow are shown in Table 36.7. 00: Image read as YCbCr 4:4:4 when the image input in the multiwindow is in the YCbCr format 01: Image read as YCbCr 4:2:2 when the image input in the multiwindow is in the YCbCr format 10: Image read as YCbCr 4:2:0 when the image input in the multiwindow is in the YCbCr format 11: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	RPKF[4:0]	00000	R/W	These bits set the RGB input data pack form when the image input to the multiwindow is in the RGB format. When the RY bit in BPKFR is 1, the image input to the multiwindow is in the RGB format, and the data pack form can be selected from the data pack forms in Table 36.8 by the setting of these bits.

Table 36.7 YCbCr Data Pack Forms

YCbCr 4:4:4 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1

YCbCr 4:2:2/4:2:0 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2

Table 36.8 RGB Data Input Pack Forms

No	RPKF [4:0]	Bit Rate [bpp]	Phase	Bit																															
				31 to 24								23 to 16								15 to 8								7 to 0							
0	B'00000	24	—	α	α	α	α	α	α	α	α	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
1	B'00001		—	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	α	α	α	α	α	α	α	α
2	B'00010		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
			1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
			2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
3	B'00011	16		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
7	B'00111	18		—	—	—	—	—	—	—	—	—	—	—	—	—	—	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
11	B'01011	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
			1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
			2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
12	B'01100	24		α	α	α	α	α	α	α	α	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
13	B'01101	16		—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0

36.3.14 BEU Blend Control Register 0 (BBLCR0)

BBLCR0 sets the blending and output mode for the BEU input systems.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	AMUX3	AMUX2	AMUX1	—	LAY[2:0]			V3AP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	V2AP[7:0]								V1AP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 29 28	AMUX3 AMUX2 AMUX1	0 0 0	R/W R/W R/W	Each bit selects whether the alpha value of the pixel data of the source display should be used or the alpha value specified in layer units by the V3AP, V2AP, or V1AP bits should be used for blending at the corresponding input system. However, when the alpha value of the pixel data is set to be used (these bits are set to 1) and an input pack form without an alpha value is specified, an alpha value of 255 is used instead. Table 36.9 shows the alpha values used at blending according to the setting of this register. 0: Alpha value in layer units set for input system n is used at blending (n = 1 to 3). 1: Alpha value of pixel data is used at blending.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 24	LAY[2:0]	000	R/W	These bits specify the overlay order of the layers at blending. The layers are overlaid in the order set by these bits as shown in Table 36.10.
23 to 16	V3AP[7:0]	H'00	R/W	These bits specify the alpha value for input system 3. A value from 0 to 255 can be set. 0 sets full transparency and 255 sets full opacity for blending.
15 to 8	V2AP[7:0]	H'00	R/W	These bits specify the alpha value for input system 2. A value from 0 to 255 can be set. 0 sets full transparency and 255 sets full opacity for blending.
7 to 0	V1AP[7:0]	H'00	R/W	These bits specify the alpha value for input system 1. A value from 0 to 255 can be set. 0 sets full transparency and 255 sets full opacity for blending.

Table 36.9 Selection of Input Alpha Values

Register Settings						Used Alpha Value
BSIFRn.MOD Graphic/OSD	**BPKFR.RY**	**BSIFRn.RPKF[4:0] CHRR[4:0] Pack Form**	**BBLCR0.AMUXn**	**BPCCR0.SAPExx**	**BPCCRxx.R/GY/B[15:0] Data Comparison**	**Alpha Value**
0: Graphic	0: YCbCr	All pack forms	0	0	—	BBLCR0.VnAP[7:0] value
				1	Match	BPCCRxx.AP[7:0] value
					No match	BBLCR0.VnAP[7:0] value
			1	0	—	Including α: α value in pixel units Not including α: 255
				1	Match	BPCCRxx.AP[7:0] value
					No match	Including α: α value in pixel units Not including α: 255
	1: RGB	Pack form other than RGB888, RGB666, or RGB565	0	—	—	BBLCR0.VnAP[7:0] value
			1	—	—	255
		RGB888, RGB666, or RGB565 pack form	0	0	—	BBLCR0.VnAP[7:0] value
				1	Match	BPCCRxx.AP[7:0] value
					No match	BBLCR0.VnAP[7:0] value
			1	0	—	Including α: α value in pixel units Not including α: 255
				1	Match	BPCCRxx.AP[7:0] value
					No match	Including α: α value in pixel units Not including α: 255
1: OSD	—	—	0	—	—	BBLCR0.VnAP[7:0] value
		—	1	—	—	α value in color units stored in CLUT

Note: n = 1 to 3, xx = 11, 12, 21, 22, 31, or 32

Table 36.10 Layer Blending Order Specified by LAY Bits

LAY Bits	Front Layer	Middle Layer	Back Layer
B'000	Input system 3	Input system 2	Input system 1
B'001	Input system 2	Input system 3	Input system 1
B'010	Input system 3	Input system 1	Input system 2
B'011	Input system 1	Input system 3	Input system 2
B'100	Input system 2	Input system 1	Input system 3
B'101	Input system 1	Input system 2	Input system 3
B'110	Setting prohibited		
B'111	Setting prohibited		

36.3.15 BEU Blend Control Register 1 (BBLCR1)

BBLCR1 sets blending of the BEU input systems.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PWD[1:0]		—	—	—	—	—	MT[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VSKIP	—	—	—	DPMD	—	—	—	PXA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
25, 24	PWD[1:0]	00	R/W	These bits specify the parent display at BEU processing. The source image read size (BSSZR1 to BSSZR3) of the input system selected by these bits as the parent display is used as the destination display size. 00: Input system 1 is set as the parent display. 01: Input system 2 is set as the parent display. 10: Input system 3 is set as the parent display. 11: Setting prohibited
23 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	MT[2:0]	001	R/W	These bits specify the output mode for BEU processing*. 001: Memory output mode 100: LCDC output mode 101: Memory and LCDC simultaneous output mode Other than above: Setting prohibited
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VSKIP	0	R/W	When this bit is 0, the number of pixels for the size specified by BSSZR1 to BSSZR3 are read at reading of the source display. The destination display size matches the size of the parent display specified by the PWD bits. When this bit is 1, the number of pixels for half of the vertical size specified by BSSZR1 to BSSZR3 are read at reading of the source display by reading only the odd-numbered lines. The destination display size becomes half the size of the parent display specified by the PWD bits. Figure 36.5 shows an example in which input system 1 is selected as the parent display. 0: All lines of the source display are read. 1: Every other line of the source display is read. Note: When this bit is 1, the vertical location of a child display also becomes half.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DPMD	0	R/W	This bit selects from MSB padding and zero padding for lower-bit extension when converting to another format with RGB565 input. There are two types of target formats, PKF = B'00011 and PKF = B'01101 RGB formats. Other formats are not supported. 0: 0 padding (backward compatibility) 1: MSB padding This setting enables input 1, 2, and 3 for all multidisplays.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PXA	0	R/W	Selects the pixel-unit alpha value of the input system or the alpha value specified in the PAD bits in BPKFR as the value to be added to the PAD field when the RGB output pack includes the PAD field. The pixel-unit alpha value of the input system refers to the alpha value selected by the AMUX1 to AMUX3 bits in BBLCR0. When the pixel-unit alpha value of the input system is to be output, the alpha value of the bottom display is given priority, as shown in Table 36.11. 0: Alpha value set in the PAD bits in BPKFR is used for the PAD field in the destination image (RGB or YCbCr image). 1: Pixel-unit alpha value of the input system is used for the PAD field in the destination image (RGB or YCbCr image).

Note: * In LCDC output mode, the BEU starts operating by activating the LCDC after the BEU has been activated.

For terminating the BEU in LCDC output mode, issue a software reset of the BEU after the LCDC has been terminated.

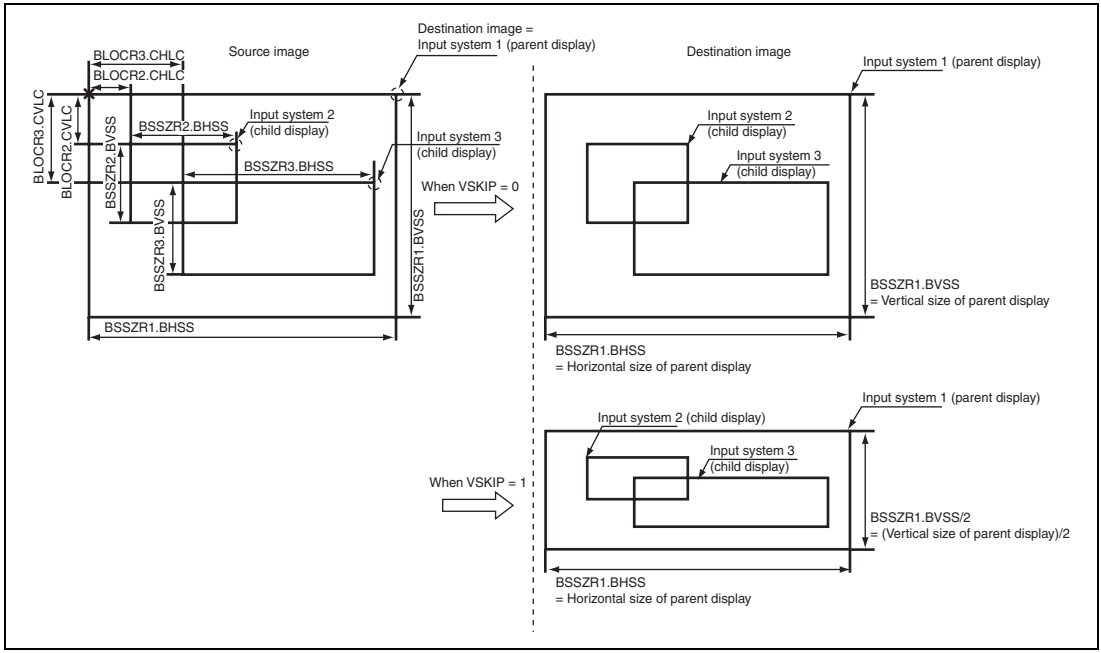


Figure 36.5 Comparison of Destination Displays with Skipping at Reading Enabled and Disabled

Table 36.11 Priority of Pixel-Unit Alpha Values of Input Systems

Top-Layer Display	Middle-Layer Display	Bottom-Layer Display	Alpha Value
Contains pixels	Contains pixels	Contains pixels	Bottom-layer display
Contains pixels	Contains pixels	No pixels	Middle-layer display
Contains pixels	No pixels	Contains pixels	Bottom-layer display
No pixels	Contains pixels	Contains pixels	Bottom-layer display
Contains pixels	No pixels	No pixels	Top-layer display
No pixels	No pixels	Contains pixels	Bottom-layer display
No pixels	Contains pixels	No pixels	Middle-layer display

36.3.16 BEU Process Control Register (BPROCR)

BPROCR controls which processings the BEU performs for an image.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BRSEL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MODE[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	BRSEL	0	R/W	Selects blending or raster operation 2. 0: Enables blending. 1: Enables raster operation 2.
30 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	MODE[1:0]	00	R/W	These bits select the drawing color processing when raster operation 2 is selected. The setting of these bits is invalidated when blending is selected. 00: Addition (adds the pixel colors of the parent display and child display) Output display = child display × child display mixing rate + parent display × parent display mixing rate 01: Replacement (replaces the pixel colors of the parent display with those of the child display) Output display = child display × child display mixing rate 10: Subtraction (subtracts the pixel colors of the child display from those of the parent display) Output display = parent display × parent display mixing rate − child display × child display mixing rate 11: Setting prohibited

In the image processing unit in Figure 36.6, either blending or raster operation 2 is selected by the BRSEL bit. Figure 36.7 shows an output example in which addition is performed when raster operation 2 is selected.

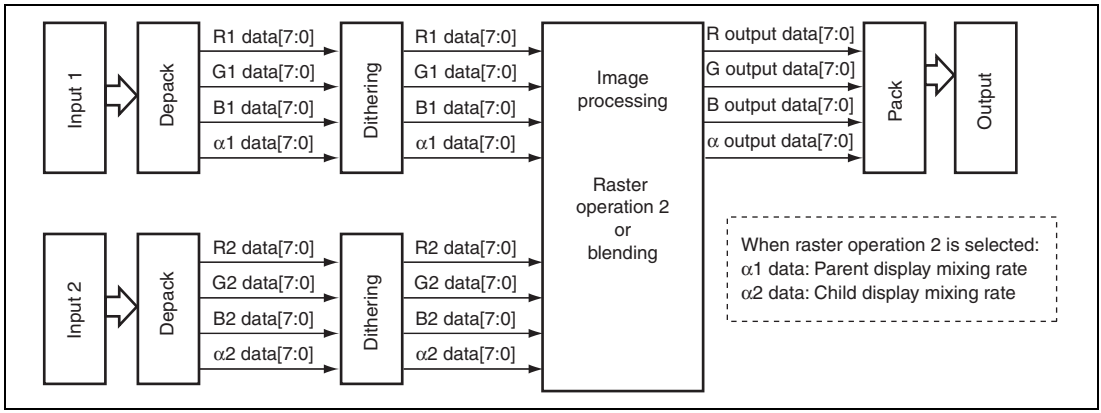


Figure 36.6 Image Processing Selection

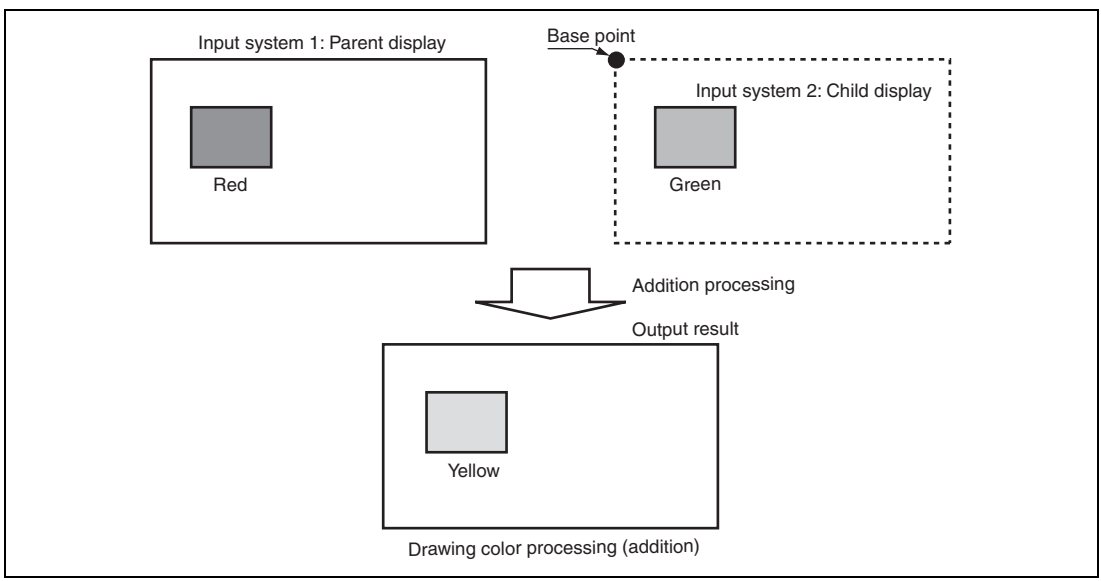


Figure 36.7 Output Example when Raster Operation 2 Processing is Addition

36.3.17 BEU Multiwindow Control Register 0 (BMWCR0)

BMWCR0 enables or disables the multiwindow function and sets whether multidisplays 1 to 4 are used or not.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MWEN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MSON4	MSON3	MSON2	MSON1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MWEN	0	R/W	Enables or disables the multiwindow function. 0: Enables the multiwindow function. 1: Disables the multiwindow function.
30 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MSON4	0	R/W	These bits set the displays to be used in the multiwindow. Up to four multidisplays can be used. The setting of these bits is invalidated when the MWEN bit is cleared to 0. MSONn (n = 1 to 4) 0: Multidisplay n is not used. 1: Multidisplay n is used.
2	MSON3	0	R/W	
1	MSON2	0	R/W	
0	MSON1	0	R/W	

36.3.18 Blend Location Registers 1 to 3 (BLOCR1 to BLOCR3)

BLOCR are registers that specify the locations for input systems 1 to 3 at BEU blending.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CVLC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CHLC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	CVLC[11:0]	H'000	R/W	These bits specify the number of pixels for the vertical offset from the base point as the blending start location of input systems 1 to 3 (see Figure 36.9, 1-pixel units).
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CHLC[11:0]	H'000	R/W	These bits specify the number of pixels for the horizontal offset from the base point as the blending start location of input systems 1 to 3.

Figures 36.8 to 36.11 show destination displays of the BEU. The destination display size matches that of the parent display size. The parent display can be selected as input system 1, 2, or 3 by the PWD bits in BBLCR1. The order in which to overlay these input systems can be specified as desired, independent of the parent display setting, by the LAY bits in BBLCR0. Blending operation is not performed for areas where the source displays do not overlap. Areas in a child display which overflow from the parent display are not written to the destination display.

The top-left corner of the parent display (= destination display) becomes the base point. The locations of the child displays are expressed by offsets from the base point. For example, if input display 1 is set as the parent display, the top-left corner of input display 1 becomes the base point, the location of input display 2 is determined by the CHLC and CVLC bits in BLOCR2, and the location of input display 3 is determined by the CHLC and CVLC bits in BLOCR3.

When operating together with the LCDC, set the parent display size to match the output size of the interacting block.

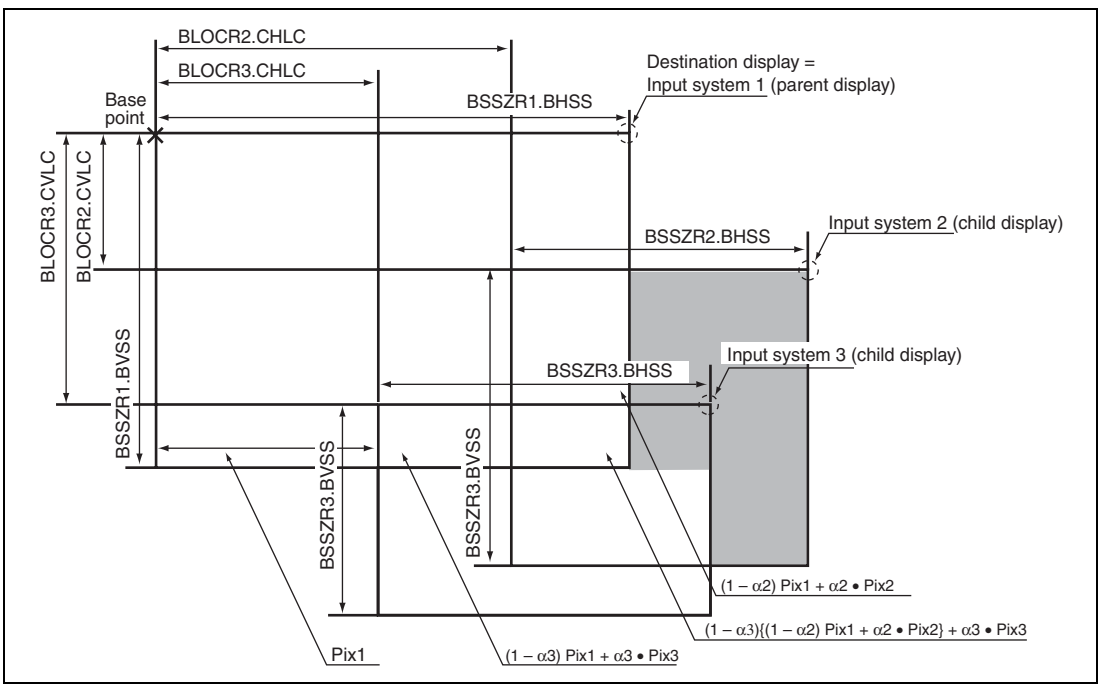


Figure 36.8 BEU Destination Display

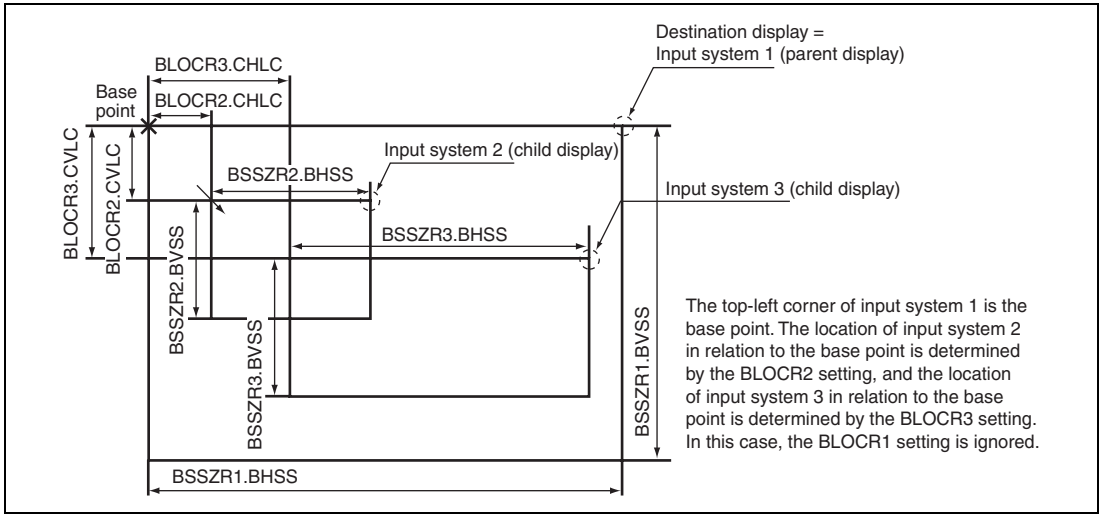


Figure 36.9 Destination Display when Input System 1 is Parent Display

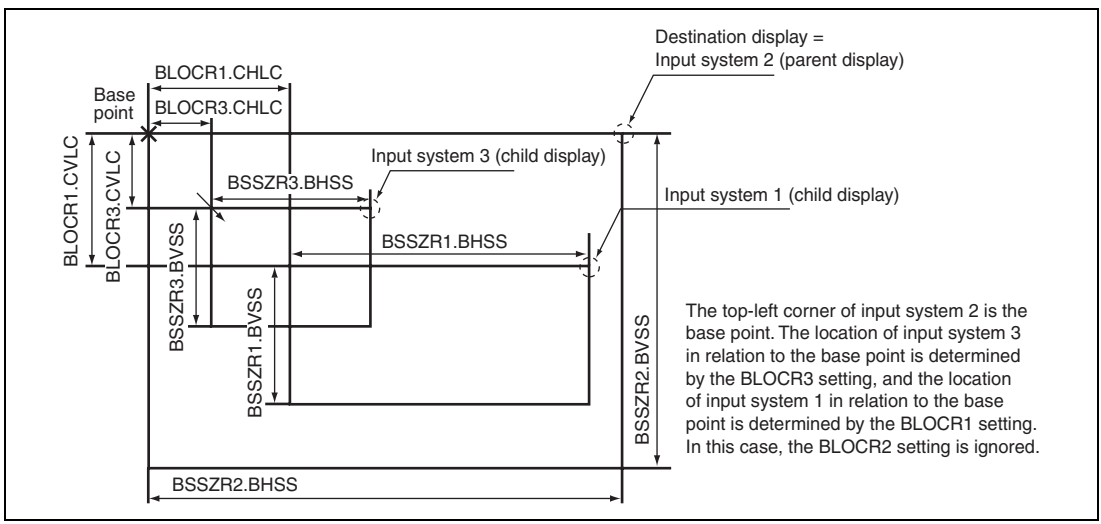


Figure 36.10 Destination Display when Input System 2 is Parent Display

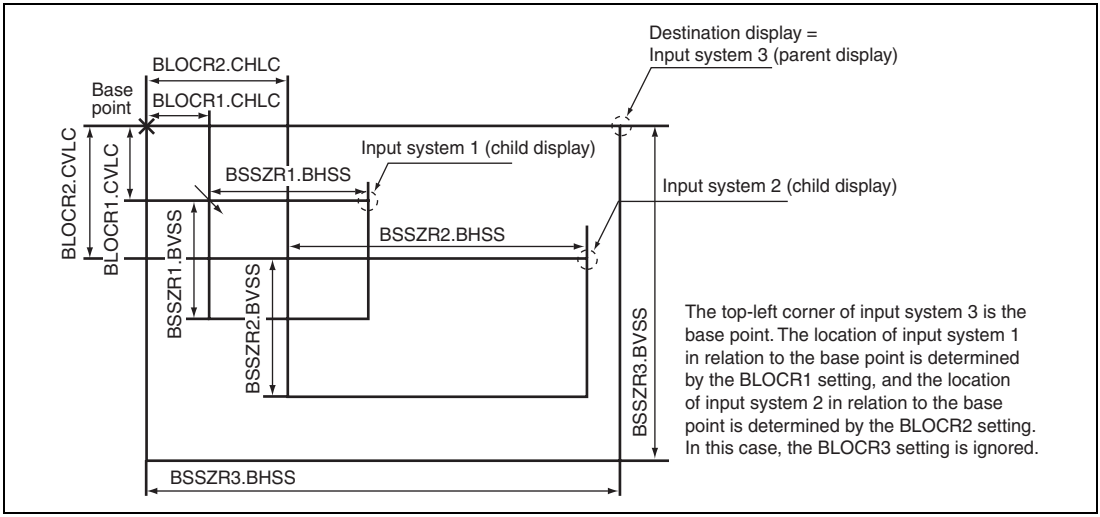


Figure 36.11 Destination Display when Input System 3 is Parent Display

36.3.19 BEU Multidisplay Location Registers 1 to 4 (BMLOCR1 to BMLOCR4)

BMLOCR are registers that specify the locations for multidisplays 1 to 4 in the multiwindow.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CVLC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CHLC[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	CVLC[11:0]	H'000	R/W	These bits specify the number of pixels for the vertical offset from the base point as the start location of a multidisplay (see Figure 36.13, 1-pixel units). A setting causing multidisplays 1 to 4 to overlap each other is prohibited.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	CHLC[11:0]	H'000	R/W	These bits specify the number of pixels for the horizontal offset from the base point as the start location of a multidisplay (see Figure 36.13, 1-pixel units). For each multidisplay, in the horizontal direction the lower 2 bits of CHLC are truncated. From that location, the value specified by bits CHLC1 and CHLC is added to the value specified by bits BHSS in BMSSZRn. The resulting value is rounded up to 4-pixel units and an area of the resulting number of pixels is treated as an image manipulation area. A setting causing multidisplays 1 to 4 to overlap each other is prohibited.

The multiwindow function overlays a window containing a maximum of four displays (multidisplays) on top of the blended result of input systems 1 to 3. The valid size matches the parent display size. The parent display can be selected as input system 1, 2, or 3 by the PWD bits in BBLCR1. Figure 36.12 shows the output result when the multiwindow is overlaid on the blended result of input systems 1 to 3.

Note that the α-blending function is not enabled in the multiwindow.

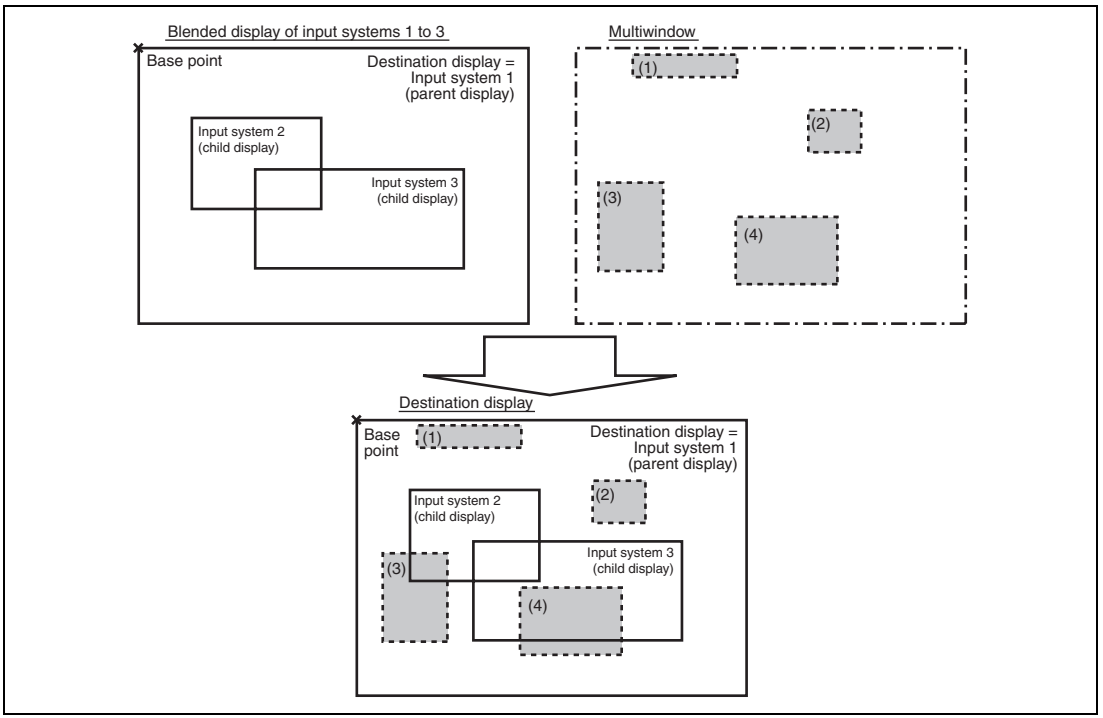


Figure 36.12 Multiwindow Output Result

The top-left corner of the parent display (= destination display) becomes the base point. The locations of the multidisplays are expressed by offsets from the base point. For example, if input display 1 is set as the parent display, the top-left corner of input display 1 becomes the base point, and the locations of multidisplays 1 to 4 are determined by the CHLC and CVLC bits in BMLOCR1 to BMLOCR4, respectively. Figure 36.13 shows where the multidisplays are located when multidisplays 1 and 2 are enabled.

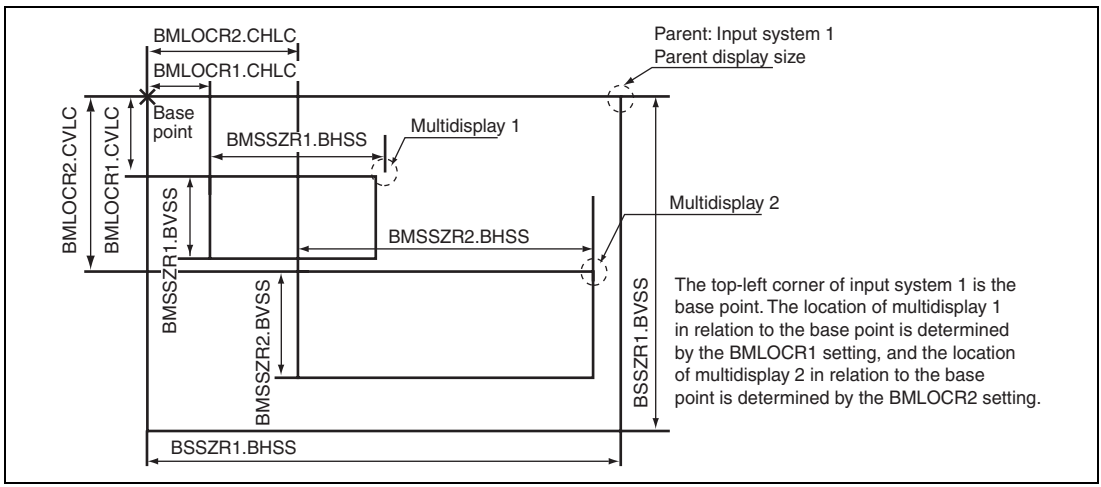


Figure 36.13 Multidisplay Locations

36.3.20 BEU Multidisplay Transparent Color Control Registers 1 and 2 (BMPCCR1 and BMPCCR2)

BMPCCR are registers that set the compare data used in transparent color comparison.

If the pack form of a BEU multidisplay is RGB565, RGB666, RGB888, or YCbCr, the input data is compared with the R, GY, and B data set in these bits. The bit size to be compared differs for each input pack form, and bits become valid from the upper side. Clear bits other than the valid bits to 0.

Note that only the Y component is to be compared in case of the YCbCr format. In case of the RGB565 pack form, for example, the R data is compared with the value in the R[7:3] bits, the G data with the value in the GY[7:2] bits, and the B data with the value in the B[7:3] bits.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GY[7:0]								B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	R[7:0]	H'00	R/W	Compare data R (RGB)
15 to 8	GY[7:0]	H'00	R/W	Compare data G (RGB) or Y (YCbCr)
7 to 0	B[7:0]	H'00	R/W	Compare data B (RGB)

36.3.21 Blend Pack Form Register (BPKFR)

BPKFR specifies the pack form for the blended BEU output.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PAD[7:0]								—	—	TM2	TM	—	—	—	DITH2
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DITH1	RY	TE	CHDS[1:0]		—	—	WP KFR	WPCK[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	PAD[7:0]	H'00	R/W	These bits specify the PAD value in the output data pack. When data is output in a pack shown in Table 36.14, set the value to be stored in the PAD field in these bits.
23, 22	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
21	TM2	0	R/W	This bit selects whether to use ITU-R BT.601 and 8-bit full-scale RGB color conversion or ITU-R BT.709 YCbCr and 8-bit full-scale RGB color conversion. 0: Use ITU-R BT.601 YCbCr and 8-bit full scale RGB color conversion. 1: Use ITU-R BT.709 YCbCr and 8-bit full scale RGB color conversion.
20	TM	0	R/W	Selects whether to use color conversion of ITU-R BT.601 defined YCbCr and 8-bit full-scale RGB or color conversion of 8-bit full-scale YCbCr and 8-bit full-scale RGB. 0: Uses color conversion of full-range RGB[0,255] and compressed-range YCbCr[16,235/240]. 1: Uses color conversion of full-range RGB[0,255] and full-range YCbCr[0,255].

Bit	Bit Name	Initial Value	R/W	Description
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	DITH2	0	R/W	Selects quantization or dithering for tone reduction in input system 2. 0: Performs quantization at tone reduction in input system 2. 1: Performs dithering at tone reduction in input system 2. Note: Dithering is performed only when the format for reading images is set to the RGB format (BPKFR.RY = 1).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	DITH1	0	R/W	Selects quantization or dithering for tone reduction in input system 1. 0: Performs quantization at tone reduction in input system 1. 1: Performs dithering at tone reduction in input system 1. Note: Dithering is performed only when the format for reading images is set to the RGB format (BPKFR.RY = 1).
11	RY	0	R/W	The TE bit turns on or off the circuit for RGB ⇔ YCbCr conversion. The RY bit selects the direction of RGB ⇔ YCbCr conversion. Table 36.12 shows the relationship between the TE and RY bit settings and the input/output data format. 00: BEU reads a YCbCr source image and does not perform YCbCr → RGB conversion. 01: BEU reads a YCbCr source image and performs YCbCr → RGB conversion. 10: BEU reads an RGB source image and does not perform RGB → YCbCr conversion. 11: BEU reads an RGB source image and performs RGB → YCbCr conversion.
10	TE	0	R/W	

Bit	Bit Name	Initial Value	R/W	Description
9, 8	CHDS[1:0]	00	R/W	These bits set the YCbCr output format of the BEU. The YCbCr data output pack forms are shown in Table 36.13. 00: Output as YCbCr 4:4:4 when the image output from input system 2 is in the YCbCr format 01: Output as YCbCr 4:2:2 when the image output from input system 2 is in the YCbCr format 10: Output as YCbCr 4:2:0 when the image output from input system 2 is in the YCbCr format 11: Setting prohibited
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
5	WPKFR	0	R/W	Sets the output data pack form when BEU output data is in RGB format. When setting WPKFR to BGR888 (3 bytes/pixel) format, set BPKFR.WPKRR =1, BPKFR.WPCK = H'15. 0: Output is pack format indicated by BPKFR.WPKF. 1: Output is BGR888 format.
4 to 0	WPCK[4:0]	00000	R/W	These bits specify the BEU output pack form. The BEU packs output data into a 32-bit pack. When data is output in the RGB format, the data is packed into one of the patterns shown in Table 36.14. Set a value in the WPCK column in Table 36.14 in these bits.

Table 36.12 Relationship between TE and RY Bits in BPKFR and Input/Output Data Format

TE Bit	RY Bit	Input	Output
0	0	YCbCr format	YCbCr format
0	1	RGB format	RGB format
1	0	YCbCr format	RGB format
1	1	RGB format	YCbCr format

Table 36.13 YCbCr Data Output Pack Forms

YCbCr 4:4:4 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cb1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1	Cr1

YCbCr 4:2:2/4:2:0 data format

Component	31 to 24								23 to 16								15 to 8								7 to 0							
Y data	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y0	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y1	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y2	Y3	Y3	Y3	Y3	Y3	Y3	Y3	Y3
C data	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cb0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cr0	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cb2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2	Cr2

Table 36.14 RGB Data Output Pack Forms

No	WPCK [4:0]	Bit Rate [bpp]	Phase	Bit																															
				31 to 24								23 to 16								15 to 8								7 to 0							
0	B'00000	8	—	R0	R0	R0	G0	G0	G0	B0	B0	R1	R1	R1	G1	G1	G1	B1	B1	R2	R2	R2	G2	G2	G2	B2	B2	R3	R3	R3	G3	G3	G3	B3	B3
1	B'00001	12	—	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
2	B'00010			R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1	0	0	0	0
6	B'00110	16	—	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
8	B'01000	18	—	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	PAD							
10	B'01010			PAD								R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0
13	B'01101			PAD								R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0
14	B'01110			R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	PAD							
16	B'10000		0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	R1	R1	R1	R1	R1	R1	G1	G1
			1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	0	0	0	0	0	0	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	B2	B2	B2	B2
			2	B2	B2	0	0	0	0	0	0	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	0	0	0	0	0	0
17	B'10001		0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
			1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
			2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
18	B'10010		0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1	0	0
			1	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2	0	0
			2	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3	0	0
19	B'10011	24	—	PAD								R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
20	B'10100			R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	PAD							
21	B'10101		0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
			1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
			2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
22	B'10110	18	—	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	0	0	0	0	0	0	0	G0	G0	G0	B0	B0	B0	B0	B0	B0
23	B'10111			0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0

36.3.22 BEU Transparent Color Control Register 0 (BPCCR0)

BPCCR0 enables or disables transparent color comparison.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CV3	CV2	CV1	CV0	—	SPIC	SAPEM2	SAPEM1	—	—	SAPE32	SAPE31	SAPE22	SAPE21	SAPE12	SAPE11
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	CV3	0	R/W	Turns on or off the multidisplay color replacement function. When the pixel value input to the multidisplay matches BMPCCR1.R, BMPCCR1.GY, and BMPCCR1.B, that pixel value and α are replaced with the BMPCCR2 value. Set BPCCR0.SAPEM2-1 = 0 when CV3 = 1. 0: Multidisplay color replacement function off 1: Multidisplay color replacement function on

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	CV2	0	R/W	CV2: Input system 3 display color replacement function
	CV1	0	R/W	CV1: Input system 2 display color replacement function
	CV0	0	R/W	CV0: Input system 1 display color replacement function When input pixel values of input system 1 to input system 3 match BPCCRx1.R, BPCCRx1.GY, and BPCCRx1.B, those pixel values and α are replaced with BPCCRx2. (They correspond to each input system at x: 1, 2, and 3.) Color replacement function is enabled when the input format for that input system is RGB 565, 666, 888, or YCbCr. Color replacement function can be set independently for each input system. When CV2 = 1, set BPCCR0.SAPE32-31 = 0. When CV1 = 1, set BPCCR0.SAPE22-21 = 0. When CV1 = 0, set BPCCR0.SAPE12-11 = 0. Note 1: Color replacement function cannot be specified when performing center enlargement (BBLCR1.SELRDD = 1) on input data. Note 2: System 1 transparent color function cannot be used when using system 1 color replacement. 0: Input system x display color replacement function off 1: Input system x display color replacement function on
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	SPIC	0	R/W	Performs data comparison when the input pack form of the BEU multidisplay is RGB565, RGB666, RGB888, or YCbCr with the SAPEM2 or SAPEM1 bit set to 1. 0: Multidisplay is selected when data comparison results in a match, and blended display is selected in a mismatch. 1: Blended display is selected when data comparison results in a match, and multidisplay is selected in a mismatch.

Bit	Bit Name	Initial Value	R/W	Description
9	SAPEM2	0	R/W	Two types of color comparison are possible for each input system as transparent color comparison. When the SAPEM2 bit is set to 1, input data of the multidisplay is compared with the data set in BMPCCR2. When the SAPEM1 bit is set to 1, input data of the multidisplay is compared with the data set in BMPCCR1. When both the SAPEM2 and SAPEM1 bits are cleared to 0, data comparison is disabled. 00: Disables data comparison 1 and 2 for BEU multidisplay. 01: Enables data comparison 1 for BEU multidisplay. 10: Enables data comparison 2 for BEU multidisplay. 11: Enables data comparison 1 and 2 for BEU multidisplay.
8	SAPEM1	0	R/W	
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
5	SAPE32	0	R/W	Two types of color comparison are possible for each input system as transparent color comparison. When the SAPE32 bit is set to 1, input data of input system 3 is compared with the data set in BPCCR32. When the SAPE31 bit is set to 1, input data of input system 3 is compared with the data set in BPCCR31. When both the SAPE32 and SAPE31 bits are cleared to 0, data comparison is disabled. 00: Disables data comparison 1 and 2 for input system 3. 01: Enables data comparison 1 for input system 3. 10: Enables data comparison 2 for input system 3. 11: Enables data comparison 1 and 2 for input system 3.
4	SAPE31	0	R/W	

Bit	Bit Name	Initial Value	R/W	Description
3	SAPE22	0	R/W	Two types of color comparison are possible for each input system as transparent color comparison. When the SAPE22 bit is set to 1, input data of input system 2 is compared with the data set in BPCCR22. When the SAPE21 bit is set to 1, input data of input system 2 is compared with the data set in BPCCR21. When both the SAPE22 and SAPE21 bits are cleared to 0, data comparison is disabled. 00: Disables data comparison 1 and 2 for input system 2. 01: Enables data comparison 1 for input system 2. 10: Enables data comparison 2 for input system 2. 11: Enables data comparison 1 and 2 for input system 2.
2	SAPE21	0	R/W	
1	SAPE12	0	R/W	Two types of color comparison are possible for each input system as transparent color comparison. When the SAPE12 bit is set to 1, input data of input system 1 is compared with the data set in BPCCR12. When the SAPE11 bit is set to 1, input data of input system 1 is compared with the data set in BPCCR11. When both the SAPE12 and SAPE11 bits are cleared to 0, data comparison is disabled. 00: Disables data comparison 1 and 2 for input system 1. 01: Enables data comparison 1 for input system 1. 10: Enables data comparison 2 for input system 1. 11: Enables data comparison 1 and 2 for input system 1.
0	SAPE11	0	R/W	

36.3.23 BEU Transparent Color Control Registers 11, 12, 21, 22, 31, and 32 (BPCCR11, BPCCR12, BPCCR21, BPCCR22, BPCCR31, and BPCCR32)

BPCCR are registers that set the compare data used in transparent color comparison and the alpha value.

If the pack form of BEU input systems 1 to 3 is RGB565, RGB666, RGB888, or YCbCr, the input data is compared with the R, GY, and B data set in these bits. The bit size to be compared differs for each input pack form, and bits become valid from the upper side. Clear bits other than the valid bits to 0.

Note that only the Y component is to be compared in case of the YCbCr format. In case of the RGB565 pack form, for example, the R data is compared with the value in the R[7:3] bits, the G data with the value in the GY[7:2] bits, and the B data with the value in the B[7:3] bits.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AP[7:0]								R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GY[7:0]								B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	AP[7:0]	H'00	R/W	These bits set the alpha value used at a data comparison match. Performs data comparison with the BPCCRxx value corresponding to each bit when the input pack form of BEU input system 1 is RGB565, RGB666, RGB888, or YCbCr with the SAPExx bit in BPCCR0 set to 1. When data comparison results in a match, the pixel-unit alpha value is replaced with the alpha value set in these bits (xx = 11, 12, 21, 22, 31, or 32).
23 to 16	R[7:0]	H'00	R/W	Compare data R (RGB)
15 to 8	GY[7:0]	H'00	R/W	Compare data G (RGB) or Y (YCbCr)
7 to 0	B[7:0]	H'00	R/W	Compare data B (RGB)

36.3.24 BEU Destination Memory Width Register (BDMWR)

BDMWR sets the memory width of the destination memory area of the BEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BDW[17:16]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 to 2	BDW[17:2]	H'00000	R/W	These bits set the width of the destination memory area of the BEU (longword units). These bits specify the horizontal width of the destination memory area where the destination image is to be located by BEU processing. Specifies more than a value for range to get the number of pixel resulting the output horizontal pixel of parent display rounded up to four-pixel units. Within the range: Data write is performed. Exceed the range: Black to be output.
1, 0	BDW[1:0]		R	

36.3.25 BEU Destination Address Y Register (BDAYR)

BDAYR sets the start address for the Y/RGB plane of the image drawn by the BEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDAY[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDAY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BDAY[31:2]	H'0000 0000	R/W	These bits set the start address for drawing the Y/RGB plane by the BEU (longword units).
1, 0	BDAY[1:0]		R	When drawing a YCbCr image, set the start address of the Y plane. When drawing an RGB image, set the start address of the RGB plane.

36.3.26 BEU Destination Address C Register (BDACR)

BDACR sets the start address for the C plane of the image drawn by the BEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BDAC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BDAC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	BDAC[31:2]	H'0000 0000	R/W	These bits set the start address for drawing the C plane by the BEU (longword units).
1, 0	BDAC[1:0]		R	When drawing a YCbCr image, set the start address of the C plane.
				Note: When drawing an RGB image, this register is not used.

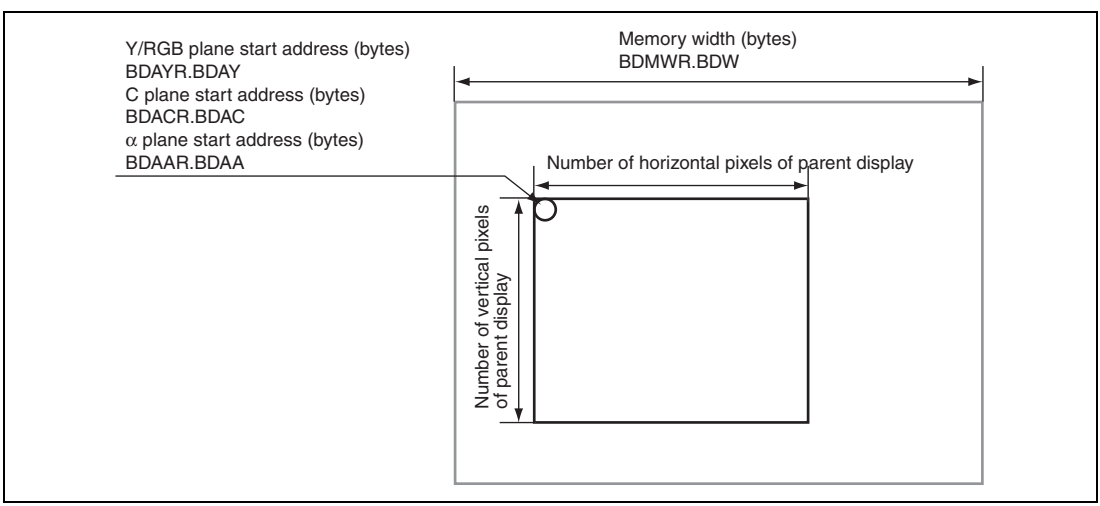


Figure 36.14 Destination Display Size and Start Address

36.3.27 BEU Address Fixed Register (BAFXR)

BAFXR sets fixed address mode for the data output from the BEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BAFIX
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BAFIX	0	R/W	When this bit is 1, the addresses to write the image output from the BEU is fixed to the values set in BDAYR and BDACR. When this bit is 0, the address is incremented according to BDMWR. 0: Output addresses of the BEU are not in fixed address mode. 1: Output addresses of the BEU are in fixed address mode. Note: Fixed address mode can be used only for RGB images. Correct operation cannot be guaranteed for YCbCr images.

36.3.28 BEU Swapping Register (BSWPR)

BSWPR sets swapping within 64-bit data at the data input/output section of the BEU.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MODSEL	—	—	—	—	BEILS4	BEIWS4	BEIBS4	—	—	—	—	—	BEILS3	BEIWS3	BEIBS3
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	BEILS2	BEIWS2	BEIBS2	—	BEOLS	BEOWS	BEOBS	—	BEILS	BEIWS	BEIBS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MODSEL	0	R/W	When MODSEL = 1 in BSWPR, swap setting is possible for each input system. When MODSEL = 0 in BSWPR, the swap settings made by BEILS, BEIWS, and BEIBS bits in BSWPR are available for all input systems. 0: Disables swap setting mode for each input system. 1: Enables swap setting mode for each input system.
30 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	BEILS4	0	R/W	0: Input system 4 (multidisplay) input data is not swapped in longword units. 1: Input system 4 (multidisplay) input data is swapped in longword units.
25	BEIWS4	0	R/W	0: Input system 4 (multidisplay) input data is not swapped in word units. 1: Input system 4 (multidisplay) input data is swapped in word units.

Bit	Bit Name	Initial Value	R/W	Description
24	BEIBS4	0	R/W	0: Input system 4 (multidisplay) input data is not swapped in byte units. 1: Input system 4 (multidisplay) input data is swapped in byte units.
23 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18	BEILS3	0	R/W	0: Input system 3 (multidisplay) input data is not swapped in longword units. 1: Input system 3 (multidisplay) input data is swapped in longword units.
17	BEIWS3	0	R/W	0: Input system 3 (multidisplay) input data is not swapped in word units. 1: Input system 3 (multidisplay) input data is swapped in word units.
16	BEIBS3	0	R/W	0: Input system 3 (multidisplay) input data is not swapped in byte units. 1: Input system 3 (multidisplay) input data is swapped in byte units.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	BEILS2	0	R/W	0: Input system 2 (multidisplay) input data is not swapped in longword units. 1: Input system 2 (multidisplay) input data is swapped in longword units.
9	BEIWS2	0	R/W	0: Input system 2 (multidisplay) input data is not swapped in word units. 1: Input system 2 (multidisplay) input data is swapped in word units.
8	BEIBS2	0	R/W	0: Input system 2 (multidisplay) input data is not swapped in byte units. 1: Input system 2 (multidisplay) input data is swapped in byte units.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	BEOLS	0	R/W	Sets swapping in longword units for output data. In longword swapping for output data, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the BEU output section. 0: BEU output data is not swapped in longword units. 1: BEU output data is swapped in longword units.
5	BEOWS	0	R/W	Sets swapping data in word units for output data. In word swapping for output data, 16 bits are swapped in word units from the MSB side or LSB side for 64-bit data in the BEU output section. 0: BEU output data is not swapped in word units. 1: BEU output data is swapped in word units.
4	BEOBS	0	R/W	Sets swapping data in byte units for output data. In byte swapping for output data, 16 bits are swapped in byte units from the MSB side or LSB side for 64-bit data in the BEU output section. 0: BEU output data is not swapped in byte units. 1: BEU output data is swapped in byte units.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	BEILS	0	R/W	Sets swapping data in longword units for input data. In longword swapping for input data, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the BEU input section (for all input systems). When MODSEL = 0 and BEILS = 1 in BSWPR, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the BEU2 input read section (for all input systems). When BEILS = 0 in BSWPR, data is not swapped in longword units. When MODSEL = 1 and BEILS = 1 in BSWPR, data is swapped in longword units within the 32 bits on the MSB side and within the 32 bits on the LSB side for 64-bit data in the BEU2 input system 1 read section. When BEILS = 0 in BSWPR, data is not swapped in longword units. 0: BEU input data is not swapped in longword units. 1: BEU input data is swapped in longword units.
1	BEIWS	0	R/W	Sets swapping data in word units for input data. In word swapping for input data, 16 bits are swapped in word units from the MSB side or LSB side for 64-bit data in the BEU input section (for all input systems). When MODSEL = 0 and BEIWS = 1 in BSWPR, data is swapped in word units within the 16 bits on the MSB or LSB side for 64-bit data in the BEU2 input read section (for all input systems). When BEIWS = 0 in BSWPR, data is not swapped in word units. When MODSEL = 1 and BEIWS = 1 in BSWPR, data is swapped in word units within the 16 bits on the MSB or LSB side for 64-bit data in the BEU2 input system 1 read section. When BEIWS = 0 in BSWPR, data is not swapped in word units. 0: BEU input data is not swapped in word units. 1: BEU input data is swapped in word units.

Bit	Bit Name	Initial Value	R/W	Description
0	BEIBS	0	R/W	Sets swapping data in byte units for input data. In byte swapping for input data, 16 bits are swapped in byte units from the MSB side or LSB side for 64-bit data in the BEU input section (for all input systems). When MODSEL = 0 and BEIBS = 1 in BSWPR, data is swapped in byte units within the 16 bits on the MSB or LSB side for 64-bit data in the BEU2 input read section (for all input systems). When BEIBS = 0 in BSWPR, data is not swapped. When MODSEL = 1 and BEIBS = 1 in BSWPR, data is swapped in byte units within the 16 bits on the MSB or LSB side for 64-bit data in the BEU2 input system 1 read section. When BEIBS = 0 in BSWPR, data is not swapped. 0: BEU input data is not swapped in byte units. 1: BEU input data is swapped in byte units.

Figure 36.15 shows the data swapping when MODSEL = 0. Setting for one system (BSWPR) is provided in the data swapping register, which is the setting of each input system.

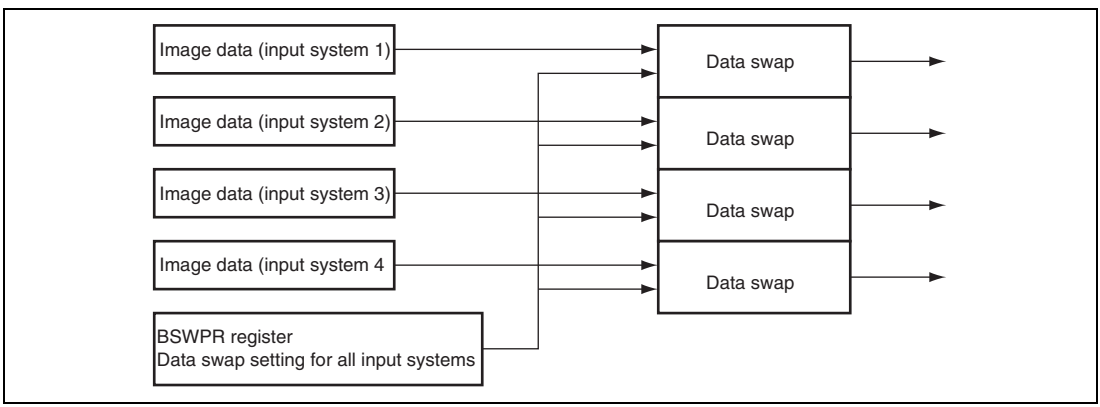


Figure 36.15 Data Swap Configuration when MODSEL = 0

Figure 36.16 shows the data swap configuration when MODSEL = 1. Data swapping method can be changed for each input system.

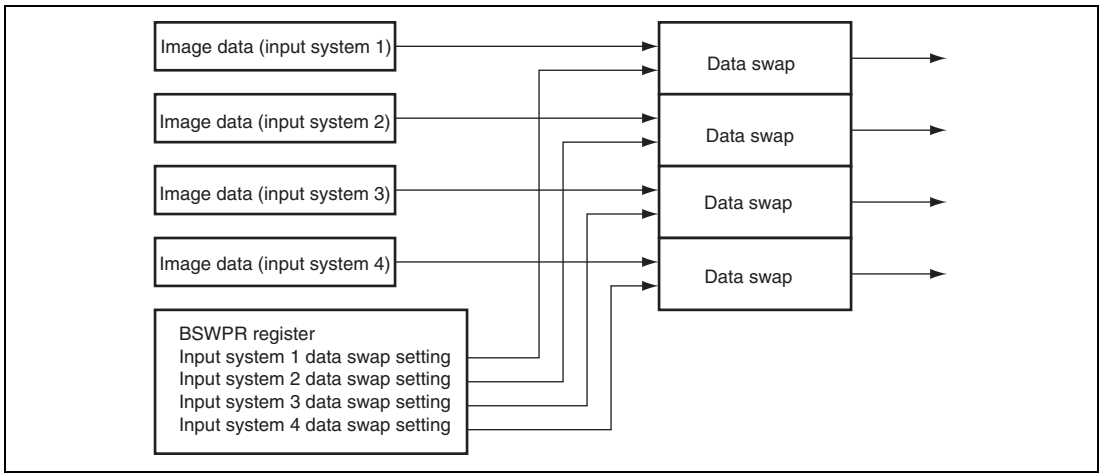


Figure 36.16 Data Swap Configuration when MODSEL = 1

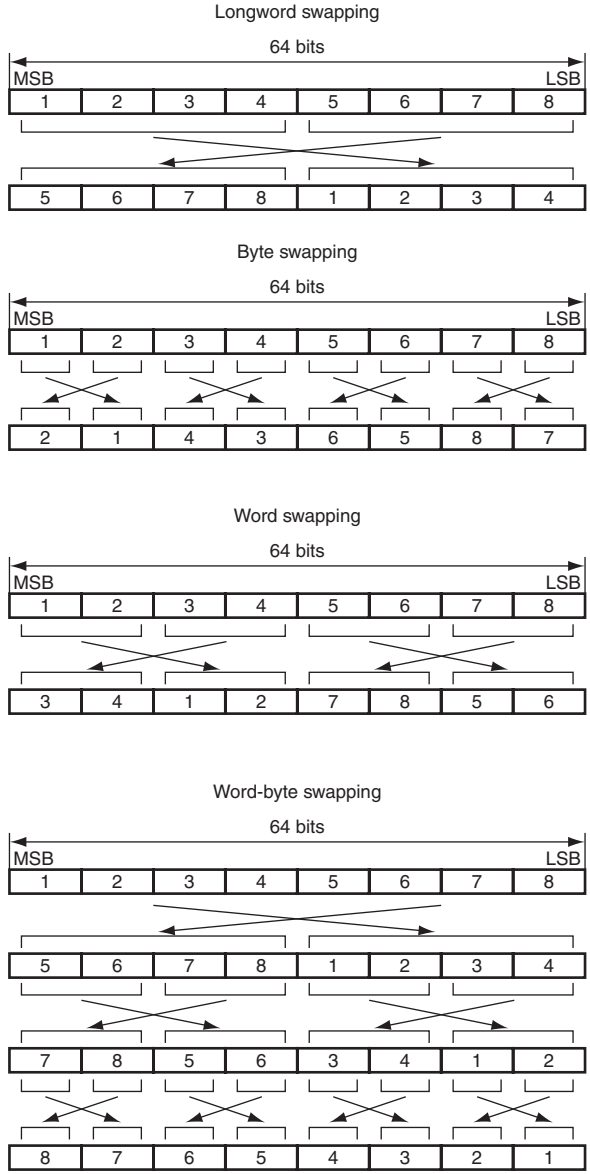


Figure 36.17 Data Swapping

36.3.29 BEU Event Interrupt Enable Register (BEIER)

BEIER enables or disables output of the interrupt signal of a BEVTR flag.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BEVIOE	BEENDE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	BEVIOE	0	R/W	Enables or disables the interrupt signal of the BEVIO bit in BEVTR to be output. 0: Disables output of the interrupt signal of the BEVIO bit in BEVTR. 1: Enables output of the interrupt signal of the BEVIO bit in BEVTR.
0	BEENDE	0	R/W	Enables or disables the interrupt signal of the BEEND bit in BEVTR to be output. 0: Disables output of the interrupt signal of the BEEND bit in BEVTR. 1: Enables output of the interrupt signal of the BEEND bit in BEVTR.

36.3.30 BEU Event Register (BEVTR)

BEVTR indicates the interrupt source when an internal interrupt occurs in the BEU. Whether output of the interrupt signal of each source in BEVTR is enabled or disabled is set by BEIER.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INT REQ
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BE VIO	BE END
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INTREQ	0	R/W	This bit is set to 1 when an interrupt source occurs in the BEU while output of the corresponding interrupt signal is enabled in BEIER. This bit remains cleared to 0 in the case where even though an interrupt source occurs in the BEU, output of the corresponding interrupt signal is disabled in BEIER, or in the case where even though output of the corresponding interrupt signal is enabled in BEIER, no interrupt source occurs in the BEU. 0: Indicates no interrupt is output externally. 1: Indicates an interrupt source has occurred in the BEU, and an interrupt is being output externally.
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	BEVIO	0	R/W	BEVIO Flag (violation status) [Reading] 0: Indicates that no violation setting has been detected, or the BEVIO flag has already been cleared. 1: Indicates that a violation setting has been detected. [Writing] 0: Clears the BEVIO flag by writing 0 to it. 1: NOP
0	BEEND	0	R/W	BEEND Flag (end status) [Reading] 0: Indicates that BEU processing has not finished, or the BEEND flag has already been cleared. 1: Indicates that BEU processing has finished. [Writing] 0: Clears the BEEND flag by writing 0 to it. 1: NOP

The BEEND flag (end status) is set when BEU operation has finished. At this time, if the BEENDE bit (enables or disables an end interrupt) in BEIER is 1, an interrupt is generated. The BEEND flag can be reset by writing 0 to the BEEND bit. The BEEND flag is not cleared unless 0 is written to the BEEND bit after BEU processing ends. Therefore, 0 must always be written to the BEEND bit to clear it before the BEU is activated again.

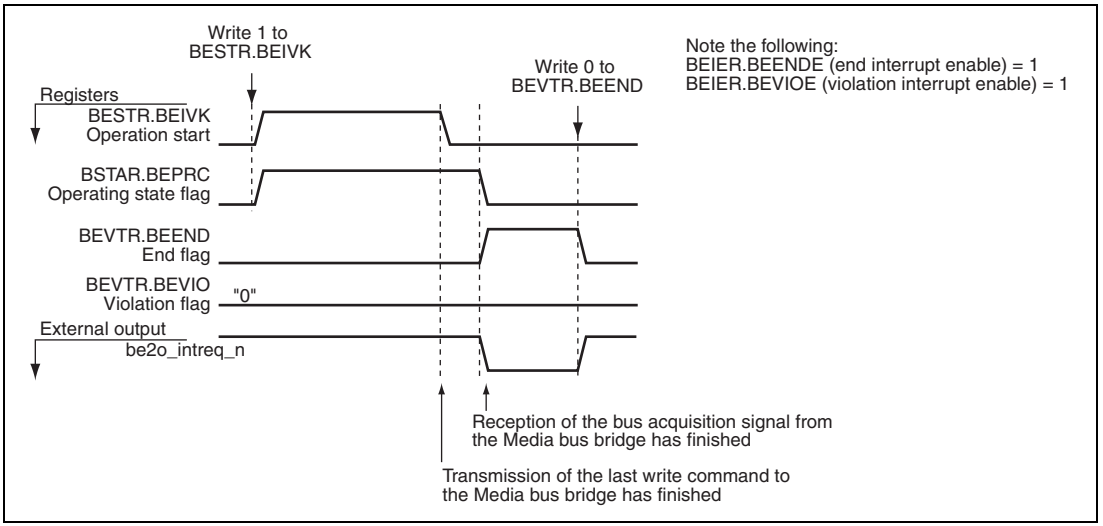


Figure 36.18 Operation Sequence without Violation Setting and Forcible Setting of End Flag

When a violation setting is made as shown in Figure 36.19, and the BEIVK bit (operation start) in BESTR is set to 1, the BEU does not start operation and the BEVIO flag (violation status) is set. At this time, if the BEVIOE bit (enables or disables a violation interrupt) in BEIER is 1, an interrupt is generated. The BEVIO flag can be reset by writing 0 to the BEVIO bit. When the BEVIO flag is set, the BEEND bit (end status) in BEVTR is not set, and the scheduling section, data transfer section, and data processing section are not activated.

The BEVIO flag is not cleared unless 0 is written to the BEVIO bit after BEU processing ends. Therefore, 0 must always be written to the BEVIO bit to clear it before the BEU is activated again.

The BEIVK bit is reset simultaneously to the BEVIO flag being set.

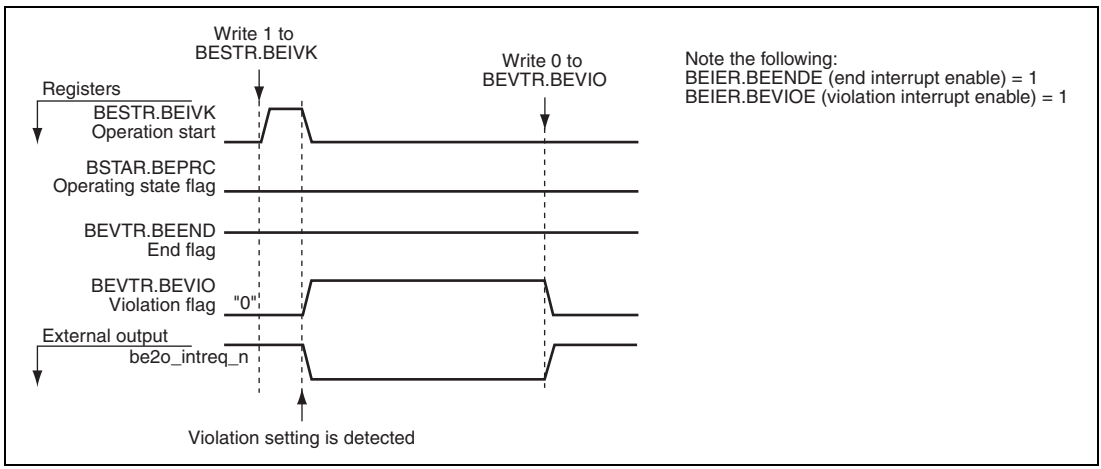


Figure 36.19 Operation Sequence with Violation Setting and Forcible Setting of Violation Flag

36.3.31 BEU Register Control Register (BRCNTR)

BRCNTR controls switching of the planes of BEU registers with a 2-plane configuration. Modifying these registers during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RS	RC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	RS	0	R/W	Specifies the register plane. Specifies which register plane is used by the BEU in synchronization with VSYNC. The setting of this bit is valid only when the RC bit is 0. 0: BEU uses plane A of the register. 1: BEU uses plane B of the register.
0	RC	0	R/W	Enables register plane switching. Specifies whether or not to switch the register plane used by the BEU in synchronization with VSYNC. If the register plane is not switched, the register plane specified by the RS bit is used. If the BEU is operating alone at register plane switching, the register plane is switched when the BEU is activated. If the BEU is operating together with the LCDC, the register plane is switched by the switching signal from the LCDC. 0: Uses the specified register plane. 1: Switches the register plane.

36.3.32 BEU Status Register (BSTAR)

BSTAR indicates the internal status of the BEU and the internal signal states.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	REGST	—	—	—	—	—	—	—	BEPRC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	REGST	0	R	Indicates which register plane is currently used. 0: BEU is using plane A of the register. 1: BEU is using plane B of the register.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BEPRC	0	R	This bit is set to 1 when the BEU is activated by the BEIVK bit in BESTR. This bit being set to 1 indicates that the BEU is operating. During this period, do not modify registers that are prohibited to be modified during operation. This bit is automatically cleared to 0 simultaneously with clearing of the BEEND flag in BEVTR. 0: BEU is halted. 1: BEU is operating.

36.3.33 BEU Module Reset Register (BBRSTR)

BBRSTR executes the module reset of the BEU.

Modifying this register during operation is prohibited. If B'1 is written to the ALLRST bit during operation, the logic circuits handshaking with the CPU bus are forcibly reset, and malfunction may also affect modules other than the BEU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ALL RST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ALLRST	0	W	When 1 is written to this bit, all internal control signals of the BEU are reset. If 1 is written to this bit during operation, the logic circuits handshaking with the CPU bus are forcibly reset, and malfunction may also affect modules other than the BEU. 0: NOP 1: Module reset of the BEU

36.3.34 BEU Register-Plane Forcible Setting Register (BRCHR)

BRCHR forcibly specifies the register plane.

Writing 1 to read-only bits is prohibited. If 1 is written to any of these bits, a malfunction may occur. Modifying this register during operation is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RCH
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RCH	0	R/W	Specifies the register plane forcibly. This bit forcibly switches the register plane currently used. Switching of the register plane can be done by this register only when the RC bit in BRCNTR is 1. To forcibly switch the register plane, set the RC bit to 1 in advance. 0: Specifies plane A of the register. 1: Specifies plane B of the register.

36.4 Linked Operation

36.4.1 Register Switching in Linked Operation

In operation linked with the LCDC, make a setting to switch the register planes (BRCNTR.RC = 1), load the register values into the internal registers at the timing of the switching signal from the LCDC, and at the same time, switch the register planes.

The output timing of the switching signal outputs the switching signal at the end of a frame. Accordingly, in LCDC-linked operation, the register planes need to be switched immediately after the BEU is activated. Figure 36.20 shows the register switching operation in LCDC-linked operation.

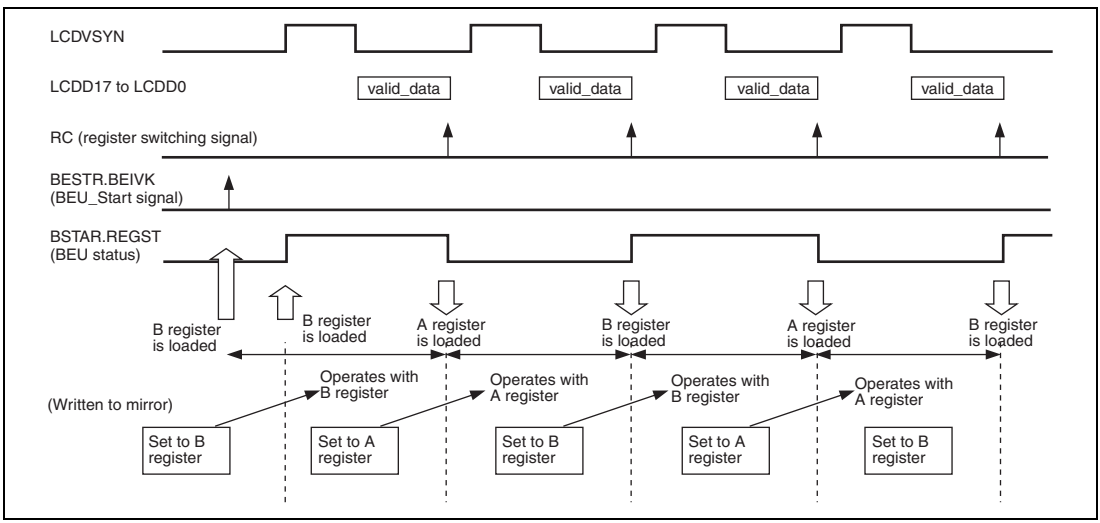


Figure 36.20 Register Switching in LCDC-Linked Operation

36.4.2 Multiwindow in Linked Operation

Since all multiwindow-related registers do not have a 2-plane configuration, modifying the registers during operation is prohibited. Therefore, in continuous-frame operation, the multiwindow image cannot be changed for each frame and the image size cannot be changed.

36.4.3 Selecting Data Format for Linked Operation

During linked operation,

LCDC expects to receive RGB format data from the BEU. The table below shows combinations of input/output format settings possible in linked operation.

Table 36.15 Relationship between TE and RY Bits in RPKFR and Possibility of Input/Output Linked Operation

TE Bit	RY Bit	BEU Input	BEU Output	LCDC Linked Operation
0	0	YCbCr format	YCbCr format	Not possible
0	1	RGB format	RGB format	Possible
1	0	YCbCr format	RGB format	Possible
1	1	RGB format	YCbCr format	Not possible

Section 37 JPEG Processing Unit (JPU)

The JPEG processing unit (JPU) incorporates the JPEG codec with an encoding and decoding function conforming to the JPEG baseline process, so that the JPU can encode image data and decode JPEG data quickly.

For the encoding and decoding processes, frame buffers or line buffers for raster block conversion are necessary in an external buffer (such as an SDRAM) which is externally connected to the LSI.

37.1 Features

The JPU has the following features.

- Conforming specification: JPEG baseline
- Operating precision: Conforming to JPEG Part 2, ISO-IEC10918-2
- Color format: YCbCr 4:2:2 (H = 2:1:1, V = 1:1:1), YCbCr 4:2:0 (H = 2:1:1, V = 2:1:1)
- Quantization tables: Four tables
- Huffman tables: Four tables (two AC tables, two DC tables)
- Target markers: Start of image (SOI), start of frame type 0 (SOF0), start of scan (SOS), define quantization tables (DQT), define Huffman tables (DHT), define restart interval (DRI), restart (RSTm), end of image (EOI)
- Image data rate: Maximum 108 Mbytes/s (54 MHz operation)
- The frame buffer can be reduced from the RAM area such as the SDRAM by using line buffer mode.

  This mode cannot be used with some pixel clock frequencies of the camera sensor or RAM bandwidth.
- Image rotation (by 90°, 180°, or 270°) can be performed in the encoding process (only in frame buffer mode).
- In reload mode, address toggling at every transfer of a specified amount of data during stream reading and writing is supported so that the buffer capacity can be reduced.
- Processing unit: 8-byte address boundary, 4-byte data length

  Note that the output data size during encoding is determined by the 16-byte boundary at the end of the address.
- Processable image size:
  Maximum 4092 (horizontal) × 4092 (vertical) pixels,
  Minimum 16 (horizontal) × 16 (vertical) pixels

  Image is processed in 4-pixel units.

Note: Do not perform unsupported color formatting or encoding or decoding of an image with an unsupported size.

Figure 37.1 shows the connection between the JPU and peripheral modules.

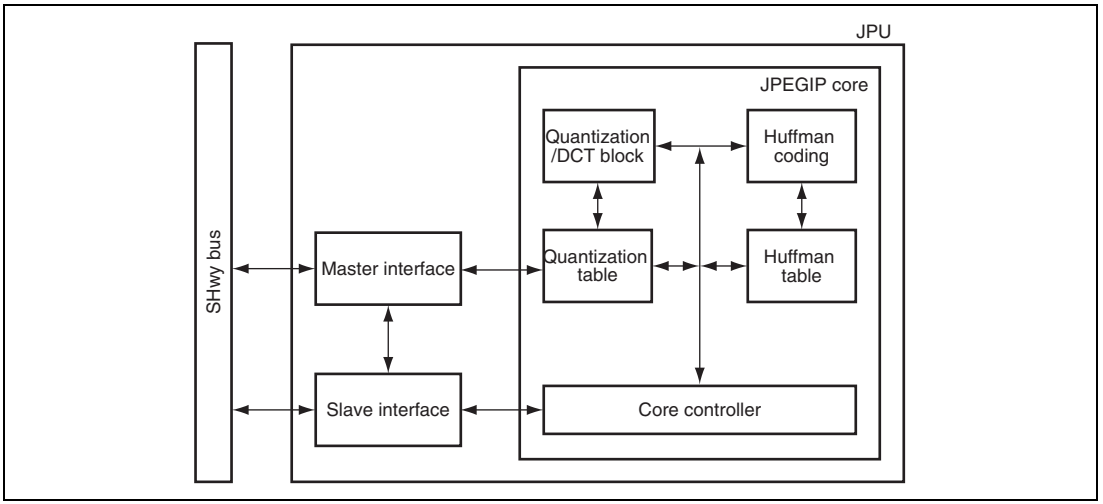


Figure 37.1 Connection between JPU and Peripheral Modules

37.2 Register Descriptions

Table 37.1 shows the JPU registers. Table 37.2 shows the register state in each processing mode.

These registers are 32-bit access registers. When modifying the reserved bits or read-only bits, write 0 to the bits. However, when clearing the bits in the JPEG interrupt status register (JINTS), write 0 only to the bits to be cleared and write 1 to the other bits. If register access which is not specified is performed, operation cannot be guaranteed.

Table 37.1 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
JPEG code mode register	JCMOD	R/W	H'FE98 0000	32
JPEG code command register	JCCMD	R/W	H'FE98 0004	32
JPEG code status register	JCSTS	R	H'FE98 0008	32
JPEG code quantization table number register	JCQTN	R/W	H'FE98 000C	32
JPEG code Huffman table number register	JCHTN	R/W	H'FE98 0010	32
JPEG code DRI upper register	JCDRIU	R/W	H'FE98 0014	32
JPEG code DRI lower register	JCDRID	R/W	H'FE98 0018	32
JPEG code vertical size upper register	JCVSZU	R/W	H'FE98 001C	32
JPEG code vertical size lower register	JCVSZD	R/W	H'FE98 0020	32
JPEG code horizontal size upper register	JCHSZU	R/W	H'FE98 0024	32
JPEG code horizontal size lower register	JCHSZD	R/W	H'FE98 0028	32
JPEG code data count upper register	JCDTCU	R	H'FE98 002C	32
JPEG code data count middle register	JCDTCM	R	H'FE98 0030	32
JPEG code data count lower register	JCDTCD	R	H'FE98 0034	32
JPEG interrupt enable register	JINTE	R/W	H'FE98 0038	32
JPEG interrupt status register	JINTS	R/W	H'FE98 003C	32
JPEG code decode error register	JCDERR	R/W	H'FE98 0040	32
JPEG code reset register	JCRST	R	H'FE98 0044	32
JPEG interface encoding control register	JIFECNT	R/W	H'FE98 0070	32
JPEG interface encode source Y address register 1	JIFESYA1	R/W	H'FE98 0074	32
JPEG interface encode source C address register 1	JIFESCA1	R/W	H'FE98 0078	32

Register Name	Abbreviation	R/W	Address	Access Size
JPEG interface encode source Y address register 2	JIFESYA2	R/W	H'FE98 007C	32
JPEG interface encode source C address register 2	JIFESCA2	R/W	H'FE98 0080	32
JPEG interface encode source memory width register	JIFESMW	R/W	H'FE98 0084	32
JPEG interface encode source vertical size register	JIFESVSZ	R/W	H'FE98 0088	32
JPEG interface encode source horizontal size register	JIFESHSZ	R/W	H'FE98 008C	32
JPEG interface encode destination address register 1	JIFEDA1	R/W	H'FE98 0090	32
JPEG interface encode destination address register 2	JIFEDA2	R/W	H'FE98 0094	32
JPEG interface encode data reload size register	JIFEDRSZ	R/W	H'FE98 0098	32
JPEG interface decoding control register	JIFDCNT	R/W	H'FE98 00A0	32
JPEG interface decode source address register 1	JIFDSA1	R/W	H'FE98 00A4	32
JPEG interface decode source address register 2	JIFDSA2	R/W	H'FE98 00A8	32
JPEG interface decode data reload size register	JIFDDRSZ	R/W	H'FE98 00AC	32
JPEG interface decode destination memory width register	JIFDDMW	R/W	H'FE98 00B0	32
JPEG interface decode destination vertical size register	JIFDDVSZ	R	H'FE98 00B4	32
JPEG interface decode destination horizontal size register	JIFDDHSZ	R	H'FE98 00B8	32
JPEG interface decode destination Y address register 1	JIFDDYA1	R/W	H'FE98 00BC	32
JPEG interface decode destination C address register 1	JIFDDCA1	R/W	H'FE98 00C0	32
JPEG interface decode destination Y address register 2	JIFDDYA2	R/W	H'FE98 00C4	32
JPEG interface decode destination C address register 2	JIFDDCA2	R/W	H'FE98 00C8	32
JPEG code quantization table 0 register	JCQTBL0	R/W	H'FE99 0000 to H'FE99 003C	32
JPEG code quantization table 1 register	JCQTBL1	R/W	H'FE99 0040 to H'FE99 007C	32

Register Name	Abbreviation	R/W	Address	Access Size
JPEG code quantization table 2 register	JCQTBL2	R/W	H'FE99 0080 to H'FE99 00BC	32
JPEG code quantization table 3 register	JCQTBL3	R/W	H'FE99 00C0 to H'FE99 00FC	32
JPEG code Huffman table DC0 register	JCHTBD0	R/W	H'FE99 0100 to H'FE99 010C	32
JPEG code Huffman table DC0 register	JCHTBD0	R/W	H'FE99 0110 to H'FE99 0118	32
JPEG code Huffman table AC0 register	JCHTBA0	R/W	H'FE99 0120 to H'FE99 012C	32
JPEG code Huffman table AC0 register	JCHTBA0	R/W	H'FE99 0130 to H'FE99 01D0	32
JPEG code Huffman table DC1 register	JCHTBD1	R/W	H'FE99 0200 to H'FE99 020C	32
JPEG code Huffman table DC1 register	JCHTBD1	R/W	H'FE99 0210 to H'FE99 0218	32
JPEG code Huffman table AC1 register	JCHTBA1	R/W	H'FE99 0220 to H'FE99 022C	32
JPEG code Huffman table AC1 register	JCHTBA1	R/W	H'FE99 0230 to H'FE99 02D0	32

Note: For the settings of the JPEG code quantization table and JPEG code Huffman table, see section 37.3.1 (6), Table Setting.

Table 37.2 Register State in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Module Standby	R-Standby	U-Standby	Sleep
JCMOD	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCCMD	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCSTS	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCQTN	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCHTN	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCDRIU	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCDRID	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCVSZU	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCVSZD	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCHSZU	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCHSZD	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCDTCU	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCDTCM	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCDTCD	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JINTE	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JINTS	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCDERR	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCRST	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFECNT	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESYA1	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESCA1	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESYA2	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESCA2	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESMW	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESVSZ	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFESHSZ	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFEDA1	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFEDA2	Initialized	Initialized	Retained	Initialized	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Module Standby	R-Standby	U-Standby	Sleep
JIFEDRSZ	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDCNT	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDSA1	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDSA2	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDRSZ	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDMW	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDVSZ	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDHSZ	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDYA1	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDCA1	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDYA2	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JIFDDCA2	Initialized	Initialized	Retained	Initialized	Initialized	Retained
JCQTBL0	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCQTBL1	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCQTBL2	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCQTBL3	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBD0	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBD0	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBA0	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBA0	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBD1	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBD1	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBA1	Undefined	Undefined	Retained	Undefined	Undefined	Retained
JCHTBA1	Undefined	Undefined	Retained	Undefined	Undefined	Retained

37.2.1 JPEG Code Mode Register (JCMOD)

JCMOD sets the operating mode before the JPU starts operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SOIC	PCTR	MSKIP[1:0]		CCNT	DSP	REDU[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SOIC	0	R/W	SOI Marker Delete This bit is valid when the MSKIP bit is B'00. In this mode, the amount of coded data indicated by registers JCDTCU, JCDTDM, and JCDTCD includes the SOI (2 bytes). 0: SIO marker exists 1: SOI marker deleted
7	PCTR	0	R/W	Image Data Input Control This bit should be fixed to 1.
6, 5	MSKIP[1:0]	00	R/W	Marker Skip Mode 00: Marker output 01: No marker output Other than above: Setting prohibited
4	CCNT	0	R/W	Code Amount Count Mode (This bit is valid only in decoding.) 0: Code amount count mode disabled 1: Code amount count mode enabled
3	DSP	0	R/W	Encoding/Decoding Set 0: Encoding process 1: Decoding process

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	REDU[2:0]	000	R/W (Encoding) R (Decoding)	Subsampling Set 001: 4:2:2 010: 4:2:0 Other than above: Setting prohibited (in encoding) Error (JPU cannot process normally in decoding.)

37.2.2 JPEG Code Command Register (JCCMD)

JCCMD sets commands. Bits of this register need not be cleared to 0 after setting a command.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SRST	RWC MD	RRC MD	LCMD1	LCMD2	BRST	—	—	—	—	JEND	JRST	JSRT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R*/W	R*/W	R*/W	R*/W	R*/W	R	R	R	R	R*/W	R*/W	R*/W

Note: * Values read from these bits are undefined.

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	SRST	0	R/W	Software Reset Setting this bit to 1 resets the JPU. This bit holds 1 until the transfer between the CPU bus and the JPU is completed; it is cleared to 0 when the transfer is completed. After a software reset is specified, do not specify another software reset or a bus reset before this bit is cleared to 0. For details, refer to section 37.6, Software Reset Processing.

Bit	Bit Name	Initial Value	R/W	Description
11	RWCMD	0	R*/W	Reload Buffer Write Restart Command This bit is valid during the encoding process in reload mode. The JPU stops writing the data to be coded to a reload buffer after one buffer of data specified by JIFEDRSZ has been transferred. Though the encoding process automatically stops, the first two reload buffers of data are automatically written. However, when JIFEDA1 is equal to JIFEDA2, only one reload buffer operates and the JPU stops writing the data to be coded after the first one buffer of data has been transferred. Restart writing the data to be coded by writing 1 to this bit.
10	RRCMD	0	R*/W	Reload Buffer Read Restart Command This bit is valid during decoding in reload mode. The JPU stops reading the coded data from a reload buffer after one buffer of data specified by JIFDDRSZ has been transferred. Restart reading the coded data by writing 1 to this bit. JPU continues reading coded data as long as the end of the code is not detected.
9	LCMD1	0	R*/W	External Line Buffer Processing Restart Command This bit is valid only for subsampling setting 4:2:0 in line buffer mode. JPU stops transferring the image data after transfer of the lines of data specified by JIFDCNT or JIFECNT is completed. Though the JPU automatically stops processing, the first two lines of data are automatically written. However, when JIFDDYA1 is equal to JIFDDYA2 and JIFDDCA1 is equal to JIFDDCA2, only one line buffer operates and the JPU stops writing the image data after the first one buffer of data has been transferred. Restart transferring by writing 1 to this bit in both the encoding and decoding processes.

Bit	Bit Name	Initial Value	R/W	Description
8	LCMD2	0	R*/W	External Line Buffer Process Restart Command This bit is valid only for subsampling setting 4:2:2 in line buffer mode. JPU stops transferring the mage data after transfer of the lines of data specified by JIFDCNT or JIFECNT is completed. Though the JPU automatically stops processing, the first two lines of data are automatically written. However, when JIFDDYA1 is equal to JIFDDYA2 and JIFDDCA1 is equal to JIFDDCA2, only one line buffer operates and the JPU stops writing the image data after the first one buffer of data has been transferred. Restart transferring by writing 1 to this bit in both the encoding and decoding processes.
7	BRST	0	R*/W	Bus Reset Setting this bit to 1 resets internal circuits. While the JPU is in operation (from JPEG core process start command setting to JINTS data transfer end interrupt occurrence), do not set this bit to 1. For the bus reset processing, see section 37.5, Bus Reset Processing.
6 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	JEND	0	R*/W	Interrupt Signal Clear Command This bit is valid for interrupt sources bits INS6, INS5, and INS3 in the JINTS register. To clear an interrupt signal, set this bit to 1.
1	JRST	0	R*/W	JPEG Core Processing-Stopped Revoke Command To clear the processing-stopped state because of requests to read the image size and subsampling setting (the INT3 bit in the JINTE register), set this bit to 1. (This bit is valid only in decoding.)
0	JSRT	0	R*/W	JPEG Core Process Start Command To start JPEG core processing, set this bit to 1. Do not write this bit to 1 again before the JPU is not completely activated.

Note: * Values read from these bits are undefined.

37.2.3 JPEG Code Status Register (JCSTS)

JCSTS indicates the internal state of the JPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0.
0	STS	0	R	Operating State This bit indicates the operating state of the JPEG core in the JPU. If its state is indicated as not in operation, the JPU may not have transferred data completely. To check whether the data has been completely transferred, use the transfer end interrupt through the JINTS register. 0: Not in operation 1: Encoding or decoding

37.2.4 JPEG Code Quantization Table Number Register (JCQTN)

JCQTN sets the quantization table number before encoding is started.

- To use quantization table No. 0 (JCQTBL0) as the first color component, set QT1 to B'00
- To use quantization table No. 1 (JCQTBL1) as the first color component, set QT1 to B'01
- To use quantization table No. 2 (JCQTBL2) as the first color component, set QT1 to B'10
- To use quantization table No. 3 (JCQTBL3) as the first color component, set QT1 to B'11

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	QT3[1:0]		QT2[1:0]		QT1[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	QT3[1:0]	00	R/W	Quantization table number for the third color component
3, 2	QT2[1:0]	00	R/W	Quantization table number for the second color component
1, 0	QT1[1:0]	00	R/W	Quantization table number for the first color component

37.2.5 JPEG Code Huffman Table Number Register (JCHTN)

JCHTN sets the Huffman table number (AC/DC) before encoding is started.

- To use DC/AC Huffman table No. 0 (JCHTBD0 and JCHTBA0) as the first color component, set bits HTA1 and HTD1 to B'0
- To use DC/AC Huffman table No. 1 (JCHTBD1 and JCHTBA1) as the first color component, set bits HTA1 and HTD1 to B'1

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	HTA3	HTD3	HTA2	HTD2	HTA1	HTD1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R\W	R\W	R\W	R\W	R\W	R\W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	HTA3	0	R/W	Huffman table number (AC) for the third color component
4	HTD3	0	R/W	Huffman table number (DC) for the third color component
3	HTA2	0	R/W	Huffman table number (AC) for the second color component
2	HTD2	0	R/W	Huffman table number (DC) for the second color component
1	HTA1	0	R/W	Huffman table number (AC) for the first color component
0	HTD1	0	R/W	Huffman table number (DC) for the first color component

37.2.6 JPEG Code DRI Upper Register (JCDRIU)

JCDRIU sets the upper bytes of the minimum coded units (MCUs) preceding an RST marker.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRIU[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	DRIU[7:0]	H'00	R/W	Upper Bytes of MCUs Preceding RST Marker When both upper and lower bytes are set to H'00, neither a DRI nor an RST marker is placed. (These bits are valid only in encoding.)

37.2.7 JPEG Code DRI Lower Register (JCDRID)

JCDRID sets the lower bytes of MCUs preceding an RST marker.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRID[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	DRID[7:0]	H'00	R/W	Lower Bytes of MCUs Preceding RST Marker When both upper and lower bytes are set to H'00, neither a DRI nor an RST marker is placed. (These bits are valid only in encoding.)

37.2.8 JPEG Code Vertical Size Upper Register (JCVSZU)

JCVSZU sets the upper bytes of the vertical image size. When rotating the image in encoding, set the size of the rotated image in this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	VSZU[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	VSZU[7:0]	H'00	R/W	Upper Bytes of Vertical Image Size Up to 4092 pixels can be set as the vertical image size. In the decoding process, a downloaded value from the JPEG coded data is set. When the vertical image size exceeds 2048 pixels, it is treated as an error.

37.2.9 JPEG Code Vertical Size Lower Register (JCVSZD)

JCVSZD sets the lower bytes of the vertical image size. When rotating the image in encoding, set the size of the rotated image in this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	VSZD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	VSZD[7:0]	H'00	R/W	Lower Bytes of Vertical Image Size In the decoding process, a downloaded value from the JPEG coded data is set.

37.2.10 JPEG Code Horizontal Size Upper Register (JCHSZU)

JCHSZU sets the upper bytes of the horizontal image size. When rotating the image in encoding, set the size of the rotated image in this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HSZU[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	HSZU[7:0]	H'00	R/W	Upper Bytes of Horizontal Image Size Up to 4092 pixels can be set as the horizontal image size. In the decoding process, a downloaded value from the JPEG coded data is set. When the horizontal image size exceeds 4092 pixels, it is treated as an error.

37.2.11 JPEG Coded Horizontal Size Lower Register (JCHSZD)

JCHSZD sets the lower bytes of the horizontal image size. When rotating the image in encoding, set the size of the rotated image in this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HSZD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	HSZD[7:0]	H'00	R/W	Lower Bytes of Horizontal Image Size In the decoding process, a downloaded value from the JPEG coded data is set.

37.2.12 JPEG Code Data Count Upper Register (JCDTCU)

The upper bytes for the counted amount of data to be encoded are set to JCDTCU. This register is valid only in an encoding process. The values of this register are reset before encoding starts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DCU[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	DCU[7:0]	H'00	R	Upper bytes for the counted amount of data to be encoded

37.2.13 JPEG Code Data Count Middle Register (JCDTCM)

The middle bytes for the counted amount of data to be encoded are set to JCDTCM. This register is valid only in an encoding process. The values of this register are reset before encoding starts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DCM[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	DCM[7:0]	H'00	R	Middle bytes for the counted amount of data to be encoded

37.2.14 JPEG Code Data Count Lower Register (JCDTCD)

The lower bytes for the counted amount of data to be encoded are set to JCDTCD. This register is valid only in an encoding process. The values of this register are reset before encoding starts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DCD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	DCD[7:0]	H'00	R	Lower bytes of the counted amount of data to be encoded

37.2.15 JPEG Interrupt Enable Register (JINTE)

JINTE enables interrupts.

When bits INT7 to INT5 are set to B'1, the INS5 bit in JINTS which indicates the error status is set to B'1, and the ERR bit in JCDERR indicates the particular error code.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	INT14	INT13	INT12	INT11	INT10	—	—	INT7	INT6	INT5	—	INT3	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	INT14	0	R/W	This flag sets whether an interrupt is generated at every address reloading for coded data reading. (This bit is valid only when reload mode is enabled.)
13	INT13	0	R/W	This flag sets whether an interrupt is generated at every address reloading for coded data writing. (This bit is valid only when reload mode is enabled.)
12	INT12	0	R/W	This flag sets whether an interrupt is generated at every transmission completion of Y components in multiples of 16 lines and C components in multiples of 8 lines. (This bit is valid only when line buffer mode is enabled.)
11	INT11	0	R/W	This flag sets whether an interrupt is generated at every transmission completion of Y components in multiples of 8 lines and C components in multiples of 8 lines. (This bit is valid only when line buffer mode is enabled.)
10	INT10	0	R/W	This flag sets whether an interrupt of transfer to the external buffer end is generated.
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7	INT7	0	R/W	This flag sets whether an interrupt is generated when the number of data in the restart interval of the Huffman-coding segment is not correct. When this flag is not set, an error code is not returned. (This bit is valid only in decoding.)
6	INT6	0	R/W	This flag sets whether an interrupt is generated when the total number of data in the Huffman-coding segment is not correct. When this flag is not set, an error code is not returned. (This bit is valid only in decoding.)
5	INT5	0	R/W	This flag sets whether an interrupt is generated when the last MCU number in the Huffman-coding segment is not correct. When this flag is not set, an error code is not returned. (This bit is valid only in decoding.)
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	INT3	0	R/W	This flag sets whether an interrupt is generated when it has been determined that the image size and the subsampling setting of the encoded data can be read through analyzing the data.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

37.2.16 JPEG Interrupt Status Register (JINTS)

JINTS identifies the interrupt sources.

The interrupts of bits INS10 to INS14 and INS3 are generated when the corresponding bits in the JINTE register are set to B'1. When the interrupt sources are bits INS14 to INS10, the interrupt signal can be negated by clearing the corresponding interrupt status bits to 0. When the interrupt sources are bits INS6, INS5, or INS3 (JPEG core), the interrupt signal can be negated by clearing the corresponding interrupt status bits to 0 and setting the JCCMD register appropriately.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	INS14	INS13	INS12	INS11	INS10	—	—	—	INS6	INS5	—	INS3	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W*	R/W*	R/W*	R/W*	R/W*	R	R	R	R/W*	R/W*	R	R/W*	R	R	R

Note: * When the bit is read as 1, write 0 to clear it.
When the bit is read as 0, write 1 to it.

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	INS14	0	R/W*	This bit is set to 1 every time the address is reloaded during stream data reading. (This bit is valid only when reload mode is enabled.)
13	INS13	0	R/W*	This bit is set to 1 every time the address is reloaded during stream data writing. (This bits is valid only when reload mode is enabled.)
12	INS12	0	R/W*	This bit is set to 1 when transfer of Y components in multiples of 16 lines and C components in multiples of 8 lines to the line buffer is completed. (Subsampling setting: 4:2:0, This bit is valid only when line buffer mode is enabled.)

Bit	Bit Name	Initial Value	R/W	Description
11	INS11	0	R/W*	This bit is set to 1 when transfer of Y components in multiples of 8 lines and C components in multiples of 8 lines to the line buffer is completed. (Subsampling setting: 4:2:2, This bit is valid only when line buffer mode is enabled.)
10	INS10	0	R/W*	This bit is set to 1 after all data on the result of JPU encoding and decoding have been completely transferred.
9 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	INS6	0	R/W*	This bit is set to 1 when the JPEG completes the encoding process normally.
5	INS5	0	R/W*	This bit is set to 1 when an encoded data error occurs. (This bit is valid only in decoding.)
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	INS3	0	R/W*	This bit is set to 1 when the image size and subsampling setting can be read. (This bit is valid only in decoding.) When an interrupt occurs, the JPU stops processing and the state is indicated by the JCRST register. To make the JPU resume processing, set the JPEG core processing-stopped release command (JRST bit in the JCCMD register).
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: * When the bit is read as 1, write 0 to clear it.
When the bit is read as 0, write 1 to it.

37.2.17 JPEG Code Decode Error Register (JCDERR)

Through the error code, JCDERR identifies the type of error which occurred in the encoded data analysis for decoding. The values of this register are reset before the JPU starts decoding.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	ERR[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0.
3 to 0	ERR[3:0]	1010	R/W	Error Code (See tables 37.4 and 37.5.)

37.2.18 JPEG Code Reset Register (JCRST)

JCRST indicates a processing-stopped state due to requests to read the image size and subsampling setting (the INTS3 bit in the JINTE register) (available in decoding). To resume processing, set the JPEG core processing-stopped release command (the JRST bit in the JCCMD register).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0.
0	RST	0	R	Operating State 0: State other than below 1: Suspended state caused by JINTE interrupt source

37.2.19 JPEG Interface Encoding Control Register (JIFECNT)

JIFECNT controls the encoding process. Note that the SWAP bit is used also in the decoding process.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PU[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RE LOAD	SWAP[1:0]		ROT[1:0]		BUF	INFT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	PU[11:0]	H'000	R/W	Processing Unit (These bits are valid only in line buffer mode.) [YCbCr 4:2:2 processing] Set the number of lines in vertical Y and C components (in multiples of 8: H'008 units). (Write 0 to the bits PU[2:0].) [YCbCr 4:2:0 processing] Set the number of lines in vertical Y components (in multiples of 16: H'010 units). (Write 0 to the bits PU[3:0].) A value equal to half of the number of lines in vertical Y components is automatically set as the number of lines in vertical C components. Note: The maximum number of settable lines is 2048 and the minimum number is 16. Values outside of this range should not be set.
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	RELOAD	0	R/W	Reload Mode 0: Reload mode disabled 1: Reload mode enabled
5, 4	SWAP[1:0]	00	R/W	Byte/Word Swap (see the JIFDCNT register) Data transferred to the JPU is swapped (image data in encoding; coded data in decoding). 00: (1) (2) (3) (4) 01: (2) (1) (4) (3) [Byte swap] 10: (3) (4) (1) (2) [Word swap] 11: (4) (3) (2) (1) [Word - byte swap]
3, 2	ROT[1:0]	00	R/W	Rotated Read Mode (These bits are valid only in frame buffer mode.) 00: 0° 01: 90° 10: 180° 11: 270°
1	BUF	0	R/W	Buffer Mode 0: Frame buffer mode 1: Line buffer mode
0	INFT	0	R/W	Subsampling Setting 0: YCbCr 4:2:2 1: YCbCr 4:2:0

Note: Bits other than SWAP can be modified only when the DSP bit in JCMOD is 0 (encoding process).

37.2.20 JPEG Interface Encode Source Y Address Register 1 (JIFESYA1)

JIFESYA1 sets the source address of the Y component.

This register should be set in 8-pixel units. When reading of an image rotated by 90°, 180°, or 270° is specified, set the origin address for the rotated image. Note that a rotated image can be read only in frame buffer mode (see section 37.3.1 (7), Rotated Read Mode Setting).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ESYA1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESYA1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	ESYA1[31:3]	H'0000 0000	R/W	Y Component Source Address (in 8-pixel units)
2 to 0	ESYA1[2:0]		R	The lower three bits should be set to 0.

37.2.21 JPEG Interface Encode Source C Address Register 1 (JIFESCA1)

JIFESCA1 sets the source address of the C component.

This register should be set in 8-pixel units. When reading of an image rotated by 90°, 180°, or 270° is specified, set the origin address for the rotated image. Note that a rotated image can be read only in frame buffer mode (see section 37.3.1 (7), Rotated Read Mode Setting).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ESCA1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESCA1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	ESCA1[31:3]	H'0000 0000	R/W	C Component Source Address (in 8-pixel units)
2 to 0	ESCA1[2:0]		R	The lower three bits should be set to 0.

37.2.22 JPEG Interface Encode Source Y Address Register 2 (JIFESYA2)

JIFESYA2 sets the source address of the Y component, and is valid in line buffer mode.

Set the start address in 8-pixel units. The source address of the Y component, which is used by the JPU, changes automatically in specified line buffer units: JIFESYA1 → JIFESYA2 → JIFESYA1 → . . . .

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ESYA2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESYA2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	ESYA2[31:3]	H'0000 0000	R/W	Y Component Source Address (These bits are valid in line buffer mode. in 8-pixel units)
2 to 0	ESYA2[2:0]		R	The lower three bits should be set to 0.

37.2.23 JPEG Interface Encode Source C Address Register 2 (JIFESCA2)

JIFESCA2 sets the source address of the C component, and is valid in line buffer mode.

Set the start address in 8-pixel units. The source address of the C component, which is used by the JPU, changes automatically in specified line buffer units: JIFESCA1 → JIFESCA2 → JIFESCA1 → . . . .

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ESCA2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESCA2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	ESCA2[31:3]	H'00000 0000	R/W	C Component Source Address (These bits are valid in line buffer mode. in 8-pixel units)
2 to 0	ESCA2[2:0]		R	The lower three bits should be set to 0.

37.2.24 JPEG Interface Encode Source Memory Width Register (JIFESMW)

JIFESMW sets the buffer memory width in which the image data is stored (see section 37.3.3, Storing Image Data). Set this register in 8-pixel units. The maximum settable width is 4092 pixels and the minimum is 16 pixels. Sizes outside of this range should not be set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ESMW[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 3	ESMW[11:3]	H'000	R/W	Buffer memory width in which image data is stored (in 8-pixel units)
2 to 0	ESMW[2:0]		R	The lower three bits should be set to 0.

37.2.25 JPEG Interface Encode Source Vertical Size Register (JIFESVSZ)

JIFESVSZ sets the vertical size of the image transferred from the external buffer (see section 37.3.3, Storing Image Data). Set the image size in 4-pixel units. The maximum settable width is 4092 pixels and the minimum is 16 pixels. Sizes outside of this range should not be set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ESVSZ[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 2	ESVSZ[11:2]	H'000	R/W	Vertical size of the image transferred from the external buffer (in 4-pixel units)
1, 0	ESVSZ[1:0]		R	The lower two bits should be set to 0.

37.2.26 JPEG Interface Encode Source Horizontal Size Register (JIFESHSZ)

JIFESHSZ sets the horizontal size of the image transferred from the external buffer (see section 37.3.3, Storing Image Data). Set the image size in 4-pixel units. The maximum settable width is 4092 pixels and the minimum is 16 pixels. Sizes outside of this range should not be set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ESHSZ[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 2	ESHSZ[11:2]	H'000	R/W	Horizontal size of the image transferred from the external buffer (in 4-pixel units).
1, 0	ESHSZ[1:0]		R	The lower two bits should be set to 0.

37.2.27 JPEG Interface Encode Destination Address Register 1 (JIFEDA1)

JIFEDA1 sets the destination address of the coded data. Set the start address in 8-pixel units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDA1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDA1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	EDA1[31:3]	H'0000 0000	R/W	Coded data destination address (in 8-pixel units)
2 to 0	EDA1[2:0]		R	The lower three bits should be set to 0.

37.2.28 JPEG Interface Encode Destination Address Register 2 (JIFEDA2)

JIFEDA2 sets the destination address of the coded data.

Set the start address in 8-pixel units. The coded data destination address, which is used by the JPU, changes every time the data transfer specified by JIFEDRSZ is completed: JIFEDA1 → JIFEDA2 → JIFEDA1 → . . . . (This register is valid only in auto-reload mode.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDA2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDA2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	EDA2[31:3]	H'0000 0000	R/W	Coded data destination address (in 8-pixel units)
2 to 0	EDA2[2:0]		R	The lower three bits should be set to 0.

37.2.29 JPEG Interface Encode Data Reload Size Register (JIFEDRSZ)

JIFEDRSZ sets the number of data (in 256-byte units) before the target address is switched. This register is valid only in auto-reload mode. This register should be set to H'0000 0000 in code amount count mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	EDRSZ[23:16]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDRSZ[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 8	EDRSZ[23:8]	H'00 0000	R/W	Coded data destination address (in 256-byte units)
7 to 0	EDRSZ[7:0]		R	The lower eight bits should be set to 0.

37.2.30 JPEG Interface Decoding Control Register (JIFDCNT)

JIFDCNT controls the decoding process. Note that the SWAP bit is also used in the encoding process.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PU[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	RE LOAD	SWAP[1:0]		BMS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	PU[11:0]	H'000	R/W	Processing Unit (These bits are valid only in line buffer mode.) [YCbCr 4:2:2 processing] Set the number of lines in vertical Y and C components (in multiples of 8: H'008 units). (Write 0 to the bits PU[2:0].) [YCbCr 4:2:0 processing] Set the number of lines in vertical Y components (in multiples of 16: H'010 units). (Write 0 to the bits PU[3:0].) A value equal to half of the number of lines in vertical Y components is automatically set as the number of lines in vertical C components. Note: The maximum number of settable lines is 2048 and the minimum number is 16. Values outside of this range should not be set.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RELOAD	0	R/W	Reload Mode 0: Reload mode disabled 1: Reload mode enabled
2, 1	SWAP[1:0]	00	R/W	Byte/Word Swap (see the JIFECNT register) Data transferred from the JPU is swapped (coded data in encoding, image data in decoding). 00: (1) (2) (3) (4) 01: (2) (1) (4) (3) [Byte swap] 10: (3) (4) (1) (2) [Word swap] 11: (4) (3) (2) (1) [Byte - word swap]
0	BMS	0	R/W	Buffer Mode 0: Frame buffer mode 1: Line buffer mode

Note: Bits other than SWAP can be modified only when the DSP bit in JCMOD is 1 (decoding process).

37.2.31 JPEG Interface Decode Source Address Register 1 (JIFDSA1)

JIFDSA1 sets the source address of the coded data. Set the start address of the coded data in 8-pixel units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSA1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSA1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DSA1[31:3]	H'0000 0000	R/W	Coded data source address (in 8-pixel units)
2 to 0	DSA1[2:0]		R	The lower three bits should be set to 0.

37.2.32 JPEG Interface Decode Source Address Register 2 (JIFDSA2)

JIFDSA2 sets the source address of the coded data. Set the start address in 8-pixel units. The source address of the coded data, which is used by the JPU, changes automatically every time the data transfer specified by JIFDDRSZ is completed: JIFDSA1 → JIFDSA2 → JIFDSA1 → . . . . (This register is valid only in auto-reload mode.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSA2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSA2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DSA2[31:3]	H'0000 0000	R/W	Coded data source address (in 8-pixel units)
2 to 0	DSA2[2:0]		R	The lower three bits should be set to 0.

37.2.33 JPEG Interface Decode Data Reload Size Register (JIFDDRSZ)

JIFDDRSZ sets the number of data (256-byte units) transferred before the address is changed (This register is valid only in auto reload mode).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	DDRSZ[23:16]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDRSZ[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 8	DDRSZ[23:8]	H'00 0000	R/W	Number of data to be transferred before the address is changed (in 256-byte units)
7 to 0	DDRSZ[7:0]		R	The lower eight bits should be set to 0.

37.2.34 JPEG Interface Decode Destination Memory Width Register (JIFDDMW)

JIFDDMW sets the memory width of the data written to the external buffer (see section 37.3.3, Storing Image Data).

Set this register in 8-pixel units. The maximum settable size is 4092 pixels and the minimum is 16 pixels. Sizes outside of this range should not be set

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDMW[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDMW[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DDMW[31:3]	H'0000 0000	R/W	Memory width of image data stored in external buffer (in 8-pixel units)
2 to 0	DDMW[2:0]		R	The lower three bits should be set to 0.

37.2.35 JPEG Interface Decode Destination Vertical Size Register (JIFDDVSZ)

JIFDDVSZ indicates the vertical size of the image written to the external buffer.

This register is valid only in decoding. Treat the image size as an error if the size exceeds the processable size.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDVSZ[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	DDVSZ [15:0]	H'0000	R	Vertical size of image written to external buffer After the JPEG coded data is analyzed, the values of JCVSZU and JCVSZD are automatically set to these bits in 4-pixel units.

37.2.36 JPEG Interface Decode Destination Horizontal Size Register (JIFDDHSZ)

JIFDDHSZ indicates the horizontal size of the image written to the external buffer. This register is valid only in decoding. Treat the image size as an error if the size exceeds the processable size.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDHSZ[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	DDHSZ [15:0]	H'0000	R	Horizontal size of image written to external buffer After the JPEG coded data is analyzed, the values of JCHSZU and JCHSZD are automatically set to these bits in 4-pixel units.

37.2.37 JPEG Interface Decode Destination Y Address Register 1 (JIFDDYA1)

JIFDDYA1 sets the destination address of the Y component. Set the address in 8-pixel units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDYA1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDYA1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DDYA1[31:3]	H'0000 0000	R/W	Y component destination address (in 8-pixel units)
2 to 0	DDYA1[2:0]		R	The lower three bits should be set to 0.

37.2.38 JPEG Interface Decode Destination C Address Register 1 (JIFDDCA1)

JIFDDCA1 sets the destination address of the C component. Set the address in 8-pixel units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDCA1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDCA1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DDCA1[31:3]	H'0000 0000	R/W	C component destination address (in 8-pixel units)
2 to 0	DDCA1[2:0]		R	The lower three bits should be set to 0.

37.2.39 JPEG Interface Decode Destination Y Address Register 2 (JIFDDYA2)

JIFDDYA2 sets the destination address of the Y component. This register is valid only in line buffer mode.

Set the address in 8-pixel units. The destination address of the Y component, which is used by the JPU, changes automatically every time the data transfer specified by JIFDDRSZ is completed: JIFDDYA1 → JIFDDYA2 → JIFDDYA1 → . . . .

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDYA2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDYA2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DDYA2[31:3]	H'0000 0000	R/W	Y component destination address (in 8-pixel units)
2 to 0	DDYA2[2:0]		R	The lower three bits should be set to 0.

37.2.40 JPEG Interface Decode Destination C Address Register 2 (JIFDDCA2)

JIFDDCA2 sets the destination address of the C component. This register is valid only in line buffer mode.

Set the address in 8-pixel units. The destination address of the C component, which is used by the JPU changes automatically every time the data transfer specified by JIFDDRSZ is completed: JIFDDCA1 → JIFDDCA2 → JIFDDCA1 → . . . .

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DDCA2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DDCA2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	DDCA2[31:3]	H'0000 0000	R/W	C Component Destination Address (in 8-pixel units)
2 to 0	DDCA2[2:0]		R	The lower three bits should be set to 0.

37.3 Operation

37.3.1 Encoding

(1) Overview of Processing

The encoding process flows are described below.

(a) Frame Buffer Mode

1. One frame of image data is transferred to the external buffer.
2. The JPEG core is activated.
   A marker is output. (After a marker is output, image data can be input.)
   Approximately 30000 cycles (necessary for making SOI to SOS markers) in normal mode
   Approximately 300 cycles (necessary for making SOS marker) in marker skip mode
3. Image data is transferred in MCUs from the external buffer to the JPU.
4. Image data is input to the JPEG core.
   The input data is processed in MCUs at any time in the JPEG core, and encoded data is output.
5. An interrupt is generated after frame data is processed completely.
   An interrupt is generated after the specified amount of data is reloaded in reload mode.

(b) Line Buffer Mode

1. The JPEG core is activated.
   A marker is output. (After a marker is output, image data can be input.)
   Approximately 30000 cycles (necessary for making SOI to SOS markers) in normal mode
   Approximately 300 cycles (necessary for making SOS marker) in marker skip mode
2. Y and C components of the image data are transferred to the external line buffer: Y components in multiples of 8 lines and C components in multiples of 8 lines (YCbCr 4:2:2), or Y components in multiples of 16 lines and C components in multiples of 8 lines (YCbCr 4:2:0).
   After transfer has been completed, a transfer end command is issued to the JPU.
   Every time the transfer to the external buffer is completed, a transfer end command is issued to the JPU.
3. Image data is transferred in MCUs from the external buffer to the JPU.
4. Image data is input to the JPEG core.
   The input data is processed in MCUs at any time in the JPEG core, and the encoded data is output.

5. After the JPU has read one external line buffer of the image data, the rest of the image data is transferred to the external line buffer.

   After all of the image data is transferred, a transfer end command is issued to the JPU (go to step 3).
6. After specified frame data processing is completed, an interrupt is generated.

   An interrupt is generated after the specified amount of data is reloaded in reload mode.

(2) Frame Buffer Mode (Encoding)

After the initial settings are completed, input image data in the external buffer and activate the JPU by setting the JSRT bit in JCCMD to B'1. After the JPU has been activated, the JPEG markers (SOI to SOS) are generated and output. It takes approximately 30000 cycles to generate the markers.

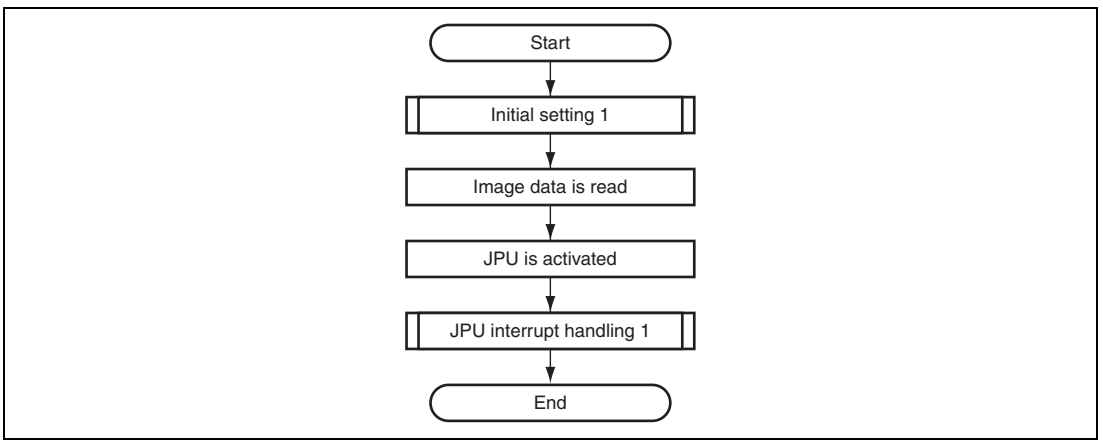
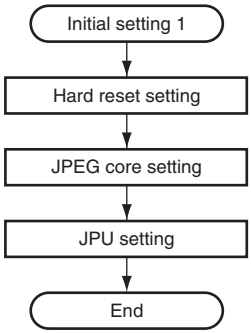


Figure 37.2 Flow of Encoding in Frame Buffer Mode

After a reset, set all necessary registers before the JPU is activated.



JPEG core settings

Subsampling setting:	REDU bit in JCMOD
Encoding/decoding setting:	DSP bit in JCMOD
Code amount count mode setting:	CCNT bit in JCMOD
Marker skip mode setting:	MSKIP bit in JCMOD
SOI marker delete setting:	SOIC bit in JCMOD
Quantization table number setting:	JCQTN
Huffman table number setting:	JCHTN
DRI setting:	JCDRIU, JCDRID
Vertical image size setting:	JCVSZU, JCVSZD
Horizontal image size setting:	JCHSZU, JCHSZD
Quantization table setting:	JCQTBL1 to JCQTBL4
Huffman table setting:	JCHTB0, JCHTB1, JCHTA0, JCHTA1

JPU settings

Interrupt setting:	INT14 to INT10 bits in JINTE
Subsampling setting:	INFT bit in JIFECNT
Buffer mode setting:	BUF bit in JIFECNT
Rotated read mode setting:	ROT bit in JIFECNT
Byte/word swap setting of read data:	SWAP bit in JIFECNT
Byte/word swap setting of write data:	SWAP bit in JIFDCNT
Reload mode setting:	RELOAD bit in JIFECNT
Y component source address setting:	JIFESYA1
C component source address setting:	JIFESCA1
Image buffer memory width setting:	JIFESMW
Vertical size of received image setting:	JIFESVSZ
Horizontal size of received image setting:	JIFESHSZ
Stream data write start address setting:	JIFEDA1, JIFEDA2
Number of data preceding address switching setting:	JIFEDRSZ

Figure 37.3 Flow of Initial Setting 1

Figure 37.4 shows the interrupt handling flow in frame buffer mode (encoding). When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.4 occur), process all interrupt sources.

- When the INS6 bit in the interrupt status register JINTS is set to B'1, the JPEG encoding has been completed. Though the JPU continues processing since the coded data to be transferred remains, the INS10 bit will be set to B'1 after the last coded data is transferred. The signal asserted by this interrupt source cannot be negated by clearing the corresponding interrupt status bit to 0. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to clear the interrupt signal.

- When the INS10 bit in the interrupt status register JINTS is set to B'1, all coded data has been transferred and the JPU has completed encoding; transfer the coded data outside the JPU. When reload mode is set, transfer the rest of the coded data. When the total size of the coded data transferred to the reload buffer matches the number of the coded data set by the JIFEDRSZ, bits INS10 and INS13 are simultaneously set to B'1. This interrupt occurs when the INT10 bit in the interrupt enable register JINTE is set to B'1.
- In encoding with reload mode, when the INS13 bit in the interrupt status register JINTS is set to B'1, it indicates that the number of coded data set by the JIFDDRSZ have been transferred. When the first interrupt occurs, transfer the coded data stored in the reload buffer to the other memory area. When the second and subsequent interrupts occur, transfer the coded data and set a command for restarting transfer to the reload buffer.

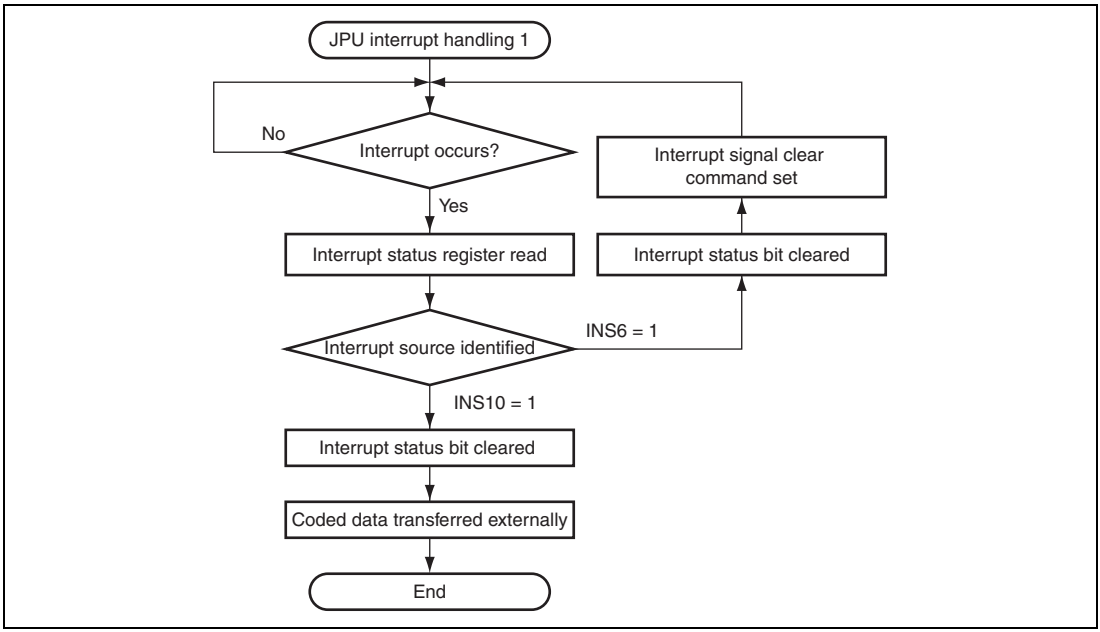


Figure 37.4 Flow of JPU Interrupt Handling 1 in Encoding

(3) Line Buffer Mode (Encoding)

After the initial settings are completed, activate the JPU by setting the JSRT bit in JCCMD to B'1. After the JPU has been activated, the JPEG markers (SOI to SOS) are generated and output. It takes approximately 30000 cycles to generate the markers. After the image data is stored in the external buffer and a line buffer write end command is issued, the JPU loads the image data from the external buffer and starts encoding it.

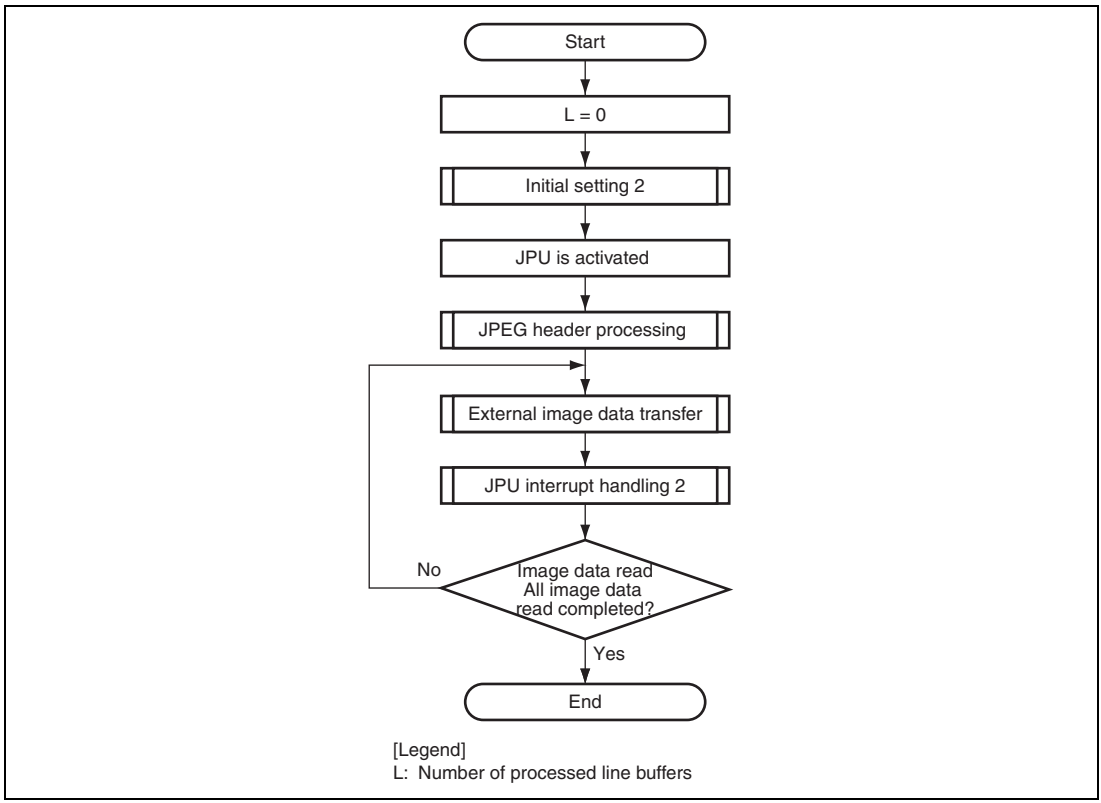
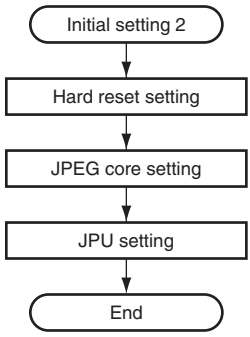


Figure 37.5 Flow of Encoding in Line Buffer Mode

After a reset, set all necessary registers before the JPU is activated.



JPEG core settings

Subsampling setting:	REDU bit in JCMOD
Encoding/decoding setting:	DSP bit in JCMOD
Code amount count mode setting:	CCNT bit in JCMOD
Marker skip mode setting:	MSKIP bit in JCMOD
SOI marker delete setting:	SOIC bit in JCMOD
Quantization table number setting:	JCQTN
Huffman table number setting:	JCHTN
DRI setting:	JCDRIU, JCDRID
Vertical image size setting:	JCVSZU, JCVSZD
Horizontal image size setting:	JCHSZU, JCHSZD
Quantization table setting:	JCQTBL1 to JCQTBL4
Huffman table setting:	JCHTB0, JCHTB1, JCHTA0, JCHTA1

JPU settings

Interrupt setting:	INT14 to INT10 bits in JINTE
Subsampling setting:	INFT bit in JIFECNT
Buffer mode setting:	BUF bit in JIFECNT
Rotated read mode setting:	ROT bit in JIFECNT
Byte/word swap setting of read data:	SWAP bit in JIFECNT
Byte/word swap setting of write data:	SWAP bit in JIFDCNT
Reload mode setting:	RELOAD bit in JIFECNT
Processing unit setting:	PU bit in JIFECNT
Y component source address setting:	JIFESYA1, JIFESYA2
C component source address setting:	JIFESCA1, JIFESYA2
Image buffer memory width setting:	JIFESMW
Vertical size of received image setting:	JIFESVSZ
Horizontal size of received image setting:	JIFESHSZ
Stream data write start address setting:	JIFEDA1, JIFEDA2
Number of data preceding address switching setting:	JIFEDRSZ

Figure 37.6 Flow of Initial Setting 2

In line buffer mode, set the line buffer processing restart command after activating the JPU. Transferring the image data to the line buffer can be done before the JPU is activated. Two line buffers are provided: one for the data to be output to the JPU and the other for the data to be transferred from the external modules. Thus, transfer the image data to the line buffer after confirming that the data stored in the buffer for outputting data to the JPU has been transferred to the JPU through interrupts.

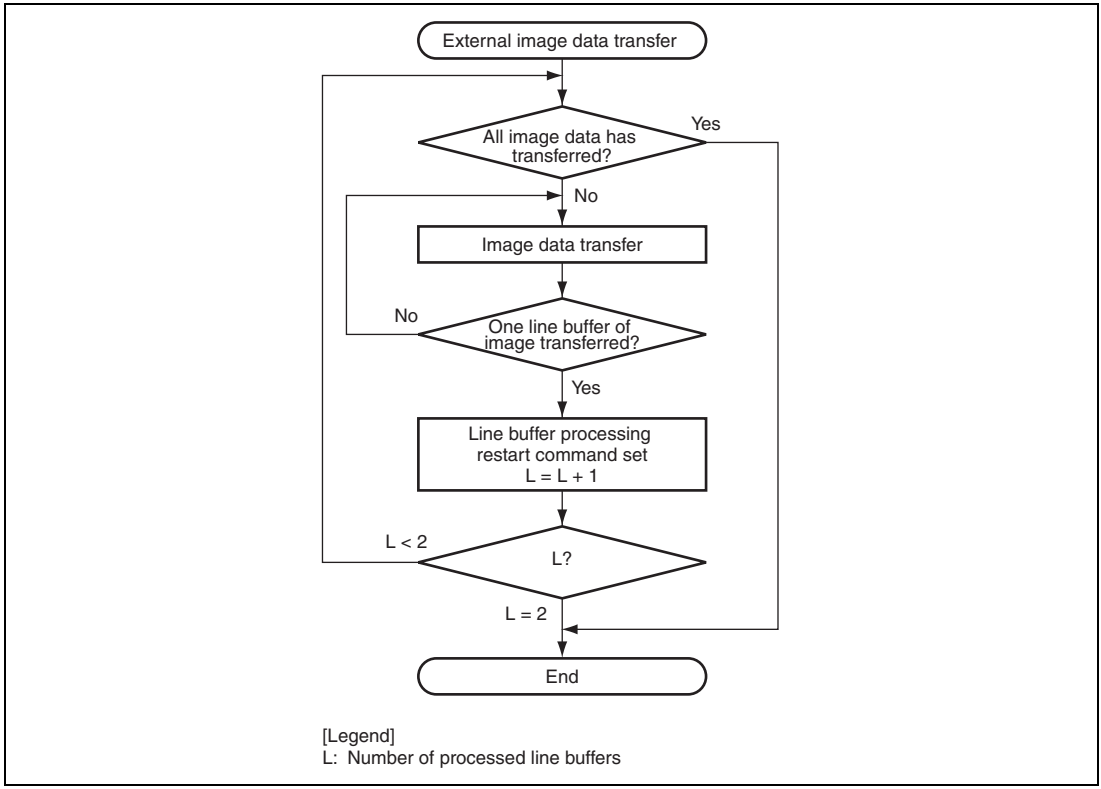


Figure 37.7 Flow of External Image Data Transfer

When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.8 occur), process all interrupt sources.

- When the INS12 bit or INS11 bit in the interrupt status register JINTS is set to B'1, the transfer of one line buffer of image data has been completed. Thus transfer new data to the line buffer. This interrupt occurs when the INT12 or INT11 bit in the interrupt enable register JINTE is set to B'1.
- When the INS6 bit in the interrupt status register JINTS is set to B'1, the JPEG encoding has been completed. Though the JPU continues processing since coded data to be transferred remains, the INS10 bit will be set to B'1 after the last coded data is transferred. The signal which is asserted by this interrupt source cannot be negated by clearing the corresponding interrupt status bit to 0. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to negate the interrupt signal.

- When the INS10 bit in the interrupt status register JINTS is set to B'1, all coded data has been transferred and the JPU has completed encoding; transfer the coded data outside the JPU. When reload mode is set, transfer the rest of the coded data. When the total size of the coded data transferred to the reload buffer matches the number of coded data set by JIFEDRSZ, bits INS10 and INS13 are simultaneously set to B'1. This interrupt occurs when the INT10 bit in the interrupt enable register JINTE is set to B'1.
- In encoding with reload mode, when the INS13 bit in the interrupt status register JINTS is set to B'1, it indicates that the number of coded data set by JIFDDRSZ have been transferred. When the first interrupt occurs, transfer the coded data stored in the reload buffer to the other memory area. When the second and subsequent interrupts occur, transfer the coded data and set a command for restarting transfer to the reload buffer.

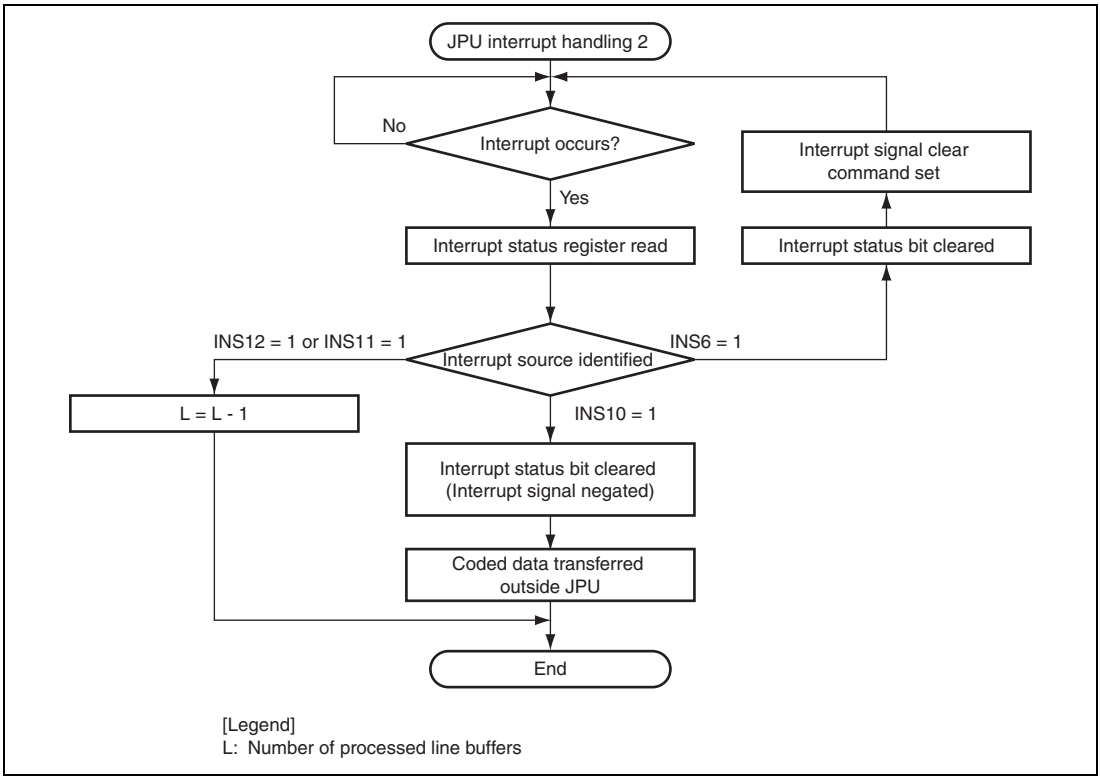


Figure 37.8 Flow of JPU Interrupt Handling 2

(4) JPEG Coded Data Format

Figure 37.9 shows the data output stream in encoding. The amount of coded data from SOI to EOI is indicated by the JCDTCU, JCDTCM, and JCDTCD registers. When the JCDRIU and JCDRID registers are set to H'0000 0000, the following markers are not output.

- DRI marker
- RST marker (in encoded image data)

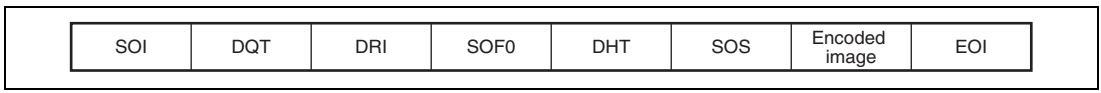


Figure 37.9 JPEG Coded Data Format

DQT: Not output for unused table

DHT: Output in order DC0, AC, DC1, and AC1. Not output for unused table.

SOF0: Component identifiers are C1 = first component, C2 = second component, and C3 = third component

SOS: Scan component selectors are CS1 = first component, CS2 = second component, and CS3 = third component

Header Volume (Reference)

- SOI: 2 bytes (FFD8)
- DQT: 134 bytes when two quantization tables are used, 199 bytes when three quantization tables are used (±65 bytes/table increase or decrease)
- DRI: 6 bytes
- SOF0: 19 bytes (4:2:2)
- DHT: 420 bytes (two tables are used)
- SOS: 14 bytes (4:2:2)
- EOI: 2 bytes (FFD9)

(a) JPEG Coded Data Format when Marker Skip Mode is Set

When MSKIP bits in JCMOD are set to B'01, SOI to DHT markers are not output. The RST marker output depends on the settings of the DRI upper and DRI lower registers. The amount of coded data from SOS to EOI is indicated by the JCDTCU, JCDTCM, and JCDTCD registers.

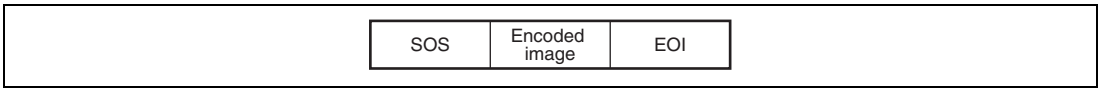


Figure 37.10 JPEG Coded Data Format when Marker Skip Mode is Set

(b) JPEG Coded Data Format when SOI Marker is Deleted

When the SOIC bit in JCMOD is set to B'1, the SOI marker is not output. This setting is valid when marker skip mode is not set. The amount of coded data from DQT to EOI equals the value indicated by JCDTCU, JCDTCM, and JCDTCD – 2.

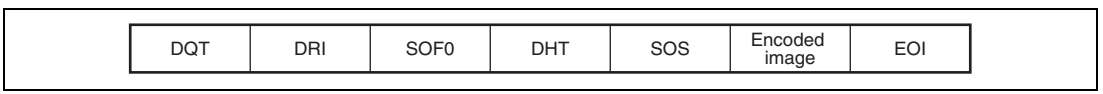


Figure 37.11 JPEG Coded Data Format when SOI Marker is Deleted

(c) Outputting End of JPEG Coded Data

The JPU transfers data as it ends at a 16-byte boundary. When the end of the coded data (the EOI marker) is not placed at the 16-byte boundary, H'FF is inserted into the data and the JPU outputs it. An H'FF byte is not included in the amount of the coded data indicated by the JCDTCU, JCDTCM, and JCDTCD registers.

(5) Code Amount Count Mode

In this mode, only the amount of coded data is counted. In encoding, set the CCNT bit in JCMOD to B'1 and enable reload mode, and set the JIFEDRSZ register (the number of data before the target address is switched) to 0; thus, the JPEG coded data is not output.

(6) Table Setting

(a) Quantization Table Specification

The order of addresses shown in 8 × 8 blocks corresponds to that of the register addresses. Do not access this table while the JPU is in processing.

Table 37.3 Quantization Table

00	01	02	03	04	05	06	07
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37
38	39	3A	3B	3C	3D	3E	3F

JCQTBL0 (H'FE99 0000) = H'0001 0203
JCQTBL0 (H'FE99 0004) = H'0405 0607
:
JCQTBL0 (H'FE99 003C) = H'3C3D 3E3F

(b) Huffman Table Specification

Examples of the Huffman table specification given in the ITU-T T81 Annex K.3.3 recommended by JPEG are shown below. In encoding, the following settings must be specified for all codes so that Huffman codes can be generated for all group numbers.

- DC Huffman table: The number of codes for each code length is 12 codes.
  The group numbers in order of frequency of occurrence are 12.
- AC Huffman table: The number of codes for each code length is 162 codes.
  The zero run length/the group numbers in order of frequency of occurrence are 162.

Do not access the following tables while the JPU is in processing.

- Table K.3/T81
  JCHTBD0 (H'FE99 0100) = H'0001 0501
  JCHTBD0 (H'FE99 0104) = H'0101 0101
  JCHTBD0 (H'FE99 0108) = H'0100 0000
  JCHTBD0 (H'FE99 010C) = H'0000 0000
  JCHTBD0 (H'FE99 0110) = H'0001 0203
  JCHTBD0 (H'FE99 0114) = H'0405 0607
  JCHTBD0 (H'FE99 0118) = H'0809 0A0B

- Table K.4/T81
  JCHTBD1 (H'FE99 0200) = H'0003 0101
  JCHTBD1 (H'FE99 0204) = H'0101 0101
  JCHTBD1 (H'FE99 0208) = H'0101 0100
  JCHTBD1 (H'FE99 020C) = H'0000 0000
  JCHTBD1 (H'FE99 0210) = H'0001 0203
  JCHTBD1 (H'FE99 0214) = H'0405 0607
  JCHTBD1 (H'FE99 0218) = H'0809 0A0B

- Table K.5/T81
  JCHTBA0 (H'FE99 0120) = H'0002 0103
  JCHTBA0 (H'FE99 0124) = H'0302 0403
  JCHTBA0 (H'FE99 0128) = H'0505 0404
  JCHTBA0 (H'FE99 012C) = H'0000 017D
  JCHTBA0 (H'FE99 0130) = H'0102 0300
  JCHTBA0 (H'FE99 0134) = H'0411 0512
  JCHTBA0 (H'FE99 0138) = H'2131 4106
  JCHTBA0 (H’FE99 013C) = H'1351 6107
  JCHTBA0 (H’FE99 0140) = H'2271 1432
  JCHTBA0 (H’FE99 0144) = H'8191 A108
  :
  JCHTBA0 (H'FE99 01C0) = H'E3E4 E5E6
  JCHTBA0 (H'FE99 01C4) = H'E7E8 E9EA
  JCHTBA0 (H'FE99 01C8) = H'F1F2 F3F4
  JCHTBA0 (H'FE99 01CC) = H'F5F6 F7F8
  JCHTBA0 (H'FE99 01D0) = H'F9FA 0000

- Table K.6/T81
  JCHTBA1 (H'FE99 0220) = H'0002 0102
  JCHTBA1 (H'FE99 0224) = H'0404 0304
  JCHTBA1 (H'FE99 0228) = H'0705 0404
  JCHTBA1 (H'FE99 022C) = H'0001 0277
  JCHTBA1 (H'FE99 0230) = H'0001 0203
  JCHTBA1 (H'FE99 0234) = H'1104 0521
  JCHTBA1 (H'FE99 0238) = H'3106 1241
  JCHTBA1 (H'FE99 023C) = H'5107 6171
  JCHTBA1 (H'FE99 0240) = H'1322 3281
  JCHTBA1 (H'FE99 0244) = H'0814 4291
  :
  JCHTBA1 (H'FE99 02C0) = H'E2E3 E4E5
  JCHTBA1 (H'FE99 02C4) = H'E6E7 E8E9
  JCHTBA1 (H'FE99 02C8) = H'EAF2 F3F4
  JCHTBA1 (H'FE99 02CC) = H'F5F6 F7F8
  JCHTBA1 (H'FE99 02D0) = H'F9FA 0000

(7) Rotated Read Mode Setting

In encoding with frame buffer mode, image rotation processing can be performed (set the ROT bit in JIFECNT). When this processing is specified, set the source addresses shown in Figure 37.12 in the JIFESYA1 and JIFECYA1 registers.

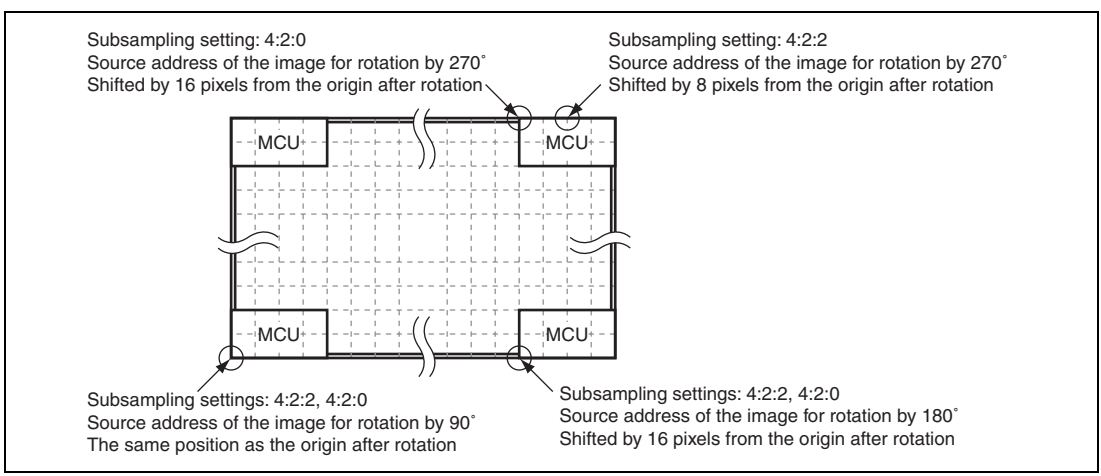


Figure 37.12 Position of Source Addresses for Rotation

37.3.2 Decoding

(1) Overview of Processing

The decoding process flows are described below.

(a) Frame Buffer Mode

1. JPEG core processing (decoding)
   Interrupts are generated every time the specified amount of data has been reloaded in reload mode.
2. Decoding data and transferring
   One frame of data can be processed at anytime and transferred.
3. An interrupt is generated after all of the processing is completed.

(b) Line Buffer Mode

1. JPEG core processing (decoding)
   Interrupts are generated every time the specified amount of data has been reloaded in reload mode.
2. Decoding data and transferring
3. Y and C components of the decoded data are transferred to the external buffer: Y components in multiples of 8 lines and C components in multiples of 8 lines (YCbCr 4:2:2), or Y components in multiples of 16 lines and C components in multiples of 16 lines (YCbCr 4:2:0).
   Every time when the transfer to the external buffer is completed, a transfer end command is issued to the JPU.
4. An interrupt is generated after all of the processing is completed.

(2) Frame Buffer Mode (Decoding)

After the initial setting is completed, activate the JPU by setting the JSRT bit in JCCMD to B'1. After the JPU has been activated, it obtains the coded data from the external buffer and starts decoding it. When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.13 occur), process all interrupt sources.

- When the INS14 bit in the interrupt status register JINTS is set to B'1, the coded data in the reload buffer has been transferred. This interrupt occurs when the INT14 bit in the interrupt enable register JINTE is set to B'1.

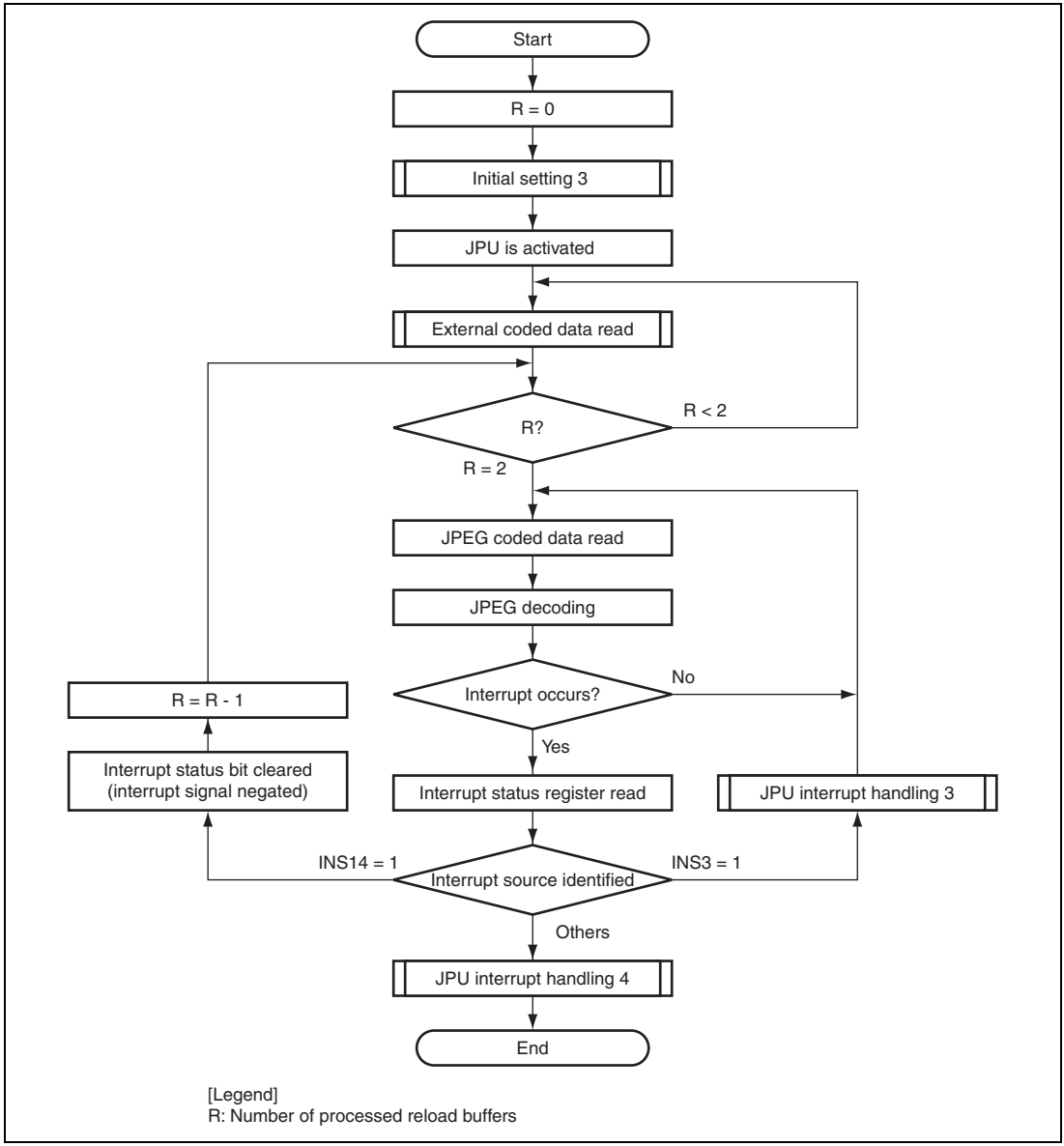
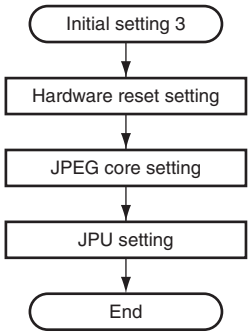


Figure 37.13 Flow of Decoding in Frame Buffer Mode

After a reset, set all necessary registers before the JPU is activated.



JPEG core settings

Encoding/decoding setting:	DSP bit in JCMOD
Interrupt setting:	INT7 to INT5, INT3 bits in JINTE

JPU settings

Interrupt setting:	INT14 to INT10 bits in JINTE
Byte/word swap setting of read data:	SWAP bit in JIFECNT
Byte/word swap setting of write data:	SWAP bit in JIFDCNT
Buffer mode setting:	BMS bit in JIFDCNT
Reload mode setting:	RELOAD bit in JIFDCNT
Stream data read start address setting:	JIFDS1, JIFDS2
Number of data preceding address switching setting:	JIFDDRSZ
Pixels in horizontal array can be stored in SDRAM:	JIFDDMW
Y component destination address setting:	JIFDDYA1
C component destination address setting:	JIFDDCA1

Figure 37.14 Flow of Initial Setting 3

Before the JPU is activated, prepare the coded data since the JPU starts reading the data immediately after being activated when reload mode is not set. If not all of the coded data is prepared, transfer the coded data to the buffer sequentially so that the JPU can read the data smoothly.

When reload mode is set, set a reload transfer write end command after the JPU is activated. The coded data can be transferred to the reload buffer before the JPU is activated. Two reload buffers are provided: one is used for the data to be output to the JPU and the other is for the data to be transferred from the external modules. Thus, transfer the data to the reload buffer after confirming that the data stored in the buffer used to output the data to the JPU has been transferred to the JPU through interrupts.

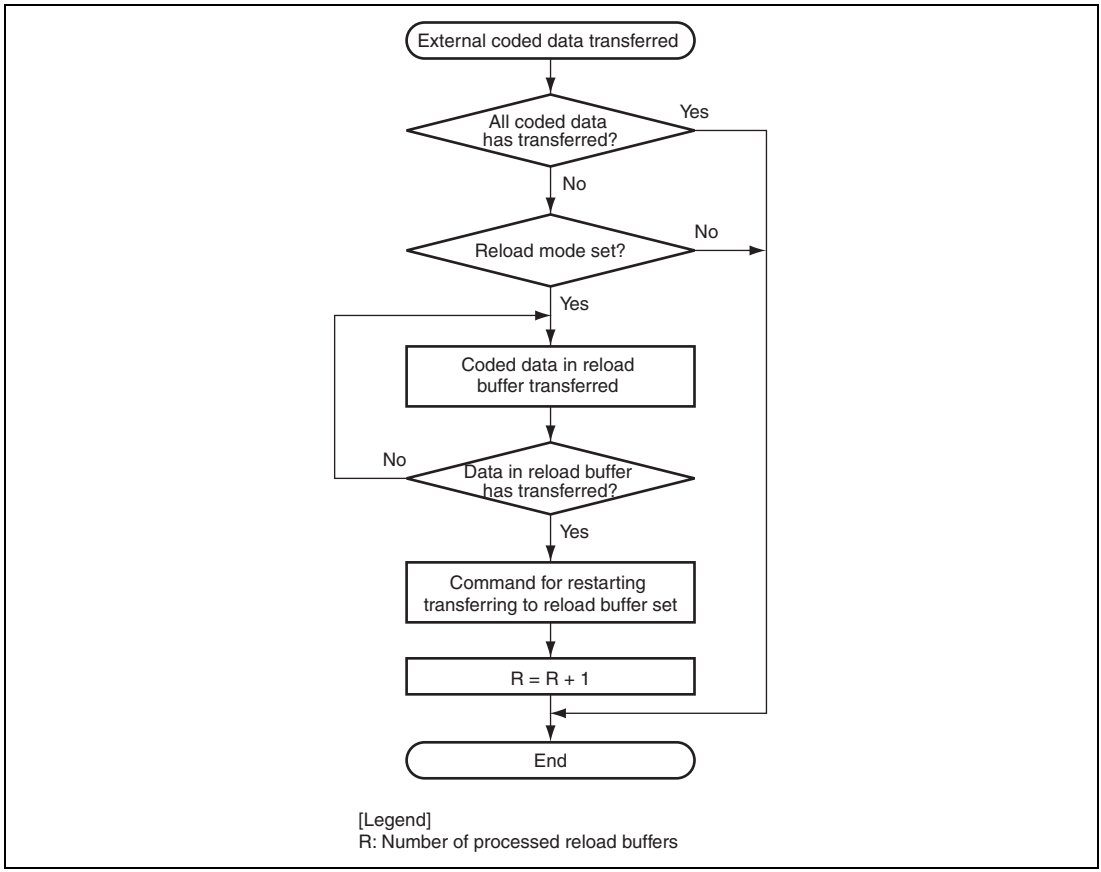


Figure 37.15 Flow of Transferring External Coded Data

When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.16 occur), process all interrupt sources.

- When the INS3 bit in the interrupt status register JINTS is set to B'1, the JPEG coded data is prepared and its image size and subsampling setting can be read. After reading of all necessary registers, set the command for revoking the processing-stopped state to resume decoding. This interrupt occurs when the INT3 bit in the interrupt enable register JINTE is set to B'1. The interrupt signal which is asserted by this interrupt source cannot be negated through clearing the corresponding interrupt status bit to 0. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to clear the interrupt signal.

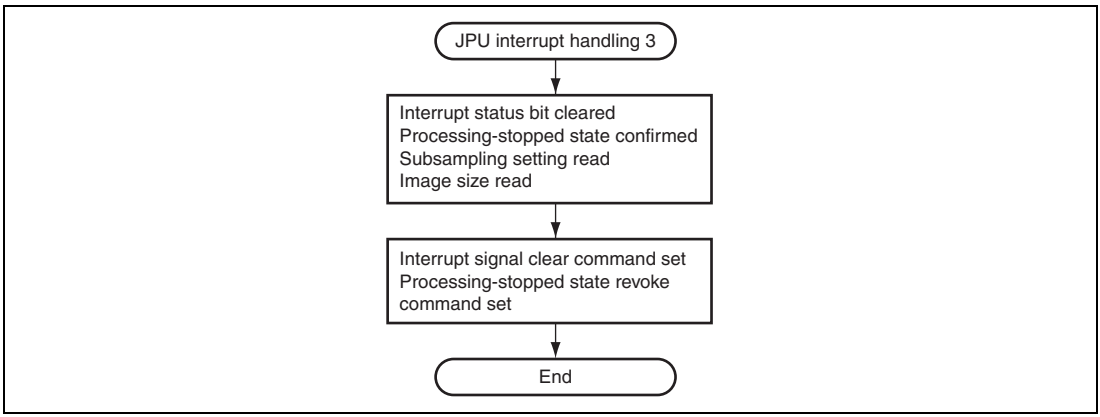


Figure 37.16 Flow of JPU Interrupt Handling 3

Figure 37.17 shows the interrupt handling flow in decoding. When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.17 occur), process all interrupt sources.

- When the INS5 bit in the interrupt status register JINTS is set to B'1, the input JPEG coded data has an error and the JPU stops the decoding process. Read ERR bits in JCDERR and identify its error source. The interrupt signal which is asserted by this interrupt source cannot be negated by clearing the corresponding interrupt status bit. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to clear the interrupt signal.
- When the INS6 bit in the interrupt status register JINTS is set to B'1, the JPEG decoding process is completed. When the INS10 bit is set to B'1, all of the image data is transferred and JPU completes decoding. The interrupt signal which is asserted by this interrupt source cannot be negated by clearing the corresponding interrupt status bit. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to clear the interrupt signal.

- When the INS10 bit in the interrupt status register JINTS is set to B'1, all of the image data has been transferred and the JPU has completed decoding; transfer the image data outside the JPU. This interrupt occurs when the INT10 bit in the interrupt enable register JINTE is set to B'1.

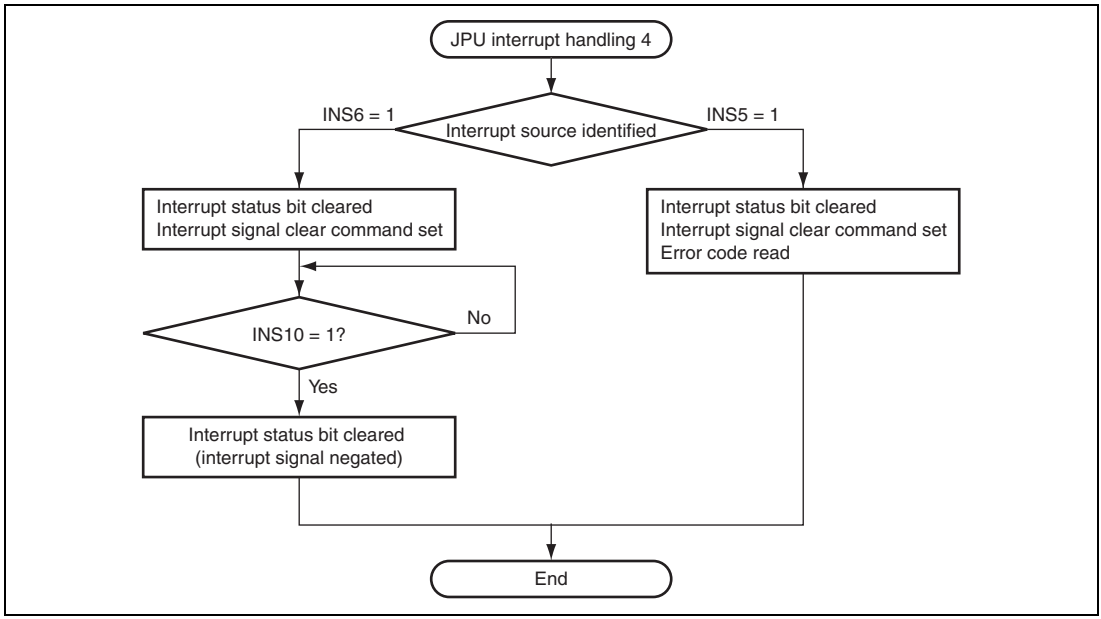


Figure 37.17 Flow of JPU Interrupt Handling 4

(3) Line Buffer Mode (Decoding)

After the initial settings are completed, activate the JPU by setting the JSRT bit in JCCMD to B'1. After the JPU has been activated, transfer the coded data to the external buffer. After the coded data has been stored in the external buffer, the JPU reads the data from the external buffer and starts decoding it.

When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.18 occur), process all interrupt sources.

- When the INS14 bit in the interrupt status register JINTS is set to B'1, the coded data in the reload buffer has been transferred. This interrupt occurs when the INT14 bit in the interrupt enable register JINTE is set to B'1.

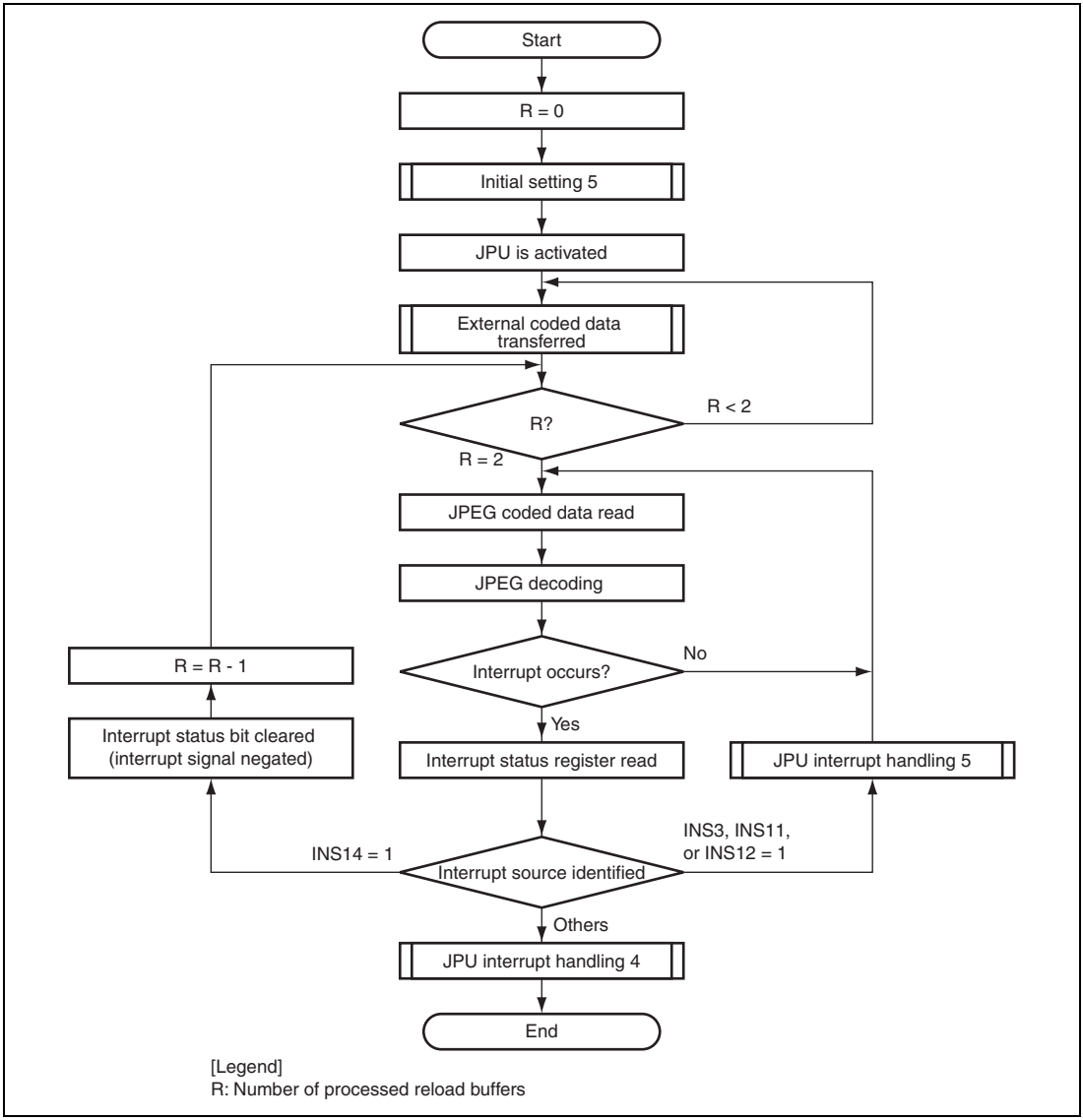
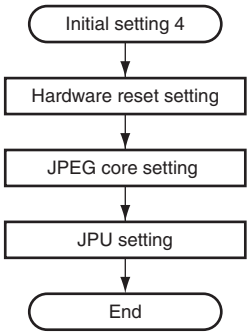


Figure 37.18 Flow of Decoding Process in Line Buffer Mode

After a reset, set all necessary registers before the JPU is activated.



JPEG core settings

Encoding/decoding setting:	DSP bit in JCMOD
Interrupt setting:	INT7 to INT5, INT3 bits in JINTE

JPU settings

Interrupt setting:	INT14 to INT10 bits in JINTE
Byte/word swap setting of read data:	SWAP bit in JIFECNT
Byte/word swap setting of write data:	SWAP bit in JIFDCNT
Buffer mode setting:	BMS bit in JIFDCNT
Reload mode setting:	RELOAD bit in JIFDCNT
Processing unit setting:	PU bit in JIFDCNT
Stream data read start address setting:	JIFDS1, JIFDS2
Number of data preceding address switching setting:	JIFDDRSZ
Pixels in horizontal array can be stored in SDRAM:	JIFDDMW
Y component destination address setting:	JIFDDYA1, JIFDDYA2
C component destination address setting:	JIFDDCA1, JIFDDCA2

Figure 37.19 Flow of Initial Setting 4

When multiple interrupts occur (also when multiple interrupts that are not shown in Figure 37.20 occur), process all interrupt sources.

- When the INS3 bit in the interrupt status register JINTS is set to B'1, the JPEG coded data is prepared and its image size and subsampling setting can be read. After reading all necessary registers, set the command for revoking the processing-stopped state to resume decoding. This interrupt occurs when the INT3 bit in the interrupt enable register JINTE is set to B'1. The interrupt signal which is asserted by this interrupt source cannot be negated by clearing the corresponding interrupt status bit to 0. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to clear the interrupt signal.
- When the INS11 or INS12 bit in the interrupt status register JINTS is set to B'1, one line buffer of image data has been transferred. When the first interrupt occurs, transfer the image data stored in the line buffer to the other memory area. When the second and subsequent interrupts occur, transfer the image data and set a command for restarting transfer to the line buffer. This interrupt occurs when the INT11 or INT12 bit in the interrupt enable register JINTE is set to B'1.

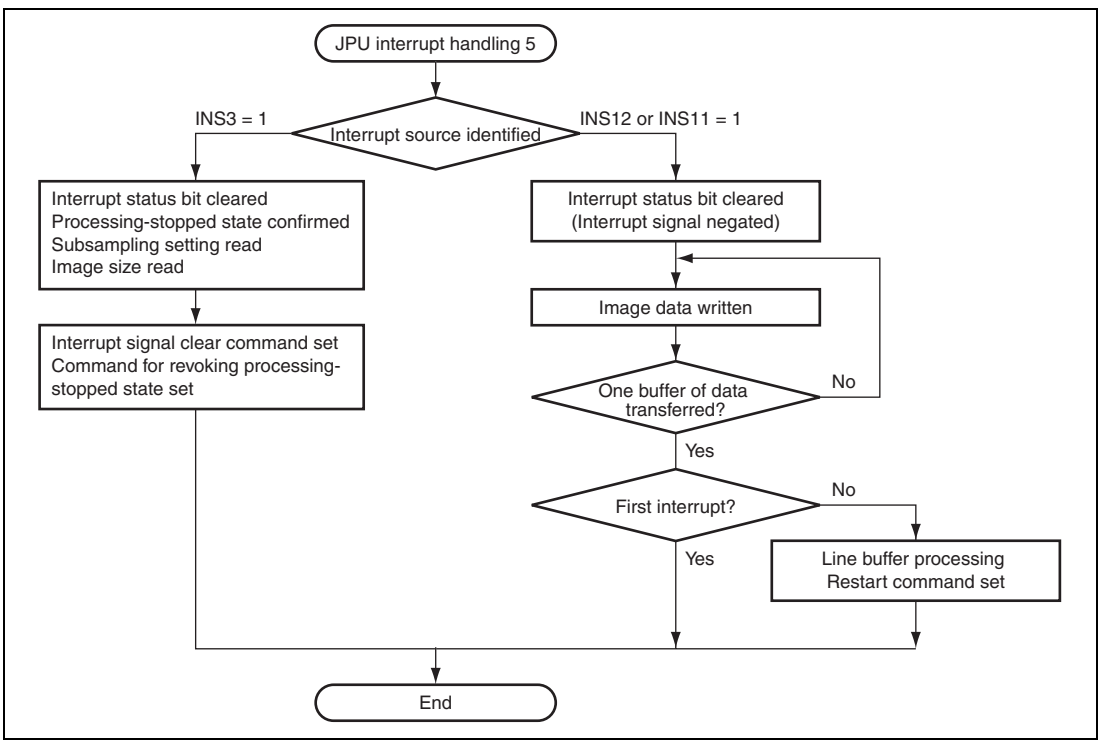


Figure 37.20 Flow of JPU Interrupt Handling 5

(4) Input JPEG Coded Data

Markers to be processed in decoding are SOS, SOF0, SOS, DQT, DHT, DRI, RSTm, and EOI. Other markers except for the error markers shown below are ignored even if they are read.

(5) JPEG Decoding Errors

(a) Error Marker

If an error marker is found while analyzing encoded data for decoding, identify the error type by the code and set the code (shown in Table 37.4) to ERR bits in JCDERR. When an error is detected, the JPU generates an interrupt signal and stops decoding. The stored code is 1010 (default value) at the start of processing for the next frame or after a bus reset.

Table 37.4 Decoding Error Codes

Code	Description
0000	Normal
0001	SIO not detected: SIO not detected until EOI detected
0010	SOF1 to SOFF detected
0011	Subsampling setting other than YCbCr 4:4:4 (H = 1:1:1, V = 1:1:1)/YCbCr 4:2:2 (H = 2:1:1, V = 1:1:1)/YCbCr 4:1:1 (H = 4:1:1, V = 1:1:1)/YCbCr 4:2:0 (H = 2:1:1, V = 2:1:1) detected
0100	SOF accuracy error: Other than 8 detected
0101	DQT accuracy error: Other than 0 detected
0110	Component error 1: The number of SOF0 header components detected is other than 1, 3, or 4
0111	Component error 2: The number of components differs between SOF0 header and SOS
1000	SOF0, DQT, and DHT not detected when SOS detected
1001	SOS not detected: SOS not detected until EOI detected
1010	EOI not detected (default)
1011	Restart interval data number error detected
1100	Image size error detected
1101	Last MCU data number error detected
1110	Block data number error detected

(b) Huffman Coded Segment Error

In decoding, if it is determined that the increase or decrease in the number of data to be decoded is caused by an error such as bit inversion or a lack of data in the Huffman coded segment through analysis of the encoded data, the error type is identified and the error code is set in ERR bits in JCDFRR. When a necessary bit among INT7 to INT5 in JINTE is set to B'1, the error code is set and decoding is terminated by an interrupt signal generation. The code to be stored is 1010 (default value) at the start of processing for the next frame or after a bus reset.

In this error detection, an increase or decrease in the number of data to be decoded is detected; an error is not detected even when any other error occurs in the Huffman coded segment and the error does not cause an increase or decrease in the number of data to be decoded.

[Example]

The number of data in a Huffman coded segment with subsampling setting 4:2:2, DRI = 2, X = 80 pixels, and Y = 8 pixels

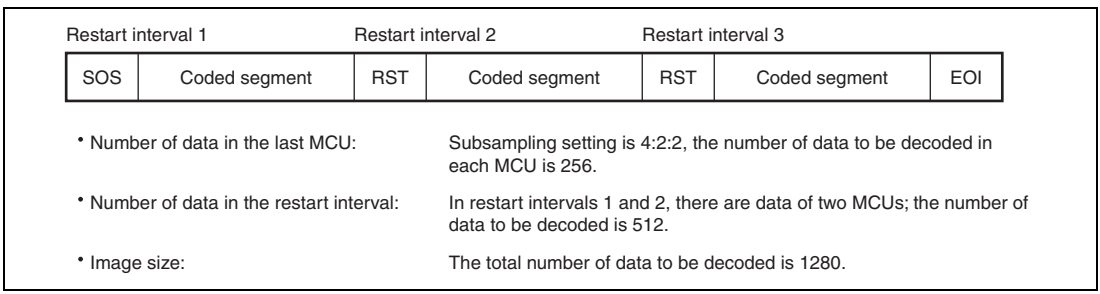


Figure 37.21 Huffman Coded Segment

Table 37.5 Segment Error Codes

Code	Description
0000	Normal
1011	Restart interval data number error: The number of data in each interval is compared with the number of data specified by the DRI marker. If an interval has more or less data that is specified by the DRI marker, the decoding error code (1011) is set. The last interval which is shorter than the restart interval is not compared. If the DRI marker segment is not placed or the specified number is 00, an error is not detected even if the RSTm marker is placed. Also an m which indicates the order of RSTm marker modulo 8 (m = 0 to 7) is exempt from the error detection analysis. When the INT7 bit in JINTE is set to B'0, this error is not detected.
1100	Image size error: The data number of an image which is calculated from the number of lines specified by the frame parameter and the number of samples per line is compared with the total number of data from SOS to EOI (in pixel units). If the numbers of data do not match, the decoding error code (1100) is set. When the INT6 bit in JINTE is set to B'0, this error is not detected. The data number of an image is shown in MCU units. Thus the number of lines and the number of samples per line need to be shown in MCU units.
1101	Last MCU data number error: Whether the number of data in the MCUs at the EOI detection is shown in MCU units is checked and fractions are detected. If error (1100) occurs simultaneously, error (1100) has priority. When the INT5 bit in JINTE is set to B'0, this error is not detected.
1110	Block data number error: Whether a block is an 8×8 array is checked; the check is performed for fractions. When bits INT7 to INT5 in JINTE are set to B'0, this error is not detected.

37.3.3 Storing Image Data

Storing the image data in the buffer and transferring the data to or from the JPU are described below.

(1) Frame Buffer Mode
(VGA: 640 × 480, Image data area ⇔ QVGA: Transferring image data 320 × 240)

- Example of register settings

	VGA → VGA
JIFESMW/JIFDDMW: H'0000 0280 (640)	H'0000 0280
JIFESYAI/JIFDDYAI: H'0001 2CA0 (76960)	H'0000 0000
JIFESHSZ/JIFDDHSZ: H'0000 0140 (320)	H'0000 0280
JIFESVSZ/JIFDDVSZ: H'0000 00F0 (240)	H'0000 01E0
JIFESCAI/JIFDDCA1: H'0005 DCA0 (384160)	H'0004 B000

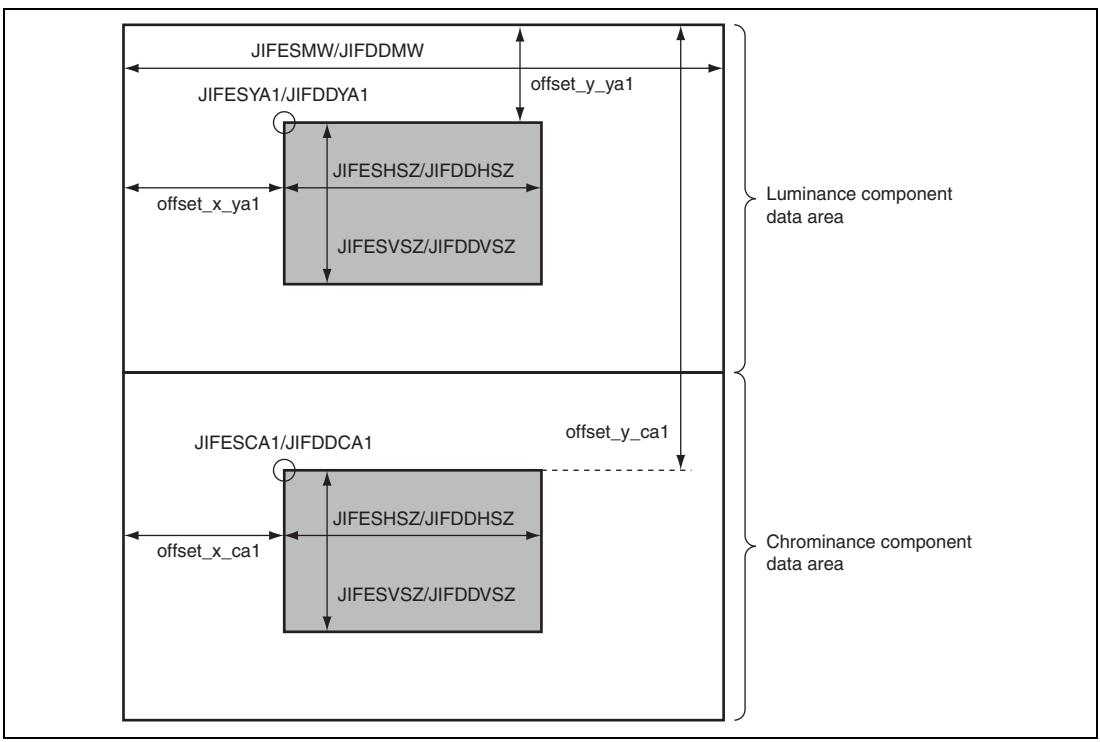


Figure 37.22 Data Allocation Image in Frame Buffer

(2) Storing Image Data

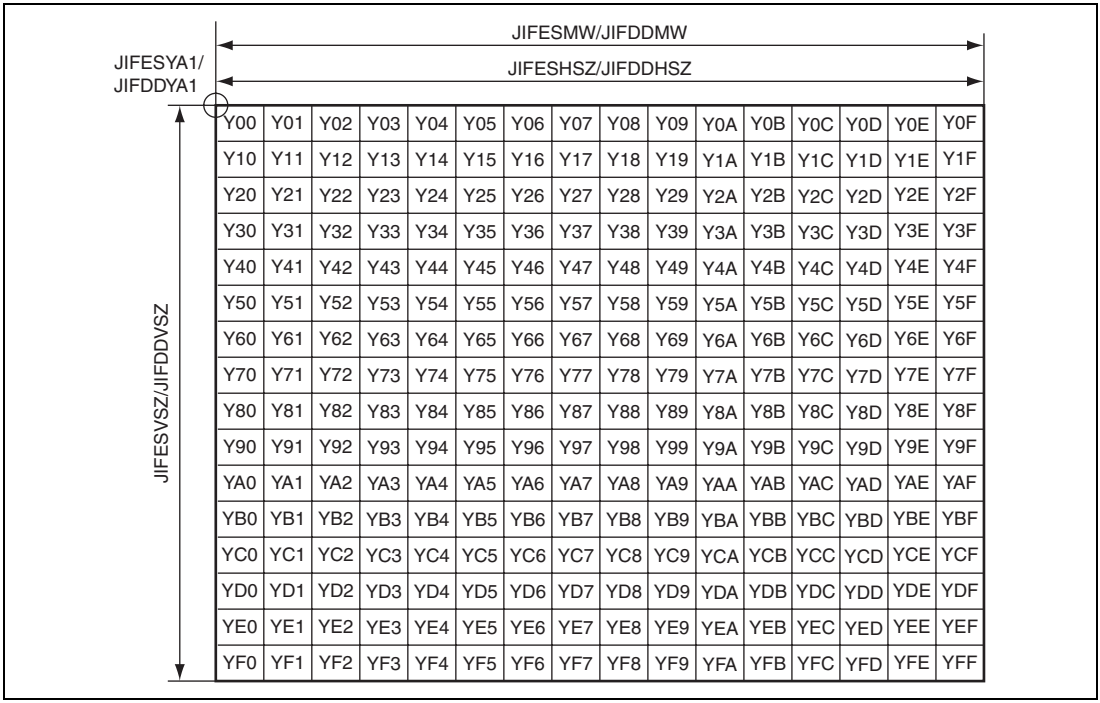
(a) Luminance component data

Specify JIFESYA1/JIFDDYA1 (JIFESYA2/JIFDDYA2) as the origin of the image. Transfer the data in the area specified by JIFESHSZ/JIFDDHSZ and JIFESVSZ/JIFDDVSZ.

- Register setting value:
  JIFESMW/JIFDDMW: H'0000 0010 (16)
  JIFESYA1/JIFDDYA1: H'0000 0000 (0)
  JIFESHSZ/JIFDDHSZ: H'0000 0010 (16)
  JIFESVSZ/JIFDDVSZ: H'0000 0010 (16)

- Data stored in buffer:
  H'0000 0000 = Y00, Y01, Y02, Y03
  H'0000 0004 = Y04, Y05, Y06, Y07
  H'0000 0008 = Y08, Y09, Y0A, Y0B
  H'0000 000C = Y0C, Y0D, Y0E, Y0F
  :
  H'0000 00F0 = YF0, YF1, YF2, YF3
  H'0000 00F4 = YF4, YF5, YF6, YF7
  H'0000 00F8 = YF8, YF9, YFA, YFB
  H'0000 00FC = YFC, YFD, YFE, YFF

- Transferring data to the JPU:
  Burst transfer is performed in 1-MCU units in the line direction. One line is 16 bytes.



Y00	Y01	Y02	Y03	Y04	Y05	Y06	Y07	Y08	Y09	Y0A	Y0B	Y0C	Y0D	Y0E	Y0F
Y10	Y11	Y12	Y13	Y14	Y15	Y16	Y17	Y18	Y19	Y1A	Y1B	Y1C	Y1D	Y1E	Y1F
Y20	Y21	Y22	Y23	Y24	Y25	Y26	Y27	Y28	Y29	Y2A	Y2B	Y2C	Y2D	Y2E	Y2F
Y30	Y31	Y32	Y33	Y34	Y35	Y36	Y37	Y38	Y39	Y3A	Y3B	Y3C	Y3D	Y3E	Y3F
Y40	Y41	Y42	Y43	Y44	Y45	Y46	Y47	Y48	Y49	Y4A	Y4B	Y4C	Y4D	Y4E	Y4F
Y50	Y51	Y52	Y53	Y54	Y55	Y56	Y57	Y58	Y59	Y5A	Y5B	Y5C	Y5D	Y5E	Y5F
Y60	Y61	Y62	Y63	Y64	Y65	Y66	Y67	Y68	Y69	Y6A	Y6B	Y6C	Y6D	Y6E	Y6F
Y70	Y71	Y72	Y73	Y74	Y75	Y76	Y77	Y78	Y79	Y7A	Y7B	Y7C	Y7D	Y7E	Y7F
Y80	Y81	Y82	Y83	Y84	Y85	Y86	Y87	Y88	Y89	Y8A	Y8B	Y8C	Y8D	Y8E	Y8F
Y90	Y91	Y92	Y93	Y94	Y95	Y96	Y97	Y98	Y99	Y9A	Y9B	Y9C	Y9D	Y9E	Y9F
YA0	YA1	YA2	YA3	YA4	YA5	YA6	YA7	YA8	YA9	YAA	YAB	YAC	YAD	YAE	YAF
YB0	YB1	YB2	YB3	YB4	YB5	YB6	YB7	YB8	YB9	YBA	YBB	YBC	YBD	YBE	YBF
YC0	YC1	YC2	YC3	YC4	YC5	YC6	YC7	YC8	YC9	YCA	YCB	YCC	YCD	YCE	YCF
YD0	YD1	YD2	YD3	YD4	YD5	YD6	YD7	YD8	YD9	YDA	YDB	YDC	YDD	YDE	YDF
YE0	YE1	YE2	YE3	YE4	YE5	YE6	YE7	YE8	YE9	YEA	YEB	YEC	YED	YEE	YEF
YF0	YF1	YF2	YF3	YF4	YF5	YF6	YF7	YF8	YF9	YFA	YFB	YFC	YFD	YFE	YFF

Figure 37.23 Luminance Data

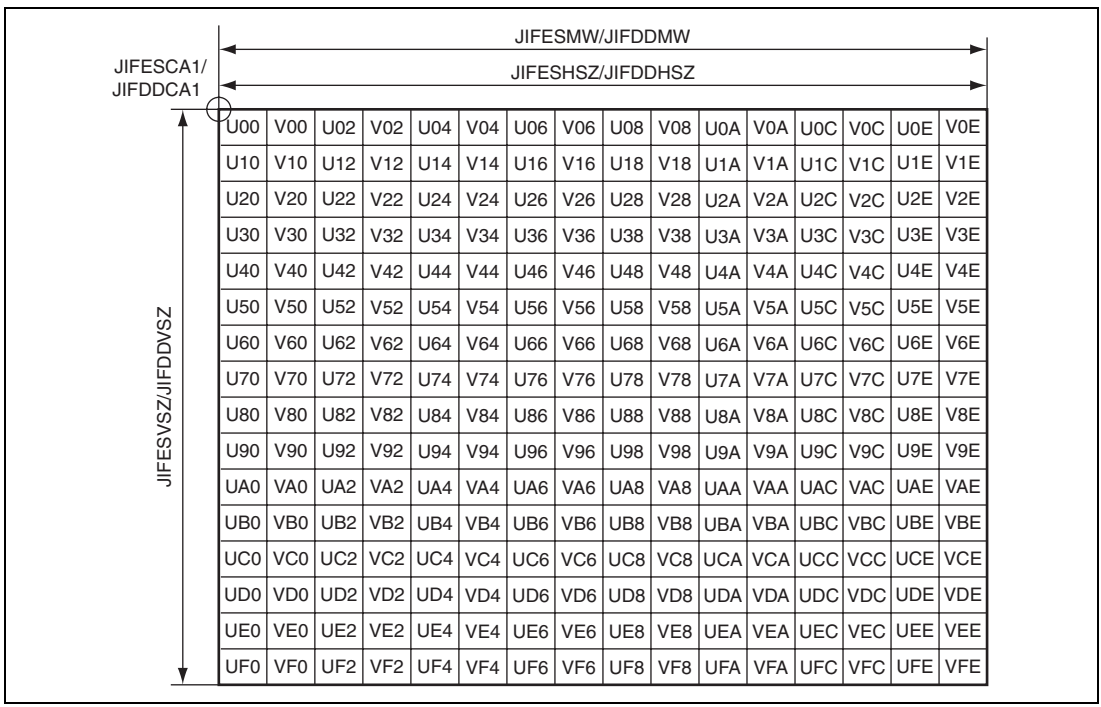
(b) Chrominance Component Data (Cb: U, Cr: V)

Specify JIFESCA1/JIFDDCA1 (JIFESCA2/JIFDDCA2) as the origin of the image, and transfer the data in the area specified by JIFESHSZ/JIFDDHSZ and JIFESVSZ/JIFDDVSZ.

- Register setting value:
  JIFESMW/JIFDDMW: H'0000 0010 (16)
  JIFESCA1/JIFDDCA1: H'0000 0100 (256)
  JIFESHSZ/JIFDDHSZ: H'0000 0010 (16)
  JIFESVSZ/JIFDDVSZ: H'0000 0010 (16)

- Data stored in buffer:
  H'0000 0010 = U00, V00, U02, V02
  H'0000 0104 = U04, V04, U06, V06
  H'0000 0108 = U08, V08, U0A, V0A
  H'0000 010C = U0C, V0C, U0E, V0E
  :
  H'0000 01F0 = UF0, VF0, UF2, VF2
  H'0000 01F4 = UF4, VF4, UF6, VF6
  H'0000 01F8 = UF8, VF8, UFA, VFA
  H'0000 01FC = UFC, VFC, UFE, VFE

- Transferring data to the JPU:
  Burst transfer is performed in 1-MCU units in the line direction. One line is 16 bytes.



U00	V00	U02	V02	U04	V04	U06	V06	U08	V08	U0A	V0A	U0C	V0C	U0E	V0E
U10	V10	U12	V12	U14	V14	U16	V16	U18	V18	U1A	V1A	U1C	V1C	U1E	V1E
U20	V20	U22	V22	U24	V24	U26	V26	U28	V28	U2A	V2A	U2C	V2C	U2E	V2E
U30	V30	U32	V32	U34	V34	U36	V36	U38	V38	U3A	V3A	U3C	V3C	U3E	V3E
U40	V40	U42	V42	U44	V44	U46	V46	U48	V48	U4A	V4A	U4C	V4C	U4E	V4E
U50	V50	U52	V52	U54	V54	U56	V56	U58	V58	U5A	V5A	U5C	V5C	U5E	V5E
U60	V60	U62	V62	U64	V64	U66	V66	U68	V68	U6A	V6A	U6C	V6C	U6E	V6E
U70	V70	U72	V72	U74	V74	U76	V76	U78	V78	U7A	V7A	U7C	V7C	U7E	V7E
U80	V80	U82	V82	U84	V84	U86	V86	U88	V88	U8A	V8A	U8C	V8C	U8E	V8E
U90	V90	U92	V92	U94	V94	U96	V96	U98	V98	U9A	V9A	U9C	V9C	U9E	V9E
UA0	VA0	UA2	VA2	UA4	VA4	UA6	VA6	UA8	VA8	UAA	VAA	UAC	VAC	UAE	VAE
UB0	VB0	UB2	VB2	UB4	VB4	UB6	VB6	UB8	VB8	UBA	VBA	UBC	VBC	UBE	VBE
UC0	VC0	UC2	VC2	UC4	VC4	UC6	VC6	UC8	VC8	UCA	VCA	UCC	VCC	UCE	VCE
UD0	VD0	UD2	VD2	UD4	VD4	UD6	VD6	UD8	VD8	UDA	VDA	UDC	VDC	UDE	VDE
UE0	VE0	UE2	VE2	UE4	VE4	UE6	VE6	UE8	VE8	UEA	VEA	UEC	VEC	UEE	VEE
UF0	VF0	UF2	VF2	UF4	VF4	UF6	VF6	UF8	VF8	UFA	VFA	UFC	VFC	UFE	VFE

Figure 37.24 Chrominance Data

37.3.4 Storing Coded Data

Storing the coded data in the buffer and transferring the coded data to or from the JPU are described below.

(1) Coded Data Sample: Coded length (SOI to EOI) is 592 bytes

Table 37.6 Coded Data Sample

ADDRESS	000102030405060708090A0B0C0D0E0F	101112131415161718191A1B1C1D1E1F
0000 00	FFD8FFDB008400100B0C0E0C0A100E0D	0E1211101318281A181616183123251D
0000 20	283A333D3C3933383740485C4E404457	453738506D51575F626768673E4D7179
0000 40	7064785C656763011112121815182F1A	1A2F634238426363636363636363636363
0000 60	6363636363636363636363636363636363	6363636363636363636363636363636363
0000 80	6363636363636363FFC000110801E002	8003012100021101031101FFC401A200
0000 A0	00010501010101010100000000000000	00010203040506070809 0A0B10000201
0000 C0	0303020403050504040000017D010203	00041105122131410613516107227114
0000 E0	328191A1082342B1C11552D1F0243362	7282090A161718191A25262728292A34
0001 00	35363738393A434445464748494A5354	55565758595A636465666768696A7374
0001 20	75767778797A838485868788898A9293	9495969798999AA2A3A4A5A6A7A8A9AA
0001 40	B2B3B4B5B6B7B8B9BAC2C3C4C5C6C7C8	C9CAD2D3D4D5D6D7D8D9DAE1E2E3E4E5
0001 60	E6E7E8E9EAF1F2F3F4F5F6F7F8F9FA01	00030101010101010101010000000000
0001 80	00010203040506070809 0A0B11000201	02040403040705040400010277000102
0001 A0	03110405213106124151076171132232	8108144291A1B1C109233352F0156272
0001 C0	D10A162434E125F11718191A26272829	2A35363738393A434445464748494A53
0001 E0	5455565758595A636465666768696A73	7475767778797A828384858687888 98A
0002 00	92939495969798999AA2A3A4A5A6A7A8	A9AAB2B3B4B5B6B7B8B9BAC2C3C4C5C6
0002 20	C7C8C9CAD2D3D4D5D6D7D8D9DAE2E3E4	E5E6E7E8E9EAF2F3F4F5F6F7F8F9FAFF
0002 40	DA000C03010002110311003F0000FFD9	XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

(2) Transferring Data at the End of Coded Data

[Encoding]

Insert H'FF up to a 16-byte address boundary.

CC: Coded data, FF-D9: EOI markers

- CC-CC-FF-D9 Insert H'FF up to a 16-byte boundary
- CC-FF-D9-FF Insert H'FF up to a 16-byte boundary
- FF-D9-FF-FF Insert H'FF up to a 16-byte boundary

[Decoding]

- The JPU continues reading until the JPU core detects the EOI marker. The JPU may read other data in addition to coded data.

(3) Reload Mode Disabled

A way of storing the data is shown in Table 37.6, Coded Data Sample.

- JIFEDA1/JIFDSA1: H'0000 0000

  As shown in Table 37.6, Coded Data Sample, the address of coded data is incremented from the start and then stored.

- JIFEDA1/JIFDSA1: H'0000 0004

```
ADDRESS    000102030405060708090A0B0C0D0E0F  101112131415161718191A1B1C1D1E1F
0000 00    XXXXXXXXFFD8FFDB008400100B0C0E0C  0A100E0D0E1211101318281A18161618
0000 20    3123251D283A333D3C3933383740485C  4E404457453738506D51575F62676867
           :
0002 20    C3C4C5C6C7C8C9CAD2D3D4D5D6D7D8D9  DAE2E3E4E5E6E7E8E9EAF2F3F4F5F6F7
0002 40    F8F9FAFFDA000C03010002110311003F  0000FFD9FFFFFFFFFFFFFFFFFFFFFF
```

- JIFEDA1/JIFDSA1: H'0000 0008

```
ADDRESS    000102030405060708090A0B0C0D0E0F  101112131415161718191A1B1C1D1E1F
0000 00    XXXXXXXXXXXXXXXXFFD8FFDB00840010  0B0C0E0C0A100E0D0E1211101318281A
0000 20    181616183123251D283A333D3C393338  3740485C4E404457453738506D51575F
           :
0002 20    B8B9BAC2C3C4C5C6C7C8C9CAD2D3D4D5  D6D7D8D9DAE2E3E4E5E6E7E8E9EAF2F3
0002 40    F4F5F6F7F8F9FAFFDA000C0301000211  0311003F0000FFD9FFFFFFFFFFFFFFFF
```

- JIFEDA1/JIFDSA1: H'0000 000C

```
ADDRESS    000102030405060708090A0B0C0D0E0F  101112131415161718191A1B1C1D1E1F
0000 00    XXXXXXXXXXXXXXXXXXXXXXXXFFD8FFDB  008400100B0C0E0C0A100E0D0E121110
0000 20    1318281A181616183123251D283A333D  3C3933383740485C4E40445745373850
           :
0002 20    B4B5B6B7B8B9BAC2C3C4C5C6C7C8C9CA  D2D3D4D5D6D7D8D9DAE2E3E4E5E6E7E8
0002 40    E9EAF2F3F4F5F6F7F8F9FAFFDA000C03  0100021103110 03F0000FFD9FFFFFFFF
```

(4) Reload Mode Enabled

- JIFEDA1/JIFDSA1: H'0000 0000
  JIFEDA2/JIFDSA2: H'0000 0100
  JIFEDRSZ/JIFDDRSZ: H'0000 0100

```
ADDRESS    000102030405060708090A0B0C0D0E0F  101112131415161718191A1B1C1D1E1F
(Reload buffer 1)
0000 00    FFD8FFDB008400100B0C0E0C0A100E0D  0E1211101318281A181616183123251D
           :
0000 E0    328191A1082342B1C11552D1F0243362  7282090A161718191A25262728292A34
(Reload buffer 2)
0001 00    35363738393A434445464748494A5354  55565758595A636465666768696A7374
           :
0001 E0    5455565758595A636465666768696A73  7475767778797A82838485868788898A
(Reload buffer 1)
0000 00    92939495969798999AA2A3A4A5A6A7A8  A9AAB2B3B4B5B6B7B8B9BAC2C3C4C5C6
0000 20    C7C8C9CAD2D3D4D5D6D7D8D9DAE2E3E4  E5E6E7E8E9EAF2F3F4F5F6F7F8F9FAFF
0000 40    DA000C0301000211031100 3F0000FFD9  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

37.3.5 Fractional Data Processing

The JPU can encode or decode the image in 4-pixel units, and sometimes it is needed to process the image that partially fills MCU (fractional data processing).

When the size of the image for coding by the JPU leaves one or more MCUs only partially filled, those MCUs are completed by filling the areas at the right and bottom relative to the origin of the image with black data (Y: H'00, CbCr: H'80). The image thus completed is then coded. When rotation is selected, MCUs are completed by filling the right and bottom areas relative to the post-rotation origin of the image with black data.

In decoding, if the image size information (parameter in the SOF0 marker) in the JPEG coded data leaves one or more MCUs only partially filled, the JPU only transfers the portions that contain actual data to the external buffer, e.g. in SDRAM. Thus, when the JPU decodes JPEG coded data that has been produced by the JPU coding data which did not completely fill all MCUs, the black data in the right and bottom areas of the incompletely filled MCUs are not transferred.

37.4 Interrupt Handling Flow

Interrupts for the JPU can be classified into interrupts for encoding/decoding and interrupts for transferring data. The way to clear an interrupt signal differs according to the interrupt sources.

37.4.1 Interrupts for Encoding and Decoding

Bits INS6, INS5, and INS3 in the interrupt status register JINTS are the interrupt sources for the JPEG core incorporated in the JPU. The interrupt signals which are asserted by these interrupt sources cannot be negated by clearing the corresponding interrupt status bits. Issue an interrupt signal clear command (set the JEND bit in JCCMD to B'1) to clear the interrupt signal.

(1) Encoding

- JPEG encoding process end

  When the INS6 bit in the interrupt status register JINTS is set to B'1, the JPEG encoding process is completed. After all of the coded data is transferred, the JPU ends the encoding process.

(2) Decoding

- JPEG decoding process end

  When the INS6 bit in the interrupt status register JINTS is set to B'1, the JPEG decoding process is completed. After all of the image data is transferred, JPU completes decoding.
- JPEG decoding error occurrence

  When the INS5 bit in the interrupt status register JINTS is set to B'1, the input JPEG coded data has an error and the JPU stops the decoding process. Read the error code (ERR bits in JCDERR) and identify the error source.
- Request for reading the image size and subsampling setting

  When the INS3 bit in the interrupt status register JINTS is set to B'1, JPEG coded data is input and information regarding its size and subsampling can be read. Since the JPEG decoding process is suspended, start the JPEG decoding process by setting the command to revoke the processing-stopped state after accessing the necessary registers. This interrupt occurs when the INT3 bit in the interrupt enable register JINTE is set to B'1.

37.4.2 Interrupts for Transferring Data

Bits INS14 to INS10 in the interrupt status register JINTS are the interrupt sources for transferring the image data and coded data. The interrupt signals asserted by these interrupt sources can be negated by clearing the corresponding interrupt status bits to 0.

(1) Encoding

- Occurs after data of one line buffer has been transferred (in line buffer mode)

  When the INS12 or INS11 bit in the interrupt status register JINTS is set to B'1, one line buffer of image data has been transferred; transfer the rest of the image data to the line buffer and resume transferring the data from the line buffer. This interrupt occurs when the INT12 or INT11 bit in the interrupt enable register JINTE is set to B'1.
- Occurs after the specified number of data have been reloaded

  When the INS13 bit in the interrupt status register JINTS is set to B'1, the coded data specified by JIFEDRSZ has been transferred. When the first interrupt occurs, transfer the coded data stored in the reload buffer to the other memory area. When the transfer is completed and the next reload interrupt occurs, transfer the coded data and set a command to restart the transfer to the reload buffer. The data is transferred to the first two reload buffers automatically and interrupts occur. These interrupts occur when the INT13 bit in the interrupt enable register JINTE is set to B'1.
- Occurs after all processes are completed

  When the INS10 bit in the interrupt status register JINTS is set to B'1, all of the coded data has been transferred and the JPU has completed encoding; transfer the coded data outside the JPU. When reload mode is set, transfer the rest of the coded data. When the total size of the coded data transferred to the reload buffer matches the number of coded data set by the JIFEDRSZ, bits INS10 and INS13 in JINTS are simultaneously set to B'1. This interrupt occurs when the INT10 bit in the interrupt enable register JINTE is set to B'1.

(2) Decoding

- Occurs after data of one line buffer has been transferred (in line buffer mode)

  When the INS12 or INS11 bit in the interrupt status register JINTS is set to B'1, one line buffer of image data has been transferred. When the first interrupt occurs, transfer the image data stored in the line buffer to the other memory area. When the transfer is completed and the next reload buffer interrupt occurs, transfer the image data and set a command to restart the transfer to the line buffer. The data is transferred to the first two reload buffers automatically and interrupts occur. These interrupts occur when the INT12 or INT11 bit in the interrupt enable register JINTE is set to B'1.
- Occurs after the specified number of data have been reloaded

  When the INS14 bit in the interrupt status register JINTS is set to B'1, the coded data in the reload buffer has been transferred; transfer the rest of the coded data to the reload buffer and set a command to resume transferring the data from the reload buffer. This interrupt occurs when the INT14 bit in the interrupt enable register JINTE is set to B'1.
- Occurs after all processes are completed

  When the INS10 bit in the interrupt status register JINTS is set to B'1, all of the image data has been transferred and the JPU has completed decoding; transfer the image data outside the JPU. This interrupt occurs when the INT10 bit in the interrupt enable register JINTE is set to B'1.

37.5 Bus Reset Processing

Issuing the bus reset command (setting the BRST bit in JCCMD to 1) causes a bus reset.

When the JPU is in operation, the bus reset command should not be issued. Registers other than below are initialized by a bus reset.

- JPEG code mode register (JCMOD)
- JPEG code command register (JCCMD)
- JPEG code quantization table number register (JCQTN)
- JPEG code Huffman table number register (JCHTN)
- JPEG code DRI upper register (JCDRIU)
- JPEG code DRI lower register (JCDRID)
- JPEG code vertical size upper register (JCVSZU)
- JPEG code vertical size lower register (JCVSZD)
- JPEG code horizontal size upper register (JCHSZU)
- JPEG code horizontal size lower register (JCHSZD)
- JPEG interrupt enable register (JINTE)
- JPEG interface control register (JIFCNT)

37.6 Software Reset Processing

Issuing the software reset command (setting the SRST bit in JCCMD to 1) causes a software reset.

After a software reset is specified, the SRST bit in JCCMD holds 1 until the current transfer in the JPU is completed. After the software reset processing, the following registers are initialized.

- JPEG code status register (JCSTS)
- JPEG code data count upper register (JCDTCU)
- JPEG code data count middle register (JCDTCM)
- JPEG code data count lower register (JCDTCD)
- JPEG interrupt status register (JINTS)
- JPEG code decode error register (JCDERR)
- JPEG code restart register (JCRST)
- JPEG interface control register (JIFCNT)

Section 38 2D Graphic Accelerator (2DG)

This section is covered by a non-disclosure agreement. Please contact a Renesas Electronics sales representative for details.

38.1 Features

- Extended 2D functions

  High-functional bold line drawing, antialias line drawing, and BITBLT type commands with ROP/alpha blending
- Upgraded control command functions

  Two command systems: GOSUB/RET and INT command, and upgraded WPR and TRAP command functions

Section 39 LCD Controller (LCDC)

The LCD controller (LCDC) reads display data from an external memory or receives display data from the blend engine unit (BEU). The LCDC uses the palette memory to determine the colors according to the settings and then sends the data to the LCD module. This LCDC allows connection of TFT LCD modules that support the RGB interface or the 80-Series CPU's bus interface (SYS interface).

39.1 Features

The LCDC has the following features.

- Supports TFT LCD modules (does not support LCD modules with the NTSC/PAL type or LVDS interface)
- LCD module interface
  RGB interface (8-/9-/12-/16-/18-/24-bit bus width)
  80-Series CPU's bus interface (SYS interface, 8-/9-/12-/16-/18-/24-bit bus width)
- SYS interface supports the input/output mode for VSYNC
- Supports 8-/12-/16-/18-/24-bpp display image data formats
- Display image data is read in continuous or one-shot mode: continuous mode where display image data is continuously read according to the refresh rate of the LCD module and one-shot mode where display image data is read at intervals of the frame rate.
- Display image data is read in full or partial screen mode: full screen mode where the size of the display image data to be read depends on the panel size of the LCD module and partial screen mode where the size of the display image data to be read depends on the size of the screen to be updated.
- Display image data can be written back to the external memory
- Each of the RGB colors can be corrected by the 256-entry, 24-bit-input/output internal color palette memory
- Supports inversion of output signals to agree with the LCD module's signal polarity
- Interrupts can be generated every frame or user-specified line
- YCbCr signals are read and converted into RGB signals for output to the LCD module
- Supports YCbCr output function mode

Table 39.1 shows the LCDC functions.

Table 39.1 LCDC Functions

		Function	Remakes
Input data format	12 bpp	RGB 444	
	16 bpp	RGB 565	
	18 bpp	RGB 666	
		BGR 666	
	24 bpp	RGB 888	
		BGR 888	
	YCbCr	YCbCr 4:2:0, 4:2:2, 4:4:4	
Output data format	RGB interface	RGB8	3 cycle/pixel
		RGB9	2 cycle/pixel
		RGB12a	2 cycle/pixel
		RGB12b	1 cycle/pixel
		RGB16	1 cycle/pixel
		RGB18	1 cycle/pixel
		RGB24	1 cycle/pixel
	SYS interface	SYS8a	3 cycle/pixel
		SYS8b	3 cycle/pixel
		SYS8c	3 cycle/pixel
		SYS8d	2 cycle/pixel
		SYS9	2 cycle/pixel
		SYS12	2 cycle/pixel
		SYS16a	1 cycle/pixel
		SYS16b	2 cycle/pixel
		SYS16c	2 cycle/pixel
		SYS18	1 cycle/pixel
		SYS24	1 cycle/pixel
	Display data write-back	WB8a	• Packed format available • Write-back operation in units of 32 bits • Byte or word swap
		WB8d	
		WB9	
		WB16	
		WB18	
		WB24	
	YCbCr output	YCbCr 4:2:2	

		Function	Remakes
LCD driver interface	RGB interface	Interface with HSYNC and VSYNC • Polarity inversion • Output pulse width and position setting	
	SYS interface	80-Series bus interface • Support of VSYNC input/output	
Dot clock	Source clock	Bus clock, peripheral clock, external clock	
	Division ratio	n/m m = 60, 54, 48, 42 $1 \leq n \leq m/3, m/2$	
Interrupt	User setting	Interrupt generated when reading of specified lines of data is completed	
	Frame	Interrupt generated when the first pixel data of a frame starts to be output	
		Interrupt generated when output of the last pixel data of a frame is completed	
	VRAM read	Interrupt generated when access to a frame of data in VRAM is completed	
		Interrupt generated when access to a line of data in VRAM is completed	
	VSYNC	Interrupt generated when VSYNC is asserted	
		Interrupt generated when VSYNC is negated	
Display image	Image data read	Image data read depending on the refresh rate of LCD module	
		Image data read depending on the frame rate of display image	
	Display image size	Full screen	
		Only the specified area is updated.	
	Image data processing	Each color of R, G, and B is converted using the color palette • 256 entries • 24-bit input/output	
Format conversion	YCbCr to RGB	YCbCr data is converted into RGB for output.	

Figure 39.1 is a block diagram of the LCDC.

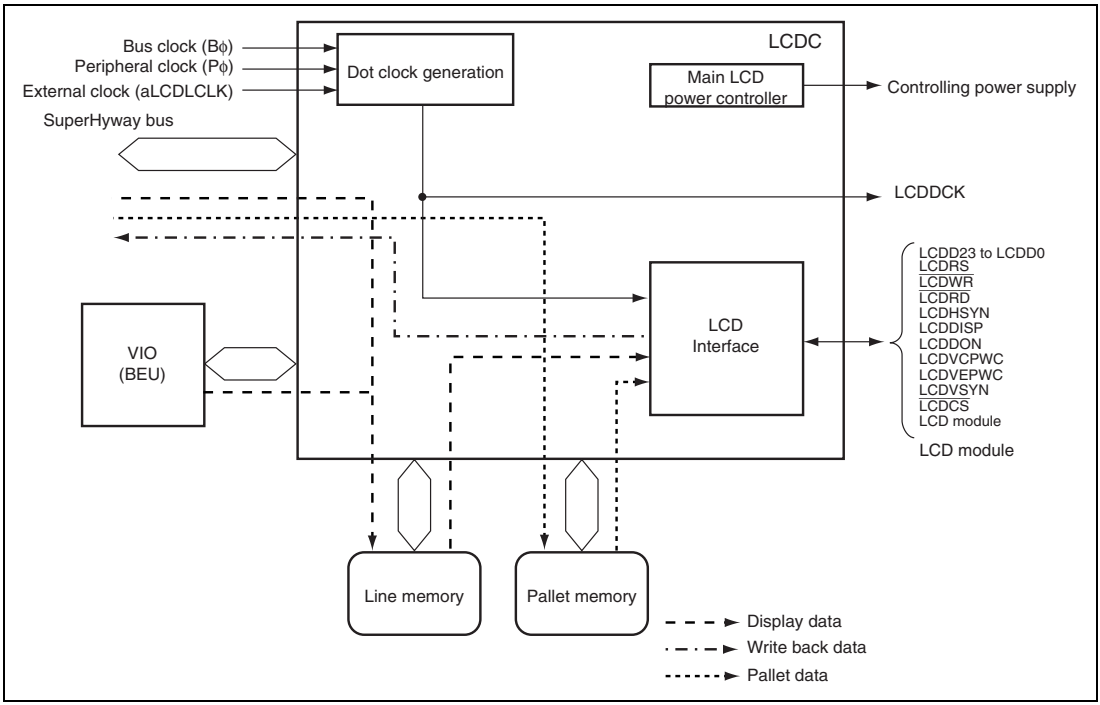


Figure 39.1 Block Diagram of LCDC

39.2 Input/Output Pins

Table 39.2 shows the pin configuration of the LCDC.

Table 39.2 Pin Configuration

Name	Function	I/O	Description
LCDDON	Display on signal	Output	Display start signal (DON)
LCDVCPWC	Power control	Output	Main LCD module power control (V_{CC})
LCDVEPWC	Power control	Output	Main LCD module power control (V_{EE})
LCDDCK/ $\overline{\text{LCDWR}}$	Dot clock/ write strobe	Output	Dot clock signal (RGB interface)/write strobe signal (SYS interface)
LCDVSYN	Vertical sync signal	Output/ I/O	Vertical sync signal (VSYNC) for main LCD (output for RGB interface, I/O for SYS interface, or output for YCbCr output mode)
LCDHSYN/ $\overline{\text{LCDCS}}$	Horizontal sync signal/chip select	Output	Horizontal sync signal (RGB interface and YCbCr output mode)/chip select signal for main LCD (SYS interface)
LCDDISP/ LCDRS	Display enable/ Register select	Output	Display enable signal (RGB interface)/register select signal (SYS interface)
$\overline{\text{LCDRD}}$	Read strobe	Output	Read strobe signal (SYS interface)
LCDD23 to LCDD0	LCD data bus	Output/ I/O	LCD panel data (output for RGB interface, I/O for SYS interface, or output for YCbCr output mode)
LCDLCLK	Input clock	Input	LCD source clock (external input)

39.3 Register Descriptions

Table 39.3 shows the register configuration of the LCDC. Table 39.4 shows the states of the registers in each mode.

Most of the LCDC registers have a double-buffered mechanism (sides A and B). The LCDC switches sides for use of the registers. A register with two sides has a mirror address to access the side not currently being used.

Table 39.3 Register Configuration

			Address			
Register Name	**Symbol**	**R/W**	**Address (Side A)**	**Address (Side B)**	**Mirror Address**	**Access Size**
LCDC palette data registers 00 to FF	LDPR00 to LDPRFF	R/W	H'FE94 0000 to H'FE94 03FC	—	—	32
Main LCD dot clock pattern register 1	MLDDCKPAT1R	R/W	H'FE94 0400	—	—	32
Main LCD dot clock pattern register 2	MLDDCKPAT2R	R/W	H'FE94 0404	—	—	32
LCDC dot clock register	LDDCKR	R/W	H'FE94 0410	—	—	32
Dot clock stop register	LDDCKSTPR	R/W	H'FE94 0414	—	—	32
Main LCD module type register 1	MLDMT1R	R/W	H'FE94 0418	H'FE94 1418	H'FE94 2418	32
Main LCD module type register 2	MLDMT2R	R/W	H'FE94 041C	H'FE94 141C	H'FE94 241C	32
Main LCD module type register 3	MLDMT3R	R/W	H'FE94 0420	H'FE94 1420	H'FE94 2420	32
Main LCD data format register	MLDDFR	R/W	H'FE94 0424	H'FE94 1424	H'FE94 2424	32
Main LCD scan mode register 1	MLDSM1R	R/W	H'FE94 0428	H'FE94 1428	H'FE94 2428	32
Main LCD scan mode register 2	MLDSM2R	R/W	H'FE94 042C	—	—	32
Main LCD display data read start address register 1	MLDSA1R	R/W	H'FE94 0430	H'FE94 1430	H'FE94 2430	32
Main LCD display data read start address register 2	MLDSA2R	R/W	H'FE94 0434	H'FE94 1434	H'FE94 2434	32
Main LCD display data storing memory line size register	MLDMLSR	R/W	H'FE94 0438	H'FE94 1438	H'FE94 2438	32
Main LCD horizontal character number register	MLDHCNR	R/W	H'FE94 0448	H'FE94 1448	H'FE94 2448	32
Main LCD horizontal sync signal register	MLDHSYNR	R/W	H'FE94 044C	H'FE94 144C	H'FE94 244C	32
Main LCD vertical line number register	MLDVLNR	R/W	H'FE94 0450	H'FE94 1450	H'FE94 2450	32
Main LCD vertical sync signal register	MLDVSYNR	R/W	H'FE94 0454	H'FE94 1454	H'FE94 2454	32
Main LCD horizontal partial display register	MLDHPDR	R/W	H'FE94 0458	H'FE94 1458	H'FE94 2458	32
Main LCD vertical partial display register	MLDVPDR	R/W	H'FE94 045C	H'FE94 145C	H'FE94 245C	32

Register Name	Symbol	R/W	Address (Side A)	Address (Side B)	Mirror Address	Access Size
			Address			
Main LCD power management register	MLDPMR	R/W	H'FE94 0460	—	—	32
LCDC palette control register	LDPALCR	R/W	H'FE94 0464	—	—	32
LCDC interrupt register	LDINTR	R/W	H'FE94 0468	—	—	32
LCDC status register	LDSR	R	H'FE94 046C	—	—	32
LCDC control register 1	LDCNT1R	R/W	H'FE94 0470	—	—	32
LCDC control register 2	LDCNT2R	R/W	H'FE94 0474	—	—	32
LCDC register side change control register	LDRCNTR	R/W	H'FE94 0478	—	—	32
LCDC input image data swap register	LDDDSR	R/W	H'FE94 047C	—	—	32
LCDC register side forcible select register	LDRCR	R/W	H'FE94 0484	—	—	32
LCDC driver write data register 0	LDDWD0R	R/W	H'FE94 0800	—	—	32
LCDC driver write data register 1	LDDWD1R	R/W	H'FE94 0804	—	—	32
LCDC driver write data register 2	LDDWD2R	R/W	H'FE94 0808	—	—	32
LCDC driver write data register 3	LDDWD3R	R/W	H'FE94 080C	—	—	32
LCDC driver write data register 4	LDDWD4R	R/W	H'FE94 0810	—	—	32
LCDC driver write data register 5	LDDWD5R	R/W	H'FE94 0814	—	—	32
LCDC driver write data register 6	LDDWD6R	R/W	H'FE94 0818	—	—	32
LCDC driver write data register 7	LDDWD7R	R/W	H'FE94 081C	—	—	32
LCDC driver write data register 8	LDDWD8R	R/W	H'FE94 0820	—	—	32
LCDC driver write data register 9	LDDWD9R	R/W	H'FE94 0824	—	—	32
LCDC driver write data register A	LDDWDAR	R/W	H'FE94 0828	—	—	32
LCDC driver write data register B	LDDWDBR	R/W	H'FE94 082C	—	—	32
LCDC driver write data register C	LDDWDCR	R/W	H'FE94 0830	—	—	32
LCDC driver write data register D	LDDWDDR	R/W	H'FE94 0834	—	—	32
LCDC driver write data register E	LDDWDER	R/W	H'FE94 0838	—	—	32
LCDC driver write data register F	LDDWDFR	R/W	H'FE94 083C	—	—	32
LCDC driver read data register	LDDRDR	R/W	H'FE94 0840	—	—	32
LCDC driver write access register	LDDWAR	R/W	H'FE94 0900	—	—	32
LCDC driver read access register	LDDRAR	R/W	H'FE94 0904	—	—	32

Table 39.4 Register States in Each Mode

Register Symbol	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
LDPR00 to LDPRFF	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDDCKPAT1R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDDCKPAT2R	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDDCKR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDDCKSTPR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDMT1R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDMT2R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDMT3R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDDFR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDSM1R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDSM2R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDSA1R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDSA2R	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDMLSR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDHCNR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDHSYNR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDVLNR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDVSYNR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDHPDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDVPDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
MLDPMR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDPALCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDINTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDSR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDCNT1R	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDCNT2R	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDRCNTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDDDSR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDRCR	Initialized	Initialized	Retained	Retained	Initialized	Retained

Register Symbol	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
LDDWD0R to LDDWDFR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDDRDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDDWAR	Initialized	Initialized	Retained	Retained	Initialized	Retained
LDDRAR	Initialized	Initialized	Retained	Retained	Initialized	Retained

39.3.1 LCDC Palette Data Registers 00 to FF (LDPR00 to LDPRFF)

LDPRnn specifies palette data in the palette memory. To access the palette memory, use these registers. Each 32-bit register has eight bits assigned to each of the R, G, and B colors. For details on the color palette specification and settings, see section 39.4.2, Color Palette Specification.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PALDDnn[32:16]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PALDDnn[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	PALDnn[23:0]	H'00 0000	R/W	Palette Data

Note: nn = 00 to FF

39.3.2 Main LCD Dot Clock Pattern Register 1 (MLDDCKPAT1R)

The LCDC generates the dot clock by dividing the source clock selected through LDDCKR. MLDDCKPAT1R and MLDDCKPAT2R in combination specify the dot clock output pattern. For details, see section 39.4.7, Dot Clock Settings.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—		DCKPAT1[59:48]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCKPAT1[47:32]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 0	DCKPAT1 [59:32]	H'000 0000	R/W	Dot Clock Pattern 1 [59:32] Specifies the upper 28 bits of the output pattern of the dot clock which is obtained from the divided source clock. Up to a 60-bit pattern can be specified in combination with these bits and the bits in MLDDCKPAT2R. When the value specified in LDDCKR.MDCDR is smaller than 60, unnecessary bits at the MSB side of the dot clock pattern are ignored.

39.3.3 Main LCD Dot Clock Pattern Register 2 (MLDDCKPAT2R)

MLDDCKPAT2R specifies the lower 32 bits of a dot clock output pattern for the main LCD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCKPAT1[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCKPAT1[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DCKPAT1 [31:0]	H'0000 0000	R/W	Dot Clock Pattern 1 [31:0] Specifies the lower 32 bits of the output pattern of the dot clock which is obtained from the divided source clock. A maximum of 60-bit pattern can be specified in combination with these bits and bits in MLDDCKPAT1R.

39.3.4 LCDC Dot Clock Register (LDDCKR)

LDDCKR selects the input clock and specifies the division ratio of the dot clock.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ICKSEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	MOSEL	MDCDR[5:0]					
Initial value:	0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

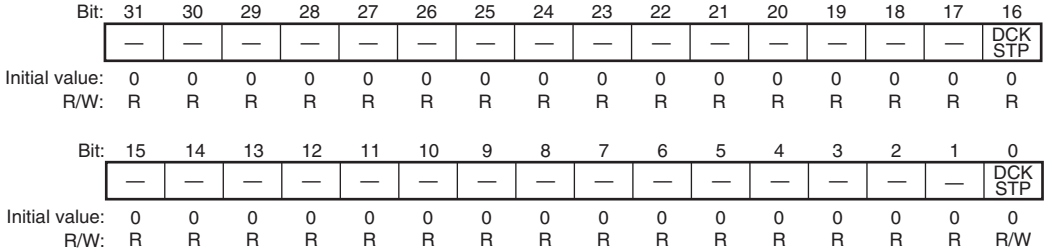
Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
17, 16	ICKSEL [1:0]	00	R/W	Input Clock Select* 00: Selects the bus clock (Bϕ) 01: Selects the peripheral clock (Bϕ) 10: Selects the external clock (LCDLCLK) 11: Setting prohibited
15, 14	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 10	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
9 to 7	—	All 0		Reserved These bits are always read as 0. The write value should always be 0.
6	MOSEL	0	R/W	Main LCD Output Clock Select Selects the division ratio of the dot clock (main LCD operating clock) from the 1/1 and value specified by bits in this register. 0: A division ratio specified by bits in this register 1: A division ratio of 1/1
5 to 0	MDCDR [5:0]	111100	R/W	Clock Division Ratio These bits specify the denominator (m) when the division ratio of the source clock is denoted as n/m. Only 60, 54, 48, or 42 can be specified for the denominator (m). The value specified in these bits should match the number of the clock pattern bits specified in MLDDCKPAT1R and MLDDCKPAT2R.

Note: * When the bus clock and peripheral clock are set to the same frequency through the CPG, select the bus clock (ICKSEL = B'00).

39.3.5 Dot Clock Stop Register (LDDCKSTPR)

LDDCKSTPR stops or restarts the LCDC dot clock. When temporarily stopping the dot clock and then restarting it, set the DCKSTP bit to 1, check that the DCKSTS bit becomes 1, make necessary settings, and then restart the dot clock.



Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	DCKSTS	0	R	Dot Clock Status This bit is set to 1 while the LCDC dot clock stops. 0: The dot clock is being supplied 1: The dot clock has stopped
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DCKSTP	0	R/W	Dot Clock Stop/Restart Control Stops or restarts the LCDC dot clock. 0: Supplies the dot clock 1: Stops the dot clock

39.3.6 Main LCD Module Type Register 1 (MLDMT1R)

The interface mode, the polarity of control signals, the data bus width, and the number of access cycles for the LCD panel can be specified by main LCD module type registers 1 to 3 in the LCDC.

MLDMT1R selects the interface from the RGB and SYS interfaces and controls the signal polarity and output.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VPOL	HPOL	DW POL	DI POL	DA POL	—	—	—	—	—	—	HS CNT	DW CNT
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	YM	IFM	—	—	—	—	—	—	—	—	MIFTYP[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	VPOL	0	R/W	Vertical Sync Signal Polarity Select In VSYNC output mode (when the VSEL bit in MLDMT2R is cleared to 0), this bit selects the polarity of the VSYNC signal for the main LCD when the RGB or SYS interface is used. In VSYNC input mode (when the VSEL bit in MLDMT2R is set to 1), this bit selects whether or not the polarity of the input VSYNC signal is toggled. Note that the polarity of the VSYNC signal is active high. In VSYNC output mode 0: Output VSYNC is active high 1: Output VSYNC is active low In VSYNC input mode 0: Polarity is not toggled 1: Polarity is toggled
27	HPOL	0	R/W	Horizontal Sync Signal Polarity Select Selects the polarity of the HSYNC signal when the RGB interface of the main LCD is used. When the RGB interface mode is selected by the IFM bit, this bit is valid. 0: HSYNC signal is active high 1: HSYNC signal is active low

Bit	Bit Name	Initial Value	R/W	Description
26	DWPOL	1	R/W	Dot Clock Polarity Select Selects the polarity of the dot clock signal when the RGB interface of the main LCD is used. 0: Data latched at the falling edge of the dot clock 1: Data latched at the rising edge of the dot clock
25	DIPOL	0	R/W	Display Enable Polarity Select Selects the polarity of the display enable signal when the RGB interface of the main LCD is used. When the RGB interface mode is selected by the IFM bit, this bit is valid. 0: Display enable signal is active high 1: Display enable signal is active low
24	DAPOL	0	R/W	Display Data Polarity Select Selects the polarity of display data when the RGB or SYS interface of the main LCD is used. The polarity is only for display data and does not affect commands or instructions for connected LCDs. 0: Display data is active high 1: Display data is active low
23 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	HSCNT	0	R/W	HSYNC Signal Output Control Controls the HSYNC output signal during vertical blanking. When the RGB interface mode is selected by the IFM bit, this bit is valid. 0: Outputs HSYNC during vertical blanking 1: Does not outputs HSYNC during vertical blanking
16	DWCNT	0	R/W	Dot Clock Control Controls the dot clock signal output during horizontal or vertical blanking when the RGB interface mode is used. 0: Outputs the dot clock signal during horizontal/vertical blanking 1: Does not output the dot clock signal during horizontal/vertical blanking
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
13	YM	0	R/W	YCbCr Output Mode Set Selects the interface mode for the YCbCr output mode. 0: Normal mode 1: YCbCr output mode
12	IFM	0	R/W	Interface Mode Set Selects the interface mode for the connected main LCD. 0: RGB interface mode 1: SYS interface mode
11 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	MIFTYP [3:0]	1010	R/W	Main LCD Module Set These bits select the type of the main LCD according to the IFM bit. See table 39.5 and Figure 39.14.

Table 39.5 Main LCD Module Settings

- IFM = B'0

MIFTYP[3:0]	Description
0000	RGB8 (24 bpp, three-time transfer, 8:8:8)
0001	Reserved
0010	Reserved
0011	Reserved
0100	RGB9 (18 bpp, two-time transfer, 9:9)
0101	RGB12a (24 bpp, two-time transfer, 12:12)
0110	RGB12b (12 bpp, one-time transfer)
0111	RGB16 (16 bpp, one-time transfer)
1000	Reserved
1001	Reserved
1010	RGB18 (18 bpp, one-time transfer)
1011	RGB24 (24 bpp, one-time transfer)
1100 to 1111	Reserved

- IFM = B'1

MIFTYP[3:0]	Description
0000	SYS8a (24 bpp, three-time transfer, 8:8:8)
0001	SYS8b (18 bpp, three-time transfer, 8:8:2)
0010	SYS8c (18 bpp, three-time transfer, 2:8:8)
0011	SYS8d (16 bpp, two-time transfer, 8:8)
0100	SYS9 (18 bpp, two-time transfer, 9:9)
0101	SYS12 (24 bpp, two-time transfer, 12:12)
0110	Reserved
0111	SYS16a (16 bpp, one-time transfer)
1000	SYS16b (18 bpp, two-time transfer, 16:2)
1001	SYS16c (18 bpp, two-time transfer, 2:16)
1010	SYS18 (18 bpp, one-time transfer)
1011	SYS24 (24 bpp, one-time transfer)
1100 to 1111	Reserved

39.3.7 Main LCD Module Type Register 2 (MLDMT2R)

MLDMT2R controls the write strobe signal and indicates access status when the SYS interface of the main LCD is used.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	CSUP[2:0]			RSV	VSEL	WCSC[7:0]							
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WCEC[7:0]								WCLW[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
28 to 26	CSUP[2:0]	001	R/W	Chip Select Setup Time Set These bits specify the setup time (the period from $\overline{\text{LCDCS}}$ assertion to $\overline{\text{LCDWR}}$ assertion) of the chip select signal for display data transfer via the SYS interface on a dot clock basis.
25	RSV	0	R/W	VSYNC Mode Register Select Polarity Set Sets the polarity of the register select signal when display data is transferred via the SYS interface. 0: Register select signal is active low 1: Register select signal is active high
24	VSEL	0	R/W	VSYNC I/O Mode Select Selects the input or output mode of the VSYNC signal of the main LCD for the SYS interface. This bit is writable when the DO bit in LDCNT2R is cleared to 0. 0: VSYNC output mode (LCDC outputs the VSYNC) 1: VSYNC input mode (LCDC outputs display data or commands according to the VSYNC input from an external device)
23 to 16	WCSC[7:0]	H'00	R/W	Setup Time Set These bits set the setup time (the period from $\overline{\text{LCDCS}}$ assertion to $\overline{\text{LCDWR}}$ assertion) of the write strobe signal ($\overline{\text{LCDWR}}$) for the SYS interface command transmission on a bus clock basis.
15 to 8	WCEC[7:0]	H'00	R/W	Write Cycle Time Set These bits set the time of a chip select ($\overline{\text{LCDCS}}$) cycle for the SYS interface command transmission on a bus clock basis.
7 to 0	WCLW[7:0]	H'00	R/W	Low Pulse Width Set These bits specify the low level width of the write strobe signal for the SYS interface command transmission on a bus clock basis.

39.3.8 Main LCD Module Type Register 3 (MLDMT3R)

MLDMT3R controls the read strobe signal via the SYS interface of the main LCD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RDLC[5:0]						RCSC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCEC[7:0]								RCLW[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 24	RDLC[5:0]	000000	R/W	Read Data Latch Set These bits specify the timing of latching data read via the SYS interface command transmission on a bus clock basis.
23 to 16	RCSC[7:0]	H'00	R/W	Setup Time Set These bits specify the setup time of the read strobe signal for the SYS interface command transmission on a bus clock basis.
15 to 8	RCEC[7:0]	H'00	R/W	Read Cycle Time Set These bits specify the time of a chip select ($\overline{\text{LCDCS}}$) cycle for the SYS interface command transmission on a bus clock basis.
7 to 0	RCLW[7:0]	H'00	R/W	Low Pulse Width Set These bits specify the low level width of the read strobe signal for the SYS interface command transmission on a bus clock basis.

39.3.9 Main LCD Data Format Register (MLDDFR)

MLDDFR sets the type of image data input to the main LCD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CF	CC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	YF[1:0]		—	—	—	PFK[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	CF	0	R/W	YCbCr → RGB Conversion Formula Select Selects the formula for YCbCr → RGB conversion. 0: Selects RGB extension 1: Selects RGB compression
16	CC	0	R/W	YCbCr → RGB Conversion Select Enables or disables the YCbCr → RGB conversion. When the data format of display data in memory is YCbCr, set this bit. 0: Enables RGB conversion 1: Disables RGB conversion
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	YF[1:0]	00	R/W	Input Image YCbCr Data Packed Form These bits select the packed format when the input image data format is YCbCr. When the CC bit in MLDDFR is set to 1, this bit is valid. 00: 4:2:0 YCbCr 01: 4:2:2 YCbCr 10: 4:4:4 YCbCr 11: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	PKF[4:0]	00000	R/W	Input Image RGB Data Packed Form These bits select the packed format when the input image data format is RGB. When the CC bit in MLDDFR is cleared to 0, this bit is valid. Tables 39.6 and 39.7 list the packed formats for the YCbCr and RGB, respectively.

Note: When the LCDC operates together with the VIO (BEU), set the PKF bits to B'00000.

Table 39.6 YCbCr Packed Format

			Bits			
YF[1:0]	**YCbCr**		**31 to 24**	**23 to 16**	**15 to 8**	**7 to 0**
B'00	4:2:0	Y data	Y0	Y1	Y2	Y3
B'00	4:2:0	C data	Cb0	Cr0	Cb2	Cr2
B'01	4:2:2	Y data	Y0	Y1	Y2	Y3
B'01	4:2:2	C data	Cb0	Cr0	Cb2	Cr2
B'10	4:4:4	Y data	Y0	Y1	Y2	Y3
B'10	4:4:4	C data	Cb0	Cr0	Cb1	Cr1
B'11	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Table 39.7 RGB Packed Format

PKF [4:0]	Bit Rate [bpp]	Phase	Bit																													
			31 to 24								23 to 16								15 to 8								7 to 0							
B'00000	24	-	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0
B'00001	24	-	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	0	0
B'00010	24	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3
B'00011	16	-	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1
B'00111	18	-	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0
B'01000	12	-	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1
B'01001	18	0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3
B'01010	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2
		2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3
B'01011	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3
B'01100	24	-	0	0	0	0	0	0	0	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0
B'01101	16	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0
B'01110 to B'11111	Reserved																																	

39.3.10 Main LCD Scan Mode Register 1 (MLDSM1R)

MLDSM1R selects the display data size from full or partial and selects the mode for reading display data from continuous or one-shot. For details of the settings, see section 39.4.4, One-Shot Mode, and section 39.4.5, Partial Screen Mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PRD	—	—	—	—	—	—	—	OS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	PRD	0	R/W	Partial Size Read Set Selects the size of display data read from external memory. 0: Full screen size (panel size) of data is read 1: Partial screen size of data is read
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OS	0	R/W	One-Shot Read Set Selects the mode for reading display data from external memory. 0: Continuous mode (display data is read according to the refresh rate of the panel) 1: One-shot mode (display data is read using the OSTRG bit in MLDSM2R as a trigger)

39.3.11 Main LCD Scan Mode Register 2 (MLDSM2R)

MLDSM2R sets the one-shot trigger of the main LCD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OSTRG
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTRG	0	R/W	One-Shot Trigger When the one-shot mode is selected for reading display data, setting this bit to 1 starts reading of one frame of data from external memory. This bit is cleared when the display operation is started. 0: — 1: One frame of data read

39.3.12 Main LCD Display Data Read Start Address Register 1 (MLDSA1R)

MLDSA1R sets the start address of the main LCD display data in memory. When the data format of display data is YCbCr, these bits set the start address of the luminance signal. If the BEU is specified as the source for data acquisition, there is no need to set this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:3]													—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	SA[31:3]	H'0000 0000	R/W	Display Data Start Address These bits set the start address of the main LCD display data in memory. When the data format of display data is YCbCr, these bits set the start address of the luminance signal.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

39.3.13 Main LCD Display Data Read Start Address Register 2 (MLDSA2R)

MLDSA2R sets the start address of the color difference signal of the main LCD display data in memory. When the data format of display data is YCbCr, this register is valid. If the BEU is specified as the source for data acquisition, there is no need to set this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:3]													—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	SA[31:3]	H'0000 0000	R/W	Display Data Start Address These bits specify the start address of the color difference signal of the main LCD display data in memory. When the data format of display data is YCbCr, these bits are valid.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

39.3.14 Main LCD Display Data Storing Memory Line Size Register (MLDMLSR)

MLDMLSR specifies the amount of one line of main LCD display data in memory. A vertical line number is incremented when the specified size of memory is read. When the data format of display data is YCbCr 4:4:4, twice the size set is used in memory. If the BEU is specified as the source for data acquisition, there is no need to set this register.

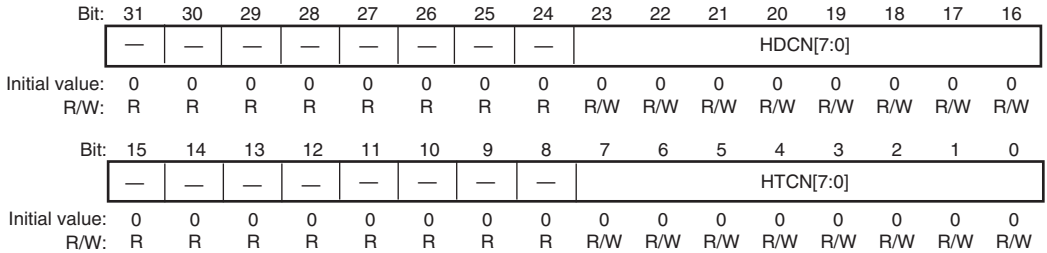
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LS[15:3]													—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 3	LS[15:3]	H'0000	R/W	Line Size These bits set the amount of one line of main LCD display data in memory. When the data format of display data is YCbCr 4:4:4, twice the size set is used in memory.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

39.3.15 Main LCD Horizontal Character Number Register (MLDHCNR)

MLDHCNR specifies the horizontal size of the main LCD and the scan size including the horizontal blanking. This register must be set in units of characters (eight dots). When the LCDC operates together with the BEU, the horizontal display size of the LCDC is the same as the horizontal output size of the BEU. In this case, set the HDCN bits to the same size as the horizontal size of the parent display specified in BSSZRn.BHSS (n: parent display) in the BEU.



Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	HDCN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HTCN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	HDCN[7:0]	H'1E	R/W	Horizontal Display Character Count These bits specify the number of horizontal display characters (one character = eight dots). For example, when an LCD module which has 240 pixels horizontally is used, set these bits to H'1E (= 240/8 = 30).
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	HTCN[7:0]	H'21	R/W	Horizontal Character Count These bits specify the number of horizontal characters (one characters = eight dots). The horizontal blanking is three characters (24 dots) at a minimum.

Notes: 1. The specified size of horizontal display data must be equal to 4 Kbytes per line or less. For example, when the amount of pixel data is specified to 32 bits by setting the PKF bits in MLDDFR to B'00000, the maximum number of horizontal pixels is 1024 (128 characters).

2. Specify the HDCN and HTCN bits so that the following expression is satisfied.

$HTCN \geq HDCN + 3$

39.3.16 Main LCD Horizontal Sync Signal Register (MLDHSYNR)

MLDHSYNR specifies the timing and size of the horizontal sync signal of the main LCD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	HSYNW[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HSYNP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	HSYNW [3:0]	H'1	R/W	Horizontal Sync Signal Pulse Width These bits specify the pulse width of the horizontal sync signal in units of characters (one character = eight dots).
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	HSYNP [7:0]	H'20	R/W	Horizontal Sync Signal Output Position These bits specify the output position of the horizontal sync signal in units of characters (one character = eight dots).

Note: Specify the HSYNW and HSYNP bits so that the following expressions are satisfied.
HTCN ≥ HSYNP + HSYNW
HSYNP ≥ HDCN + 1

39.3.17 Main LCD Vertical Line Number Register (MLDVLNR)

MLDVLNR specifies the vertical size excluding or including blanking of the main LCD. When the LCDC operates together with the BEU, the vertical display size of the LCDC is the same as the vertical output size of the BEU. In this case, set the VDLN bits to the same size as the vertical size of the parent display specified in BSSZRn.BVSS (n: parent display) in the BEU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	VDLN[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VTLN[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	VDLN[10:0]	H'140	R/W	Vertical Display Line Count These bits specify the number of display lines in the vertical direction in units of lines. For example, when an LCD having 320 vertical lines is used, set these bits to H'140 = 320.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	VTLN[10:0]	H'142	R/W	Vertical Line Total Count These bits specify the total number of lines in the vertical direction in units of lines. The minimum total number of lines is 3.

Notes: 1. Specify the VTLN and VDLN bits so that the following expression is satisfied.
$VTLN \geq VDLN + 1$

2. For VSYNC input mode, the interval of the VSYNC input must be shorter than one frame specified by the VTLN bits. The vertical blanking during which commands can be issued is defined from the completion of display operation to the end of the frame specified by the VTLN bits.

39.3.18 Main LCD Vertical Sync Signal Register (MLDVSYNR)

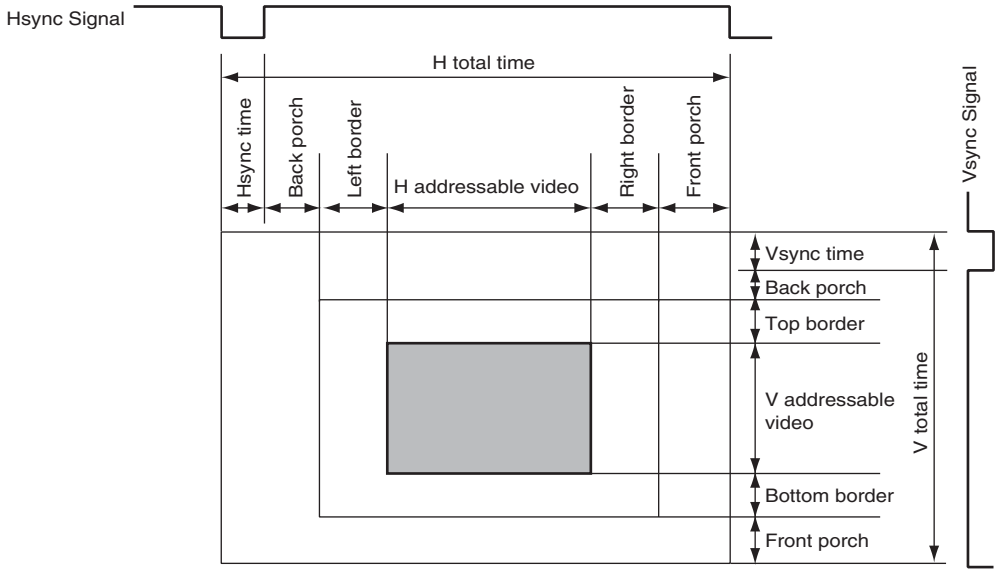
MLDHSYNR specifies the timing and width of the vertical sync signal of the main LCD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	VSYNW[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VSYNP[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	VSYNW [3:0]	H'1	R/W	Vertical Sync Signal Width These bits specify the pulse width of vertical sync signal assertion in units of lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	VSYNP [10:0]	H'141	R/W	Vertical Sync Signal Output Position These bits specify the output position of the vertical sync signal in units of lines.

Note: Specify the VSYNP bits so that the following expressions is satisfied.
VTLN ≥ VSYNP
VSYNP ≥ VDLN + 1



Active video = Top/left border + addressable video + bottom/right border
Total H blank = Hsync time + back porch + front porch
Total V blank = Vsync time + back porch + front porch

LDHCNR.HDCN = H addressable video
LDHCNR.HTCN = H total time
LDHSYNR.HSYNW = Hsync time
LDHSYNR.HSYNP = H addressable video + right border + front porch

LDVLNR.VDLN = V addressable video
LDVLNR.VTLN = V total time
LDVSYNR.VSYNW = Vsync time
LDVSYNR.VSYNP = V addressable video + bottom border + front porch

Figure 39.5 Valid Display and Blanking Periods

39.3.19 Main LCD Horizontal Partial Display Register (MLDHPDR)

MLDHPDR specifies the number of horizontal display characters and the number of offset characters in the main LCD partial display mode. For details on settings, see section 39.4.5, Partial Screen Mode.

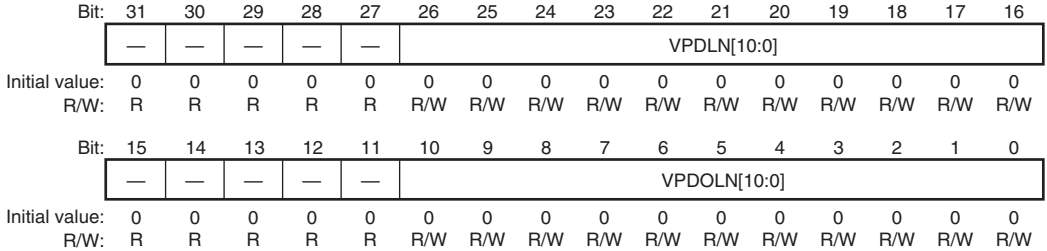
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	HPDCN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HPDOCN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	HPDCN [7:0]	H'00	R/W	Horizontal Partial Display Character Count These bits specify the number of horizontal display characters of the partial display in partial display mode (one character = eight dots).
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	HPDOCN [7:0]	H'00	R/W	Horizontal Partial Display Offset Character Count These bits specify the horizontal offset of the partial display from the display size in partial display mode in units of characters (one character = eight dots).

39.3.20 Main LCD Vertical Partial Display Register (MLDVPDR)

MLDVPDR specifies the number of vertical display lines and the number of offset lines in the main LCD partial display mode.



Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	VPDLN [10:0]	H'000	R/W	Vertical Partial Display Line Count These bits specify the number of vertical display lines of the partial display in partial display mode (one character = eight dots).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	VPDOLN [10:0]	H'000	R/W	Vertical Partial Display Offset Line Count These bits specify the vertical offset of the partial display from the display size in partial display mode in units of lines.

39.3.21 Main LCD Power Management Register (MLDPMR)

MLDPMR controls power supply circuits for the main LCD. For details, see section 39.4.6, Power Management Function. When the main LCD module is not used, be sure to clear this register to H'0000 0000.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ONA[3:0]				ONB[3:0]				ONC[3:0]				OFFD[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OFFE[3:0]				OFFF[3:0]				—	VC	VE	DO	—	—	LPS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	ONA[3:0]	0000	R/W	LCDC Switching-On Sequence Interval A These bits specify the time in units of frames until the display data (LCDD) and timing signals (LCDHSYN, LCDDCK, LCDDISP, LCDVSYN, and $\overline{\text{LCDRD}}$) start to be output after an assertion of the LCDVCPWC signal in the LCDC switching-on sequence. When the VC bit is cleared to 0, clear these bits to H'0. For details, see section 39.4.6, Power Management Function.
27 to 24	ONB[3:0]	0000	R/W	LCDC Switching-On Sequence Interval B These bits specify the time in units of frames until the LCDVEPWC signal is asserted after the display data (LCDD) and timing signals (LCDHSYN, LCDDCK, LCDDISP, LCDVSYN, and $\overline{\text{LCDRD}}$) start to be output in the LCDC switching-on sequence. When the VE bit is cleared to 0, clear these bits to H'0. For details, see section 39.4.6, Power Management Function.
23 to 20	ONC[3:0]	0000	R/W	LCDC Switching-On Sequence Interval C These bits specify the time in units of frames from assertion of the LCDVEPWC signal to assertion of the LCDDON signal in the LCDC switching-on sequence. For details, see section 39.4.6, Power Management Function.

Bit	Bit Name	Initial Value	R/W	Description
19 to 16	OFFD[3:0]	0000	R/W	LCDC Switching-Off Sequence Interval A These bits specify the time in units of frames from a negation of the LCDDON signal to a negation of the LCDVEPWC signal in the LCDC switching-off sequence. When the VE bit is cleared to 0, clear these bits to H'0. For details, see section 39.4.6, Power Management Function.
15 to 12	OFFE[3:0]	0000	R/W	LCDC Switching-Off Sequence Interval B These bits specify the time in units of frames until the display data (LCDD) and timing signals (LCDHSYN, LCDDCK, LCDDISP, LCDVSYN, and $\overline{\text{LCDRD}}$) stop output after a negation of the LCDVEPWC signal in the LCDC switching-off sequence. For details, see section 39.4.6, Power Management Function.
11 to 8	OFFF[3:0]	0000	R/W	LCDC Switching-Off Sequence Interval C These bits specify the time in units of frames until the LCDVCPWC signal is negated after the display data (LCDD) and timing signals (LCDHSYN, LCDDCK, LCDDISP, LCDVSYN, and $\overline{\text{LCDRD}}$) stop output in the LCDC switching sequence. When the VC bit is cleared to 0, clear these bits to H'0. For details, see section 39.4.6, Power Management Function.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	VC	0	R/W	LCDVCPWC Signal Enable Enables or disables LCDC switching-on and -off sequences using the LCDVCPWC signal. 0: LCDVCPWC fixed low 1: LCDVCPWC assertion and negation controlled by a given sequence
5	VE	0	R/W	LCDVEPWC Signal Enable Enables or disables LCDC switching-on and -off sequences using the LCDVEPWC signal. 0: LCDVEPWC fixed low 1: LCDVEPWC assertion and negation controlled by a given sequence

Bit	Bit Name	Initial Value	R/W	Description
4	DO	0	R/W	LCDDON Signal Enable Enables or disables LCDC switching-on and -off sequences using the LCDDON signal. 0: LCDDON fixed low 1: LCDDON assertion and negation controlled by a given sequence
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	LPS[1:0]	00	R	LCD Module Power Supply State These bits indicate the state of the power supply for the LCD module while LCDC switching-on and -off sequences are enabled. 00: Power is not supplied to the LCD module 11: Power is supplied to the LCD module

39.3.22 LCDC Palette Control Register (LDPALCR)

LDPALCR specifies the access mode to the palette memory. For details on the color palette specification and settings, see section 39.4.2, Color Palette Specification.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	MM	0	R/W	Main LCD Palette Mode Enables the main LCD to use the palette memory. 0: Disables use of the palette memory by the main LCD 1: Enables use of the palette memory (24-bit input and output palette memory) by the main LCD
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PE	0	R/W	Palette Read/Write Enable Specifies the source of access to the palette memory. 0: The LCDC uses the palette memory (display mode) 1: The host (CPU) uses the palette memory (CPU access mode)

39.3.23 LCDC Interrupt Register (LDINTR)

LDINTR specifies the start timing of interrupts and indicates the interrupt status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	UILN[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SE	UE	ME	MHE	FE	VSE	VEE	—	SS	US	MS	MHS	FS	VSS	VES
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	UILN[10:0]	H'000	R/W	User-Specified Interrupt Line Number These bits specify the position to generate a user-specified interrupt. A user-specified interrupt occurs when image data for the specified number of lines has been read from the external memory. The number of lines is specified by this register.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	SE	0	R/W	Frame Start Interrupt Enable Enables or disables an interrupt to be generated when the first pixel data of a frame is output to the LCD. This setting is valid only for output frame in one-shot mode. 0: Disables an interrupt when the first pixel data is output to the LCD 1: Enables an interrupt when the first pixel data is output to the LCD
13	UE	0	R/W	User-Specified Interrupt Enable Enables or disables an LCDC user-specified interrupt. 0: Enables a user-specified interrupt 1: Disables a user-specified interrupt

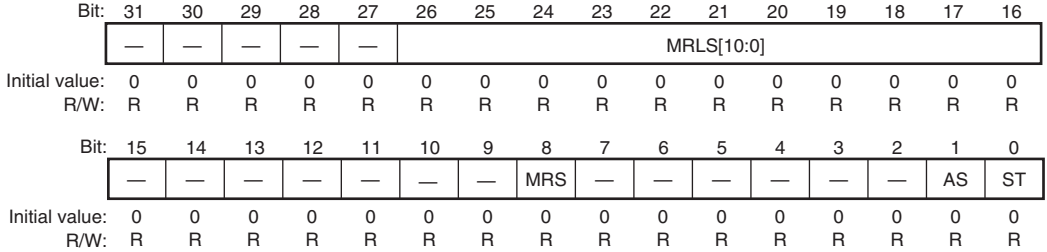
Bit	Bit Name	Initial Value	R/W	Description
12	ME	0	R/W	Memory Access Interrupt Enable Enables or disables an interrupt to be generated on completion of one frame of data access to external memory. 0: An interrupt is not generated on completion of one-frame data access to external memory 1: An interrupt is generated on completion of one-frame data access to external memory
11	MHE	0	R/W	Data Memory Read Interrupt Enable Enables or disables an interrupt to be generated on completion of one line of data access to the external memory. 0: An interrupt is not generated on completion of one-line data access to the external memory 1: An interrupt is generated on completion of one-line data access to the external memory
10	FE	0	R/W	Frame End Interrupt Enable Enables or disables an interrupt to be generated when the last pixel of a frame is output to the LCD. 0: Disables an interrupt when the last pixel of a frame is output to the LCD 1: Enables an interrupt when the last pixel of a frame is output to the LCD
9	VSE	0	R/W	VSYNC Start Interrupt Enable Enables or disables a VSYNC start interrupt to be generated. 0: A VSYNC start interrupt is not generated 1: A VSYNC start interrupt is generated
8	VEE	0	R/W	VSYNC End Interrupt Enable Enables or disables a VSYNC end interrupt to be generated. 0: A VSYNC end interrupt is not generated 1: A VSYNC end interrupt is generated

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	SS	0	R/W	Frame Start Interrupt Status Indicates that a frame start interrupt has been generated. In one-shot mode, an interrupt is generated only for an output frame. Write 0 to this bit to clear it in the frame start interrupt handler. 0: Indicates that a frame start interrupt has not occurred or the interrupt handling has been completed 1: Indicates that a frame start interrupt has occurred or the interrupt handling has not been completed
5	US	0	R/W	User-Specified Interrupt Status Indicates that a user-specified interrupt has been generated. Write 0 to this bit to clear it in the user-specified interrupt handler. 0: Indicates that a user-specified interrupt has not occurred or the interrupt handling has been completed 1: Indicates that a user-specified interrupt has occurred or the interrupt handling has not been completed
4	MS	0	R/W	Memory Access Interrupt Status Indicates that a frame end interrupt has been generated for access to external memory. Write 0 to this bit to clear it in the frame end interrupt handler of the external memory access. 0: Indicates that an external memory access frame end interrupt has not occurred or the interrupt handling has been completed 1: Indicates that an external memory access frame end interrupt has occurred or the interrupt handling has not been completed

Bit	Bit Name	Initial Value	R/W	Description
3	MHS	0	R/W	Data Memory Read Interrupt Status Indicates that a line end interrupt has been generated for access to external memory. Write 0 to this bit to clear it in the line end interrupt handler of the external memory access. 0: Indicates that an external memory access line end interrupt has not occurred or the interrupt handling has been completed 1: Indicates that an external memory access line end interrupt has occurred or the interrupt handling has not been completed
2	FS	0	R/W	Frame End Interrupt Status Indicates that a frame end interrupt has been generated. In one-shot mode, an interrupt is generated only for an output frame. Write 0 to this bit to clear it in the frame end interrupt handler. 0: Indicates that a frame end interrupt has not occurred or the interrupt handling has been completed 1: Indicates that a frame end interrupt has occurred or the interrupt handling has not been completed
1	VSS	0	R/W	VSYNC Start Interrupt Status Indicates that a VSYNC start interrupt has been generated. Write 0 to this bit to clear it in the VSYNC end interrupt handler. 0: Indicates that a VSYNC start interrupt has not occurred or the interrupt handling has been completed 1: Indicates that a VSYNC start interrupt has occurred or the interrupt handling has not been completed
0	VES	0	R/W	VSYNC End Interrupt Status Indicates that a VSYNC end interrupt has been generated. Write 0 to this bit to clear it in the VSYNC end interrupt handler. 0: Indicates that a VSYNC end interrupt has not occurred or the interrupt handling has been completed 1: Indicates that a VSYNC end interrupt has occurred or the interrupt handling has not been completed

39.3.24 LCDC Status Register (LDSR)

LDSR indicates status related to the LCDC operations



Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MRLS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MRS	—	—	—	—	—	—	AS	ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0.
26 to 16	MRLS[10:0]	H'000	R	Memory Read Line Status These bits indicate the number of lines which have been read from external memory.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	MRS	0	R	Main LCD Register Side Status Indicates the register side of the main LCD used by the LCDC. 0: Side A is used 1: Side B is used
7 to 2	—	All 0	R	Reserved These bits are always read as 0.
1	AS	0	R	SYS Interface Access Status Indicates the access status of the SYS interface. When the status indicates bus busy, do not issue the next transaction. Otherwise, operation is not guaranteed. 0: Bus idle 1: Bus busy (read or write access is being executed or is waiting)

Bit	Bit Name	Initial Value	R/W	Description
0	ST	0	R	Operating Status Indicates the LCDC operating status. 0: LCDC is idle 1: LCDC is operating

39.3.25 LCDC Control Register 1 (LDCNT1R)

LDCN1R specifies whether to output display data to an LCD during LCDC operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DE	1	R/W	Display Enable Specifies whether to output display data to the LCD during LCDC operation. When only write-back operation is necessary, writing 0 to this bit fixes the data output to the LCD. When this bit is 0, LCDC operation other than output of display data is carried out. 0: Does not output display data (fixes output data low or high) 1: Outputs display data

39.3.26 LCDC Control Register 2 (LDCNT2R)

LDCNT2R specifies a reset and the display operation of the main LCDs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SR	BR	—	—	—	—	MD	SE	ME	DO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	SR	0	R/W	Software Reset Setting this bit to 1 initializes all the LCDC internal circuits except for registers. Bus transactions are guaranteed. However, when the LCDC operates together with the BEU, its handshake is not guaranteed. Even if handshaking, the LCDC is immediately initialized by a reset. Initializing by a software reset must be performed more than once. The BEU should be initialized before the software reset when operating together with the LCDC. 0: LCDC is in normal operation 1: Writing 1 initializes the LCDC
8	BR	0	R/W	Module Reset Setting this bit to 1 initializes all the LCDC internal circuits except for registers. Bus transactions are guaranteed. However, when the LCDC operates together with the BEU, its handshake is not guaranteed. Even if handshaking, the LCDC is immediately initialized by a reset. The BEU should be initialized before the software reset during operation. 0: LCDC is in normal operation 1: Writing 1 initializes the LCDC

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MD	0	R/W	Display Data Read Device Specifies the device from which display data of the main LCD is read. Specify the display data format to RGB888 by clearing the PKF bit in MLDDFR to 0. 0: Display data is read from memory 1: Display data is read from the BEU
2	SE	0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
1	ME	0	R/W	Main LCD Enable Enables or disables whether the LCDC operates the main LCD while the display is turned on. 0: LCDC does not operate the main LCD 1: LCDC operates the main LCD
0	DO	0	R/W	Display On Starts or stops the LCDC display operation. The state of the switching-on process can be read from the LPS bit in MLDPMR and SODPMR. When the DO bit is set to 1, set either the SE or ME bit to 1. 0: Stops the LCDC operation (display-off mode) 1: Starts the LCDC operation (display-on mode)

The display switching-on and switching-off sequences are shown below.

- Display switching-on sequence (when changing the DO bit from 0 to 1)

  A. Start the LCDC display operation.

  B. Turn on the LCD power according to the method specified by MLDPMR and LDCNT2R.

  When a change from B'00 to B'11 in the LPS bit is confirmed, the switching-on sequence is completed.

  Do not access the DO bit until the completion of the sequence.
- Display switching-off sequence (when changing the DO bit from 1 to 0)

  A. Turn off the LCD power according to the method specified by MLDPMR and LDCNT2R.

  B. Stop the LCDC display operation.

  When a change from B'11 to B'00 in the LPS bit is confirmed, the switching-off sequence is completed.

  Do not access the DO bit until the completion of the sequence.

39.3.27 LCDC Register Side Change Control Register (LDRCNTR)

LDRCNTR specifies register sides of the main LCDS.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MRS	MRC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	MRS	0	R/W	Main LCD Register Side Select Selects the main LCD register side the LCDC uses. Register sides are changed when display of one frame of data is completed. This bit is valid when the MRC bit is cleared to 0. 0: Side A registers are used for the main LCD 1: Side B registers are used for the main LCD
0	MRC	0	R/W	Main LCD Register Side Change Enable Selects the main LCD register side the LCDC uses when display of one frame of data is completed. When this bit is cleared to 0, the side selected by the SRS bit is used. When this bit is set to 1, the side is toggled. 0: Register side is selected by the SRS bit 1: Register side is toggled

39.3.28 LCDC Input Image Data Swap Register (LDDDSR)

LDDDSR enables or disables byte, word, and longword swap on input image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	LS	WS	BS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	LS	0	R/W	Input Image Data Longword Swap Select Enables or disables input image data to be swapped in longwords. 0: Disables longword swap 1: Enables longword swap
1	WS	0	R/W	Input Image Data Word Swap Select Enables or disables input image data to be swapped in words. 0: Disables word swap 1: Enables word swap
0	BS	0	R/W	Input Image Data Byte Swap Select 0: Disables byte swap 1: Enables byte swap

Note: The LCDC handles data in big endian by default.

39.3.29 LCDC Register Side Forcible Select Register (LDRCR)

LDRCR immediately selects the register side when changing register sides.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MR	0	R/W	Main LCD Register Side Select Selects the main LCD register side the LCDC uses. 0: Side A registers are used 1: Side B registers are used

39.3.30 LCDC Driver Write Data Registers 0 to F (LDDWD0R to LDDWDFR)

LDDWDnR specifies the write data and the register select signal (LCDRS) polarity for the LCD driver.

For details on write data stored in the DWD bits such as bits used as a valid instruction, refer to the specifications for the LCD driver used.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	WDACT	—	—	—	RSW	—	—	—	—	—	—	DWD[17:16]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DWD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	WDACT	0	R/W	Data Write Select Selects whether write data set in the DWD bits is output when the SYS interface is used. 0: Write data in DWD is not output to the LCD driver 1: Write data in DWD is output to the LCD driver
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	RSW	0	R/W	Write Register Select Polarity Selects the polarity of the register select signal (LCDRS) when data is written to via the SYS interface by the CPU. 0: Register select signal is active low 1: Register select signal is active high
23 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 to 0	DWD[17:0]	H'0 0000	R/W	Write Data These bits store data to be written to the LCD driver via the SYS interface.

39.3.31 LCDC Driver Read Data Register (LDDRDR)

LDDRDR specifies the read data and the register select signal (LCDRS) polarity for the LCD driver.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	RSR	—	—	—	—	—	—	DRD[17:16]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	RSR	0	R/W	Read Register Select Polarity Selects the polarity of the register select signal (LCDRS) when data is read by the CPU via the SYS interface. 0: Register select signal is active low 1: Register select signal is active high
23 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 to 0	DRD[17:0]	H'0 0000	R	Read Data These bits store data to be read from the LCD via the SYS interface.

39.3.32 LCDC Driver Write Access Register (LDDWAR)

The LCDC issues a write transaction to an LCD when the WA bit is set to 1. Data stored in the DWD bits in LDDWDnR in which the WDACT bit is set to 1 is used. The data is written to in the following order: LDDWD0R, LDDWD1R, LDDWD2R, ... When the WDACT bit is read as 0, the write access is stopped. If all the WDACT bits are read as 1, the write access is stopped when a write transaction of LDDWDFR is issued. Before a write transaction is issued, set the WDACT bit in LDDWD0R to 1. After the write transaction is issued, the WA bit is automatically cleared to 0.

Notes: 1. To issue a write transaction to an LCD via the SYS interface during display operation, the vertical blanking time must be equal to two lines or more.
2. To issue a write transaction, specify the vertical blanking time so that access to the LCD can be completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	WA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	WA	0	R/W	Write Transaction Issue Issues a write transaction. 0: — 1: A write transaction issued

39.3.33 LCDC Driver Read Access Register (LDDRAR)

The LCDC issues a read transaction to an LCD when the RA bit is set to 1. When this register is accessed, read data is stored in the DRD bits in LDDRDR.

Notes: 1. To issue a read transaction to an LCD via the SYS interface during display operation, the vertical blanking time must be equal to two lines or more.

2. To issue a read transaction, specify the vertical blanking time so that access to the LCD driver can be completed. If a write access is not completed within the vertical blanking time, the CA bit in LDINTR is set to 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RA	0	R/W	Read Transaction Issue Issues a read transaction. 0: — 1: A read transaction issued

39.4 Operation

39.4.1 LCDC Display Features

- Supported LCD Module
  TFT panel. The RGB interface and 80-Series CPU bus interface (SYS interface) are supported as the module interface.
- Display Resolution
  Up to XGA and HD (1280 × 720) supported.
- Display Colors
  16,777,216 colors max. (24-bit colors)
- Display Color Control
  The 256-entry palette memory supports 24-bit input and 24-bit output.
- Data Polarity Inversion
  The polarity of the RGB interface signals can be inverted. Display data can also be inverted.

39.4.2 Color Palette Specification

The LCDC has a color palette which has 256 entries. The data of each entry can be input and output in units of 24 bits, and 256 entries can be used simultaneously. The color palette must be specified while the LCDC is not operated.

The color palette can be specified through the following procedure.

- Clear the PE bit in LDPALCR to B'0 (initial value).
  The normal display mode is entered.
- Set the PE bit in LDPALCR to B'1.
  The color palette set mode is entered.
- Set data in PALDnn bits in LDPRnn (nn = H'00 to H'FF)
- Clear the PE bit in LDPALCR to B'0.
  The normal display mode is returned.

Figure 39.6 shows a data format of the color palette.

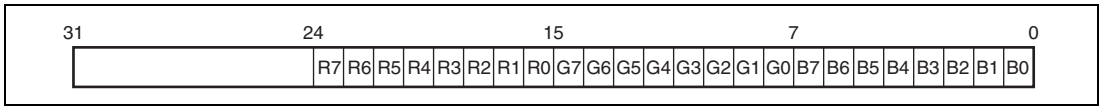


Figure 39.6 Data Format of Color Palette

For PALDnn bits in LDPRnn, bits 23 to 16 correspond to R data, bits 15 to 8 correspond to G data, and bits 7 to 0 correspond to B data. Bits 31 to 24 are reserved.

39.4.3 Display Timing Control

The display timing control is specified by the horizontal character number registers (MLDHCNR and SLDHCNR), horizontal sync signal registers (MLDHSYNR and SLDHSYNR), vertical line number registers (MLDVLNR and SLDVLNR), and vertical sync signal registers (MLDVSYNR and ASLDVSYNR).

The LCDC dot clock register (LDDCKR) specifies the division ratio. The frame rate of an LCD module is determined by the following factors: the display size and blanking period for a screen, the clock frequency used, and the division ratio.

The LCDC can generate an interrupt when the last pixel data of a frame is output after one frame of data has been read from memory. An interrupt can also be generated at intervals of user-specified lines. The LCDC interrupt register (LDINTR) specifies these settings.

39.4.4 One-Shot Mode

The LCDC supports the one-shot mode in which display data is read depending on the frame rate of the display image.

Examples and notes for use are shown below.

- Usage example 1
  1. Clear the OS bit in MLDSM1R to 0: continuous mode (initial value) is selected.
  2. Set to the OS bit to 1: a transition to the one-shot mode is made.
  3. Set the OSTRG bit in MLDSM2R to 1 when the image should be displayed: one frame of display data is read.
  4. Repeat step 3.
  5. Clear the OS bit to 0: the continuous mode is returned.

- Usage example 2: When continuously reading display data in one-shot mode
  1. Clear the OS bit in MLDSM1R to 0: continuous mode (initial value) is selected.
  2. Set to the OS bit to 1: a transition to the one-shot mode is made.
  3. Read the frame end interrupt status bit (FS) in LDINTR.
  4. When the FS bit is read as 1, set the OSTRG bit in MLDSM2R to 1: one frame of display data is read.
  5. Clear the FS bit to 0: the frame end interrupt status is cleared.
  6. Repeat steps 3 to 5.
  7. Clear the OS bit to 0: continuous mode is returned.

Note: When an offset (other than a value of H'F) is specified in the ONA, ONB, or ONC bits in MLDPMR, the trigger must be output first. The image data corresponding to the frame is not output.

39.4.5 Partial Screen Mode

The LCDC supports the partial screen mode in which display data only for an area to be updated is read.

Examples and notes for use are shown below.

- Example
  1. Specify the size of the partial screen through MLDHPDR and MLDVPDR.
  2. Clear the PRD bit in MLDSM1R to 0 and set the OS bit to 1: full screen size in one-shot mode.
  3. Set the DO bit in LDCNT2R to 1: the LCDC is started.
  4. Display data for the full screen size is read in one-shot mode.
  5. To use the partial screen mode, set both the PRD and OS bits to 1 while display data is not being read: partial screen in one-shot mode.
  6. Set the OSTRG bit in MLDSM2R to 1: display data for the partial screen size is read.
  7. Repeat step 6.
  8. Clear the PRD bit to 0 and set the OS bit to 1: the full screen size in one-shot mode.

Notes: 1. The screen size obtained from the partial screen size + offset specified in MLDHPDR and MLDVPDR must not exceed the full screen size.

2. Screen size change (a transition between full screen size and partial screen size modes or change of the partial screen sizes) cannot be used in continuous mode. Read screen sizes can be changed only while the LCDC is stopped (the LPS bits in MLDPMR are cleared to B'00) or while display data is not being read in one-shot mode.

39.4.6 Power Management Function

Most LCD modules require particular sequences to turn the power on and off. In the LCDC, the main LCD power management register (MLDPMR), LCDC control register 1 (LDCNT1R), and LCDC control register 2 (LDCNT2R) can be specified to control the LCD power supply control pins (LCDVCPWC, LCDVEPWC, and LCDDON) and execute various power control sequences using these pins. Figure 39.7 shows an example of a timing chart for switching-on and switching-off sequences.

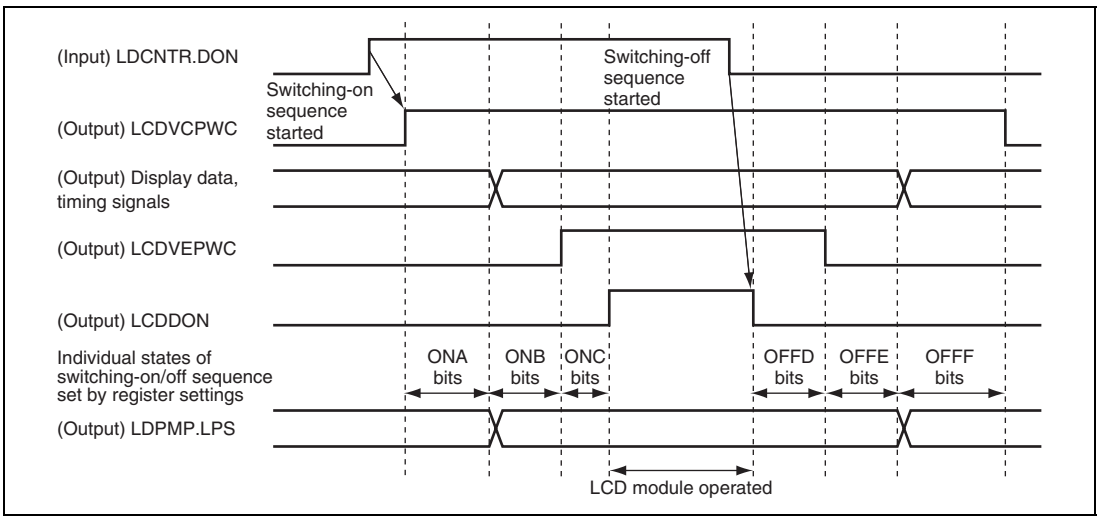


Figure 39.7 Switching-On and -Off Sequences and the LCD Module Operating State

39.4.7 Dot Clock Settings

Setting the dot clock is described. Figure 39.8 shows an example of a dot clock pattern to make the refresh rate 55 Hz. In this example, registers are set to a 33-MHz source clock, 304 dots in the horizontal direction (the HTCN bits in MLDHCNR), and 330 lines in the total vertical direction (the VTLN bit in MLDVLNR). When the two-time or three-time transfer mode is used, specify the registers so that the dot clock is two or three times the refresh rate.

The registers to set the dot clock (LDDCKR, MLDDCKPAT1R, and MLDDCKPAT2R) must be accessed while the LCDC stops the operation. Otherwise, the operation is not guaranteed.

For the denominator (m) of the division ratio, only 60, 54, 48, or 42 can be specified.

33 MHz ÷ (304 × 330 × 55 Hz) = 5.98 ≈ 6 → Specify the division ratio n/m to 10/60.

The MDCDR bits in LDDCKR = H'3C

The DCKPAT1 bits in MLDDCKPAT1R = H'1C71C71

The DCKPAT1 bits in MLDDCKPAT2R = H'C71C71C7

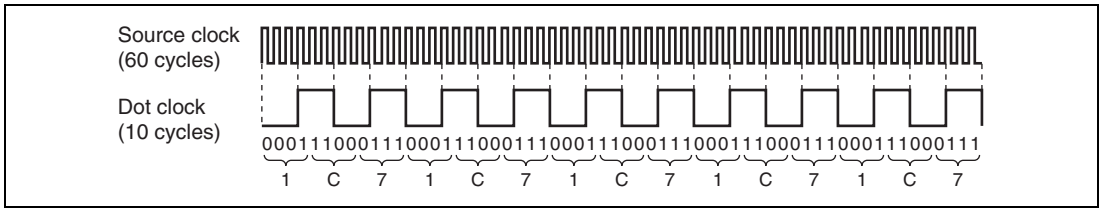


Figure 39.8 Example of Dot Clock Pattern

When the SYS interface is in use, the write strobe signal to be output is the inverse of the dot clock signal in the pattern that has been set up.

Note: For the source clock of the dot clock, Bϕ, Pϕ, or LCDLCLK can be selected. When selecting Pϕ, set Pϕ to a lower frequency than Bϕ and do not set the clock ratio to 1:1. When selecting LCDLCLK, set LCDLCLK to the same as or lower than the Bϕ frequency, including jitter.

39.4.8 Operation Coupled with the BEU

The LCDC can acquire display data from the BEU and form the corresponding display on the LCD panel.

Start Procedure:

1. Make settings for the BEU registers. Here, select the LCDC as the output destination of the BEU. For details on the register settings, see section 36.3, Description of BEU Registers.
2. Activate the BEU. The BEU waits for a request from the LCDC.
3. Make settings for the LCDC registers. Set the LCDC to acquire data from the BEU.
4. Activate the LCDC. The LCDC starts operating with the BEU.

The following is an example of settings. In this example, a QCIF (176 × 144) size image is output without blending.

Settings of BEU registers:

BSMWR1	← H'000000B0 (Image memory width: 176)
BSSZR1	← H'009000B0 (Image size: 176 × 144)
BSAYR1	← H'0C000000 (Storage area of image (Y): 0x0C000000)
BSACR1	← H'0C400000 (Storage area of image (CbCr): 0x0C400000)
BSIFR1	← H'00000000 (Image format: YCbCr444)
BBLCR0	← H'00000000 (Blend)
BBLCR1	← H'00040000 (Parent display and LCDC output settings)
BESTR	← H'00000101 (BEU activation setting)

Settings of LCDC registers:

Make normal register settings. (Dot clock settings, LCD panel settings, etc.)

MLDDFR ← H'00000000 (Specifies RGB88 as the input data format)

MLDHCNR.HDCN ← H'16 (Horizontal size: 22 characters [176 pixels])

MLDVLNR.VDLN ← H'90 (Vertical size: 144 lines)

LDCNT2R ← H'0000000B (Specifies BEU as the source for data acquisition, LCDC activation setting)

End Procedure:

1. Place the LCDC in display-off mode (set DO in LDCNT2R to 0).
2. Wait for the LCDC to complete its operations (wait for LPS in MLDPMR to be cleared to 0).
3. End the BEU operation (clear BESTR to 0).

Notes:
1. When resetting the BEU and the LCDC during coupled operation, reset the BEU before resetting the LCDC.
2. Specify the same values for the sizes of the input image for the LCDC and the output image from the BEU.
3. During coupled operation of the BEU and the LCDC, the RGB888 data format is used. Specify RGB888 (xLDDFR = 0:0RGB) as the input data format.
4. When acquiring display data from the BEU, specify the number of pixels in the horizontal direction as 1024 dots (= 128 characters) or less. When register planes are switched, planes are switched for the LCDC and BEU in synchronization. Use the frame end interrupt of the LCDC to make register settings for the LCDC and the BEU.

39.4.9 YCbCr Output Operation

This section explains the YCbCr output mode of operation. This is the operating mode when the setting of MLDMT1R.YM = 1. Input data in the YCbCr 4:2:0 format are not converted into RGB data but are output as YCbCr 4:2:2 data with the specified timing. Figure 39.9 shows an example of the output signals, and Figure 39.10 shows an example of the valid display and blanking periods.

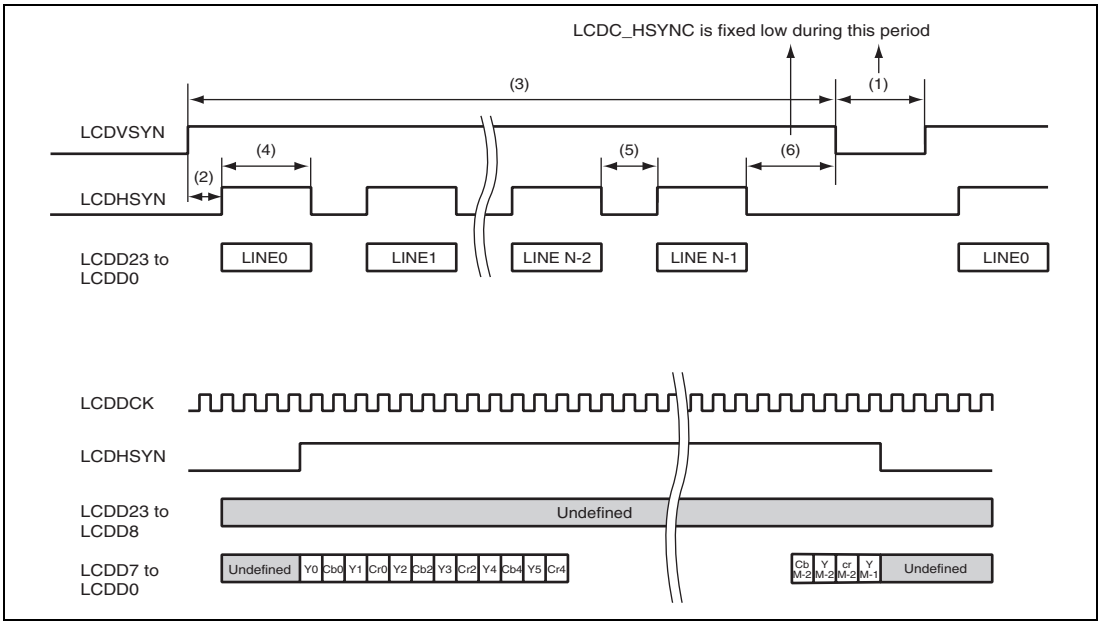


Figure 39.9 Example of Output Signals in YCbCr Operating Mode

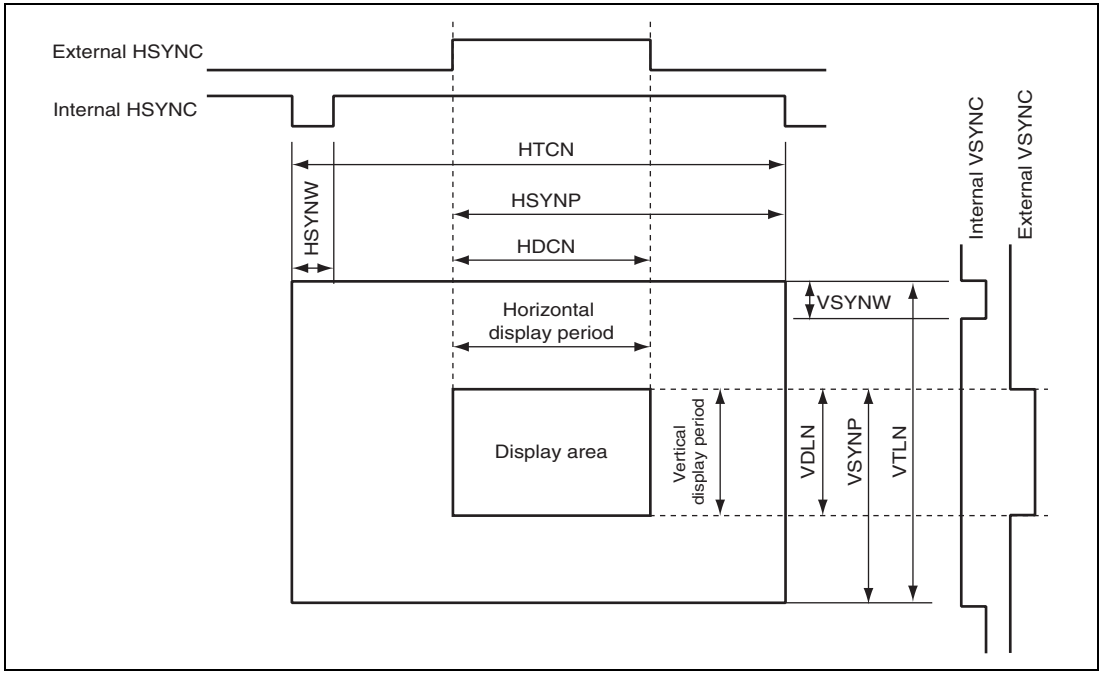


Figure 39.10 Valid Display and Blanking Periods in YCbCr Operating Mode

The valid display and the blanking periods in YCbCr operating mode are the same as those for the RGB interface. Although the internal horizontal and vertical sync signals are generated according to the specified timing, the sync signals for output to external devices are completely different signals. The horizontal sync signal for output to an external device is asserted during the horizontal display period, and only during those horizontal display periods that correspond to the vertical display period (assertion of VSYNC). The vertical sync signal for output to an external device is only asserted during the vertical display period.

The following restrictions of settings apply in this mode.

LDHCNR.HTCN = LDHCNR.HDCN + 20

LDHSYNR.HSYNP = LDHCNR.HDCN + 10

HSYNW = 3

LDVLNR.VTLN = LDVLNR.VDLN + 20

LDVSYNR.VSYNP = LDVLNR.VDLN + 10

VSYNW = 4

The numbers of cycles in (1) to (6) of Figure 39.9 are listed below.

(1) = LDHCNR.HTCN × 320

(2) = 160

(3) = LDVLNR.VDLN × LDHCNR.HTCN × 8 × 2

(4) = LDHCNR.HDCN × 8 × 2

(5) = 320

(6) = 160

Make the same dot clock setting as for the two-time transfer mode of the RGB interface. This means that the dot clock setting should be made so that the dot clock is two times the refresh rate.

In YCbCr operating mode, all input data must be YCbCr 4:2:0. In this operating mode, RGB conversion is not performed and the data is packed as YCbCr 4:2:2 for output.

Figure 39.11 shows the data format.

Output data	LCDD 23	LCDD 22	LCDD 21	LCDD 20	LCDD 19	LCDD 18	LCDD 17	LCDD 16	LCDD 15	LCDD 14	LCDD 13	LCDD 12	LCDD 11	LCDD 10	LCDD 9	LCDD 8	LCDD 7	LCDD 6	LCDD 5	LCDD 4	LCDD 3	LCDD 2	LCDD 1	LCDD 0
1st pixel (1st, Y[7:0])																	Y[7]	Y[6]	Y[5]	Y[4]	Y[3]	Y[2]	Y[1]	Y[0]
1st pixel (2nd, Cb[7:0])																	Cb[7]	Cb[6]	Cb[5]	Cb[4]	Cb[3]	Cb[2]	Cb[1]	Cb[0]
2nd pixel (1st, Y[7:0])																	Y[7]	Y[6]	Y[5]	Y[4]	Y[3]	Y[2]	Y[1]	Y[0]
2nd pixel (2nd, Cr[7:0])																	Cr[7]	Cr[6]	Cr[5]	Cr[4]	Cr[3]	Cr[2]	Cr[1]	Cr[0]

Figure 39.11 Output Format in YCbCr Operating Mode

Only the main LCD can be operated in YCbCr operating mode. Furthermore, the partial display mode and color palette functions are not available. Make settings so that these functions are not in use.

In YCbCr operating mode, supported image sizes are SQCIF (128 × 96), QCIF (176 × 144), QVGA (320 × 240), CIF (352 × 288), and VGA (640 × 480).

39.5 Setting Registers

39.5.1 Two-Side Register Switching

The LCDC has two sets of main LCD registers.

Register sides are changed when the main LCD completes its display of one frame of data and a frame end interrupt occurs.

Figure 39.12 shows an example of the timing when register sides are changed and frame end interrupts occur. Set a register side for the next frame every time a frame end interrupt occurs.

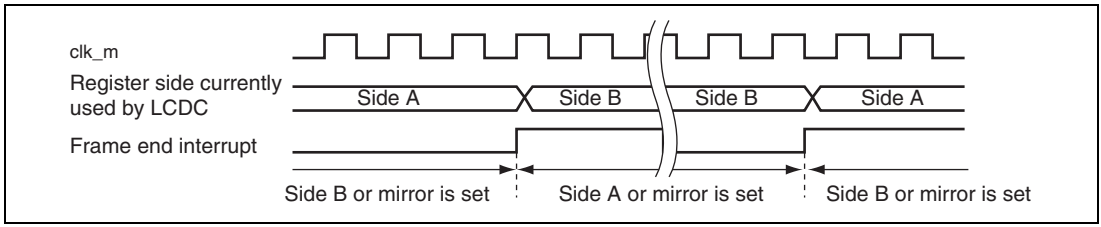


Figure 39.12 Timing of Interrupts and Register Setting

39.6 Examples of Clock and LCD Data Signals

- 16-Bit Data Bus, One 1-Cycle Transfer, TFT LCD Module, 240 × 320 Pixels, (RGB interface)

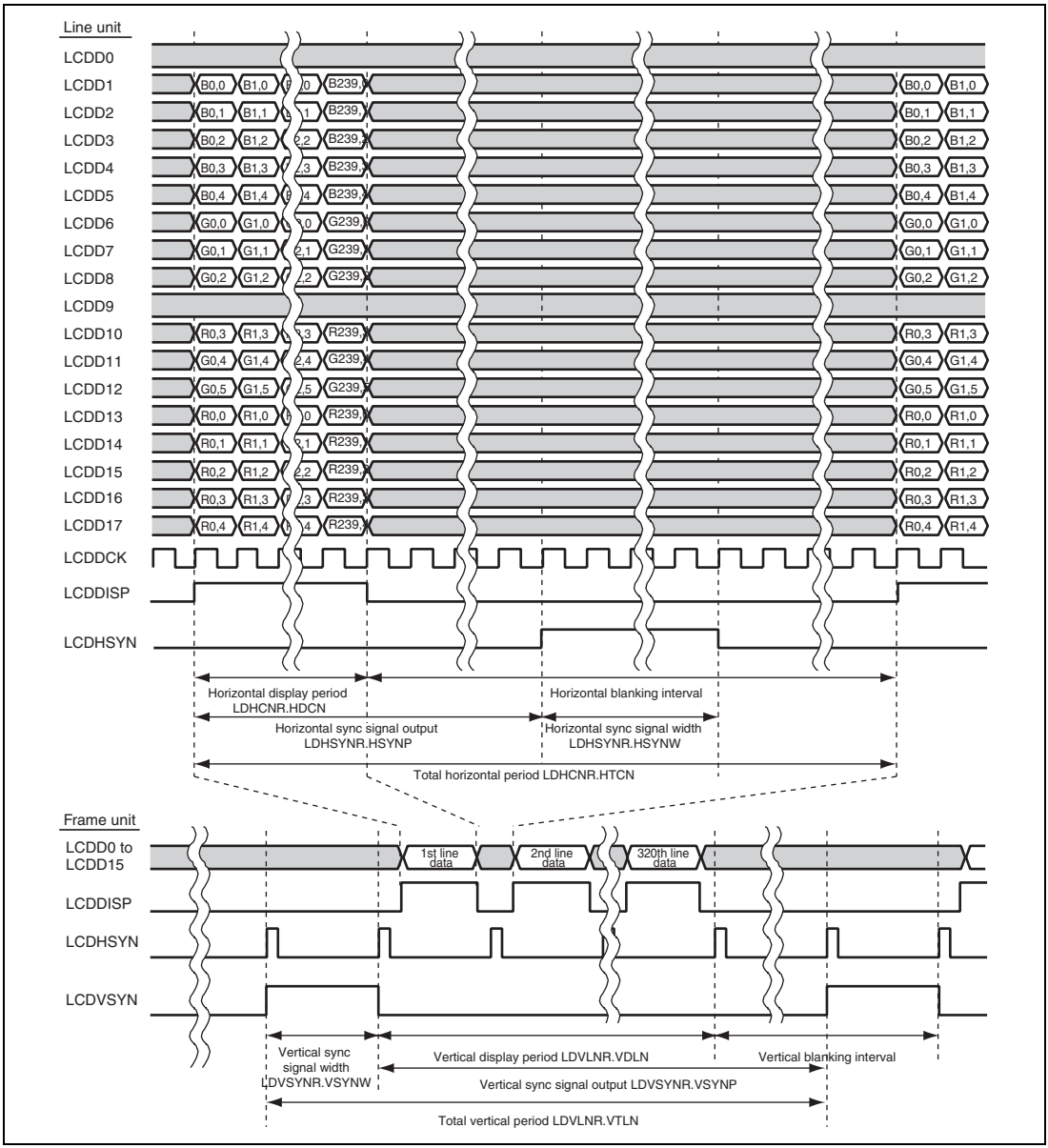


Figure 39.13 Example of Clock and LCD Data Signals (1)

- 16-Bit Data Bus, One 1-Cycle Transfer, TFT LCD Module, 240 × 320 Pixels, (SYS interface)

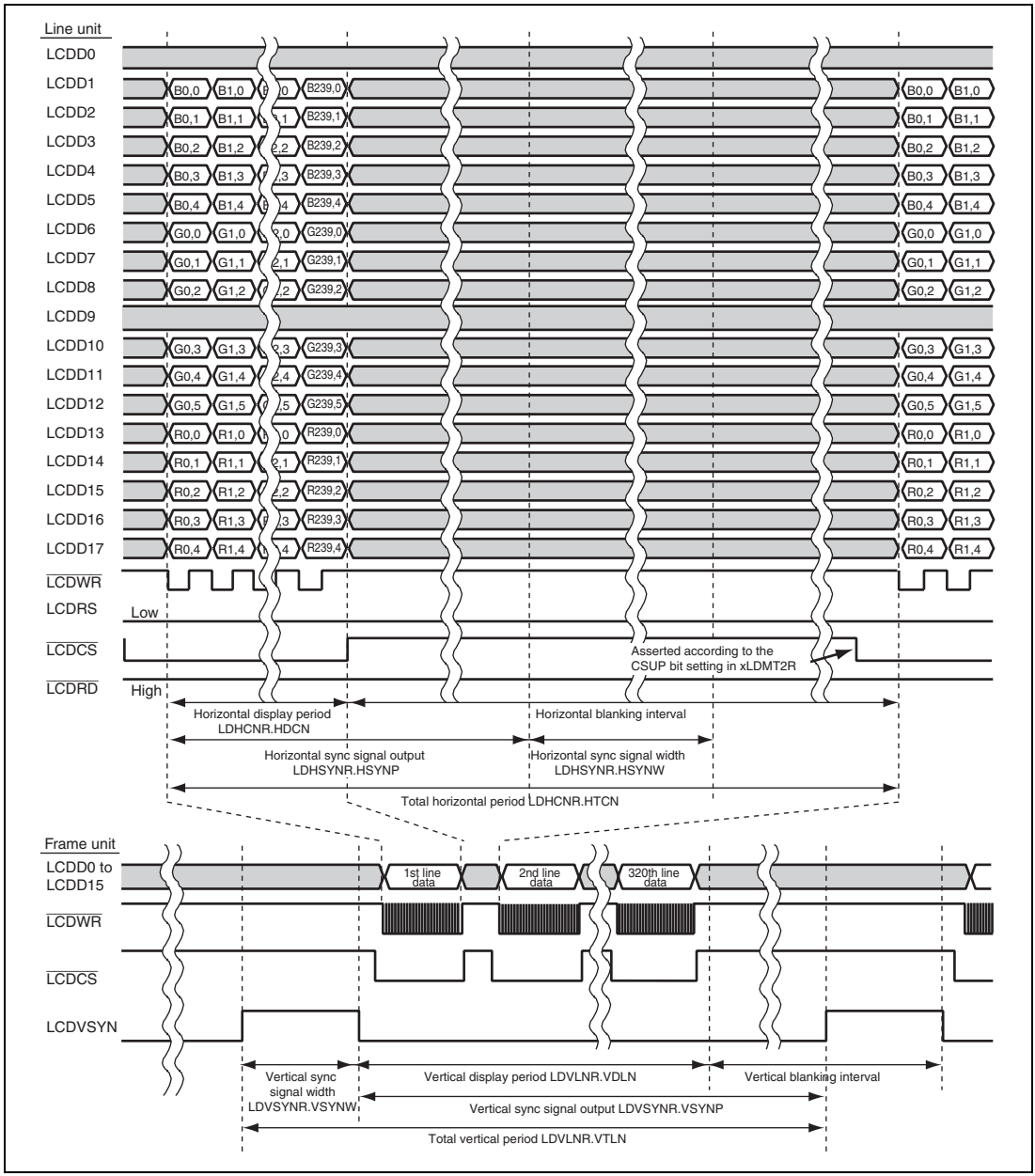
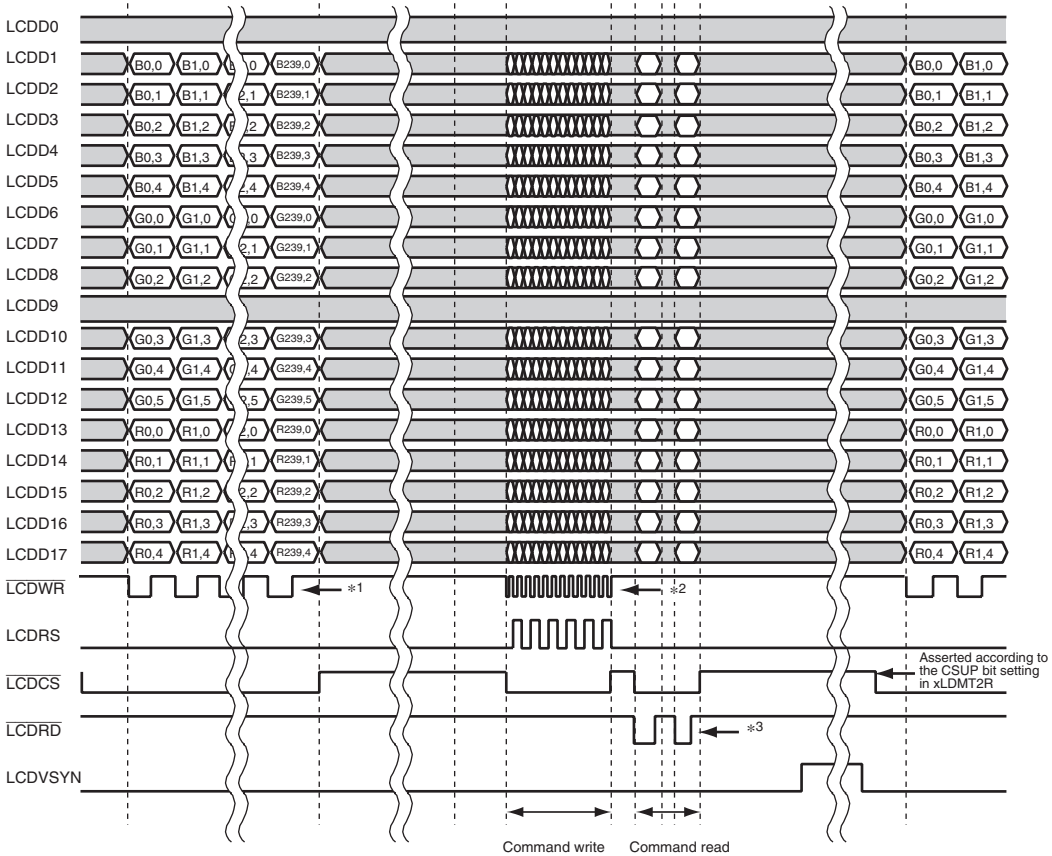


Figure 39.14 Example of Clock and LCD Data Signals (2)

- 16-Bit Data Bus, One 1-Cycle Transfer, TFT LCD Module, 240 ×320 Pixels, (SYS interface, command transfer)



Notes: 1. The signal is output in synchronization with the source clock selected by the ICKSEL bits in LDDCKR according to the xLDCKPAT1R and xLDCKPAT2R settings.
2. The signal is output in synchronization with the bus clock according to the WCSC, WCEC, and WCLW bit settings in xLDMT2R.
3. The signal is output in synchronization with the bus clock according to the RCSC, RCEC, and RCLW bit settings in xLDMT3R.

Figure 39.15 Example of Clock and LCD Data Signals (3)

39.7 Data Format

39.7.1 Main LCD Output Data Format

Figure 39.16 shows the format of the data output to the main LCD module.

Output data		LCDD 23	LCDD 22	LCDD 21	LCDD 20	LCDD 19	LCDD 18	LCDD 17	LCDD 16	LCDD 15	LCDD 14	LCDD 13	LCDD 12	LCDD 11	LCDD 10	LCDD 9	LCDD 8	LCDD 7	LCDD 6	LCDD 5	LCDD 4	LCDD 3	LCDD 2	LCDD 1	LCDD 0
SYS8a(1st)	24bpp							R7	R6	R5	R4	R3	R2	R1	R0										
SYS8a(2nd)								G7	G6	G5	G4	G3	G2	G1	G0										
SYS8a(3rd)								B7	B6	B5	B4	B3	B2	B1	B0										
SYS8b(1st)	18bpp							R7	R6	R5	R4	R3	R2	G7	G6										
SYS8b(2nd)								G5	G4	G3	G2	B7	B6	B5	B4										
SYS8b(3rd)														B3	B2										
SYS8c(1st)	18bpp													R7	R6										
SYS8c(2nd)								R5	R4	R3	R2	G7	G6	G5	G4										
SYS8c(3rd)								G3	G2	B7	B6	B5	B4	B3	B2										
SYS8d(1st)	16bpp							R7	R6	R5	R4	R3	G7	G6	G5										
SYS8d(2nd)								G4	G3	G2	B7	B6	B5	B4	B3										
SYS9(1st)	18bpp							R7	R6	R5	R4	R3	R2	G7	G6	G5									
SYS9(2nd)								G4	G3	G2	B7	B6	B5	B4	B3	B2									
SYS12(1st)	24bpp							R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4						
SYS12(2nd)								G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0						
SYS16a(1st)	16bpp							R7	R6	R5	R4	R3	G7	G6	G5		G4	G3	G2	B7	B6	B5	B4	B3	
SYS24(1st)	24bpp	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0
SYS16b(1st)	18bpp							R7	R6	R5	R4	R3	R2	G7	G6		G5	G4	G3	G2	B7	B6	B5	B4	
SYS16b(2st)								B3	B2																
SYS16c(1st)	18bpp							R7	R6																
SYS16c(2nd)								R5	R4	R3	R2	G7	G6	G5	G4		G3	G2	B7	B6	B5	B4	B3	B2	
SYS18(1st)	18bpp							R7	R6	R5	R4	R3	R2	G7	G6	G5	G4	G3	G2	B7	B6	B5	B4	B3	B2
RGB8(1st)	24bpp							R7	R6	R5	R4	R3	R2	R1	R0										
RGB8(2nd)								G7	G6	G5	G4	G3	G2	G1	G0										
RGB8(3rd)								B7	B6	B5	B4	B3	B2	B1	B0										
RGB9(1st)	18bpp							R7	R6	R5	R4	R3	R2	G7	G6	G5									
RGB9(2nd)								G4	G3	G2	B7	B6	B5	B4	B3	B2									
RGB12a(1st)	24bpp							R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4						
RGB12a(2nd)								G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0						
RGB12b(1st)	12bpp							R7	R6	R5	R4	G7	G6	G5	G4	B7	B6	B5	B4						
RGB16(1st)	16bpp							R7	R6	R5	R4	R3	G7	G6	G5		G4	G3	G2	B7	B6	B5	B4	B3	
RGB18(1st)	18bpp							R7	R6	R5	R4	R3	R2	G7	G6	G5	G4	G3	G2	B7	B6	B5	B4	B3	B2
RGB24(1st)	24bpp	R7	R6	R5	R4	R3	R2	R1	R0	G7	G6	G5	G4	G3	G2	G1	G0	B7	B6	B5	B4	B3	B2	B1	B0

Figure 39.16 Main LCD Output Data Format

39.7.2 Write-Back Data Format

Figure 39.17 shows the data format when the display data is written back.

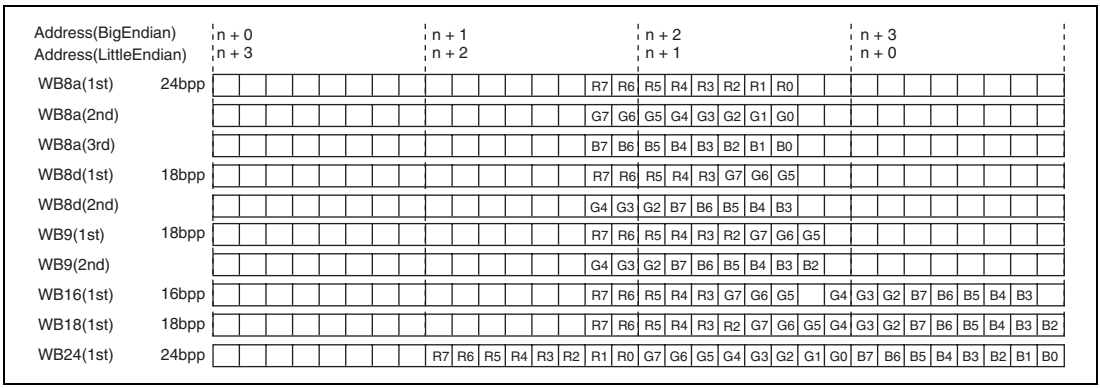


Figure 39.17 Write-Back Data Format

39.8 Usage Note

39.8.1 Notes on User-Specified Interrupts

When the format of the image data input to the LCDC is YCbCr420, user-specified interrupts work as follows.

1. When a user-specified interrupt is set at a line number equal to or smaller than half the number of lines in the vertical direction of the image, the user-specified interrupt is generated twice in one frame. The first interrupt position is correct; detect the first interrupt only.
2. When a user-specified interrupt is set at a line number greater than half the number of lines in the vertical direction of the image, no user-specified interrupt is generated. To set an interrupt at a line number (n) greater than half the number of lines in the vertical direction of the image, specify (n – vertical image size / 2) instead of n, and detect the second interrupt only.

Section 40 Video Output Unit (VOU)

The video output unit (VOU) converts image data that is obtained from the blend engine unit (BEU) or memory and outputs it as ITU-R BT.601 or ITU-R BT.656 digital data.

The VOU also scales up images.

40.1 Features

The VOU has the following features.

- Supported video system: NTSC, PAL
- Output digital level: Conforms to ITU-R BT.601, ITU-R BT.656
- Output interface: 16-bit Y/C interface, 8-bit multiplexed YC interface
- Output timing: 13.5 MHz in 16-bit Y/C interface, 27 MHz in 8-bit multiplexed YC interface
- Output pixel frequency: 13.5 MHz, 27 MHz
- Supported source image: sub-QCIF, QVGA, WQVGA, VGA
- Maximum destination image size: 720 × 240 per field
- Source image format: YCbCr 4:2:2, YCbCr 4:2:0, YCbCr 4:4:4, RGB
- Scaling up of images
  Horizontal factor: 1, 1.125, 2, 2.25, or 4
  Vertical factor: 1, 2, or 4
- RGB → YCbCr conversion function: Outputs YCbCr after converting obtained RGB data
- Double-buffered register: Efficient register access through a double-buffered mechanism

Note: The image is enlarged by 4 pixels in the horizontal and vertical directions.

Figure 40.1 shows a block diagram of the VOU.

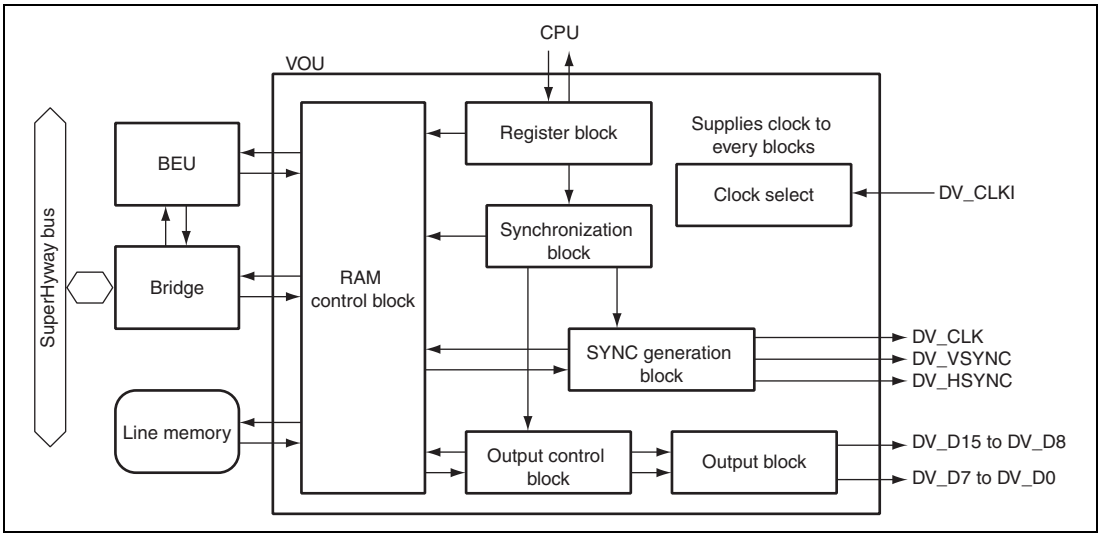


Figure 40.1 VOU Block Diagram

40.2 Pin Configuration

The VOU pin configuration is shown in Table 40.1.

Table 40.1 Pin Configuration

Name	Function	I/O	Description
DV_CLK	Pixel clock output	Output	Pixel clock output (13.5 MHz, 27 MHz)
DV_VSYNC	Vertical sync signal	Output	VOU vertical sync signal output
DV_HSYNC	Horizontal sync signal output	Output	VOU horizontal sync signal output
DV_D15 to DV_D8	Data output	Output	Upper pixel data (Y: 16-bit interface) (YC: 8-bit multiplexed YC interface) (Rec. 656 output)
DV_D7 to DV_D0	Data output	Output	Lower pixel data (C: 16-bit interface) (0: 8-bit multiplexed YC interface) (0: Rec. 656 output)
DV_CLKI	Video clock input	Input	Video clock input pin (27 MHz)

40.3 Register Descriptions

Table 40.2 shows the VOU register configuration. Table 40.3 shows the register state in each processing mode.

Some VOU registers have a double-buffered mechanism (side A and side B). The VOU switches two sides of a register. The mirror address is provided for accessing one of the two sides which is not currently being used.

Table 40.2 Register Configuration

			Address			
Register Name	Abbr.	R/W	Address (Side A)	Address (Side B)	Mirror Address	Access Size
VOU execution register	VOUER	R/W	H'FE96 0000	—	—	32
VOU control register	VOUCR	R/W	H'FE96 0004	H'FE96 1004	H'FE96 2004	32
VOU status register	VOUSTR	R/W	H'FE96 0008	—	—	32
VOU video control register	VOUVCR	R/W	H'FE96 000C	H'FE96 100C	H'FE96 200C	32
VOU source image size register	VOUISR	R/W	H'FE96 0010	H'FE96 1010	H'FE96 2010	32
VOU background color register	VOUBCR	R/W	H'FE96 0014	H'FE96 1014	H'FE96 2014	32
VOU display position register	VOUDPR	R/W	H'FE96 0018	H'FE96 1018	H'FE96 2018	32
VOU display size register	VOUDSR	R/W	H'FE96 001C	H'FE96 101C	H'FE96 201C	32
VOU valid pixel start position register	VOUVPR	R/W	H'FE96 0020	H'FE96 1020	H'FE96 2020	32
VOU interrupt register	VOUIR	R/W	H'FE96 0024	—	—	32
VOU reset register	VOUSRR	R/W	H'FE96 0028	—	—	32
VOU mode setting register	VOUMSR	R/W	H'FE96 002C	H'FE96 102C	H'FE96 202C	32
VOU horizontal sync interval register	VOUHIR	R/W	H'FE96 0030	H'FE96 1030	H'FE96 2030	32
VOU source image data format register	VOUDFR	R/W	H'FE96 0034	H'FE96 1034	H'FE96 2034	32
VOU source image data destination start address register 1	VOUAD1R	R/W	H'FE96 0038	H'FE96 1038	H'FE96 2038	32
VOU source image data destination start address register 2	VOUAD2R	R/W	H'FE96 003C	H'FE96 103C	H'FE96 203C	32
VOU source image data address increment register	VOUAIR	R/W	H'FE96 0040	H'FE96 1040	H'FE96 2040	32
VOU source image data swap register	VOUSWR	R/W	H'FE96 0044	—	—	32
VOU register side switch register	VOURCR	R/W	H'FE96 0048	—	—	32
VOU register side forcedly specify register	VOURPR	R/W	H'FE96 0050	—	—	32

Table 40.3 Register State in Each Processing Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby Mode	Module Standby Mode	R/U-Standby	Sleep
VOUER	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUSTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUVCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUISR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUBCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUDPR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUDSR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUVPR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUIR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUSRR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUMSR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUHIR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUDFR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUAD1R	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUAD2R	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUAIR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOUSWR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOURCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
VOURPR	Initialized	Initialized	Retained	Retained	Initialized	Retained

40.3.1 VOU Execution Register (VOUER)

VOUER activates and terminates the VOU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SEN	—	—	—	—	—	OM	IS	ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SEN	0	R/W	Sync Signal Enable Starts or stops outputting sync signals from the VOU. When the ST bit is set to 1, this bit must also be set to 1. 0: Sync signal output is halted 1: Sync signal output is started
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	OM	0	R/W	Operating Mode Specify Specifies the operating mode when the VOU obtains the source image from the memory. Operates in either frame units or field units. When the operating mode in which the VOU operates in field units is specified, the start address of each of the fields in VOUAD1R and VOUAD2R should be specified. In this case, the VSYNC interrupt and switching registers should be synchronized with the field. When the operating mode in which the VOU operates in frame units is specified, the start address of each of the frames in VOUAD1R and VOUAD2R should be specified. The source image should be stored as a frame image. In this case, the VOU automatically obtains the images in both the top and bottom fields. The VSYNC interrupt and switching registers should be synchronized with the frame. Note that when treating a YCbCr 4:2:0 image, specify that the VOU operates in frame units. 0: Operates in field units 1: Operates in frame units
1	IS	0	R/W	Display Data Source Selects whether the display data is obtained from the BEU or memory. When the display data is obtained from the BEU, the data is obtained as YCbCr 4:4:4 regardless of the VOU source image data setting. The BEU outputs the display data to the VOU as YCbCr 4:4:4 regardless of the data packed format for output. The display data should be stored as a frame image when the data is obtained from the memory and scaled up vertically. 0: Display data is obtained from BEU 1: Display data is obtained from memory
0	ST	0	R/W	Start Activates or terminates the VOU. Setting this bit to 1 activates the VOU. To stop the VOU, clear this bit to 0. The VOU is actually terminated when the immediately following frame is finished, and the EXE bit in VOUSTR is retained at 1 until that frame finishes. 0: VOU is terminated 1: VOU is activated

40.3.2 VOU Control Register (VOUCR)

VOUCR specifies the output mode, enables interrupts, and selects the sync signal width.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD[2:0]			CKPL	HPOL	VPOL	HSC[1:0]		—	HW[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VW[4:0]					—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	MD[2:0]	000	R/W	Output Mode Selects the format of data output from the VOU and the pixel clock frequency. 000: NTSC, 16-bit interface, 13.5-MHz clock output 001: NTSC, 8-bit interface, 27-MHz clock output 011: NTSC, 8-bit interface (Rec. 656), 27-MHz clock output 101: PAL, 8-bit interface, 27-MHz clock output Other than above: Reserved
28	CKPL	0	R/W	Pixel Clock Polarity Changes the synchronization edge of the pixel clock for the data and sync signals output from the VOU. 0: Data and sync signals are synchronized on the rising edge of the pixel clock 1: Data and sync signals are synchronized on the falling edge of the pixel clock
27	HPOL	0	R/W	Horizontal Sync Signal Polarity Changes the polarity of the horizontal sync signal output from the VOU. 0: DV_HSYNC is high-active 1: DV_HSYNC is low-active
26	VPOL	0	R/W	Vertical Sync Signal Polarity Changes the polarity of the vertical sync signal output from the VOU. 0: DV_VSYNC is high-active 1: DV_VSYNC is low-active

Bit	Bit Name	Initial Value	R/W	Description
25, 24	HSC [1:0]	00	R/W	Horizontal Synchronization Suppression Set one of these bits to 1 when the horizontal sync signal is to be output from the VOU only during a valid period. 00: Normal operation 01: Outputs DV_HSYNC only while the data is valid 10: Outputs DV_HSYNC only while DV_VSYNC is negated 11: Setting prohibited
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22 to 16	HW[6:0]	H'40	R/W	Horizontal Sync Signal Assert Width These bits change the assert width of the horizontal sync signal output from the VOU. Specify the assert width as a number of 13.5-MHz clock cycles. The initial value is 64.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	VW[4:0]	H'03	R/W	Vertical Sync Signal Assert Width These bits change the assert width of the vertical sync signal output from the VOU. Specify the assert width as a number of lines. The initial value is 3.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.3 VOU Status Register (VOUSTR)

VOUSTR clears the interrupt sources.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EXE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FLD	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	EXE	0	R	Execution Flag This bit is set to 1 when processing of a top field subsequent to setting the ST bit in VOUER to 1 starts. This bit is cleared to 0 when the frame subsequent to clearing the ST bit to 0 has finished. 0: VOU has halted 1: VOU is operating
30 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	RS	0	R	Register Side Status Indicates the side being used. 0: Side A is being used 1: Side B is being used
15	FLD	0	R	Field Flag Indicates the field being processed. 0: The field being processed is a top field 1: The field being processed is a bottom field
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.4 VOU Video Control Register (VOUVCR)

VOUVCR makes various settings regarding video control.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CLB	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HME	VME	—	—	—	—	—	—	—	—	HM[1:0]		—	—	—	VM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
30 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	CLB	0	R/W	Color Bar When this bit is set to 1, settings of all bits in VOUVCR except for this bit are disabled, and the destination image becomes a color bar. The settings of VOUDPR and VOUBCR are also disabled. The color bar output has the pattern shown in Figure 40.2. 0: Normal image is output 1: Color bar is output
22 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	HME	0	R/W	Horizontal Scale-Up Enable When this bit is set to 1, the horizontal size of the image is scaled up. The scale-up factor is determined by the setting of the HM1 and HM0 bits. 0: Horizontal size of the destination image is the same as that of the source image. 1: Horizontal size of the destination image is scaled up. The scale-up factor is set by the HM1 and HM0 bits. Note: When scaling up the horizontal size of the image, the HVP bit in VOUVPR + the HP bit in VOUDPR should be more than 9.

Bit	Bit Name	Initial Value	R/W	Description
14	VME	0	R/W	Vertical Scale-Up Enable When this bit is set to 1, the vertical size of the image is scaled up. The scale-up factor is determined by the setting of the VM bit. To scale up the vertical size of the display data, which is obtained from the memory, store the data as a frame image. 0: Vertical size of the destination image is the same as that of the source image. 1: Vertical size of the destination image is scaled up. The scale-up factor is set by the VM.
13 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	HM[1:0]	00	R/W	Scale-Up Factor of Horizontal Image Size These bits specify the scale-up factor for the horizontal size of the image. The setting of these bits is valid only when the HME bit is set to 1. Specify the scale-up factor so that the scaled-up image size is 720 or less. 00: Scale-up factor is 1.125 01: Scale-up factor is 2.25 10: Scale-up factor is 2 11: Scale-up factor is 4
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	VM	0	R/W	Scale-Up Factor of Vertical Image Size These bits specify the scale-up factor for the vertical size of the image. The setting of this bit is valid only when the VME bit is set to 1. Specify the scale-up factor so that the scaled-up image size is 240 or less. 0: Scale-up factor is 2 1: Scale-up factor is 4

Figure 40.2 shows the color bar pattern output according to the CLB bit setting.

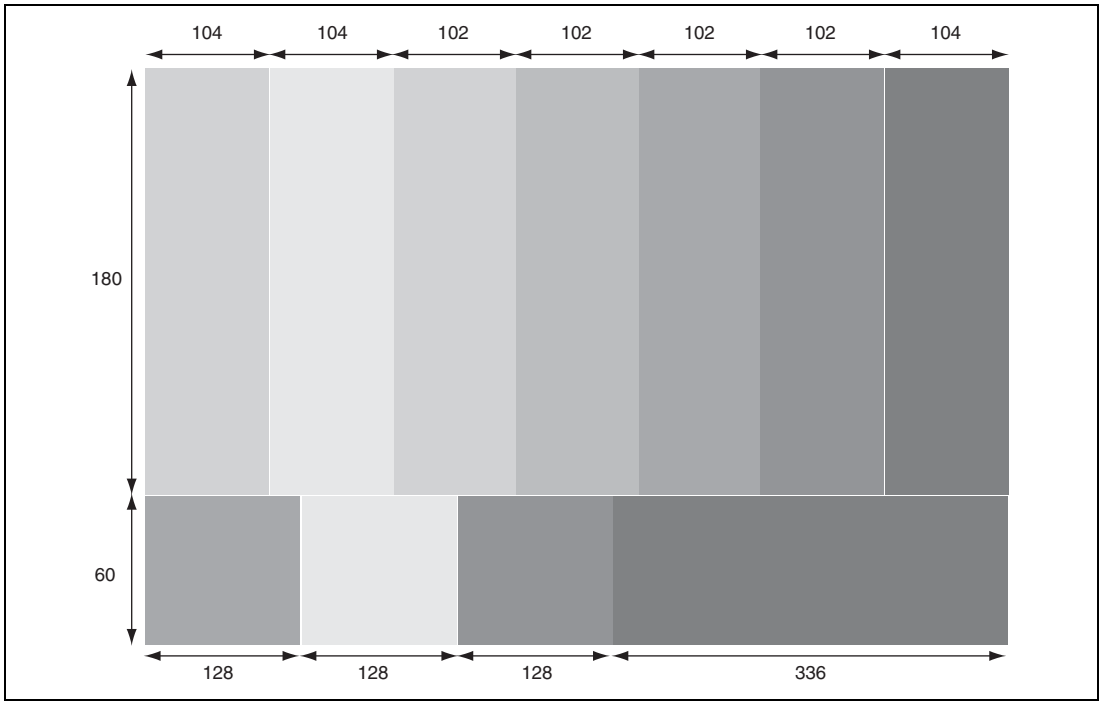


Figure 40.2 Output Color Bar Pattern

40.3.5 VOU Source Image Size Register (VOUISR)

VOUISR specifies the horizontal and vertical sizes of the source image.

When the frame-unit operating mode is selected (the OM bit in VOUER is 1), set VSZ to half the vertical size of the source image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	HSZ[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VSZ[8:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 18	HSZ[9:2]	H'000	R/W	Horizontal Size of Source Image
17, 16	HSZ[1:0]		R	Specify the horizontal size of the source image. Note that since the size is limited to 4n, the lower two bits should be 0. When the VOU operates in conjunction with the BEU, specify the same size as the horizontal size of the image output from the BEU.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 0	VSZ[8:0]	H'000	R/W	Vertical Size of Source Image Specify the vertical size of the source image. When the VOU operates in conjunction with the BEU, specify the same size as the vertical size of the image output from the BEU.

40.3.6 VOU Background Color Register (VOUBCR)

VOUBCR specifies the luminance, chrominance red, and chrominance blue of the background color.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	Y[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Cr[7:0]								Cb[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	Y[7:0]	H'10	R/W	Luminance These bits specify the luminance of the background color. The initial value is 16.
15 to 8	Cr[7:0]	H'80	R/W	Chrominance Red These bits specify the chrominance red of the background color. The initial value is 128.
7 to 0	Cb[7:0]	H'80	R/W	Chrominance Blue These bits specify the chrominance blue of the background color. The initial value is 128.

40.3.7 VOU Display Position Register (VOUDPR)

VOUDPR specifies the horizontal and vertical positions to output the destination image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	HP[9:0]									
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VP[8:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 17	HP[9:1]	H'080	R/W	Horizontal Position
16	HP[0]		R	These bits specify the horizontal position to output the image. A value from 0 to 858 (for NTSC) or a value from 0 to 864 (for PAL) can be set. Note that since the horizontal output position is limited to 2n, the lowest bit should be cleared to 0. The initial value is 128.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 0	VP[8:0]	H'010	R/W	Vertical Position These bits specify the vertical position to output the image. A value from 0 to 262 (for NTSC) or a value from 0 to 312 (for PAL) can be set. The initial value is 16.

40.3.8 VOU Display Size Register (VOUDSR)

VOUDSR specifies the horizontal and vertical image sizes.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	HDS[9:0]									
Initial value:	0	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	VDS[8:0]								
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 17	HDS[9:1]	H'2D0	R/W	Horizontal Image Size
16	HDS[0]		R	These bits specify the horizontal image size. Use these bits to adjust the display. The VOU outputs data for the size specified by these bits. Note that since the horizontal image size is limited to 2n, the lowest bit must be cleared to 0. The initial value is 720.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 0	VDS[8:0]	H'0F0	R/W	Vertical Image Size These bits specify the vertical image size. The VOU outputs lines for the size specified by these bits. The initial value is 240.

40.3.9 VOU Valid Pixel Start Position Register (VOUVPR)

VOUVPR specifies the horizontal and vertical positions where the valid pixel starts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	HVP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VVP[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 17	HVP[7:1]	H'78	R/W	Horizontal Valid Pixel Start Position
16	HVP[0]		R	These bits specify the horizontal position where output data becomes valid in pixel clock cycles. The sum of the setting of these bits and the setting of the HDS[9:0] bits in VOUDSR should be 858 or less (for NTSC), or 864 or less (for PAL). Note that since the horizontal valid pixel start position is limited to 2n, the lowest bit must be cleared to 0. The initial value is 120. Note: When scaling up the horizontal size of the image, the HVP bit in VOUVPR + the HP bit in VOUDPR should be more than 9.
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	VVP[6:0]	H'14	R/W	Vertical Valid Pixel Start Position These bits specify the vertical position where output data becomes valid in lines. The sum of the setting of these bits and the setting of the VDS[7:0] bits in VOUDSR should be 262 or less (for NTSC), or 312 or less (for PAL). Note that for the sake of data processing, a value higher than 3 should be set. The initial value is 20.

Figure 40.3 shows the relationship between the VOUDPR, VOUVPR, and VOUDSR settings and the destination image.

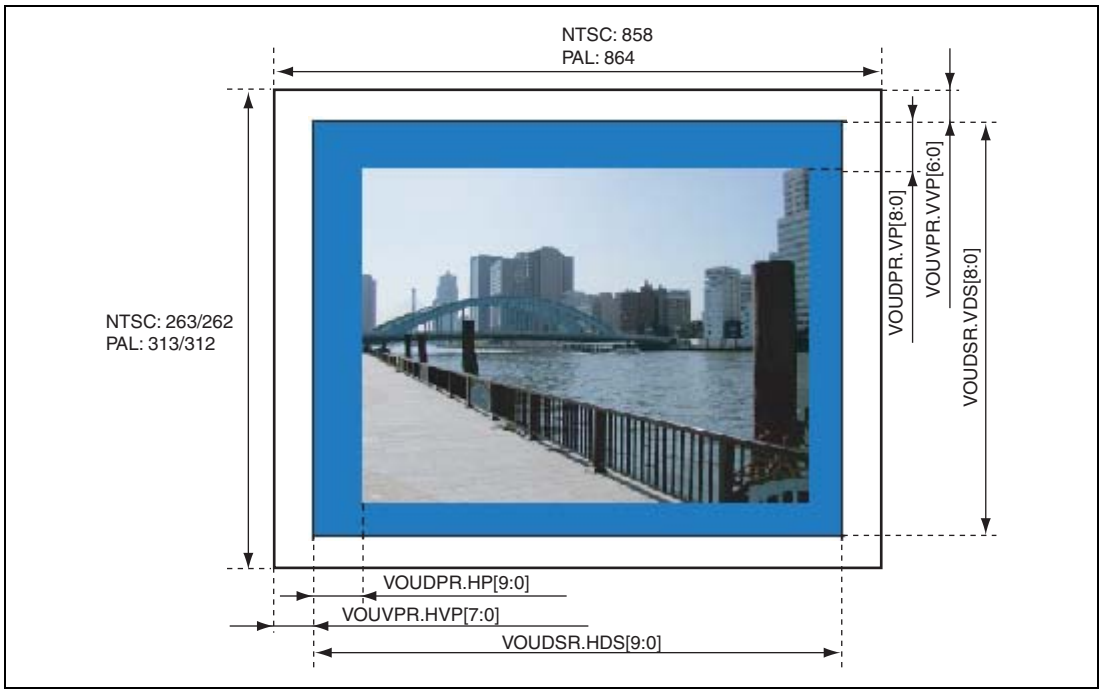


Figure 40.3 Relationship between VOUDPR, VOUVPR, and VOUDSR Settings and Destination Image

40.3.10 VOU Interrupt Register (VOUIR)

VOUIR specifies the settings of interrupts and their states.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FI	VI
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FS	VS	—	—	—	—	—	VT	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	FI	0	R/W	Field End Interrupt Enable Specifies the generation of a field-end interrupt. 0: Field-end interrupt is not generated 1: Field-end interrupt is generated
16	VI	0	R/W	VSYNC Synch Interrupt Enable Specifies the generation of an interrupt synchronized with the VSYNC. The interrupt is generated at the same time when the side of register is switched in synchronization with the VSYNC. 0: VSYNC synch interrupt is not generated 1: VSYNC synch interrupt is generated
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	FS	0	R/W	Field-End Interrupt State Indicates the state of a field-end interrupt. 0: Field-end interrupt is not generated 1: Field-end interrupt is generated

Bit	Bit Name	Initial Value	R/W	Description
8	VS	0	R/W	VSYNC Synch Interrupt State Indicates the state of a VSYNC synch interrupt. 0: VSYNC synch interrupt is not generated 1: VSYNC synch interrupt is generated
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	VT	0	R/W	VSYNC Synch Interrupt Timing Specifies the generation timing to generate a VSYNC synch interrupt. 0: VSYNC synch interrupt is generated in synchronization with the VSYNC for field 1: VSYNC synch interrupt is generated in synchronization with the VSYNC for frame
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.11 VOU Reset Register (VOUSRR)

VOUSRR resets the VOU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	HR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SR	—	—	—	—	—	—	—	BR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	HR	0	R/W	BEU Handshake Reset Forcibly stops handshaking with the BEU. When the ST bit is set to 1 in the subsequent field, handshaking with the BEU is performed again. When writing this bit to 1, stop the BEU. This bit is automatically cleared to 0 after 1 is written. 0: No operation 1: Forcibly stops handshaking
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SR	0	R/W	Software Reset Executes software reset of the VOU. Setting this bit to 1 initializes the VOU. The transaction of the bus is guaranteed. Handshaking with the BEU is forcibly stopped. When the VOU operates in conjunction with the BEU, reset the BEU in advance of the VOU. This bit is automatically cleared to 0 after 1 is written.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BR	0	R/W	Module Reset Executes the module reset of the VOU. Setting this bit to 1 initializes the VOU. Neither handshaking with the BEU nor bus transaction is guaranteed. When the VOU operates in conjunction with the BEU, reset the BEU in advance of the VOU. This bit is automatically cleared to 0 after 1 is written. 0: No operation 1: VOU is initialized except for the SEN bit in VOUER

40.3.12 VOU Mode Setting Register (VOUMSR)

VOUMSR specifies operating modes, arranges the VSYNC assert position, and selects the horizontal synch interval.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LMD	—	—	—	—	—	—	—	VSA	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HIS	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	LMD	0	R/W	LCDC Compatible Mode Specifies the VOU mode as LCDC compatible mode. LCDC compatible mode should be used with a 24.4545-MHz input clock. The HDS bit in VOUDSR should be set to 640. 0: Normal operating mode 1: LCDC compatible mode
30 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	VSA	0	R/W	VSYNC Assert Position Arrange Selects whether the VSYNC is asserted in synchronization with the HSYNC assertion or between the HSYNC and the subsequent HSYNC in the bottom field in NTSC mode. 0: VSYNC is output synchronized with HSYNC 1: VSYNC is output at the same timing in NTSC mode
22 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15	HIS	0	R/W	Horizontal Sync Interval Select Selects whether the HSYNC interval is fixed or changes according to the VOUHIR setting. 0: HSYNC interval is fixed When NTSC mode is selected, HSYNC is output at 858-pixel cycles. When PAL mode is selected, HSYNC is output at 864-pixel cycles. When LCDC compatible mode is selected, HSYNC is output at 780-pixel cycles. 1: HSYNC is output at the pixel cycle set by the HI[9:0] bits in VOUHIR.
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.13 VOU Horizontal Sync Interval Register (VOUHIR)

VOUHIR sets the horizontal sync signal interval.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	HI[9:0]									
Initial value:	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 17	HI[9:1]	H'35A	R/W	Horizontal Sync Interval
16	HI[0]		R	These bits specify the horizontal sync signal interval in pixels. The setting of these bits is valid only when the HIS bit in VOUMSR is set to 1. Note that since the sync signal interval is limited to 2n, the lowest bit must be cleared to 0. The initial value is 858.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

40.3.14 VOU Source Image Data Format Register (VOUDFR)

VOUDFR converts the source image data format and specifies the data packed format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	YF[1:0]		—	—	—	PKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CC	0	R/W	RGB to YCbCr Conversion Specifies the conversion from RGB to YCbCr. When display data in memory is in the RGB format, conversion is specified. 0: RGB to YCbCr conversion not performed 1: RGB to YCbCr conversion performed
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	YF[1:0]	0	R/W	Source Image YCbCr Data Packed Format Specifies the YCbCr packed format of the source image data. This bit is valid only when the CC bit is 0. 00: Source image data is YCbCr 4:2:0 01: Source image data is YCbCr 4:2:2 10: Source image data is YCbCr 4:4:4 11: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	PKF[4:0]	00000	R/W	Source Image RGB Data Packed Format Specifies the RGB format of source image data. This bit is valid when the CC bit is 1.

Table 40.4 shows the YCbCr packed format and Table 40.5 shows the RGB packed format.

Table 40.4 YCbCr Packed Format

YF[1:0]	YCbCr		31 to 24	23 to 16	15 to 8	7 to 0
B'00	4:2:0	Y data	Y0	Y1	Y2	Y3
		C data	Cb0	Cr0	Cb2	Cr2
B'01	4:2:2	Y data	Y0	Y1	Y2	Y3
		C data	Cb0	Cr0	Cb2	Cr2
B'10	4:4:4	Y data	Y0	Y1	Y2	Y3
		C data	Cb0	Cr0	Cb1	Cr1
B'11	Reserved	Reserved	Reserved	Reserved	Reserved	Reserved

Table 40.5 RGB Packed Format

PKF [4:0]	Bit Rate [bpp]	Phase	Bit																																					
			31 to 24								23 to 16								15 to 8								7 to 0															
B'00000	24		0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0						
B'00001	24		R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	0	0	0	0	0	0	0	0						
B'00010	24	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1						
		1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2						
		2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3						
B'00011	16		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1						
B'00111	18		0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0						
B'01000	12	-	0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0	0	0	0	0	R1	R1	R1	R1	G1	G1	G1	G1	B1	B1	B1	B1						
B'01001	18	0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1						
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2						
		2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3						
B'01010	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1						
		1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2						
		2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3						
B'01011	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1						
		1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2						
		2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3						
B'01100	24		0	0	0	0	0	0	0	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0						
B'01101 to B'11111	Reserved																																							

40.3.15 VOU Source Image Data Destination Start Address Register 1 (VOUAD1R)

VOUAD1R specifies the start address of the luminance signal or RGB signal display data for the source image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	ADR[31:2]	H'00000000	R/W	Specify the start address of the luminance signal or RGB signal display data when the VOU obtains the source image data from the memory. Store the image data as a frame image. Note that the lower two bits should be set to 0. Note: Whether the cash is enabled or not, the address in the P0 area should be specified in this register.
1, 0	ADR[1:0]		R	

40.3.16 VOU Source Image Data Destination Start Address Register 2 (VOUAD2R)

VOUAD2R specifies the start address of the chrominance signal display data for the source image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	ADR[31:2]	H'00000000	R/W	Specify the start address of the chrominance signal for display data when the VOU obtains source image data from the memory. Store image data as a frame image. Note that the lower two bits should be set to 0. Note: Whether the cash is enabled or not, the address in the P0 area should be specified in this register.
1, 0	ADR[1:0]		R	

40.3.17 VOU Source Image Data Address Increment Register (VOUAIR)

VOUAIR specifies the increment value of the address for source image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AIR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 2	AIR[15:2]	H'0000	R/W	Specify the increment value of the one-line address when the VOU obtains source image data from the memory. When the chrominance signal for display data is in the YCbCr 4:4:4 format, twice the specified increment value will be specified. Note that the lower two bits should be set to 0.
1, 0	AIR[1:0]		R	

40.3.18 VOU Source Image Data Swap Register (VOUSWR)

VOUSWR specifies byte swap, word swap, and longword swap for source image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	LS	WS	BS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	LS	0	R/W	Source Image Data Longword Swap Longword-swaps source image data (Figure 40.4). 0: Longword swapping not performed 1: Longword swapping performed
1	WS	0	R/W	Source Image Data Word Swap Word-swaps source image data (Figure 40.5). 0: Word swapping not performed 1: Word swapping performed
0	BS	0	R/W	Source Image Data Byte Swap Byte-swaps source image data (Figure 40.6). 0: Byte swapping not performed 1: Byte swapping performed

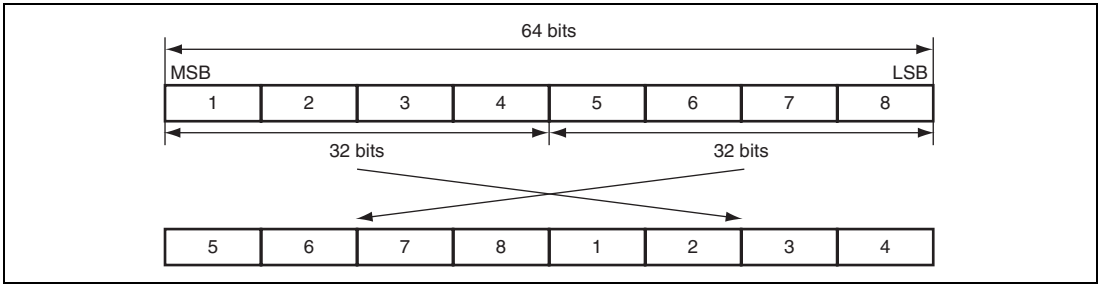


Figure 40.4 Longword Swap

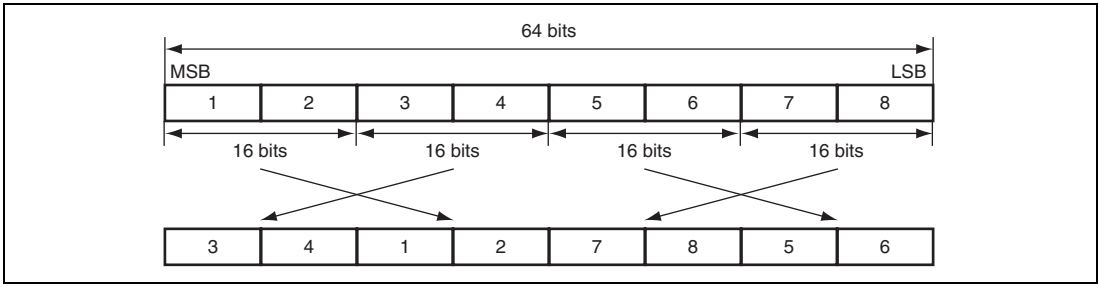


Figure 40.5 Word Swap

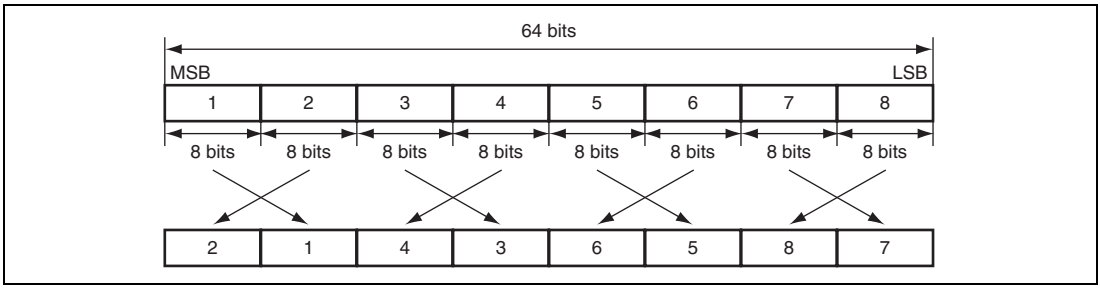


Figure 40.6 Byte Swap

40.3.19 VOU Register Side Switch Register (VOURCR)

VOURCR specifies the register side which the VOU uses and the timing for switching sides.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CT	RS	RC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	CT	0	R/W	Switch Timing Specify Specifies whether the sides are switched in synchronization with the VSYNC for field or frame. 0: Register side switch timing is synchronized with VSYNC for field 1: Register side switch timing is synchronized with VSYNC for frame
1	RS	0	R/W	Register Side Specify Specifies the register side, which the VOU uses, synchronized with the VSYNC for field or frame. Which VSYNC is used is specified by the CT bit. This bit is valid only when the RC bit is 0. 0: Side A is used 1: Side B is used
0	RC	0	R/W	Register Side Switch Enable Specifies whether the register side which the VOU uses is switched in synchronization with the VSYNC for field or frame or it is not switched. Which VSYNC is used is specified by the CT bit. When the register sides are not switched, the side specified by the RS bit is used. 0: Register side specified synchronized with VSYNC used 1: Register sides switched synchronized with VSYNC

40.3.20 VOU Register Side Forcedly Specify Register (VOURPR)

VOURPR specifies the register side which the VOU uses.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RP	0	R/W	Register Side Specify Specifies the register side to be used. 0: Side A is specified 1: Side B is specified

Table 40.6 shows the registers to be set to specify whether the source image data is obtained from the memory or the BEU, and to identify whether the source image data format is YCbCr or RGB.

Table 40.6 Registers to be Set

	Obtained from Memory		Obtained from BEU
Register Name	**YCbCr**	**RGB**	**YCbCr**
VOUER	√	√	√
VOUCR	√	√	√
VOUSTR	—	—	—
VOUVCR	√	√	√
VOUISR	√	√	√
VOUBCR	√	√	√
VOUDPR	√	√	√
VOUDSR	√	√	√
VOUVPR	√	√	√
VOUIR	√	√	√
VOUSRR	—	—	—
VOUMSR	√	√	√
VOUHIR	√	√	√
VOUDFR	√	√	X
VOUAD1R	√	√	X
VOUAD2R	√	X	X
VOUAIR	√	√	X
VOURCR	√	√	√

[Legend]

√: Need to be set.

X: Must not be set.

—: Can be set arbitrarily.

40.4 Operation

The VOU scales up the images obtained from the BEU or memory, converts their format, and outputs them to an external device. The destination image has the YCbCr 4:2:2 format and conforms to ITU-R BT.601 and ITU-R BT.656. The SYNC signals at output are generated from the pixel frequency clock input.

Images can be easily output to a TV by connecting the VOU output signals to a digital video encoder.

40.4.1 SYNC Signal Generation

Output of the SYNC signals (DV_HSYNC and DV_VSYNC) and pixel clock (DV_CLK) begins when the SEN bit in VOUER is set to 1. The sync signals are created from the 13.5-MHz clock signal.

Output of the SYNC signals and pixel clock is stopped when the SEN bit in VOUER is cleared to 0.

Whether the field is top or bottom can be identified by the assert location of the vertical sync signal.

40.4.2 Activation and Termination of VOU

The VOU is activated by setting the ST bit in VOUER to 1. The SEN bit should also be set to 1 when setting the ST bit to 1. The EXE bit is set to 1 at the start of the first top field following the ST bit being set to 1, and the VOU starts operation.

To stop the VOU, clear the ST bit to 0. The EXE bit is cleared to 0 at the end of the field following the ST being cleared to 0.

40.4.3 Handshake with BEU

While display image data is being obtained via the BEU when the EXE bit in VOUER is set to 1, do not stop the BEU since the VOU and BEU might be handshaking.

When the VOU is reset during operation (the HR bit, SR bit, or HR bit in VOUSRR is set to 1), the BEU must also be reset because the handshake with the VOU input unit is broken.

40.4.4 Modifying Registers during VOU Operation

The VOU registers have two sides. While the VOU is operating, register modification can be performed basically to register side (or mirror addresses) which are not being used by the VOU. If an attempt is made to write to any of the registers being used by the VOU, except the IE bit in VOUCR, VOUER, VOUSRR, and VOURCR, correct operation cannot be guaranteed. When the RC bit in VOURCR is 1, register sides are switched in synchronization with the VSYNC. When the RC bit is 0, the register sides specified by the RS bit in VOURCR which is synchronized with the VSYNC are used.

When the BEU and VOU operate in conjunction with each other, the registers in the BEU and VOU switch their sides simultaneously. The signal for switching register sides is output from the VOU to the BEU.

40.5 VOU Setting Procedure

The procedure for using the VOU to output an image is described below.

Figure 40.7 shows an example of setting the VOU to output a QVGA image which is not scaled up via the BEU.

The source image is in the YCbCr 4:2:2 format. The Y data is stored from address H'0C00 0000 and the CbCr data from address H'0C40 0000.

First, set up the BEU to output the image. In this setting example, one image is output to the VOU. The BEU can blend images and output them. To blend images, the BEU should be set for blending. As the last setting of the BEU, set the BEU activation bit to make the BEU wait for an activation signal from the VOU.

Next, make the VOU settings. After enabling or disabling interrupts and setting the display position, activate the VOU.

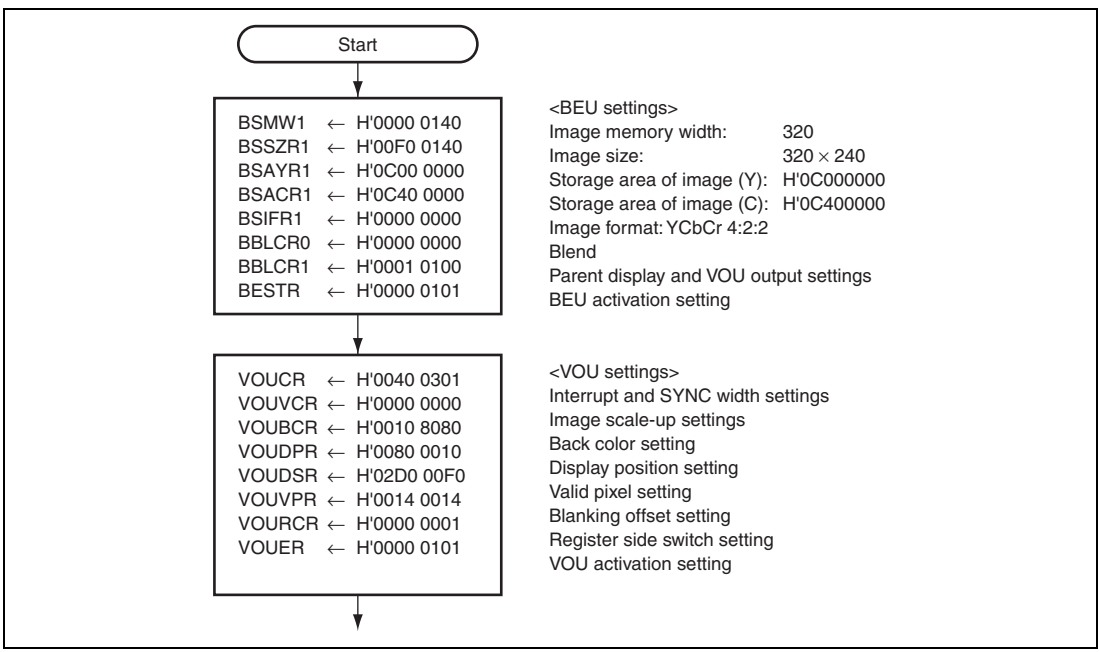


Figure 40.7 Setting Example of QVGA Output without Scaling up

40.6 Timing

40.6.1 Synchronous Timing

Figure 40.8 shows the timing of the HSYNC and the data valid period. The assert width of the HSYNC is specified by HW bits in VOUCR. The number of clocks between the HSYNC issuance and the data valid period is specified by HVP bits in VOUVPR. The data valid period is specified by HDS bits in VOUDSR.

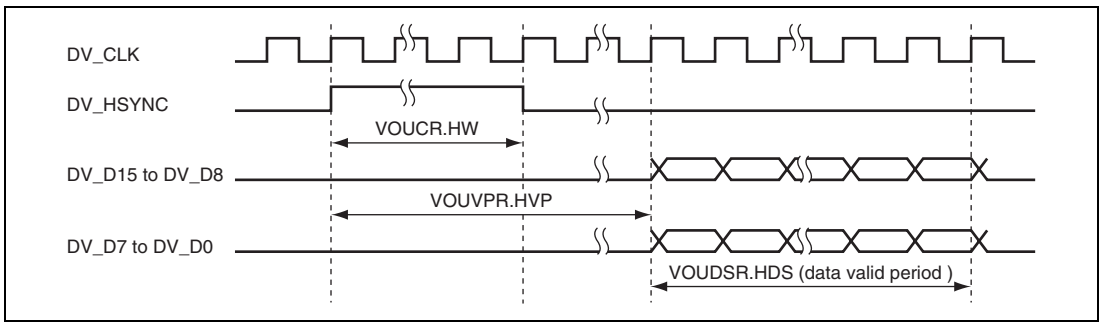


Figure 40.8 Timing of HSYNC and Data Valid Period

Figure 40.9 shows the timing of the HSYNC and the VSYNC generation. The assert width of the VSYNC is the number of lines specified by VW bits in VOUCR. At the top field, the VSYNC is output synchronized with the HSYNC. At the bottom field, the VSYNC is output between the HSYNC and the subsequent HSYNC.

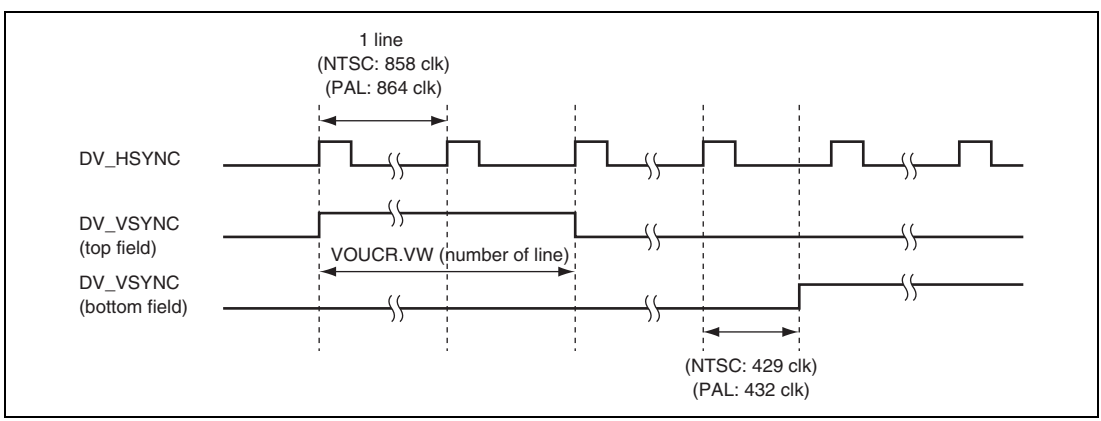


Figure 40.9 Timing of HSYNC and VSYNC Generation

40.6.2 Field End Interrupt Occurrence Timing

Figure 40.10 shows the timing at which a field end interrupt occurs.

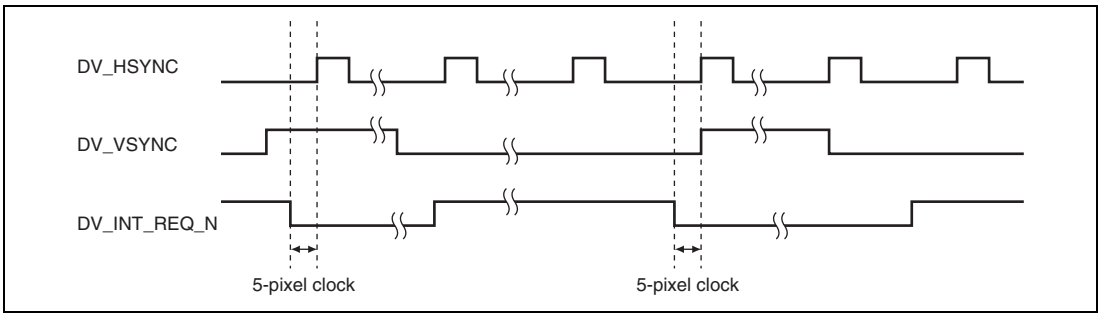


Figure 40.10 Interrupt Occurrence Timing

40.6.3 Switching Timing for Two- Side Registers

The VOU registers have two sides. The sides are switched two clocks after the VSYNC for frame or field has generated. The VSYNC interrupt occurs at the same time as register sides are switched. Figure 40.11 shows the timing of the register side switching and the VSYNC interrupt occurrence. After the VSYNC interrupt has occurred, specify the register for the subsequent frame or field.

The VSYNC interrupt (VOUIR.VI) and field interrupt (VOUIR.FI) should not be used simultaneously.

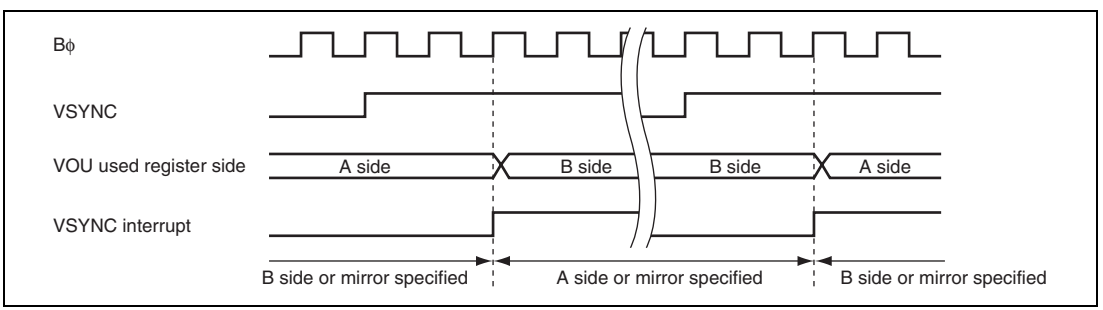


Figure 40.11 Timing of Interrupt and Specifying Timing

Section 41 Media RAM (MERAM)

This LSI has a 128-Kbyte media RAM (MERAM). Up to 32 interconnect buffers (ICBs) can be defined on the MERAM. Each ICB functions as a read buffer or write buffer for transactions of an associated image module (hereafter referred to as an associated module), and also allows data transmission/reception between associated image modules through the ICB. The use of ICBs enables the number of accesses to the SDRAM to be reduced and the access efficiency to be improved.

Furthermore, by combining multiple areas in the MERAM, it can be defined as a frame buffer cache.

Accesses from the following associated modules are supported: Two CEUs, two VEUs, two BEUs, VPU, JPU, and LCDC.

When using this function, bit 7 in the physical address space control register (PASCR) should be set to 1. (For details on PASCR, refer to section 7, Memory Management Unit (MMU).)

41.1 Features

(1) Features of Interconnect Buffers (ICBs)

- Up to 32 buffers can be defined.
- Each ICB functions as a read buffer, which prefetches data from the specified area of the SDRAM and then stores the data in the buffer according to read transactions from an associated module. The ICB can also start reading before a transaction from the associated module occurs.
- Each ICB also functions as a write buffer, which receives write transactions from a module and stores them in the buffer. Whether to write data back to the SDRAM or not is selectable.
- The write buffer for an associated module can be used as the read buffer for another associated module, which allows data transfer in the MERAM. Data write to the SDRAM is also possible.

(2) Features of Frame Buffer Cache

- An area of combined ICBs can be defined as a frame buffer cache.

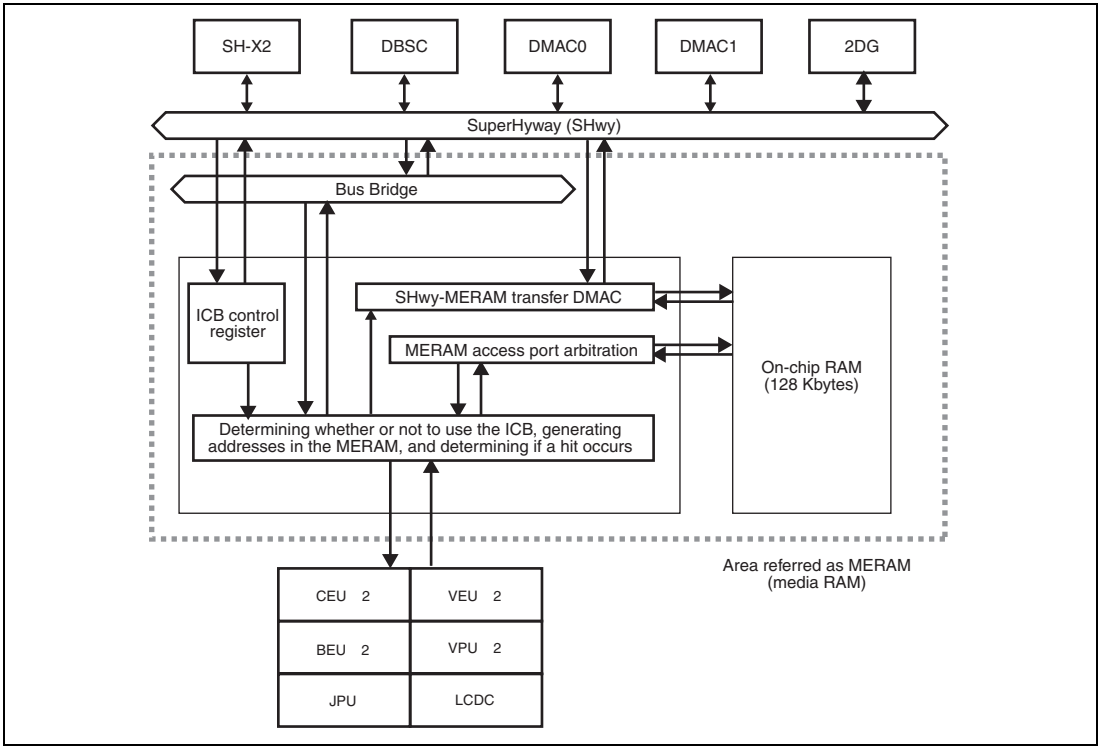


Figure 41.1 Relationship between MERAM and Other Modules

41.2 Register Descriptions

Table 41.1 shows the register configuration and memory configuration of the MERAM. Table 41.2 shows the register status in each processing mode.

Table 41.1 Register Configuration and Memory Configuration

Register Name	Abbreviation	R/W	Address	Access Size
ICB control register 0	MEVCR0	R/W	H'E800 0000	32
ICB control register 1	MEVCR1	R/W	H'E800 0004	32
ICB transfer end interrupt control register	METEIE	R/W	H'E800 0008	32
ICB transaction error interrupt control register	MEILIE	R/W	H'E800 000C	32
ICB active status register	MEACTST	R/W	H'E800 0010	32
ICB transfer end status register	METEST	R/W	H'E800 0014	32
ICB write transaction error status register	MEILWST	R/W	H'E800 0018	32
ICB read transaction error status register	MEILRST	R/W	H'E800 001C	32
ICB00 buffer control register	ME00CTRL	R/W	H'E800 0400	32
ICB00 frame size register	ME00BSIZE	R/W	H'E800 0404	32
ICB00 MERAM set register	ME00MCNF	R/W	H'E800 0408	32
ICB00 reserved	—	R	H'E800 040C	32
ICB00 external memory start address register A	ME00SSARA	R/W	H'E800 0410	32
ICB00 external memory start address register B	ME00SSARB	R/W	H'E800 0414	32
ICB00 external memory buffer size register	ME00SBSIZE	R/W	H'E800 0418	32
ICB00 reserved	—	R	H'E800 041C	32
ICB01 ****	ME01****	R/W	H'E800 0420 to H'E800 043C	32
ICB02 ****	ME02****	R/W	H'E800 0440 to H'E800 045C	32
ICB03 ****	ME03****	R/W	H'E800 0460 to H'E800 047C	32
ICB04 ****	ME04****	R/W	H'E800 0480 to H'E800 049C	32

Register Name	Abbreviation	R/W	Address	Access Size
ICB05 ****	ME05****	R/W	H'E800 04A0 to H'E800 04BC	32
ICB06 ****	ME06****	R/W	H'E800 04C0 to H'E800 04DC	32
ICB07 ****	ME07****	R/W	H'E800 04E0 to H'E800 04FC	32
ICB08 ****	ME08****	R/W	H'E800 0500 to H'E800 051C	32
ICB09 ****	ME09****	R/W	H'E800 0520 to H'E800 053C	32
ICB10 ****	ME10****	R/W	H'E800 0540 to H'E800 055C	32
ICB11 ****	ME11****	R/W	H'E800 0560 to H'E800 057C	32
ICB12 ****	ME12****	R/W	H'E800 0580 to H'E800 059C	32
ICB13 ****	ME13****	R/W	H'E800 05A0 to H'E800 05BC	32
ICB14 ****	ME14****	R/W	H'E800 05C0 to H'E800 05DC	32
ICB15 ****	ME15****	R/W	H'E800 05E0 to H'E800 05FC	32
ICB16 ****	ME16****	R/W	H'E800 0600 to H'E800 061C	32
ICB17 ****	ME17****	R/W	H'E800 0620 to H'E800 063C	32
ICB18 ****	ME18****	R/W	H'E800 0640 to H'E800 065C	32
ICB19 ****	ME19****	R/W	H'E800 0660 to H'E800 067C	32
ICB20 ****	ME20****	R/W	H'E800 0680 to H'E800 069C	32
ICB21 ****	ME21****	R/W	H'E800 06A0 to H'E800 06BC	32
ICB22 ****	ME22****	R/W	H'E800 06C0 to H'E800 06DC	32

Register Name	Abbreviation	R/W	Address	Access Size
ICB23 ****	ME23****	R/W	H'E800 06E0 to H'E800 06FC	32
ICB24 ****	ME24****	R/W	H'E800 0700 to H'E800 071C	32
ICB25 ****	ME25****	R/W	H'E800 0720 to H'E800 073C	32
ICB26 ****	ME26****	R/W	H'E800 0740 to H'E800 075C	32
ICB27 ****	ME27****	R/W	H'E800 0760 to H'E800 077C	32
ICB28 ****	ME28****	R/W	H'E800 0780 to H'E800 079C	32
ICB29 ****	ME29****	R/W	H'E800 07A0 to H'E800 07BC	32
ICB30 ****	ME30****	R/W	H'E800 07C0 to H'E800 07DC	32
ICB31****	ME31****	R/W	H'E800 07E0 to H'E800 07FC	32
MERAM direct access space	—	R/W	H'E808 0000 to H'E809 FFFF	8/16/32

Note: Any address from H'E800 0000 to H'E80F FFFF is also accessible as a mirror address. However, when making an access from the SH-X2 core, only the mirror address range is accessible.

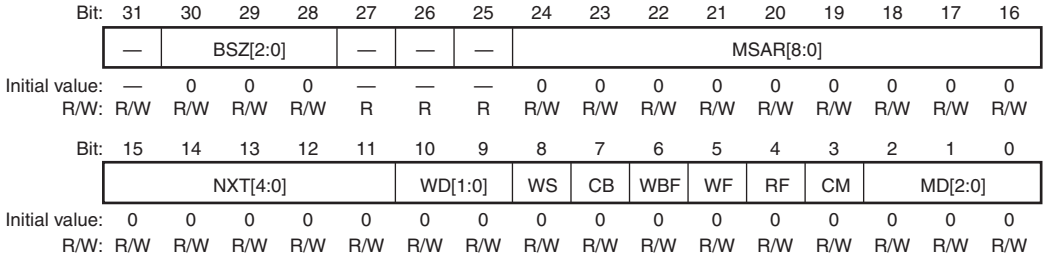
Table 41.2 Register Status in Each Processing Mode

Register Name	Power-On Reset	Manual Reset	Software Standby	Module Standby	R-Standby/ U-Standby	Sleep
MEVCR0	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEVCR1	Initialized	Initialized	Retained	Retained	Initialized	Retained
METEIE	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEILIE	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEACTST	Initialized	Initialized	Retained	Retained	Initialized	Retained
METEST	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEILWST	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEILRST	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEnnCNTRL	Initialized	Initialized	Retained	Retained	Initialized	Retained
MEnnBSIZE	Undefined	Undefined	Retained	Retained	Undefined	Retained
MEnnMCNF	Undefined	Undefined	Retained	Retained	Undefined	Retained
MEnnSSARA	Undefined	Undefined	Retained	Retained	Undefined	Retained
MEnnSSARB	Undefined	Undefined	Retained	Retained	Undefined	Retained
MEnnSBSIZE	Undefined	Undefined	Retained	Retained	Undefined	Retained

Note: nn = 00 to 31

41.2.1 ICBnn Control Register (MEnnCTRL)

MEnnCTRL (nn = 00 to 31) specifies the operating mode of ICBnn.



Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	BSZ[2:0]			—	—	—	MSAR[8:0]								
Initial value:	—	0	0	0	—	—	—	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXT[4:0]					WD[1:0]		WS	CB	WBF	WF	RF	CM	MD[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	Undefined	R/W	Reserved The write value should always be 0.
30 to 28	BSZ[2:0]	000	R/W	Indicate the number of lines that compose a block line. The 2^BSZ[2:0] line is treated as a block line. These bits specify a block line consisting of 1, 2, 4, 8, 16, 32, 64, or 128 lines.
27 to 25	—	Undefined	R	Reserved The write value should always be 0.
24 to 16	MSAR[8:0]	H'000	R/W	Specify the start address of a memory area reserved in the MERAM in units of Kbytes (0 to 511). Set a multiple of the number calculated by rounding up (MEnnBSIZE.XSZM1[15:0] + 1) into units of 2^n Kbytes (units of 1 Kbyte are also included). For the memory area size used in the MERAM, see section 41.2.3, ICBnn MERAM Set Register (MEnnMCNF).
15 to 11	NXT[4:0]	00000	R/W	Specify the next ICB number when configuring a frame buffer cache. Setting the current buffer number indicates that the buffer is the end buffer of the frame buffer cache. The value of these bits is used only when the MD[2:0] bits are B'111. For details, see section 41.4.4, Settings as Frame Buffer Cache.

Bit	Bit Name	Initial Value	R/W	Description
10, 9	WD[1:0]	00	R/W	Specify the write operation end detection method. The value of these bits is used only when the MD[2:0] bits are B'010, B'011, or B'100. 00: The write operation is regarded as completed when the associated module notifies the write operation end and the output address equals the end address. 01: The write operation is regarded as completed when the output address equals the end address. 10: The write operation is regarded as completed when the associated module notifies the write operation end. 11: The write operation end is not detected. When the set completion conditions are detected with these bits set to a value other than B'11, the WF bit in this register is set to 1 indicating the completion of data write to the last line in the ICB. At this time, when the WF bit is set to 1 with the MD[2:0] bits set to B'010 or B'011, data write back to the last line starts. In addition, setting the WF bit to 1 with the MD[2:0] bits set to B'011 or B'100 allows data reading up to the last line. Notes: 1. The end address means the 8-byte addresses including the pixels determined by the YSZM1 and XSZM1 bits in MEnnBSIZE. 2. The value of these bits depends on the characteristics or operation settings of associated modules. For details, see section 41.5, Settings for Each Associated Module.

Bit	Bit Name	Initial Value	R/W	Description
8	WS	0	R/W	Specifies whether to wait until the completion of a write transaction including the final pixel. The value of this bit is used only when the MD[2:0] bits are B'010, B'011, or B'100. Furthermore, this bit functions only for write transactions that meet the write operation end conditions specified by the WD[1:0] bits (other than B'11). 0: Upon receiving a write transaction that indicates a write operation end, the MERAM reflects the write transaction in the ICB and then suspends the operation of the module that has output the write transaction until the use of the ICB is completed. 1: The MERAM does not interrupt the operation of the module that has output the write transaction even if the write transaction indicates a write operation end. Note: The value of this bit depends on the characteristics or operation settings of associated modules. For details, see section 41.5, Settings for Each Associated Module.
7	CB	0	R/W	Indicates 0 while the ICB is using register set A, and indicates 1 while the ICB is using register set B. The write value should always be 0. Unless otherwise specified with a special application example, the value of this bit will not be used.
6	WBF	0	R/W	When the MD[2:0] bits are B'001 or B'111, this bit indicates that the data read fill operation in the ICB is completed. When the MD[2:0] bits are B'010 or B'011, this bit indicates that the write back operation of the ICB data is completed. The write value should always be 0. If 1 is written to this bit while the ICB is operating, the MERAM terminates the operation upon completion of the ongoing read fill or write back operation. Note: If 1 is written to this bit while the ICB is operating, the data written back to the SDRAM and the data filled into the ICB are not guaranteed. When writing 1 to this bit, also write 1 to the WF and RF bits simultaneously to quit using the buffer.

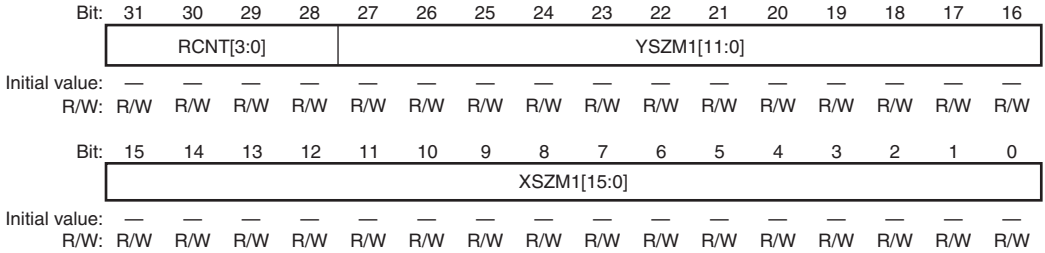
Bit	Bit Name	Initial Value	R/W	Description
5	WF	0	R/W	When the MD[2:0] bits are B'010, B'011, or B'100, this bit indicates that the data write operation in the ICB is completed. Upon detection of write operation end by the WD[1:0] bits, this bit is automatically set to 1. The write value should always be 0. If 1 is written to this bit while the ICB is operating, it is regarded that the completion of data write to the ICB has been notified. For modules that cannot automatically detect write operation end using the WD[1:0] bits, write 1 to this bit after receiving the operation completion of the module, and then notify manually that the data write to the ICB is completed. Setting this bit to 1 when the MD[2:0] bits are B'010 or B'011 starts write back operation of the data remaining in the ICB. Setting this bit to 1 when the MD[2:0] bits are B'011 or B'100 allows data reading up to the last line. When this bit is 1 and the following conditions are met, the use of the ICB is regarded as completed, the ACnn bit in MEACTST is negated, and the TEnn bit in METEST is set to 1. At the same time, the WBF, WF, and RF bits are cleared. [Conditions] When MD[2:0] = B'010: The WBF bit is 1 (write back completed). When MD[2:0] = B'011: The WBF bit is 1 (write back completed) and the RF bit is 1 (read operation completed). When MD[2:0] = B'100: The RF bit is 1 (read operation completed). Note: Even when writing is performed using an address with an offset (line N) as the start address, data for line 0 to line N is written back at completion of the write operation.

Bit	Bit Name	Initial Value	R/W	Description
4	RF	0	R/W	When the MD[2:0] bits are B'001, B'011, or B'100, this bit indicates that the data read operation in the ICB is completed. Upon detection of data read completion by the RCNT[3:0] bits in MEnnBSIZE, this bit is automatically set to 1. The write value should always be 0. If 1 is written while the ICB is operating, it is regarded that the completion of data read from the ICB has been notified. For modules that cannot automatically detect read operation end using the RCNT[3:0] bits in MEnnBSIZE, write 1 to this bit after receiving the operation completion of the module, and then notify that the data read from the ICB is completed. When this bit is 1 and the following conditions are met, the use of the ICB is regarded as completed, the ACnn bit in MEACTST is negated, and the TEnn bit in METEST is set to 1. At the same time the WBF, WF, and RF bits are cleared. [Conditions] When MD[2:0] = B'001: The WBF bit is 1 (read fill completed). When MD[2:0] = B'011: The WBF bit is 1 (write back completed) and the WF bit is 1 (write operation completed). When MD[2:0] = B'100: The WF bit is 1 (write operation completed). When MD[2:0] = B'111: The WBF bit is 1 (read fill completed).

Bit	Bit Name	Initial Value	R/W	Description
3	CM	0	R/W	Consecutive Address Mode Specify how addresses from associated modules are converted to line numbers and in-line offsets. For details, see section 41.3, Making Access to MERAM. 0: XSZM1 in MEnnBSIZE < 4096: Bits [11:0] of the address indicate the in-line offset, and the bits from bit [12] indicate the line number. 4096 <= XSZM1 in MEnnBSIZE: Bits [14:0] of the address indicate the in-line offset, and the bits from bit [15] indicate the line number. 1: XSZM1 in MEnnBSIZE < 1024: Bits [9:0] of the address indicate the in-line offset, and the bits from bit [10] indicate the line number. 1024 <= XSZM1 in MEnnBSIZE < 2048: Bits [10:0] of the address indicate the in-line offset, and the bits from bit [11] indicate the line number. 2048 <= XSZM1 in MEnnBSIZE < 4096: Bits [11:0] of the address indicate the in-line offset, and the bits from bit [12] indicate the line number. 4096 <= XSZM1 in MEnnBSIZE < 8192: Bits [12:0] of the address indicate the in-line offset, and the bits from bit [13] indicate the line number. Note: When the CM bit in MEnnCTRL is 1, set the XSZM1 bits in MEnnBSIZE to less than 8192.
2 to 0	MD[2:0]	000	R/W	Specify ICBnn usage mode. 000: No ICB is used. 001: Read buffer mode 010: Write buffer mode 011: Interconnect buffer (with write back) mode 100: Interconnect buffer (without write back) mode 111: Frame buffer cache mode

41.2.2 ICBnn Frame Size Register (MEnnBSIZE)

MEnnBSIZE (nn = 00 to 31) specifies the size of the frame buffer corresponding to ICBnn.



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	RCNT[3:0]	Undefined	R/W	Specify the read operation end detection method. When 0 is set, the read operation end is not detected. When a value “n” instead of 0 is set, the read operation is regarded as completed when the output address equals the end address for the n-th time. When the set read operation end conditions are detected with these bits set to a value other than 0, the RF bit in MEnnCNTRL is set to 1 indicating that the data read is completed. Notes: 1. The end address means the 8-byte addresses including the pixels determined by the YSZM1 and XSZM1 bits in MEnnBSIZE. 2. The value of these bits depends on the characteristics or operation settings of associated modules. For details, see section 41.5, Settings for Each Associated Module.
27 to 16	YSZM1 [11:0]	Undefined	R/W	Specify the number of lines − 1 (0 to 4095) of the frame buffer corresponding to ICBnn (nn = 0 to 31).
15 to 0	XSZM1 [15:0]	Undefined	R/W	Specify the actual data size − 1 in units of bytes (0 to 65535 bytes) in one line of the frame buffer corresponding to ICBnn (nn = 0 to 31). Note: When the CM bit in MEnnCTRL is 1, set the XSZM1 bits in MEnnBSIZE to less than 8192.

41.2.3 ICBnn MERAM Set Register (MEnnMCNF)

MEnnMCNF (nn = 00 to 31) is used for settings of buffer areas in the MERAM used by ICBnn.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	KWBNM[3:0]				KRBNM[3:0]				BNM[7:0]							
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	KWBNM [3:0]	Undefined	R/W	Specify the number of lines to be retained during writing when the MD[2:0] bits in MEnnCTRL are B'010, B'011, or B'100. Specify the number of lines in units of block lines specified by the BSZ[2:0] bits in MEnnCTRL, and the number of lines to be retained during writing is treated as (KWBNM[3:0] + 1) << BSZ[2:0] in MEnnCTRL. When the MERAM detects data write to line n, it determines that the data on the line (n – number of lines retained during writing) is fixed, and no more data writing to the line (n – number of lines retained during writing) is generated. Note: The value of these bits depends on the characteristics or operation settings of associated modules. For details, see section 41.5, Settings for Each Associated Module.

Bit	Bit Name	Initial Value	R/W	Description
27 to 24	KRBNM [3:0]	Undefined	R/W	Specify the number of lines to be retained during reading when the MD[2:0] bits in MEnnCTRL are B'001, B'011, B'100, or B'111. Specify the number of lines in units of block lines specified by the BSZ[2:0] bits in MEnnCTRL, and the number of lines to be retained during reading is treated as (KRBNM[3:0] + 1) << BSZ[2:0] in MEnnCTRL. When the MERAM detects data read on line n, it determines that the data on the line (n – number of lines retained during reading) can be released, and no more data reading on the line (n – number of lines retained during reading) is generated. Note: The value of these bits depends on the characteristics or operation settings of associated modules. For details, see section 41.5, Settings for Each Associated Module.
23 to 16	BNM[7:0]	Undefined	R/W	Specify the number of lines to be allocated in the MERAM. (Lines of the set number + 1 are allocated.) Specify the number of lines irrespective of the value of the BSZ[2:0] bits in MEnnCTRL. Allocate at least the following number of lines. MD = B'001: (Number of lines retained during reading + 1 block line) MD = B'010: (Number of lines retained during writing + 1 block line) MD = B'011, B'100: (Number of lines retained during writing + number of lines retained during reading + 1 block line) MD = B'111: See the following, (1) Calculating Memory Area Reserved in the MERAM.
15 to 0	—	Undefined	R	Reserved The write value should always be 0.

(1) Calculating Memory Area Reserved in the MERAM

When ICBnn is defined, the following amount of memory area which begins with the start address specified by the MSAR[8:0] bits in MEnnCTRL is allocated.

Memory size allocated (Kbytes) = (Number obtained by rounding up (MEnnBSIZE.XSZM1[15:0] + 1) to units of 2^n Kbytes (units of 1 Kbyte are also included)) × (MEnnMCNF.BNM[7:0] + 1)

Since no detection is performed with respect to whether the memory area overlaps with the memory area allocated by another ICB, write a value to the MSAR[8:0] bits to avoid such an overlap.

Note: Specify numbers so that the total number of lines that are allocated in the MERAM, retained during reading, and retained during writing is less than 256.

41.2.4 ICBnn External Memory Start Address Register A (MEnnSSARA)

MEnnSSARA (nn = 00 to 31) indicates the start address of the frame buffer (register set A) corresponding to ICBnn in the external memory. The value of the SSARA bits must be a multiple of 8.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSARA[31:16]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSARA[15:3]													—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	SSARA [31:3]	Undefined	R/W	Indicate the start address (in 8-byte units) of the frame buffer (register set A) corresponding to ICBnn (nn = 0 to 31) in the external memory.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

41.2.5 ICBnn External Memory Start Address B (MEnnSSARB)

MEnnSSARB (nn = 00 to 31) indicates the start address of the frame buffer (register set B) corresponding to ICBnn in the external memory. The value of the SSARB bits must be a multiple of 8.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SSARB[31:16]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSARB[15:3]													—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	SSARB [31:3]	Undefined	R/W	Indicate the start address (in 8-byte units) of the frame buffer (register set B) corresponding to ICBnn (nn = 0 to 31) in the external memory.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

41.2.6 ICBnn External Memory Buffer Size Register (MEnnSBSIZE)

MEnnSBSIZE (nn = 00 to 31) indicates the one-line width of the frame buffer corresponding to ICBnn in the external memory in units of bytes. The one-line width must be a multiple of 8.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SBSIZE[15:3]													—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 3	SBSIZE [15:3]	Undefined	R/W	Indicate the one-line width (in 8-byte units) of the frame buffer.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

41.2.7 ICB Control Register 0 (MEVCR0)

MEVCR0 is a register for future expansion. Data write to this register is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	—	All 0	R	Reserved Data write to this register is prohibited.

41.2.8 ICB Control Register 1 (MEVCR1)

MEVCR1 initializes ICBs and controls direct writing to the flag registers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RST	WD	AMD	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	RST	0	R/W	ICB Reset When this bit is set to 1, all ICBs are initialized. This bit is automatically cleared after initialization has completed. Reset all the associated modules using an ICB, and then set this bit to 1. Use this bit when an error occurs in the software program that uses an ICB.
30	WD	0	R/W	Write Direction When writing to MEACTST, METEST, MEILWST, or MEILRST, the bits to which 1 has been written are cleared usually. But when this bit is set to 1, the bits to which 1 has been written are set.
29	AMD	0	R/W	Address Mode Switch 0: The ICB function is used when the addresses from associated modules are within H'80000000 to H'BFFFFFFF. 1: The ICB function is used when the addresses from associated modules are within H'C0000000 to H'DFFFFFFF. For details, see section 41.3, Making Access to MERAM.
28 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

41.2.9 ICB Transfer End Interrupt Set Register (METEIE)

METEIE controls interrupt requests made by the transfer end flag (TEnn bits in METEST) of each ICB (nn = 0 to 31).

For modules that cannot automatically detect a write operation, when the WF bit in MEnnCTRL is manually set to 1, the MERAM can wait until the write back operation ends using this interrupt function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TInn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TInn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TInn	H'0000 0000	R/W	When any of the TInn (nn = 0 to 31) bits is set to 1, an interrupt occurs while the ICB transfer end flag (TEnn) in METEST is 1. After an interrupt occurs, it can be canceled by clearing the TEnn flag or the TInn bits in this register.

41.2.10 ICB Transaction Error Interrupt Set Register (MEILIE)

MEILIE controls interrupt requests made by the error status flag (IWnn in MEILWST or IRnn in MEILRST) of each ICB (nn = 0 to 31). For error status conditions, see the descriptions on MEILWST and MEILRST.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IInn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IInn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IInn	H'0000 0000	R/W	When any of these bits (nn = 0 to 31) is set to 1 and the error end flag (IWnn or IRnn) of each ICB in MEILWST or MEILRST is 1, an interrupt occurs. An interrupt is generated by the flag on the write side or read side. After an interrupt occurs, it can be canceled by clearing the IWnn and IRnn flags or clearing the corresponding IInn bits in this register.

41.2.11 ICB Active Status Register (MEACTST)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ACnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ACnn	H'0000 0000	R/W	Indicate that ICBnn (nn = 0 to 31) is in use. When 1 is written to this register while the WD bit in MEVCR1 is 0, the bits to which 1 has been written are cleared to 0 and the bits to which 0 has been written remain unchanged. When 1 is written to this register while the WD bit in MEVCR1 is 1, the bits to which 1 has been written are set to 1 and the bits to which 0 has been written remain unchanged.

(1) State Transitions of ICBnn

The ACnn bits (nn = 0 to 31) indicate that ICBnn (nn = 0 to 31) is in use.

When a transaction occurs successfully from an associated module to ICBnn, the ACnn bit is set to 1 and the ICBnn becomes active.

The active ICBnn must be driven to inactive mode upon completion of processing for one plane. The completion of processing for one plane means the following.

- When MD = B'001 or B'111: The RF and WBF bits in MEnnCTRL are 1.

  The RF bit indicates that the read operation of the module using ICBnn is completed. WBF = 1 shows that the read fill operation for ICBnn is completed up to the last line.

- When MD = B'010: The WF and WBF bits in MEnnCTRL are 1.
  The WF bit indicates that the write operation of the module using ICBnn is completed. WBF = 1 shows that the write back of the data written to ICBnn to the external memory is completed up to the last line.
- When MD = B'011: The WF, WBF, and RF bits in MEnnCTRL are 1.
- When MD = B'100: The WF and RF bits in MEnnCTRL are 1.

When the processing for one plane of ICBnn is completed, the WBF, WF, and RF bits in MEnnCTRL are cleared and the corresponding bit (TEnn) in METEST is set to 1. At the same time, the corresponding bit (ACnn) in MEACTST is cleared and ICBnn enters inactive mode.

The WF and RF bits can be set by manually writing 1 to the WF and RF bits of all active ICBnn when the end interrupt of the associated module occurs. Do not set the WBF bit to 1 to avoid discontinuation of write back operation.

Furthermore, the function to automatically detect the completion of processing for one plane is available for each write and read operation in some cases. For details on the function, see the descriptions on the WD[1:0] bits in MEnnCTRL, the WS bit in MEnnCTRL, and the RCNT[3:0] bits in MEnnBSIZE, and also see section 41.5, Settings for Each Associated Module. When the auto-detection function is available, the WF and RF bits in MEnnCTRL are automatically set to 1 by the execution of the last transaction in the write or read operation, which does not require manual setting.

An ICBnn that returns to inactive mode can deal with the next-plane transaction sent from the associated module. If the next-plane transaction is generated while ICBnn is active, a transaction error occurs usually. But the processing can be continued in the following cases.

- When the register set A/register set B specification of a transaction to ICBnn differs from the register set A/register set B indication of the active ICBnn
- When the register set A/register set B specification of a transaction to ICBnn matches the register set A/register set B indication of the active ICBnn, but the transaction line number is 0, the transaction processing to the last line is already completed, and line 0 is not included in the lines to be retained during reading/writing

With these conditions, an occurrence of this transaction is regarded as the completion of use of ICBnn, and therefore the completion processing starts automatically. In the case of a read transaction, the MERAM sets the RF bit in one or two buffers that the module is using for the read operation, waits until the buffers become inactive, and then processes the generated transaction. In the case of a write transaction, the MERAM sets the WF bit in one or two buffers that the module is using for the write operation, waits until the buffers become inactive, and then processes the generated transaction.

Note: Two or more ICBs are available for reading in the BEU and for reading/writing in the 2D-DMAC. But the completion processing for an ICB used for another purpose may be performed upon occurrence of the next-plane transaction to an ICBnn. For this reason, this function is not available when using two or more ICBs for each read and write operation of the associated module.

When a transaction error is detected, the corresponding bits (IWnn and IRnn) in MEILWST and MEILRST are set to 1 and the corresponding bit (ACnn) in MEACTST is negated. For transaction errors, see the descriptions on MEILWST and MEILRST.

(2) Writing to Register for Active ICB

When writing data to MEACTST, the AC bits corresponding to the bits to which 1 has been written are usually negated. When the WD bit in MEVCR1 is 1, the AC bits corresponding to the bits to which 1 has been written are set. Unless otherwise specified with a special application example, the AC bits need not be controlled by writing to this register.

While ACnn is 1, writing to the set register of an ICBnn is performed as follows:

- During the MEnnCTRL write operation, only writing 1 to the WBF, WF, and RF bits is effective and the other bits remain unchanged. When 0 is written to all the WBF, WF, and RF bits, every bit remains unchanged, but the set register of ICBnn is reevaluated, the buffers are restarted, and the read fill operation is started in advance.
  For restarting the buffers, see the description on clearing the error flag in MEILRST and restarting. For starting the read fill operation in advance of this, see (3) Starting Read Fill Operation in Advance.
- Writing to other set registers of ICBnn is possible, but only the register on the opposite plane from the register (MEnnSSARA or MEnnSSARB) specified by the CB bit in MEnnCTRL can be modified. The other registers should not be modified.

(3) Starting Read Fill Operation in Advance

When the MD[2:0] bits in MEnnCTRL are B'001, the read fill operation to the MERAM is started after a read transaction is generated from a module to ICBnn and then the ACnn bit in MEACTST is set to 1. Starting the read fill operation before a read transaction from a module to ICBnn occurs is made possible by the following procedure.

1. Set ICBnn.
2. Set the ACnn bits in MEACTST manually with the WD bit in MEVCR1 set to 1.
3. Write 0 to all bits in MEnnCTRL.

Since the ACnn bit in MEACTST is 1, no data is written to MEnnCTRL, but the read fill operation is started by reevaluating the ICBnn setting.

Note: When using ICBnn for the first time, write H'00000000 at addresses of MEnnCTRL + H'0C and MEnnCTRL + H'1C in step 1.

41.2.12 ICB Transfer End Status Register (METEST)

The TEnn bit (nn = 0 to 31) in this register indicates that the processing for one plane of ICBnn (nn = 0 to 31) is completed.

For the definition of the completion of processing for one plane of ICBnn, see the descriptions on the WF and RF bits in MEnnCTRL.

In writing to METEST, the TE bits corresponding to the bits to which 1 has been written are usually negated. When the WD bit in MEVCR1 is 1, the TE bits corresponding to the bits to which 1 has been written are set.

When the interrupt function of METEIE is used and an interrupt occurs, clear the bit that generated the interrupt. When the interrupt function is not used, the TEnn bit needs not be cleared.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TEnn	H'0000 0000	R/W	Indicate that the processing for one plane of ICBnn (nn = 0 to 31) is completed. The bits in this register to which 1 has been written are cleared.

41.2.13 ICB Write Transaction Error Status Register (MEILWST)

The IWnn bit (nn = 0 to 31) indicates that an unprocessable write transaction to ICBnn (nn = 0 to 31) has occurred. When this bit is set to 1, the ACnn bit in MEACTST is negated. A write transaction error occurs under the following conditions.

Conditions for write transaction error:

- When a write transaction occurs to ICBnn for which the MD[2:0] bits in MEnnCTRL are set to B'000
- When a write transaction occurs to ICBnn in which a read transaction error or write transaction error has already occurred (IRnn bit in MEILRST or IWnn bit in MEILWST is 1)
- When a write transaction occurs to ICBnn in the state where the ACnn bit in MEACTST of an active ICBnn is negated by writing to MEACTST
- When a write transaction occurs to a plane different from the register set A/register set B indication (value of the CB bit in MEnnCTRL) of an active ICBnn (ACnn bit in MEACTST is 1). However, when a module uses the same ICBnn repeatedly, the mismatch of register set A/register set B means the completion of processing for the former plane, which is not an error.

In writing to MEILWST, the IW bits corresponding to the bits to which 1 has been written are usually negated. When the WD bit in MEVCR1 is 1, the IW bits corresponding to the bits to which 1 has been written are set.

When the interrupt function of MEILIE is used and an interrupt occurs, clear the bit that generated the interrupt. Although the interrupt function is not used, no transaction to ICBnn for which the IWnn bit is set to 1 is processed after that. Therefore, the IWnn bit needs to be cleared. See section 41.2.14 (1), Clearing the Error Flag and Restarting.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IWnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IWnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IWnn	H'0000 0000	R/W	Indicate that an error occurred during a write transaction to ICBnn (nn = 0 to 31). The bits to which 1 has been written are cleared.

41.2.14 ICB Read Transaction Error Status Register (MEILRST)

The IRnn bit (nn = 0 to 31) indicates that an unprocessable read transaction to ICBnn (nn = 0 to 31) has occurred. When this bit is set to 1, the ACnn bit in MEACTST is negated. A read transaction error occurs under the following conditions.

Conditions for read transaction error:

- When a read transaction occurs to ICBnn for which the MD[2:0] bits in MEnnCTRL are set to B'000
- When a read transaction occurs to ICBnn in which a write transaction error or read transaction error has already occurred (IRnn bit in MEILRST or IWnn bit in MEILWST is 1)
- When a read transaction occurs to ICBnn in the state where the ACnn bit in MEACTST of an active ICBnn is negated by writing to MEACTST
- When a read transaction occurs to a plane different from the register set A/register set B indication (value of the CB bit in MEnnCTRL) of an active ICBnn (ACnn bit in MEACTST is 1). However, when a module uses the same ICBnn repeatedly, the mismatch of register set A/register set B means the completion of processing for the former plane, which is not an error.

In writing to MEILRST, the IR bits corresponding to the bits to which 1 has been written are usually negated. When the WD bit in MEVCR1 is 1, the IR bits corresponding to the bits to which 1 has been written are set.

When the interrupt function of MEILIE is used and an interrupt occurs, clear the bit that generated the interrupt. Although the interrupt function is not used, no transaction to ICBnn for which the IRnn bit is set to 1 is processed after that. Therefore, the IRnn bit needs to be cleared. See section 41.2.14 (1), Clearing the Error Flag and Restarting.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IRnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRnn															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	IRnn	H'0000 0000	R/W	Indicate that an error occurred during a read transaction to ICBnn (nn = 0 to 31). The bits to which 1 has been written are cleared.

(1) Clearing the Error Flag and Restarting

A transaction where the IWnn bit in MEILWST or the IRnn bit in MEILRST has been set to 1 remains in the ICB module and is not cleared until the transaction is processed or all ICBs are initialized. Therefore, the error module and all ICBs must be initialized except for the special case where the property of the transaction that has set the error flag can be anticipated.

1. If there is any module that uses an ICB other than ICBnn, change the register setting of the module so as not to use any ICB temporarily.
2. Issue the software reset command to all modules that use ICBnn and may cause an error.
3. Set the RST bit in MEVCR1 to 1 to reset all ICBs.

When the property of the transaction that has set the error flag can be anticipated and the transaction can be completed, the following procedure is available.

[Example] In the case of an error due to issuing of a transaction with the MD[2:0] bits in MEnnCTRL set to 000:

1. Set ICBnn so that the transaction can be handled with the IWnn flag in MEILWST or the IRnn flag in MEILRST set to 1, and then write a value other than B'000 to the MD[2:0] bits.
2. Clear the IWnn bit in MEILWST and the IRnn bit in MEILRST.
3. Set the WD bit in MEVCR1 to 1, and then set the corresponding ACnn bit in MEACTST to 1.
4. Write 0 to all bits in MEnnCTRL. Since the ACnn bit is 1, no data writing is made actually, but the ICBnn setting is reevaluated by writing to MEnnCTRL, and the remaining transaction is processed.

41.3 Making Access to MERAM

41.3.1 Direct Access from SuperHyway Initiator

The 128-Kbyte space of the application domain addresses H'E808 0000 to H'E809 FFFF is directly accessible to the MERAM memory. However, accesses are made irrespective of the ICB active status. Therefore, make accesses only to areas that are not used as ICBs.

41.3.2 Direct Access from Associated Image Module

The addresses H'0000 0000 to H'7FFF FFFF and H'C000 0000 to H'FFFF FFFF from an associated module (two CEUs, two VEUs, two BEUs, VPU, LCDC, or JPU) are output on the SHwy. Therefore, when making a direct access from an associated module to the MERAM, specify the 128-Kbyte space of addresses H'E808 0000 to H'E809 FFFF. However, accesses are made irrespective of the ICB active status. Therefore, make accesses only to areas that are not used as ICBs.

41.3.3 Access to ICB from Associated Image Module

When the AMD bit in MEVCR1 is 0, accesses from an associated module (two CEUs, two VEUs, two BEUs, VPU, LCDC, or JPU) to addresses H'8000 0000 to H'BFFF FFFF are treated as accesses to ICBs.

When the AMD bit in MEVCR1 is 1, accesses from an associated module to addresses H'C000 0000 to H'DFFF FFFF are treated as accesses to ICBs.

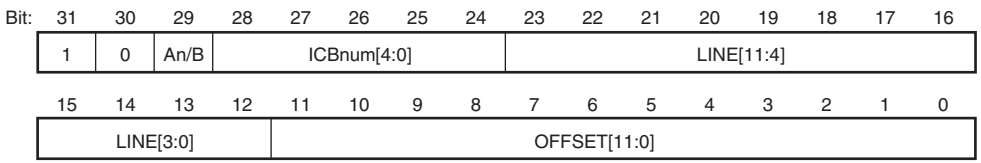
The ICB is used by interpreting the address from an associated image module as follows:

Bit field meaning:

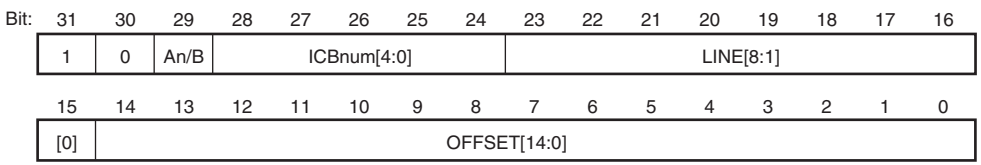
- An/B — Indicates register set A by 0 and register set B by 1.
- ICBnum[4:0] — Indicates the ICB buffer number.
- LINE[n:0] — Indicates the line number.
- OFFSET[m:0] — Indicates the in-line offset (in 1-byte units).

(a) AMD in MEVCR1 = 0, CM in MEnnCTRL = 0, Associated Image Module is Not VPU

- **(a-1) XSZM1 in MEnnBSIZE < 4096**

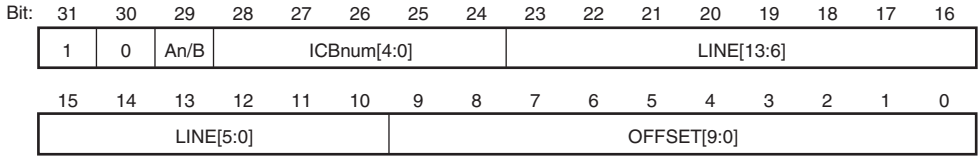


- **(a-2) 4096 <= XSZM1 in MEnnBSIZE**

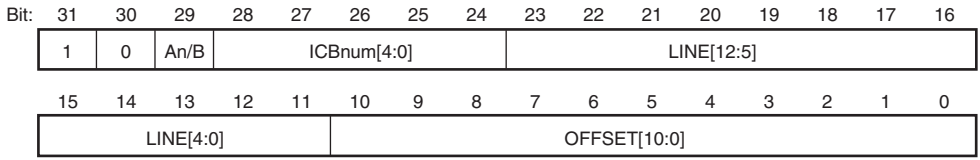


(b) AMD in MEVCR1 = 0, CM in MEnnCTRL = 1, Associated Image Module is Not VPU

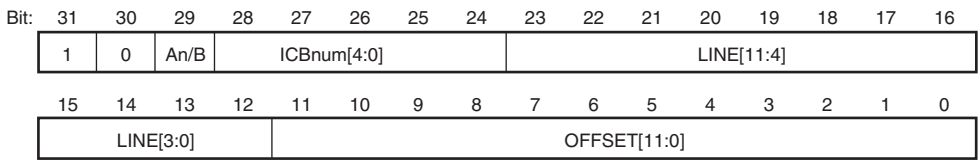
- **(b-1) XSZM1 in MEnnBSIZE < 1024**



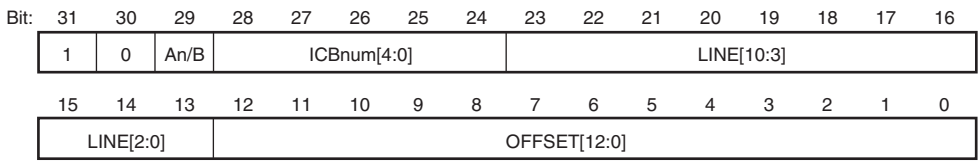
- **(b-2) 1024 <= XSZM1 in MEnnBSIZE < 2048**



- **(b-3) 2048 <= XSZM1 in MEnnBSIZE < 4096**



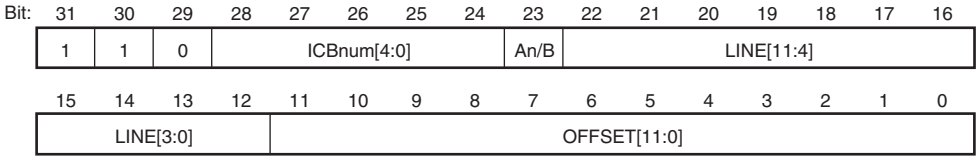
(b-4) 4096 <= XSZM1 in MEnnBSIZE < 8192



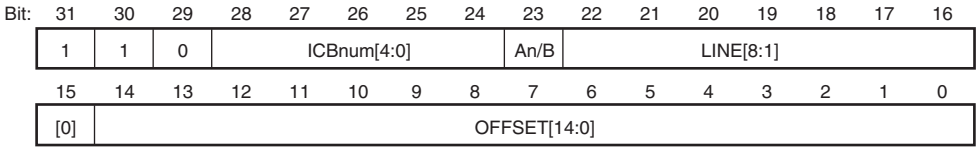
Note: When the CM bit in MEnnCTRL is 1, set the XSZM1 bits in MEnnBSIZE to less than 8192.

(c) AMD in MEVCR1 = 1, CM in MEnnCTRL = 0, Associated Image Module is Not VPU

- **(c-1) XSZM1 in MEnnBSIZE < 4096**

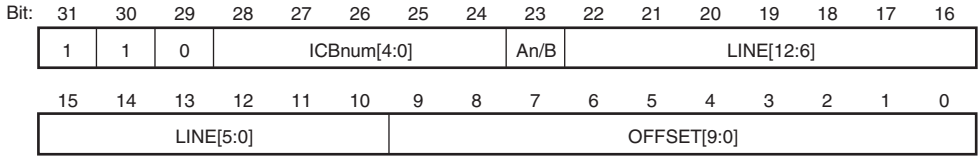


(c-2) 4096 <= XSZM1 in MEnnBSIZE

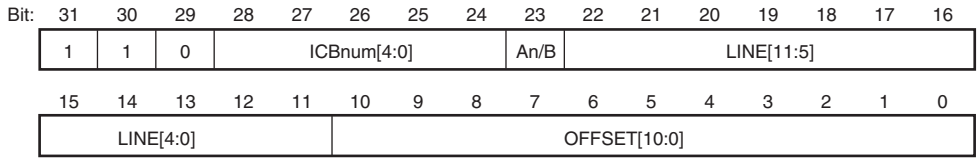


(d) AMD in MEVCR1 = 0, CM in MEnnCTRL = 1, Associated Image Module is Not VPU

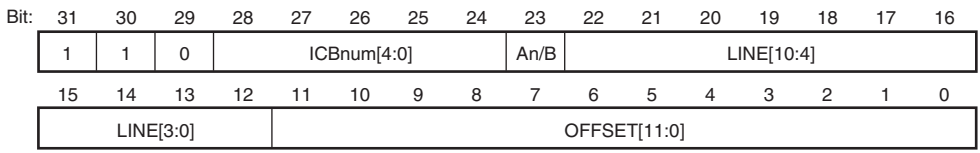
- **(d-1) XSZM1 in MEnnBSIZE < 1024**



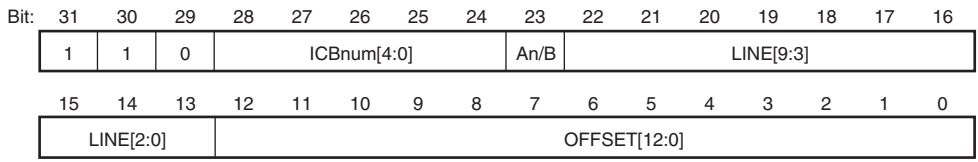
- **(d-2) 1024 <= XSZM1 in MEnnBSIZE < 2048**



- **(d-3) 2048 <= XSZM1 in MEnnBSIZE < 4096**



- **(d-4) 4096 <= XSZM1 in MEnnBSIZE < 8192**



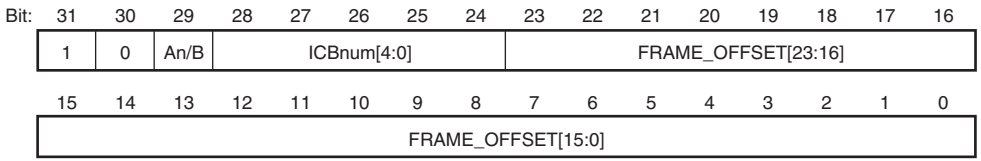
Note: When the CM bit in MEnnCTRL is 1, set the XSZM1 bits in MEnnBSIZE to less than 8192.

(e) When Associated Image Module is VPU

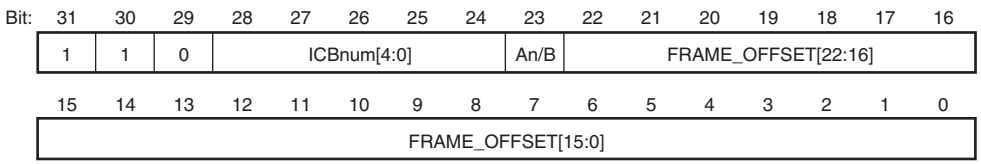
Since there is no line width set register for the VPU, the address outputs from the VPU cannot be generated separately into line numbers and in-line offsets. For the VPU, therefore, line numbers and in-line offsets are separated by using the value set in the XSZM1 bits in MEnnBSIZE. The CM bit in MEnnCTRL has no meaning in this process.

When the AMD bit in MEVCR1 is 0, bits [23:0] in an address are treated as the offset from the start frame. When the AMD bit in MEVCR1 is 1, bits [22:0] in an address are treated as the offset from the start frame. Register set A/B and ICB numbers are specified in the same way as for other modules.

- **(e-1) AMD in MEVCR1 = 0, associated image module is VPU**



(e-2) AMD in MEVCR1 = 1, associated image module is VPU



41.3.4 Additional Remarks on Line Numbers and Offset Addresses

The descriptions in (a) to (e) above are about how the output addresses from an image IP which is not the VPU are interpreted to be used for the ICB function. Addresses are generated in the image IP in the following two methods:

(1) Adjusting Stride Length of Image Modules to Interpretation as ICB Function

The IPs other than the VPU have registers for setting the stride length. The addresses output from modules can be interpreted as the ICB function by setting H'1000 or H'8000 in these registers ((a-1)(a-2), (c-1)(c-2)). (Where appropriate, examples are provided in this chapter to assist the description.)

The actual data width and stride length are set in the XSZM1 bits in MEnnBSIZE and the SBSIZE bits in MEnnSBSIZE.

In this method, the volume of data transfers between the MERAM and external memory is equal to the amount of the screen data set in the XSZM1 bits in MEnnBSIZE and the SBSIZE bits in MEnnSBSIZE. Since a buffer area with a size of 1 Kbyte, 2 Kbytes, or 4 Kbytes, that is obtained by rounding up (MEnnBSIZE.XSZM1 + 1), is allocated for each line in the MERAM, some of those areas may be wasted depending on the value of the XSZM1 bits in MEnnBSIZE.

(2) Adjusting Stride Length of Image Modules to Values in External Memory

Set the same value for the stride length of an image module as in the external memory (hereafter called as W). This allows the addresses from the image IP to be generated in the form of ICB buffer identifiers and in-frame offsets.

An example below describes how to set values in this method when the CM bit in MEnnCTRL is set to 1, the XSZM1 bits in MEnnBSIZE are set to H'3FF, and the SBSIZE bits in MEnnSBSIZE are set to H'400. The ICB interprets the offset from the start of frame contained in the output address from the image module as a line in 1-Kbyte units to perform transfers with the external memory.

In this method, the source of the MERAM to SDRAM transfer includes invalid data areas (W minus H, where H is the actual data width) in the memory. This means that the larger the size of “W – H” is, the more volume of transfers is wasted. The buffer area in the MERAM is used effectively in 1-Kbyte units except for the invalid data area in it. Consequently, the buffer area in the MERAM can be smaller by using method (2) when handling image data in which “W – H” is small.

The following summarizes how to use the ICB in this method.

- Set CM in MEnnCTRL to 1 to specify the consecutive address mode.
- Set XSZM1 in MEnnBSIZE to H'3FF to specify the line width of ICBs as 1-Kbyte units.
- Set SBSIZE in MEnnSBSIZE to H'400 to specify the line width of ICBs as 1-Kbyte units.
- Set YSZM1 in MEnnBSIZE to “roundup((W × (V – 1) + H)/1KB) – 1” to include the 1 Kbyte that contains the last pixel, where V is the number of lines in the external memory and roundup(f) is the calculation to round up f to an integer.
- Specify the number of buffer lines to be allocated in the BNM bits in MEnnBSIZE in 1-Kbyte units.
- Set the values calculated as follows in the KRBNM bits in MEnnBSIZE and the KWBNM bits in MEnnBSIZE. Use n as the desired number of lines to be retained for each IP that is described in section 41.5, Settings for Each Associated Module.
  - Calculate the desired number of lines to be retained (N) that is converted in 1-Kbyte units, assuming “N = roundup(n × W/1KB + 1)”. However, calculation can be performed with n =0, thus becoming, N = 1, in case where the number of lines to be retained is specified as 1 and also the input image is not the C plane in the YUV420 format, excluding cases where VEU images are input.
  - When 16 >= N, assume BSZ in MEnnCTRL = 0 and Ns = N.
  - When 32 >= N > 16, assume BSZ in MEnnCTRL = 1 and Ns = roundup(N/2).
  - When 64 >= N > 32, assume BSZ in MEnnCTRL = 2 and Ns = roundup(N/4).
  - (Hereinafter the same applies. Set the BSZ value so that Ns <= 16 is satisfied.)
  - Set KRBNM in MEnnBSIZE and KWBNM in MEnnBSIZE to Ns – 1.

Note the following:

- In the last line, access made to the external memory reaches the area that is rounded up the last pixel of an image to a 1-Kbyte boundary. So, make sure to allocate areas in the external memory to include the range.
- The auto-detection function for detecting the completion of access described in section 41.5, Settings for Each Associated Module, cannot be used in this method. Notify the completion of access manually.
- For the cases where the associated image module is the VPU, see the additional remarks in section 41.5.6, Image Input of VPU.

41.4 Setting ICB

Note: This chapter is written in accordance with section 41.3.4 (1), Adjusting Stride Length of Image Modules to Interpretation as ICB Function. Also, refer to section 41.3.4 (2), Adjusting Stride Length of Image Modules to Values in External Memory, since it is also applicable.

41.4.1 Settings as Read Fill Buffer

The following example shows the setting when ICBs are used as a read fill buffer for read accesses to the VEU0.

- ICB00 setting (for Y data)

```
ME00CNTRL    =   H'0000 0701 ;  // Read fill mode, no completion detection
                                // Allocate the buffer area from H'E558 0000
ME00BSIZE    =   H'005F 007F ;  // Vertical 96 lines and horizontal 128 pixels
ME00MCNF     =   H'0F1F 0000 ;  // Number of lines retained during reading = 16, total
                                // number of buffer lines = 32
ME00SSARA    =   H'5800 0000 ;  // Start address of Y data in the external memory (register
                                // set A)
ME00SSARB    =   H'5900 0000 ;  // Start address of Y data in the external memory (register
                                // set B)
ME00SBSIZE   =   H'0000 0080 ;  // Screen width in the external memory (128 bytes)
```

- ICB01 setting (for C data)

```
ME01CNTRL    =   H'0020 0701 ;  // Read fill mode, no completion detection
                                // Allocate the buffer area from H'E558 0000
ME01BSIZE    =   H'005F 007F ;  // Vertical 96 lines and horizontal 128 pixels
ME01MCNF     =   H'0F1F 0000 ;  // Number of lines retained during reading = 16, total
                                // number of buffer lines = 32
ME01SSARA    =   H'5800 8000 ;  // Start address of C data in the external memory
                                // (register set A)
ME01SSARB    =   H'5900 8000 ;  // Start address of C data in the external memory
                                // (register set B)
ME01SBSIZE   =   H'0000 0080 ;  // Screen width in the external memory (128 bytes)
```

- VEU0 setting

```
VESSR_0  =   H'0060 0080 ;    // Vertical 96 lines and horizontal 128 pixels
VSAYR_0  =   H'8000 0000 ;    // Use ICB00 (register set A) for reading Y data
VSACR_0  =   H'8100 0000 ;    // Use ICB01 (register set A) for reading C data
VESWR_0  =   H'0000 1000 ;    // Set H'1000 as screen width when using ICBs
VRFSR_0  =   H'0060 0080 ;    // Clip the output. Vertical 96 lines and horizontal
                              // 128 pixels
VDAYR_0  =   H'5801 0000 ;    // Start address in the external memory to store Y data
                              // (register set A)
VDACR_0  =   H'5801 8000 ;    // Start address in the external memory to store C data
                              // (register set A)
VEDWR_0  =   H'0000 0080 ;    // Screen width in the external memory (128 bytes)
VEIER_0  =   H'0000 0001 ;    // Enable interrupts
```

(Other registers should be set according to the desired VEU0 functions.)

After setting these, start the VEU0 by setting VESTR_0 = H'0000 0001. The read access request from the VEU0 points to the addresses using ICB00 (register set A)/ICB01 (register set A), so the MERAM starts a read fill operation from the external memory address (H'5800 0000/H'5800 8000).

This read fill operation continues to be performed automatically until all read fill operations for the lines specified in YSZM1 in ME00BSIZE/YSZM1 in ME01BSIZE completes or the buffer areas for the buffer lines specified in BNM in ME00BSIZE/BNM in ME01BSIZE get full.

If the read access request from the VEU0 is to the lines that are already filled into the buffer area, the MERAM returns data to the VEU0. Otherwise, the request is suspended until the lines have been filled into the buffer area.

The VEU0 sometimes performs read fill operations on a per-block basis, so the target line number is not always incremented serially and the target line at up to 16 lines ago may be read again. For this reason, the numbers of lines retained during reading that are set in KRBNM in ME00MCNF should be more than 16.

If the MERAM returns the first n lines corresponding to an access request from the VEU0, the data in "n-(number of lines retained during reading)" lines is considered as unnecessary and discarded from the buffer area. The area is reused for another read fill operation.

When an interrupt from the VEU0 is detected, notify the MERAM of the completion of the read operation to ICB00/ICB01 by setting as follows:

```
ME00CNTRL    =   H'0000 0711 ;     // RF bit=1
ME01CNTRL    =   H'0020 0711 ;     // RF bit=1
```

When the RF bits are set, ICB00/ICB01 are freed and bits [1:0] in MEACTST return to 0. This enables the VEU0 to be set for the next operation or ICB00/ICB01 to be set for other modules.

When the start addresses of read operations for the VEU0 are set as follows, data is filled from H'5900 0000/H'5900 8000 in the external memory:

```
VSAYR_0  =   H'A000 0000 ;    // Use ICB00 (register set B) for reading Y data
VSACR_0  =   H'A100 0000 ;    // Use ICB01 (register set B) for reading C data
```

Combining with the setting of the VEU0 two-set register can reduce resetting of the MERAM or VEU0 registers.

41.4.2 Settings as Write Back Buffer

The following example shows the setting when ICBs are used as a write back buffer for write accesses to the VEU1.

- ICB02 setting (for Y data)

```
ME02CNTRL    =   H'0040 0702 ;  // Write back mode, no completion detection
                                // Allocate the buffer area from H'E559 0000
ME02BSIZE    =   H'005F 007F ;  // Vertical 96 lines and horizontal 128 pixels
ME02MCNF     =   H'F01F 0000 ;  // Number of lines retained during writing = 16, total
                                // number of buffer lines = 32
ME02SSARA    =   H'5802 0000 ;  // Start address in the external memory to store Y data
                                // (register set A)
ME02SSARB    =   H'5902 0000 ;  // Start address in the external memory to store Y data
                                // (register set B)
ME02SBSIZE   =   H'0000 0080 ;  // Screen width in the external memory (128 bytes)
```

- ICB03 setting (for C data)

```
ME03CNTRL    =   H'0060 0702 ;  // Write back mode, no completion detection
                                // Allocate the buffer area from H'E559 8000
ME03BSIZE    =   H'005F 007F ;  // Vertical 96 lines and horizontal 128 pixels
ME03MCNF     =   H'F01F 0000 ;  // Number of lines retained during writing = 16, total
                                // number of buffer lines = 32
ME03SSARA    =   H'5802 8000 ;  // Start address in the external memory to store C data
                                // (register set B)
ME03SSARB    =   H'5902 8000 ;  // Start address in the external memory to store C data
                                // (register set B)
```

```
ME03SBSIZE   =   H'0000 0080 ;  // Screen width in the external memory (128 bytes)
```

- VEU1 setting

```
VESSR_1      =   H'0060 0080 ;  // Vertical 96 lines and horizontal 128 pixels
VSAYR_1      =   H'5801 0000 ;  // Start address of Y data in the external memory
                                // (This is an example of setting when the output result
                                // from the VEU0 is to be processed further.)
VSACR_1      =   H'5801 8000 ;  // Start address of C data in the external memory
                                // (This is an example of setting when the output result
                                // from the VEU0 is to be processed further.)
VESWR_1      =   H'0000 0080 ;  // Screen width in the external memory (128 bytes)
VRFSR_1      =   H'0060 0080 ;  // Clip the output. Vertical 96 lines and horizontal 128
                                // pixels
VDAYR_1      =   H'8200 0000 ;  // Use ICB02 (register set A) for writing Y data
VDACR_1      =   H'8300 0000 ;  // Use ICB03 (register set A) for writing C data
VEDWR_1      =   H'0000 1000 ;  // Set H'1000 as screen width when using ICBs
VEIER_1      =   H'0000 0001 ;  // Enable interrupts
```

(Other registers should be set according to the desired VEU1 functions.)

After setting these, start the VEU1 by setting VESTR_1 = H'0000 0001. The write access request from the VEU1 points to the addresses using ICB02 (register set A)/ICB03 (register set A), so the MERAM processes write data in a write operation to the buffer area allocated in the MERAM. This write data can be written until the buffer areas for the buffer lines specified in BNM in ME02BSIZE/BNM in ME03BSIZE get full.

The VEU1 sometimes performs write operations on a per-block basis, so the target line number is not always incremented serially and the target line at up to 16 lines ago may be written again. For this reason, the numbers of lines retained during writing that are set in KWBNM in ME02MCNF/KWBNM in ME03MCNF should be more than 16.

If the MERAM returns the first n lines corresponding to an access request from the VEU1, the data in "n-(number of lines retained during writing)" is considered to have been written and it is written back to the external memory. The area that contains data that has been written back is freed and reused for write data. Thus, processing the write operation from the VEU1 continues with reusing buffer areas until the last line specified in YSZM1 in ME02BSIZE/YSZM1 in ME03BSIZE is reached.

When an interrupt from the VEU1 is detected, notify the MERAM of the completion of the write operation to ICB02/ICB03 by setting as follows:

```
ME02CNTRL    =   H'0040 0722 ;  // WF bit=1
ME03CNTRL    =   H'0060 0722 ;  // WF bit=1
```

When the WF bits are set, ICB02/ICB03 are freed and bits [3:2] in MEACTST return to 0 after the MERAM writes back data that has not been done yet to the external memory. Note that the data on the lines to be retained during writing before the last line is not written back unless the WF bit is set. This enables the VEU1 to be set for the next operation or ICB02/ICB03 to be set for other modules.

When the start addresses of write operations for the VEU1 are set as follows, data is written back from H'5902 0000/H'5902 8000 in the external memory:

```
VSAYR_1  =   H'A200 0000 ;    // Use ICB02 (register set B) for writing Y data
VSACR_1  =   H'A300 0000 ;    // Use ICB03 (register set B) for writing C data
```

Combining with the setting of the VEU1 two-set register can reduce resetting of the MERAM or VEU1 registers.

41.4.3 Settings as Interconnect Buffer

The following example shows the setting when the VEU0 and VEU1 are interconnected via ICBs.

- ICB04 setting (for Y data)

```
ME04CNTRL    =   H'0080 0703 ;  // Interconnect mode with write back, no completion
                                // detection
                                // Allocate the buffer area from H'E55A 0000
ME04BSIZE    =   H'005F 007F ;  // Vertical 96 lines and horizontal 128 pixels
ME04MCNF     =   H'FF2F 0000 ;  // Number of lines retained during writing = 16, number of
                                // lines retained during reading = 16, total number of
                                // buffer lines = 32
ME04SSARA    =   H'5801 0000 ;  // Start address (register set A) in the external memory to
                                // store the Y data of the processing result from the VEU0
ME04SSARB    =   H'5901 0000 ;  // Start address in the external memory to store Y data
                                // (register set B)
ME04SBSIZE   =   H'0000 0080 ;  // Screen width of the processing result from the VEU0 in
                                // the memory (128 bytes)
```

- ICB05 setting (for C data)

```
ME05CNTRL    =   H'00A0 0703 ;  // Interconnect mode with write back, no completion
                                // detection
                                // Allocate the buffer area from H'E55A 8000
ME05BSIZE    =   H'005F 007F ;  // Vertical 96 lines and horizontal 128 pixels
ME05MCNF     =   H'FF2F 0000 ;  // Number of lines retained during writing = 16, number of
                                // lines retained during reading = 16, total number of
                                // buffer lines=48
ME05SSARA    =   H'5803 8000 ;  // Start address (register set A) in the external memory to
                                // store the C data of the processing result from the VEU0
ME05SSARB    =   H'5903 8000 ;  // Start address in the external memory to store C data
                                // (register set B)
ME05SBSIZE   =   H'0000 0080 ;  // Screen width of the processing result from the VEU0 in
                                // the memory (128 bytes)
```

- VEU0 setting (differences only)

```
VDAYR_0      =   H'8400 0000 ;  // Use ICB04 (register set A) for writing Y data
VDACR_0      =   H'8500 0000 ;  // Use ICB05 (register set A) for writing C data
VEDWR_0      =   H'0000 1000 ;  // Set H'1000 as screen width when using ICBs
```

- VEU1 setting (differences only)

```
VSAYR_1      =   H'8400 0000 ;  // Use ICB04 (register set A) for reading Y data
VSACR_1      =   H'8500 0000 ;  // Use ICB05 (register set A) for reading C data
VESWR_1      =   H'0000 1000 ;  // Set H'1000 as screen width when using ICBs
```

After setting these, start the VEU0 and VEU1 by setting VESTR_0 = H'00000001 and VESTR_1 = H'00000001. With these settings, the output data from the VEU0 is input to the VEU1 via buffer areas in the MERAM. The output data from the VEU0 is also written back to the external memory.

When MD in ME04CNTRL/MD in ME05CNTRL are set to B'100, the output data from the VEU0 is stored in the MERAM only temporarily and no access is made to the external memory.

The VEU0 completes its operation and an interrupt is generated. But at this point, the data from the VEU0 that is on the lines to be retained during writing before the last line is considered not to be fixed, so the operation for the VEU1 is not complete. When an interrupt from the VEU0 is detected, notify the MERAM of the completion of the write operation to ICB04/ICB05 by setting as follows:

```
ME04CNTRL    =   H'0080 0723 ;  // WF bit=1
ME05CNTRL    =   H'00A0 0723 ;  // WF bit=1
```

When the WF bits are set, the MERAM determines that all data is fixed and the VEU0 starts its remaining operation to write back the data that has not been done yet to the external memory. When an interrupt is generated from the VEU0, notify the MERAM of the completion of the read operation to ICB04/ICB05 by setting as follows:

```
ME04CNTRL    =   H'0080 0713 ;  // RF bit=1
ME05CNTRL    =   H'00A0 0713 ;  // RF bit=1
```

If the RF bits are set to 1 and the write back to the external memory is complete, ICB04/ICB05 are freed and bits [5:4] in MEACTST return to 0. This enables the VEU0 and VEU1 to be set for the next operation or ICB04/ICB05 to be set for other modules.

It is distinguished whether the output result from the VEU0 is written back to the register set A or register set B of ICB04/ICB05, but the same register set should be selected for the VEU0/VEU1.

41.4.4 Settings as Frame Buffer Cache

ICBs can be used as a frame buffer cache by combining two ICBs. One is called marker buffer, and the other is called cache buffer.

- Marker buffer

  Only ICB28 to ICB31 buffers can be used as the marker buffer. In other words, when the MD bits in MEmmCTRL (mm = 28 to 31 in this case) are B'111 for any of ICB28 to ICB31, such an ICB is treated as a marker buffer. Set the corresponding cache buffer number to the NXT[4:0] bits in MEmmCTRL of the marker buffer.
- Cache buffer

  When the line number 0 access is made to the marker buffer, the NXT[4:0] bits in MEmmCTRL are referenced, and the set value (cc) is regarded that ICBcc is used.

  ICBcc functions like the read fill buffer. When the read fill operation for the buffer lines specified by the BNM[7:0] bits in MEccMCNT is completed, the control is shifted to the buffer specified by the NXT[4:0] bits in MEccCTRL so that the filled data remains in the MERAM memory without being overwritten. Specify the marker buffer number (mm) again in the NXT[4:0] bits in MEccCTRL.

When processing shifts from ICBcc to ICBmm, the active bit (ACcc in MEACTST) of ICBcc remains 1. When the ACcc bit in MEACTST is 1 at an access to the cache buffer and the register set A/register set B indication by the CB bit in MEccCTRL matches the register set A/register set B shown at the address accessed, it is regarded that data is already contained in ICBcc, and therefore the read fill operation is not performed. If the register set A/register set B indication does not match or when the ACcc bit in MEACTST is 0, it is regarded that the data is updated or the access is the first one, and therefore the ACTcc bit is set, the CB bit is modified, and the read fill operation is performed again.

If the control is returned to the marker buffer for other than line number 0, the marker buffer continues operation as a read fill buffer. The marker buffer keeps operating to the last line while reusing the MERAM buffers in the same way as the normal read fill buffer. Allocate at least two lines for the marker buffer to function as a read fill buffer.

When all display data for a single frame is stored in the ICB cache buffer, the display data becomes invalid. Even in a case where the screen size is small and all display data for a single frame can be stored in the cache buffer, divide the display data when storing it.

The following describes settings specific to frame buffer cache mode for the marker buffer and cache buffer.

- Marker buffer settings

MEmmCTRL:	Set the MD[2:0] bits to B'111, and the NXT[4:0] bits to cc. Specify the start address of the buffer in the MERAM to be used by the marker buffer in the MSAR[8:0] bits.
MEmmBSIZE:	Set “total number of lines of the frame buffer (all screens) – 1” in the YSZM1[11:0] bits. Set the RCNT[3:0] bits to 1 to enable detection of the completion of read operation. The completion can also be notified manually.
MEmmMCNF:	Set at least 1 (to allocate two lines) in the BNM[7:0] bits. Specify 0 or larger (to allocate one line or more) in the KRBNM[3:0] bits.
MEmmSSARA/MEmmSSARB:	Set the start address each for register set A and register set B in the external memory, which cannot be retained in the cache buffer.
MEmmSBSIZE:	Specifies the screen width in the external memory in units of bytes in the same way as other modes.

- Cache buffer settings

MEccCTRL:	Set the MD[2:0] bits to B'111, and the NXT[4:0] bits to mm. Specify the start address in units of Kbytes in the MERAM to be allocated as a cache area in the MSAR[8:0] bits.
MEccBSIZE:	Set “total number of lines of the frame buffer (all screens) – 1” in the YSZM1[11:0] bits. When a cache buffer size enough for the data of all screens is allocated, set the RCNT[3:0] bits to 1 to enable detection of the completion of read operation in the cache buffer.
MEccMCNF:	Set the number of lines that can be allocated in the MERAM in the BNM[7:0] bits. Specify 0 or larger (to allocate one line or more) in the KRBNM[3:0] bits.
MEccSSARA/MEccSSARB:	Set the start address each for register set A and register set B in the external memory.
MEccSBSIZE:	Specifies the screen width in the external memory in units of bytes in the same way as other modes.

To perform processing manually for one register set for the frame buffer cache, write 1 to the RF bit in MEmmCTRL of the marker buffer. At this time, the active state ACcc bit in MEACTST of ICBcc is not cleared. To stop using the entire frame buffer cache or to change the frame buffer cache setting, manually clear the ACcc bit in MEACTST.

- Supplement: Reusing data in the cache buffer

  When the frame buffer cache is used with the MERAM containing image data, such as when waking up from LCD refresh mode, the first read fill operation could be omitted by specifying the cache buffer to have already been filled. In this case, make the following settings in addition to the normal cache buffer settings.

```
MEccCTRL                ← Normal initial setting | H'0000 0040 ;// Set WBF bit
MEccCTRL + address H'0C ← H'0000 0000 ;                          //
MEccCTRL + address H'1C ← H'0000 0000 ;                          //
MEVCR1                  ← H'4000 0000 ;                          // Enter ACT bit set mode
MEACTST[cc]             ← B'1 ;                                  // Set cache buffer
                                                                 // active bit
MEVCR1                  ← H'0000 0000 ;                          // Return to ACT bit
                                                                 // clear mode
```

41.5 Settings for Each Associated Module

Note: This chapter is written in accordance with section 41.3.4 (1), Adjusting Stride Length of Image Modules to Interpretation as ICB Function. Also, refer to section 41.3.4 (2), Adjusting Stride Length of Image Modules to Values in External Memory, since it is also applicable.

41.5.1 Image Output of VEU

This LSI is provided with two VEU modules that use the ICB function in the same way. The notation of “VEUn” which means the n-th VEU is used in the following.

(1) Registers That Can Use ICBs and Conditions

VDAYR and VDACR in VEUn can be set for using ICBs. Also set VEDWR in VEUn to H'1000.

Notes: 1. Since VEDWR in VEUn is common to the Y image and CbCr image, a setting such that one of these images uses ICBs and the other uses no ICB is prohibited. Furthermore, when the YCbCr 4:4:4 format is used, BDWR cannot be used because the handling of the BDWR value differs between the Y image and CbCr image.
2. No ICB can be used for image output under settings including 90-degree turn, 270-degree turn, or vertical inversion. ICBs are available only for horizontal inversion.
3. Bundled mode is not supported.

(2) Conditions for the Number of Lines Retained

Set the following values as the number of lines retained during writing, depending on the processing in the VEU.

- Equal to or larger than 1 when the processing in the VEU satisfies the condition of “VESTR.VBE || (VENHR.ENH || VENHR.ENHV) || (VRFCR!=0)”.
- Equal to or larger than 16 when the above condition is not satisfied.

(3) Detecting Completion of Processing for One Side

The VEU is a module that notifies the writing to the final pixel when outputting an image. Therefore, a combination of settings “MEnnCTRL.WD[1:0] = B'10 and MEnnCTRL.WS = B'0” is recommended.

Note: It is also possible to set the WD[1:0] bits to B'11 and the WS bit to B'1, which does not require waiting for the completion of write back to the external memory or the completion of reading from another module in the ICB. In this case, write 1 to the WF and RF bits in MEnnCTRL for all ICBnn used by VEUn upon receiving the end interrupt from VEUn, notify the completion of the use of all ICBnn, and then confirm the transfer end of all ICBnn using the transfer end interrupt from the ICB or other means.

41.5.2 Image Input of VEU

(1) Registers That Can Use ICBs and Conditions

VSAYR and VSACR in VEUn can be set for using ICBs. Also set VESWR in VEUn to H'1000.

Notes: 1. Since VESWR in VEUn is common to the Y image and CbCr image, a setting such that one of these images uses ICBs and the other uses no ICB is prohibited. Furthermore, when the YCbCr 4:4:4 format is used, BDWR cannot be used because the handling of the BDWR value differs between the Y image and CbCr image.
2. ICBs can be used for image input under settings including 90-degree turn, 270-degree turn, or vertical inversion.
3. Bundled mode is not supported.

(2) Conditions for the Number of Lines Retained

Set the following values as the number of lines retained during reading, depending on the processing in the VEU.

- Equal to or larger than 1 when the processing in the VEU satisfies the condition of “VESTR.VBE || (VENHR.ENHH || VENHR.ENHV) || (VRFCR!=0)”.
- When the above condition is not satisfied:
  - Equal to or larger than 16 if there is no LPF filter. However, if there is an MED filter, equal to or larger than 18.
  - Equal to or larger than 18 if there is a 3-tap LPF filter. However, if there is an MED filter, equal to or larger than 20.
  - Equal to or larger than 20 if there is a 5-tap LPF filter. However, if there is an MED filter, equal to or larger than 22.

(3) Detecting Completion of Processing for One Side

Since read access patterns of the VEU may become complex depending on usage conditions, it is difficult for the VEU to automatically detect the completion of read access for one register set. The method to inform the ICB of the read access end by manually writing 1 to the RF bit in MEnnCTRL upon completion of the VEU operation is recommended.

41.5.3 Image Output of BEU

This LSI is provided with two BEU modules that have the same ICB-related functions. The notation of “BEUn” which means the n-th BEU is used in the following.

(1) Registers That Can Use ICBs and Conditions

BDAYR and BDACR in BEUn can be set for using ICBs. Also set BDWR in BEUn to H'1000.

Note: Since BDWR in BEUn is common to the Y image and CbCr image, a setting such that one of these images uses ICBs and the other uses no ICB is prohibited. Furthermore, when the YCbCr 4:4:4 format is used, BDWR cannot be used because the handling of the BDWR value differs between the Y image and CbCr image.

(2) Conditions for the Number of Lines Retained

Set the number of lines retained during writing, which is equal to or larger than 1.

(3) Detecting Completion of Processing for One Side

The BEU is a module that notifies the writing to the final pixel when outputting an image. Therefore, a combination of settings “MEnnCTRL.WD[1:0] = B'10 and MEnnCTRL.WS = B'0”is recommended.

Note: It is also possible to set the WD[1:0] bits to B'11 and the WS bit to B'1, which does not require waiting for the completion of write back to the external memory or the completion of reading from another module in the ICB. In this case, write 1 to the WF and RF bits in MEnnCTRL for all ICBnn used by BEUn upon receiving the end interrupt from BEUn, notify the completion of the use of all ICBnn, and then confirm the transfer end of all ICBnn using the transfer end interrupt from the ICB or other means.

41.5.4 Image Input of BEU

(1) Registers That Can Use ICBs and Conditions

BSAYR1, BSACR1, BSAYR2, BSACR2, BSAYR3, and BSACR3 in BEUn can be set for using ICBs. Also set BSWR1, BSWR2, and BSWR3 in BEUn to H'1000.

Notes: 1. Since BSWR1 to BSWR3 in BEUn are common to the Y image and CbCr image, a setting such that one of these images uses ICBs and the other uses no ICB is prohibited. Furthermore, when the YCbCr 4:4:4 format is used, BDWR cannot be used because the handling of the BDWR value differs between the Y image and CbCr image.
2. The multiscreen function is not supported.
3. When ICBs are used for the BEU input, use ICBs for the entire input system.

(2) Conditions for the Number of Lines Retained

Set the number of lines retained during reading, which is equal to or larger than 1.

(3) Detecting Completion of Processing for One Side

Since read access patterns of the BEU may become complex depending on usage conditions, it is difficult for the VEU to automatically detect the completion of read access for one side. The method to inform the ICB of the read access end by manually writing 1 to the RF bit in MEnnCTRL upon completion of the BEU operation is recommended.

41.5.5 Image Output of VPU

(1) Registers That Can Use ICBs and Conditions

For using ICBs, VP4_DWY_ADDR, VP4_DWC_ADDR, VP4_D2WY_ADDR, and VP4_D2WC_ADDR in the VPU can be set. A setting such that one of VP4_DWY_ADDR and VP4_DWC_ADDR uses ICBs and the other uses no ICB is prohibited. A setting such that one of VP4_D2WY_ADDR and VP4_D2WC_ADDR uses ICBs and the other uses no ICB is prohibited.

(2) Conditions for the Number of Lines Retained

Set the following values as the number of lines retained during writing, in accordance with the VPU output registers.

- Buffer corresponding to VP4_DWY_ADDR in the VPU: Equal to or larger than 16
- Buffer corresponding to VP4_DWC_ADDR in the VPU: Equal to or larger than 8
- Buffer corresponding to VP4_D2WY_ADDR in the VPU: Equal to or larger than 20
- Buffer corresponding to VP4_D2WC_ADDR in the VPU: Equal to or larger than 12

Note: Since the output image of the VPU is in the YUV420 format, the number of lines of the CbCr plane is indicated as half of the Y plane. Note however that the buffer corresponding to VP4_D2WC_ADDR needs at least 12 lines as the number of lines retained during writing.

(3) Detecting Completion of Processing for One Side

The VPU module notifies the completion of write operation at each output end of each Y plane and C plane of one macro block. To detect the completion of processing for one screen, the address comparison is also required.

When the ICB for the Y plane is defined as ICByy, and the ICB for the C plane is defined as ICBcc, set the following:

WD[1:0] bits in MEyyCTRL = B'11 (not detecting), WS bit in MEyyCTRL = 1 (not stalling), WD[1:0] bits in MEccCTRL = B'00 (detecting by the write end signal or matching of the end address), and WS bit in MEccCTRL = 0 (stalling).

At this time, also set the following:

YSZM1[11:0] bits in MEccBSIZE = (HD[11:0] in VP4_IMAGE_SIZE of VPU) – 1
XSZM1[15:0] bits in MEccBSIZE = (WD[11:0] in VP4_IMAGE_SIZE of VPU) – 1

Note: It is also possible to set the WD[1:0] bits to B'11 and the WS bit to B'1, which does not require waiting for the completion of write back to the external memory or the completion of reading from another module in the ICB. In this case, write 1 to the WF and RF bits in MEnnCTRL for all ICBnn used by BEUn upon receiving the end interrupt from the VPU, notify the completion of the use of all ICBnn, and then confirm the transfer end of all ICBnn using the transfer end interrupt from the ICB or other means.

41.5.6 Image Input of VPU

(1) Registers That Can Use ICBs and Conditions

For using ICBs, VP4_CPY_ADDR and VP4_CPC_ADDR in the VPU can be set. Note that whether ICBs are used should be common to both Y/C.

For using ICBs, VP4_R0Y_ADDR and VP4_R0C_ADDR in the VPU can be set. Note that whether ICBs are used should be common to both Y/C.

(2) Conditions for the Number of Lines Retained

For VP4_CPY_ADDR and VP4_CPC_ADDR, set the number of lines retained during reading equal to or larger than 48.

For VP4_R0Y_ADDR and VP4_R0C in the VPU set as follows:

When the FF[2:0] bits in VP4_VOP_CTRL in the VPU is 1, set the number of lines retained during reading equal to or larger than 48.

When the FF[2:0] bits in VP4_VOP_CTRL in the VPU is 2, set the number of lines retained during reading equal to or larger than 80.

(3) Detecting Completion of Processing for One Side

Set the RCNT[3:0] bits in MEnnBSIZE to B'0010 (detection by two-time matches of the end address) for each ICBnn to enable automatic detection of the completion of reading for one register set.

At this time, also set the following:

YSZM1[11:0] bits in MEnnBSIZE = (HD[11:0] in VP4_IMAGE_SIZE of VPU) – 1
XSZM1[15:0] bits in MEnnBSIZE = (WD[11:0] in VP4_IMAGE_SIZE of VPU) – 1

Note: For the VPU, the size of the buffer area in the MERAM can be reduced by setting ICBs used for image output and input as follows:
XSZM1[15:0] bits in MEnnBSIZE = H'3FF
YSZM1[11:0] bits in MEnnBSIZE = roundup(VP4_IMAGE_SIZE.HD[11:0] × VP4_IMAGE_SIZE.WD[11:0]/1KB) – 1
KRBNM[3:0] bits in MEnnBSIZE: Set the necessary number of lines to be retained in 1-Kbyte units, which can be obtained by calculating "roundup(48 × VP4_IMAGE_SIZE.WD[11:0]/1KB)".
If the obtained number exceeds 16, adjust BSZ in MEnnCTRL.
SBSIZE bits in MEnnSBSIZE = H'400
Note that in the last line, access made to the external memory reaches the area that is rounded up the last pixel of an image to a 1-Kbyte boundary. So, make sure to allocate areas in the external memory to include the range. Although each line has a 1-Kbyte boundary in the example above, it can be set to have a 2-Kbyte boundary (one line cannot have more than 4 Kbytes).

41.5.7 Image Output of CEU

This LSI is provided with two CEU modules that have the same ICB-related functions. The notation of "CEUn" which means the n-th CEU is used in the following.

(1) Registers That Can Use ICBs and Conditions

For using ICBs, CDAYR, CDACR, CDBYR, and CDBCR in the CEU can be set. Also set CDWDR in the CEU to H'1000.

Note: Since CDWDR in the CEU is common to the Y image and CbCr image, a setting such that one of these images uses ICBs and the other uses no ICB is prohibited.

(2) Conditions for the Number of Lines Retained

Set the number of lines retained during writing, which is equal to or larger than 1.

(3) Detecting Completion of Processing for One Side

The CEU is a module that notifies the writing to the final pixel when outputting an image. Therefore, a combination of settings "MEnnCTRL.WD[1:0] = B'10 and MEnnCTRL.WS = B'0" is recommended.

Note: It is also possible to set the WD[1:0] bits to B'11 and the WS bit to B'1, which does not require waiting for the completion of write back to the external memory or the completion of reading from another module in the ICB. In this case, write 1 to the WF bit in MEnnCTRL for all ICBnn used by the CEU upon receiving the end interrupt from the CEU, notify the completion of the use of all ICBnn, and then confirm the transfer end of all ICBnn using the transfer end interrupt from the ICB or other means.

41.5.8 Image Output of JPU

(1) Registers That Can Use ICBs and Conditions

For using ICBs, JIFDDYA1, JIFDDYA2, and JIFDDYC2 in the JPU can be set. Also set JIFDDMW in the JPU to H'1000.

Note: Since JIFDDMW in JPU is common to these registers, a setting such that one of these register uses ICBs and the others use no ICB is prohibited.

(2) Conditions for the Number of Lines Retained

Set the number of lines retained during writing, which is equal to or larger than 8.

(3) Detecting Completion of Processing for One Side

Do not use the auto-detection function. Set the WD[1:0] bits in MEnnCTRL to B'11 (not detecting). Write 1 to the WF and RF bits in MEnnCTRL for all ICBnn used in the processing of the JPU upon receiving the end interrupt from the JPU, notify the completion of the use of all ICBnn, and then confirm the transfer end of all ICBnn using the transfer end interrupt from the ICB or other means.

41.5.9 Image Input of JPU

(1) Registers That Can Use ICBs and Conditions

For using ICBs, JIFESYA1, JIFESYC1, JIFESYA2, and JIFESYC2 in the JPU can be set. Also set JIFESMW in the JPU to H'1000.

Note: Since JIFESMW in JPU is common to these registers, a setting such that one of these register uses ICBs and the others use no ICB is prohibited.

(2) Conditions for the Number of Lines Retained

Set the number of lines retained during reading, which is equal to or larger than 8.

(3) Detecting Completion of Processing for One Side

Do not use the auto-detection function. Set the WD[1:0] bits in MEnnCTRL to B'11 (not detecting). Write 1 to the WF and RF bits in MEnnCTRL for all ICBnn used in the processing of the JPU upon receiving the end interrupt from the JPU, notify the completion of the use of all ICBnn, and then confirm the transfer end of all ICBnn using the transfer end interrupt from the ICB or other means.

41.5.10 Image Input of LCDC

When using ICBs in frame buffer cache mode, see section 41.4.4, Settings as Frame Buffer Cache. This subsection describes the conditions when ICBs are used in read fill mode.

Note: Make sure not to use the LCDC in interconnect mode.

(1) Registers That Can Use ICBs and Conditions

For using ICBs, MLDSA1R and MLDSA2R in the LCDC can be set. Also set MLDMLSR in the LCDC to H'1000.

Note: Since MLDMLSR in the LCDC is common to the Y image and CbCr image, a setting such that one of these images uses ICBs and the other uses no ICB is prohibited. Furthermore, when the YCbCr 4:4:4 format is used, MLDMLSR cannot be used because the handling of the MLDMLSR value differs between the Y image and CbCr image.

(2) Conditions for the Number of Lines Retained

Set the number of lines retained during reading, which is equal to or larger than 1.

(3) Detecting Completion of Processing for One Side

Set the RCNT[3:0] bits in MEnnBSIZE to B'0001 (detecting by single match of the end address) to enable automatic detection of the completion of reading for one register set, where the ICB number corresponding to MLDSA1R (or MLDSA2R when using MLDSA2R)is nn.

At this time, also set the following:

YSZM1[11:0] bits in MEnnBSIZE = (VDLN[10:0] in MLDVCNR of LCDC) – 1
XSZM1[15:0] bits in MEnnBSIZE = (HDCN[7:0] in MLDHCNR of LCDC) × 8 × (byte/pixel) – 1

41.6 Processing Procedure of Access to External Memory

Each ICB generates a read fill request or write back request to the external memory. This chapter describes the procedure of processing for these read fill/write back requests (hereafter referred to as external memory access requests).

1. External memory access requests made by each ICB are processed for each line. Every time the processing for one line is complete, the ICB to be processed is switched.
2. External memory access requests made by each ICB are accumulated in three types of queues. Once an external memory access request is stored in the queue, the next request is not generated until the processing of the stored request ends.
   - Queue 0: Stores external memory access requests made by ICB00 to ICB15.
   - Queue 1: Stores external memory access requests made by ICB16 to ICB23.
   - Queue 2: Stores external memory access requests made by ICB24 to ICB31.

   However, with respect to ICBs used as a cache buffer in frame buffer cache mode, external memory access requests are stored in queue 2 even if ICB00 to ICB23 are used.

   External memory access requests accumulated in queue 0 to queue 2 are processed in the following order.

   queue 2 → queue 1 → queue 2 → queue 0 → queue 2 → - - -

   That is, external memory access requests stored in queue 2 are processed once in twice, and those stored in queue 1 and queue 0 are processed once in four times. The processing rate of queue 0 is the same as that of queue 1, but up to sixteen external memory access requests from ICB00 to ICB15 are accumulated in queue 0 and up to eight requests from ICB16 to ICB23 are accumulated in queue 1. Due to this difference in the amount of accumulated requests, queue 1 can deal with requests more quickly than queue 0.

   For this reason, assign ICB24 to ICB31, ICB16 to 23, and ICB0 to 15 in the order of processing priority (highest priority should be assigned to from ICB24 to ICB31, ICB16 to ICB23, and ICB0 to ICB15).

Section 42 Image Extraction Direct Memory Access Controller (2D-DMAC)

This LSI has an image extraction direct memory access controller (2D-DMAC). This module reads the image data in the frame buffer memory, performs image extraction, rotation/inversion, magnification, and format conversion, and then writes the processed image data to the frame buffer memory.

42.1 Features

- Image extraction

  The 2D-DMAC extracts an image in the rectangular area from a point (Sx, Sy) shifted from the source image data origin in the frame memory, and then writes the extracted image data to another frame memory. Image data can be extracted in one-pixel units for YCbCr and RGB formats. This function is available for image clipping and image motion compensation.
- Image rotation/inversion

  Vertical/horizontal inversion and 90°/270° rotation can be performed.
- Simple magnification

  When writing a destination image, it can simply be enlarged twice in the X and Y directions.
- Format conversion

  RGB formats can be converted to each other.

  YCbCr formats (YCbCr4:2:0 and YCbCr4:2:2) can be converted to each other.

  No format conversion is possible between RGB and YCbCr.

  The format conversion method is equivalent to that of VIO.
- Channels

  Settings for eight channels are enabled. For the YCbCr formats, settings for one channel each for Y plane and C plane are required. Therefore, concurrent processing is possible for up to four planes (YCbCr formats) or up to eight planes (RGB formats).

  Transition of processing between channels is performed upon completion of each line processing (each macro-block line processing when rotation is specified).
- Interrupt

  Each channel has interrupt request signals that can be output at data transfer half end and data transfer end respectively.

Figure 42.1 shows a block diagram of the 2D-DMAC and figure 42.2 shows a flow in the pixel processing block.

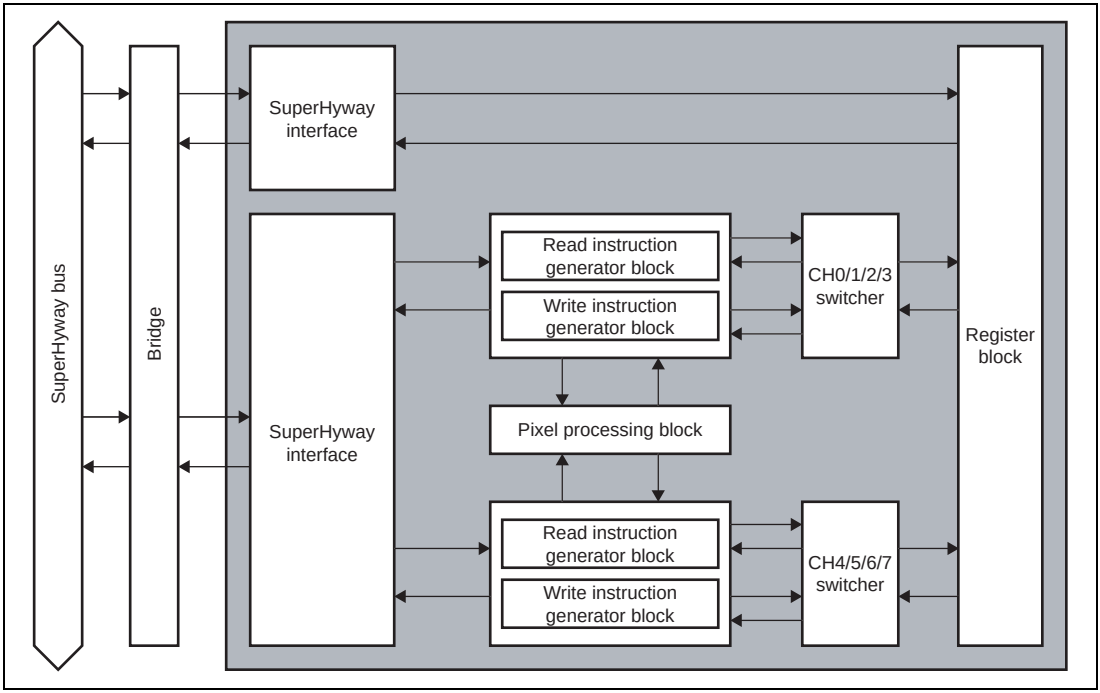


Figure 42.1 Block Diagram of 2D-DMAC

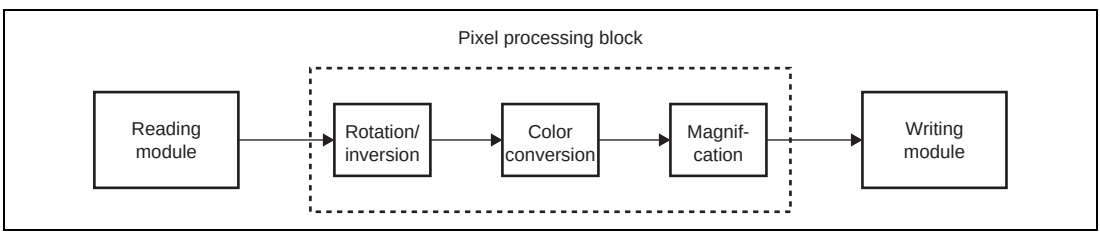


Figure 42.2 Processing Flow in Pixel Block

42.2 Register Descriptions

Table 42.1 lists the registers used in the 2D-DMAC. Table 42.2 shows the register status in each processing mode.

Table 42.1 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Interrupt status clear register	CHSTCLR	R/W	H'FEA0 0010	32
Channel priority switch register	CHPRI	R/W	H'FEA0 0014	32
CH0 control register	CH0CTRL	R/W	H'FEA0 0020	32
CH1 control register	CH1CTRL	R/W	H'FEA0 0024	32
CH2 control register	CH2CTRL	R/W	H'FEA0 0028	32
CH3 control register	CH3CTRL	R/W	H'FEA0 002C	32
CH4 control register	CH4CTRL	R/W	H'FEA0 0120	32
CH5 control register	CH5CTRL	R/W	H'FEA0 0124	32
CH6 control register	CH6CTRL	R/W	H'FEA0 0128	32
CH7 control register	CH7CTRL	R/W	H'FEA0 012C	32
CH0 input/output swap register	CH0SWAP	R/W	H'FEA0 0030	32
CH1 input/output swap register	CH1SWAP	R/W	H'FEA0 0034	32
CH2 input/output swap register	CH2SWAP	R/W	H'FEA0 0038	32
CH3 input/output swap register	CH3SWAP	R/W	H'FEA0 003C	32
CH4 input/output swap register	CH4SWAP	R/W	H'FEA0 0130	32
CH5 input/output swap register	CH5SWAP	R/W	H'FEA0 0134	32
CH6 input/output swap register	CH6SWAP	R/W	H'FEA0 0138	32
CH7 input/output swap register	CH7SWAP	R/W	H'FEA0 013C	32
CH0 source address register	CH0SAR	R/W	H'FEA0 0080	32
CH0 destination address register	CH0DAR	R/W	H'FEA0 0084	32
CH0 destination pixel register	CH0DPXL	R/W	H'FEA0 0088	32
CH0 source format register	CH0SFMT	R/W	H'FEA0 008C	32
CH0 destination format register	CH0DFMT	R/W	H'FEA0 0090	32
CH0 source line address register	CH0SARE	R	H'FEA0 0094	32
CH0 destination line address register	CH0DARE	R	H'FEA0 0098	32
CH0 destination pixel processing register	CH0DPXLE	R	H'FEA0 009C	32

Register Name	Abbreviation	R/W	Address	Access Size
CH1 source address register	CH1SAR	R/W	H'FEA0 00A0	32
CH1 destination address register	CH1DAR	R/W	H'FEA0 00A4	32
CH1 destination pixel register	CH1DPXL	R/W	H'FEA0 00A8	32
CH1 source format register	CH1SFMT	R/W	H'FEA0 00AC	32
CH1 destination format register	CH1DFMT	R/W	H'FEA0 00B0	32
CH1 source line address register	CH1SARE	R	H'FEA0 00B4	32
CH1 destination line address register	CH1DARE	R	H'FEA0 00B8	32
CH1 destination pixel processing register	CH1DPXLE	R	H'FEA0 00BC	32
CH2 source address register	CH2SAR	R/W	H'FEA0 00C0	32
CH2 destination address register	CH2DAR	R/W	H'FEA0 00C4	32
CH2 destination pixel register	CH2DPXL	R/W	H'FEA0 00C8	32
CH2 source format register	CH2SFMT	R/W	H'FEA0 00CC	32
CH2 destination format register	CH2DFMT	R/W	H'FEA0 00D0	32
CH2 source line address register	CH2SARE	R	H'FEA0 00D4	32
CH2 destination line address register	CH2DARE	R	H'FEA0 00D8	32
CH2 destination pixel processing register	CH2DPXLE	R	H'FEA0 00DC	32
CH3 source address register	CH3SAR	R/W	H'FEA0 00E0	32
CH3 destination address register	CH3DAR	R/W	H'FEA0 00E4	32
CH3 destination pixel register	CH3DPXL	R/W	H'FEA0 00E8	32
CH3 source format register	CH3SFMT	R/W	H'FEA0 00EC	32
CH3 destination format register	CH3DFMT	R/W	H'FEA0 00F0	32
CH3 source line address register	CH3SARE	R	H'FEA0 00F4	32
CH3 destination line address register	CH3DARE	R	H'FEA0 00F8	32
CH3 destination pixel processing register	CH3DPXLE	R	H'FEA0 00FC	32
CH4 source address register	CH4SAR	R/W	H'FEA0 0180	32
CH4 destination address register	CH4DAR	R/W	H'FEA0 0184	32
CH4 destination pixel register	CH4DPXL	R/W	H'FEA0 0188	32
CH4 source format register	CH4SFMT	R/W	H'FEA0 018C	32
CH4 destination format register	CH4DFMT	R/W	H'FEA0 0190	32
CH4 source line address register	CH4SARE	R	H'FEA0 0194	32
CH4 destination line address register	CH4DARE	R	H'FEA0 0198	32
CH4 destination pixel processing register	CH4DPXLE	R	H'FEA0 019C	32

Register Name	Abbreviation	R/W	Address	Access Size
CH5 source address register	CH5SAR	R/W	H'FEA0 01A0	32
CH5 destination address register	CH5DAR	R/W	H'FEA0 01A4	32
CH5 destination pixel register	CH5DPXL	R/W	H'FEA0 01A8	32
CH5 source format register	CH5SFMT	R/W	H'FEA0 01AC	32
CH5 destination format register	CH5DFMT	R/W	H'FEA0 01B0	32
CH5 source line address register	CH5SARE	R	H'FEA0 01B4	32
CH5 destination line address register	CH5DARE	R	H'FEA0 01B8	32
CH5 destination pixel processing register	CH5DPXLE	R	H'FEA0 01BC	32
CH6 source address register	CH6SAR	R/W	H'FEA0 01C0	32
CH6 destination address register	CH6DAR	R/W	H'FEA0 01C4	32
CH6 destination pixel register	CH6DPXL	R/W	H'FEA0 01C8	32
CH6 source format register	CH6SFMT	R/W	H'FEA0 01CC	32
CH6 destination format register	CH6DFMT	R/W	H'FEA0 01D0	32
CH6 source line address register	CH6SARE	R	H'FEA0 01D4	32
CH6 destination line address register	CH6DARE	R	H'FEA0 01D8	32
CH6 destination pixel processing register	CH6DPXLE	R	H'FEA0 01DC	32
CH7 source address register	CH7SAR	R/W	H'FEA0 01E0	32
CH7 destination address register	CH7DAR	R/W	H'FEA0 01E4	32
CH7 destination pixel register	CH7DPXL	R/W	H'FEA0 01E8	32
CH7 source format register	CH7SFMT	R/W	H'FEA0 01EC	32
CH7 destination format register	CH7DFMT	R/W	H'FEA0 01F0	32
CH7 source line address register	CH7SARE	R	H'FEA0 01F4	32
CH7 destination line address register	CH7DARE	R	H'FEA0 01F8	32
CH7 destination pixel processing register	CH7DPXLE	R	H'FEA0 01FC	32

Table 42.2 Register Status in Each Processing Mode

Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
CHSTCLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHPRI	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnCTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnSWAP	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnSAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnDAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnDPXL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnSFMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnDFMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnSARE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnDARE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CHnDPXLE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

n: 0 to 7

42.2.1 Control Registers (CHnCTRL)

CHnCTRL (n = 0 to 7) controls transfer mode and interrupt output in each channel.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HMRR	VMRR	ROTL	ROTR	—	MX	—	MY	HIE	HE	TIE	TE	—	—	STP	DMAEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	HMRR	0	R/W	Horizontal Inversion (symmetric to the vertical axis) The source image inverted horizontally is output. See table 42.5 for details on the relations between this bit and inversion/rotation. 0: NOP 1: Horizontally inverted source image is output.
14	VMRR	0	R/W	Vertical Inversion (symmetric to the horizontal axis) The source image inverted vertically is output. See table 42.5 for details on the relations between this bit and inversion/rotation. 0: NOP 1: Vertically Inverted source image is output.
13	ROTL	0	R/W	270° Rotation (clockwise) The source image rotated 270° is output. When this bit is 1, set the ROTR bit to 0. See table 42.5 for details on the relations between this bit and inversion/rotation. 0: NOP 1: Source image rotated 270° is output.

Bit	Bit Name	Initial Value	R/W	Description
12	ROTR	0	R/W	90° Rotation (clockwise) The source image rotated 90° is output. When this bit is 1, set the ROTL bit to 0. See table 42.5 for details on the relations between this bit and inversion/rotation. 0: NOP 1: Source image rotated 90° is output.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	MX	0	R/W	Magnify X Direction 0: No magnification 1: Outputs with double magnification in the X direction. When ROTR or ROTL is 1, set this bit to 0.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	MY	0	R/W	Magnify Y Direction 0: No magnification 1: Outputs with double magnification in the Y direction. When ROTR or ROTL is 1, set this bit to 0.
7	HIE	0	R/W	Half End Interrupt Enable Specifies whether to output an interrupt request when the half end flag is set. 0: A half end interrupt is disabled. 1: A half end interrupt is enabled.

Bit	Bit Name	Initial Value	R/W	Description
6	HE	0	R/W	Half End Indicates that data transfer for half the specified lines has been completed. This bit is set to 1 when the DVPXLE bit value in CHnDPXLE becomes half (a value of 1-bit right shift) of the DVPXL bit value in CHnDPXL that was set before starting transfer. When ROTR or ROTL is 1, the HE bit is set to 1 when the DHPXLE bit value in CHnDPXLE becomes half of the DHPXL bit value in CHnDPXL that was set before starting transfer. Only clearing to 0 after reading 1 is enabled. Clearing by the CHSTCLR register is also enabled.
5	TIE	0	R/W	Transfer End Interrupt Enable Specifies whether to output an interrupt request when the transfer end flag is set. 0: A transfer end interrupt is disabled. 1: A transfer end interrupt is enabled.
4	TE	0	R/W	Transfer End This bit is set to 1 when the DVPXLE bit value in CHnDPXLE matches the DVPXL bit value in CHnDPXL that was set before starting transfer and the entire DMA transfer is completed. When this bit is set to 1, the DMAEN bit is automatically cleared. Only clearing to 0 after reading 1 is enabled. Clearing by the CHSTCLR register is also enabled.
3, 2	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	STP	0	R/W	Transfer Stop Stops DMA transfer temporarily. When this bit is set to 1, DMA transfer is suspended upon completion of the line that is being transferred, and the DMAEN bit is cleared to 0. At this time, this bit is also cleared. To restart DMA transfer after that, set the DMAEN bit to 1. To reset the operation of a channel, rewrite data to the SAR, DAR, and DPXL registers of the channel. [Writing] 0: NOP 1: Stops DMA transfer temporarily. [Reading] 0: Transfer is in progress or suspended. 1: Pending until the line being transferred is completed.
0	DMAEN	0	R/W	DMA Transfer Enable Enables DMA transfer. Only writing 1 is enabled. This bit shows 1 during transfer, and is automatically cleared when a DMA transfer is completed. Writing 0 to this bit is ignored. A DMA transfer can be suspended by setting the STP bit to 1. [Writing] 0: NOP 1: Performs DMA transfer. [Reading] 0: No DMA transfer is in progress. 1: DMA transfer is in progress.

42.2.2 Input/Output Swap Register (CHnSWAP)

CHnSWAP (n = 0 to 7) controls 64-bit data swapping in the data input/output section of the 2D-DMAC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	OLS	OWS	OBS	—	ILS	IWS	IBS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	OLS	0	R/W	Output Longword Swap Setting In the output section of the 2D-DMAC, the 32-bit blocks at the MSB and LSB sides in 64-bit data are swapped in longword units. 0: Output longword swap is not performed. 1: Output longword swap is performed.
5	OWS	0	R/W	Output Word Swap Setting In the output section of the 2D-DMAC, data is swapped in word units within each 32-bit block at the MSB and LSB sides in 64-bit data. 0: Output word swap is not performed. 1: Output word swap is performed.
4	OBS	0	R/W	Output Byte Swap Setting In the output section of the 2D-DMAC, data is swapped in byte units within each 16-bit block in 64-bit data. 0: Output byte swap is not performed. 1: Output byte swap is performed.

Bit	Bit Name	Initial Value	R/W	Description
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	ILS	0	R/W	Input Longword Swap Setting In the input section of the 2D-DMAC, the 32-bit blocks at the MSB and LSB sides in 64-bit data are swapped in longword units. 0: Input longword swap is not performed. 1: Input longword swap is performed.
1	IWS	0	R/W	Input Word Swap Setting In the input section of the 2D-DMAC, data is swapped in word units within each 32-bit block at the MSB and LSB sides in 64-bit data. 0: Input word swap is not performed. 1: Input word swap is performed.
0	IBS	0	R/W	Input Byte Swap Setting In the input section of the 2D-DMAC, data is swapped in byte units within each 16-bit block in 64-bit data. 0: Input byte swap is not performed. 1: Input byte swap is performed.

When the system is operated in little-endian mode, this register should be used to select the appropriate swapping method according to the input/output format. The data before and after swapping in each swapping method is shown in figures 42.3, 42.4, and 42.5.

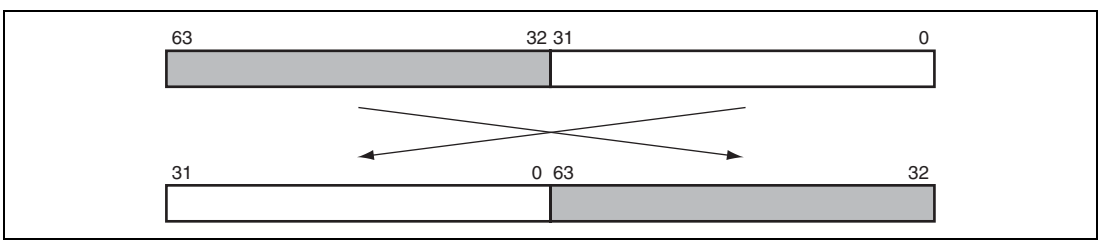


Figure 42.3 Data Before and After Longword Swapping

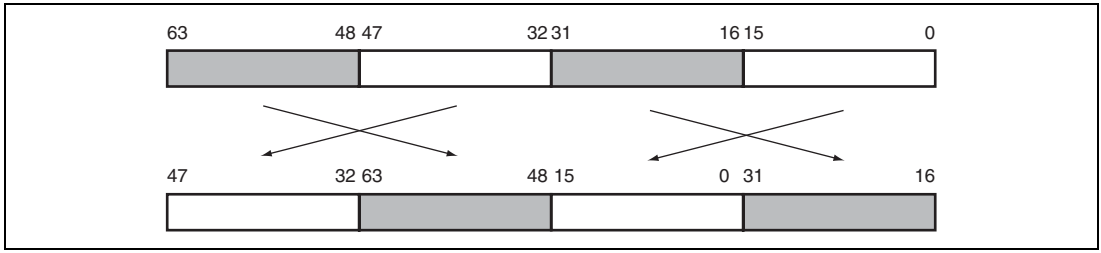


Figure 42.4 Data Before and After Word Swapping

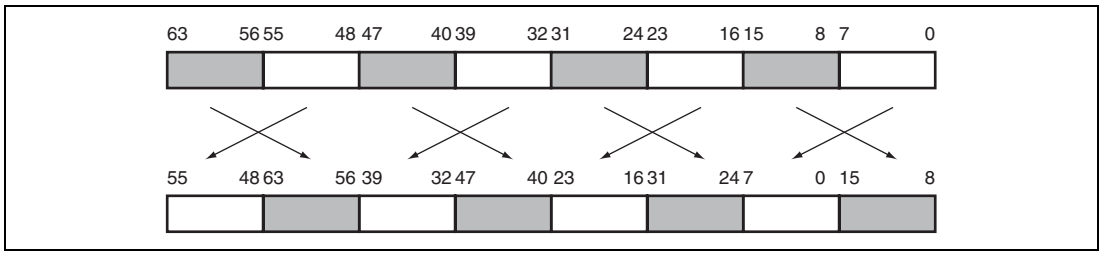


Figure 42.5 Data Before and After Byte Swapping

42.2.3 Source Format Registers (CHnSFMT)

CHnSFMT (n = 0 to 7) specifies the format of the source image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HWTH[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	MD[1:0]		PKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	HWTH[15:0]	H'0000	R/W	Source Image Horizontal Byte Count These bits specify the one-line width of the source image in units of bytes. This value should be a multiple of the pack size. When ROTR or ROTL is 1, set these bits to 16n.
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6, 5	MD[1:0]	00	R/W	Source Image Format 00: RGB data 01: Y data 10: CbCr data (YCbCr4:2:0) 11: CbCr data (YCbCr4:2:2)
4 to 0	PKF[4:0]	00000	R/W	Source Image RGB Data Packed Format These bits specify the packed format for source image data in the RGB format. This setting is valid only when MD = B'00. Table 42.3 shows the packed RGB formats.

Table 42.3 Packed RGB Formats

PKF [4:0]	Format	Bit Rate [bpp]	Phase	Bit																																					
				31 to 24								23 to 16								15 to 8								7 to 0													
B'00000	ARGB8888	24		a	a	a	a	a	a	a	a	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0								
B'00001	RGBA8888	24		R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	a	a	a	a	a	a	a	a								
B'00010	RGB888	24	0	R0	R0	R0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0	B0	R1	R1	R1	R1	R1	R1	R1	R1								
			1	G1	G1	G1	G1	G1	G1	G1	G1	B1	B1	B1	B1	B1	B1	B1	B1	R2	R2	R2	R2	R2	R2	R2	R2	G2	G2	G2	G2	G2	G2	G2	G2								
			2	B2	B2	B2	B2	B2	B2	B2	B2	R3	R3	R3	R3	R3	R3	R3	R3	G3	G3	G3	G3	G3	G3	G3	G3	B3	B3	B3	B3	B3	B3	B3	B3								
B'00011	RGB565	16																		R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0								
B'00100	RGB332	8																										R0	R0	R0	G0	G0	G0	B0	B0								
B'00111	pRGB14-666	18		0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0	B0	B0								
B'01000	pRGB4-444	12																		0	0	0	0	R0	R0	R0	R0	G0	G0	G0	G0	B0	B0	B0	B0								
B'01001	RGB666	18	0	0	0	R0	R0	R0	R0	R0	R0	0	0	G0	G0	G0	G0	G0	G0	0	0	B0	B0	B0	B0	B0	B0	0	0	R1	R1	R1	R1	R1	R1								
			1	0	0	G1	G1	G1	G1	G1	G1	0	0	B1	B1	B1	B1	B1	B1	0	0	R2	R2	R2	R2	R2	R2	0	0	G2	G2	G2	G2	G2	G2								
			2	0	0	B2	B2	B2	B2	B2	B2	0	0	R3	R3	R3	R3	R3	R3	0	0	G3	G3	G3	G3	G3	G3	0	0	B3	B3	B3	B3	B3	B3								
B'01010	BGR666	18	0	0	0	B0	B0	B0	B0	B0	B0	0	0	G0	G0	G0	G0	G0	G0	0	0	R0	R0	R0	R0	R0	R0	0	0	B1	B1	B1	B1	B1	B1								
			1	0	0	G1	G1	G1	G1	G1	G1	0	0	R1	R1	R1	R1	R1	R1	0	0	B2	B2	B2	B2	B2	B2	0	0	G2	G2	G2	G2	G2	G2								
			2	0	0	R2	R2	R2	R2	R2	R2	0	0	B3	B3	B3	B3	B3	B3	0	0	G3	G3	G3	G3	G3	G3	0	0	R3	R3	R3	R3	R3	R3								
B'01011	BGR888	24	0	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0	B1	B1	B1	B1	B1	B1	B1	B1								
			1	G1	G1	G1	G1	G1	G1	G1	G1	R1	R1	R1	R1	R1	R1	R1	R1	B2	B2	B2	B2	B2	B2	B2	B2	G2	G2	G2	G2	G2	G2	G2	G2								
			2	R2	R2	R2	R2	R2	R2	R2	R2	B3	B3	B3	B3	B3	B3	B3	B3	G3	G3	G3	G3	G3	G3	G3	G3	R3	R3	R3	R3	R3	R3	R3	R3								
B'01100	ABGR8888	24		a	a	a	a	a	a	a	a	B0	B0	B0	B0	B0	B0	B0	B0	G0	G0	G0	G0	G0	G0	G0	G0	R0	R0	R0	R0	R0	R0	R0	R0								
B'01101	RGB565	16		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	R0	R0	R0	R0	R0	G0	G0	G0	G0	G0	G0	B0	B0	B0	B0	B0								

42.2.4 Source Address Registers (CHnSAR)

CHnSAR (n = 0 to 7) specifies the transfer start address of the source image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SAR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SAR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SAR[31:0]	H'00000000	R/W	Source Image Extraction Start Address These bits specify the address of the extraction start pixel in the source image data.

42.2.5 Destination Format Registers (CHnDFMT)

CHnDFMT (n = 0 to 7) specifies the destination image information such as the format.

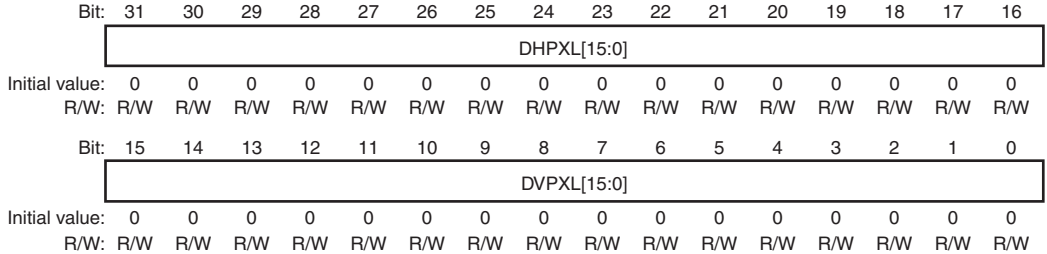
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	HWTH[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KALPHA[7:0]								AV	MD[1:0]		PKF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	HWTH[15:0]	H'0000	R/W	Destination Image Horizontal Byte Count These bits specify the one-line width of the destination image in units of bytes. Set a multiple of 16. When CTRL.MX is 1, set a value after magnification.
15 to 8	KALPHA [7:0]	H'00	R/W	Alpha Value These bits specify an alpha value used for output when the source image is in a format without alpha or when the source image is in a format with alpha and the AV bit is 1.
7	AV	0	R/W	Alpha Enable 0: The alpha value in the source image is ignored when the source image is in a format with alpha. When the output is in a format with alpha, the KALPHA value is output. 1: The alpha value in the source image is valid when the source image is in a format with alpha. When the output is in a format with alpha, the alpha value is output.
6, 5	MD[1:0]	00	R/W	Destination Image Format 00: RGB data 01: Y data 10: CbCr data (YCbCr4:2:0) 11: CbCr data (YCbCr4:2:2)
4 to 0	PKF[4:0]	00000	R/W	Destination Image RGB Data Packed Format These bits specify the packed format for destination image data in the RGB format. This setting is valid only when MD = B'00. Table 42.3 shows the packed RGB formats.

42.2.6 Destination Pixel Registers (CHnDPXL)

CHnDPXL (n = 0 to 7) specifies the horizontal pixel count and vertical pixel count of the destination image. Respecify the value in CHnDPXL every time transfer is initiated by setting the DMAEN bit, even if the value is unchanged.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DHPXL [15:0]	H'0000	R/W	Destination Image Horizontal Pixel Count These bits specify the horizontal pixel count of the destination image. Even when the MX bit in CTRL is 1, set a value before magnification. For the CbCr plane in YCbCr4:2:2 or YCbCr4:2:0, set to half of the number of pixels. The minimum pixel count is 1; set 1 or a larger value.
15 to 0	DVPXL [15:0]	H'0000	R/W	Destination Image Vertical Pixel Count These bits specify the vertical pixel count of the destination image. Even when the MX bit in CTRL is 1, set a value before magnification. For the CbCr plane in YCbCr4:2:0, set to half of the number of lines. The minimum pixel count is 1; set 1 or a larger value.

42.2.7 Destination Address Registers (CHnDAR)

CHnDAR (n = 0 to 7) specifies the transfer start address of the destination image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DAR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DAR [31:0]	H'00000000	R/W	These bits specify the start address of the destination image. The point whose address should be specified here changes according to the settings of the VMRR, HMRR, ROTR, ROTL, and MY bits. See section 42.3.2, Image Rotation/Inversion, for details. Only a multiple of 16 can be set.

42.2.8 Source Line Address Registers (CHnSARE)

CHnSARE (n = 0 to 7) shows the start address of the line in processing of the source image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SARE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SARE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SARE [31:0]	H'00000000	R	This read-only register used for internal processing shows the start address of the line in processing of the source image.

42.2.9 Destination Line Address Registers (CHnDARE)

CHnDARE (n = 0 to 7) shows the start address of the line in processing of the destination image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DARE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DARE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DARE [31:0]	H'00000000	R	This read-only register used for internal processing shows the start address of the line in processing of the destination image.

42.2.10 Destination Pixel Processing Registers (CHnDPXLE)

CHnDPXLE (n = 0 to 7) shows the number of unprocessed lines of the destination image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DHPXLE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DVPXLE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	DHPXLE [15:0]	H'0000	R	This read-only register used for internal processing shows the number of unprocessed horizontal lines of the destination image.
15 to 0	DVPXLE [15:0]	H'0000	R	This read-only register used for internal processing shows the number of unprocessed lines of the destination image.

42.2.11 Interrupt Status Clear Register (CHSTCLR)

CHSTCLR indicates the status of the HE and TE bits in each channel. By writing 1 to a bit in CHSTCLR, the HE or TE bit in the corresponding channel is to be cleared.

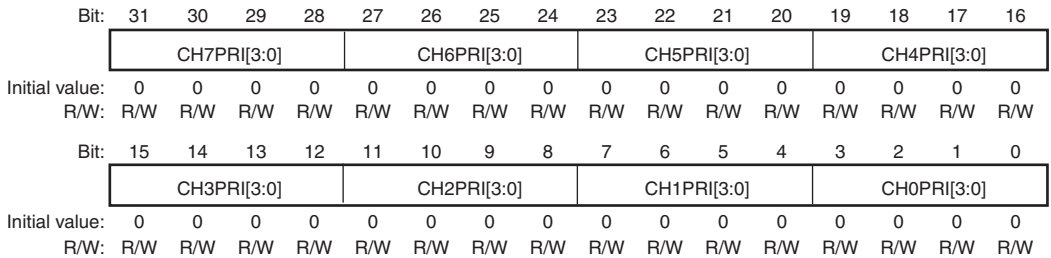
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CHnHE[7:0]								CHnTE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	CHnHE[7:0]	H'00	R/W	Indicates the status of the HE bit in CHn (n = 0 to 7). Writing 1 to a bit clears the HE bit in the corresponding channel. [Writing] 0: NOP 1: HE bit is cleared. [Reading] 0: 0 is set in the HE bit. 1: 1 is set in the HE bit.
7 to 0	CHnTE[7:0]	H'00	R/W	Indicates the status of the TE bit in CHn (n = 0 to 7). Writing 1 to a bit clears the TE bit in the corresponding channel. [Writing] 0: NOP 1: TE bit is cleared. [Reading] 0: 0 is set in the TE bit. 1: 1 is set in the TE bit.

42.2.12 Channel Priority Switch Register (CHPRI)

CHPRI specifies the priority level for each channel. The priority level for all channels should be set to 1 when starting the 2D-DMAC operation.



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	CH7PRI[3:0]	H'0	R/W	CH7 Process Priority Set to 1 when starting the 2D-DMAC operation.
27 to 24	CH6PRI[3:0]	H'0	R/W	CH6 Process Priority Set to 1 when starting the 2D-DMAC operation.
23 to 20	CH5PRI[3:0]	H'0	R/W	CH5 Process Priority Set to 1 when starting the 2D-DMAC operation.
19 to 16	CH4PRI[3:0]	H'0	R/W	CH4 Process Priority Set to 1 when starting the 2D-DMAC operation.
15 to 12	CH3PRI[3:0]	H'0	R/W	CH3 Process Priority Set to 1 when starting the 2D-DMAC operation.
11 to 8	CH2PRI[3:0]	H'0	R/W	CH2 Process Priority Set to 1 when starting the 2D-DMAC operation.
7 to 4	CH1PRI[3:0]	H'0	R/W	CH1 Process Priority Set to 1 when starting the 2D-DMAC operation.
3 to 0	CH0PRI[3:0]	H'0	R/W	CH0 Process Priority Set to 1 when starting the 2D-DMAC operation.

42.3 Operation

This module can extract image data of a required size from the source image and store it as a destination image. It also can convert the format and rotate/invert and magnify the image during extraction.

42.3.1 Image Extraction

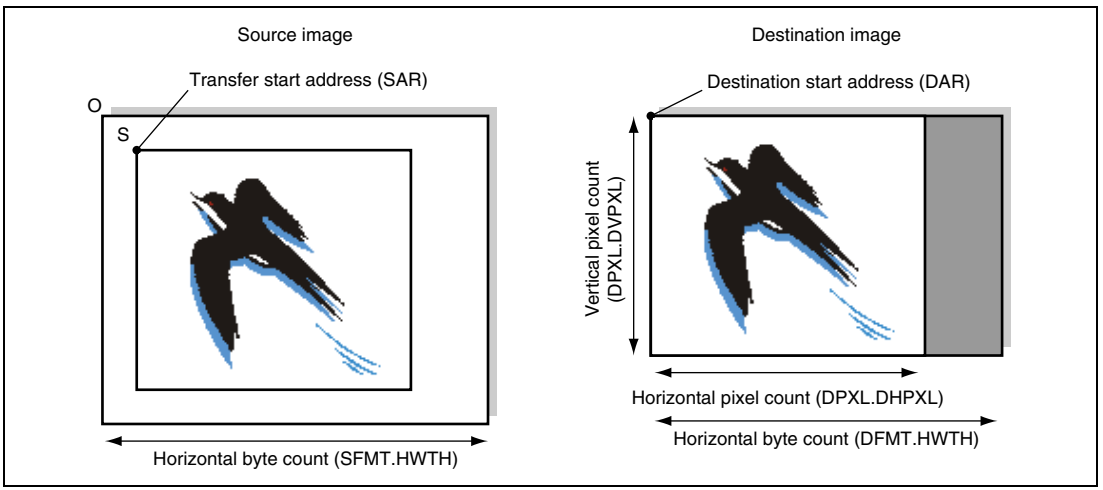


Figure 42.6 Schematic Diagram of Image Extraction

Figure 42.6 shows a schematic diagram of image extraction.

Image formats YCbCr4:2:0, YCbCr4:2:2, and RGB are supported for image extraction. Extraction of data is possible in one-pixel units for YCbCr and RGB formats. However, since the Y data and C data are independently stored in memory in the YCbCr format, set them in different channels respectively.

For source images, the address of extraction start coordinates S and the horizontal byte count must be set in respective registers. Set the start address of the extraction start pixel in SAR. If the X coordinate of extraction start coordinates S (SAR) is an odd value relative to origin O in the YCbCr format, the Y data and C data will be misaligned horizontally. Furthermore, in the YCbCr4:2:0 format, if the Y coordinate of extraction start coordinates S is an odd value relative to origin O, the Y data and C data will be misaligned vertically.

For destination images, the destination start address, numbers of horizontal and vertical pixels, and horizontal byte count must be set. Set an integer multiple of 16 (≥ DHPXL in DPXL × bytes per pixel) for the horizontal byte count (HWTH in DFMT). Discard the gray area in figure 42.6 as invalid data. If the horizontal pixel count of Y data is an odd value in the YCbCr format, invalid data may be generated along the right side of the image. Likewise, the vertical pixel count of Y data is an odd value in the YCbCr4:2:0 format, invalid data may be generated along the bottom of the image.

Table 42.4 lists these restrictions.

Table 42.4 Restrictions for Each Format

Format	Item	Restrictions
Common	Destination image horizontal byte count (CHnDFMT.HWTH)	Set an integer multiple of 16 (byte).
Y data (YCbCr4:2:2)	Source image horizontal extraction start position (CHnSAR.SAR)	If the X coordinate of starting coordinates S (SAR) relative to coordinate origin O of the source image is an odd value, Y and CbCr are horizontally misaligned.
	Destination image horizontal pixel count (CHnDPXL.DHPXL)	If the horizontal pixel count of the destination image is an odd value, invalid data may be generated along the right side of the image.
Y data (YCbCr4:2:0)	Source image horizontal extraction start position (CHnSAR.SAR)	If the X coordinate of starting coordinates S (SAR) relative to coordinate origin O of the source image is an odd value, Y and CbCr are horizontally misaligned.
	Source image vertical extraction start position (CHnSAR.SAR)	If the Y coordinate of starting coordinates S (SAR) relative to coordinate origin O of the source image is an odd value, Y and CbCr are vertically misaligned.
	Destination image horizontal pixel count (CHnDPXL.DHPXL)	If the horizontal pixel count of the destination image is an odd value, invalid data may be generated along the right side of the image.
	Destination image vertical pixel count (CHnDPXL.DVPXL)	If the vertical pixel count of the destination image is an odd value, invalid data may be generated along the bottom of the image.

42.3.2 Image Rotation/Inversion

Table 42.5 Combinations of Bits VMRR, HMRR, ROTL, and ROTR and Rotation/Inversion

VMRR Bit	HMRR Bit	ROTL Bit	ROTR Bit	Point Specified by DAR	Rotation/Inversion
0	0	0	0	A	Rotation/inversion is not performed
0	0	0	1	C	Clockwise 90° rotation
0	0	1	0	B	Clockwise 270° rotation
0	1	0	1	A	Clockwise 90° rotation, then horizontal inversion
1	0	1	0		(Clockwise 270° rotation, then vertical inversion)
1	0	0	1	D	Clockwise 90° rotation, then vertical inversion
0	1	1	0		(Clockwise 270° rotation, then horizontal inversion)
0	1	0	0	C	Horizontal inversion
1	0	0	0	B	Vertical inversion
1	1	0	0	D	180° rotation
Others				—	Setting prohibited

Desired rotation in 90-degree units and inversion can be specified through the combination of bits VMRR, HMRR, ROTR, and ROTL as shown in table 42.5. Figure 42.7 shows the source image and destination image when rotation/inversion is specified and figure 42.8 shows the case when only inversion is specified.

For rotation/inversion, the point whose address should be specified in CHnDAR changes according to the settings of bits VMRR, HMRR, ROTR, and ROTL. See figure 42.9 for details on the point that should be specified in each processing.

Use the following formulas to specify the address.

offset_add	Address of the top left corner of the destination image.
dest_hwidth	Horizontal byte count of the destination image (CHnDFMT.DHWTH)
dest_vpxl	Vertical pixel count of the destination image (CHnDPXL.DVPXL)

A DAR = offset_add

B DAR = offset_add + (dest_hwidth × (dest_vpxl − 1))

C DAR = offset_add + dest_hwidth

D DAR = offset_add + (dest_hwidth × dest_vpxl)

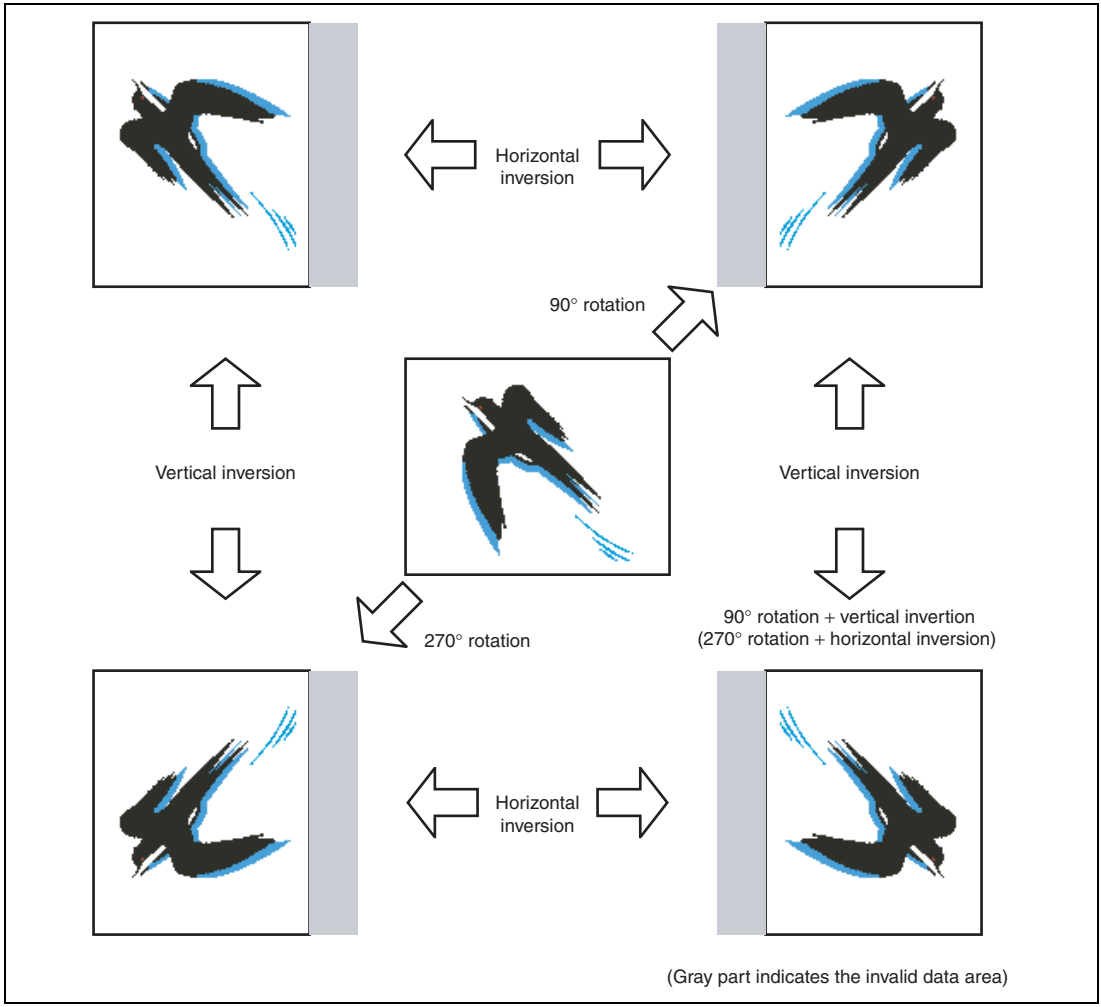


Figure 42.7 Relations between Source Image and Destination Image after Rotation/Inversion

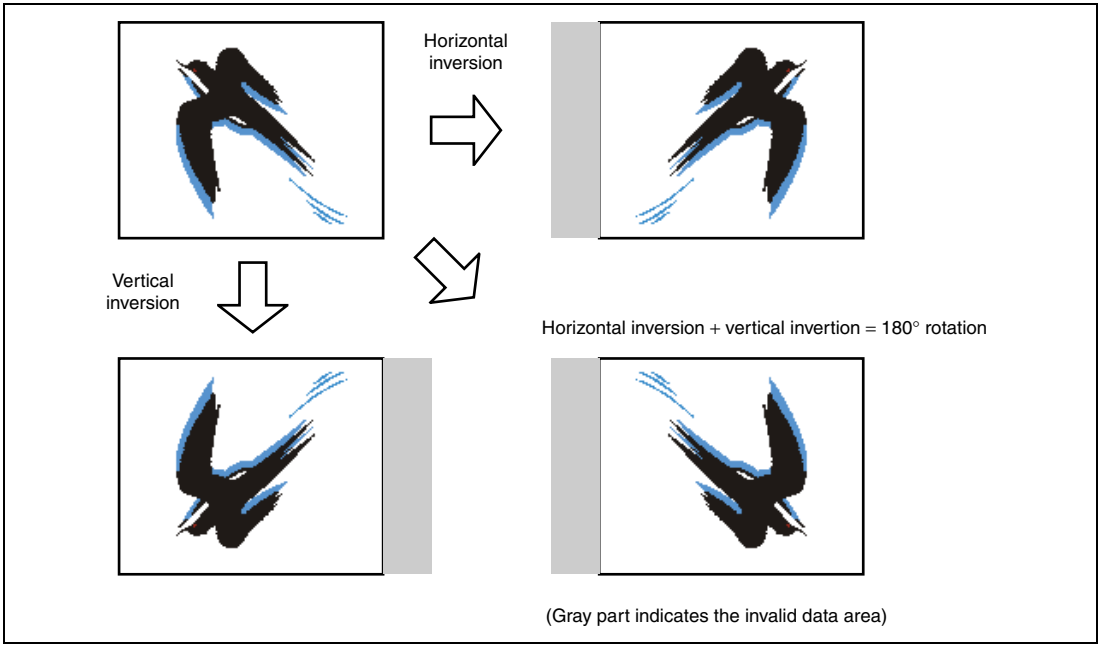


Figure 42.8 Relations between Source Image and Destination Image after Inversion

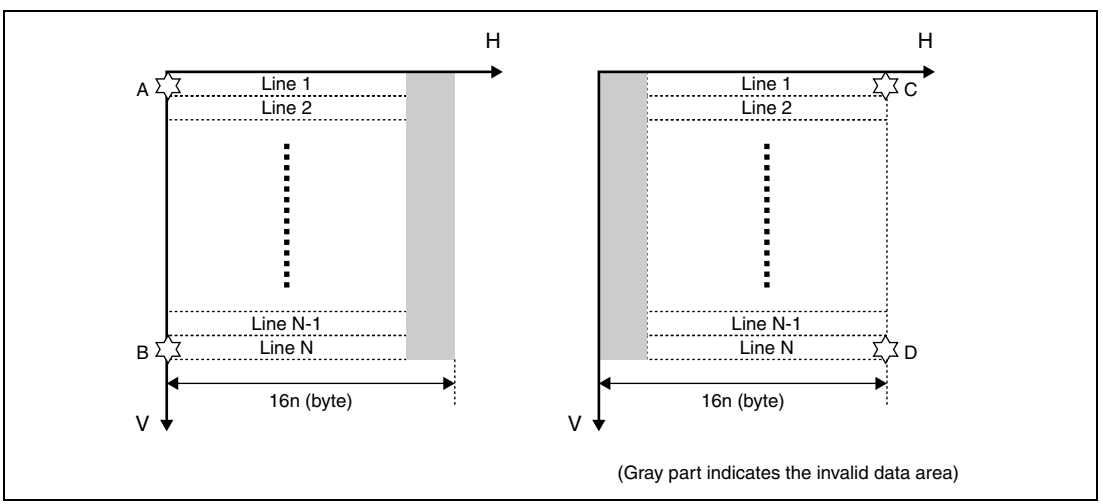


Figure 42.9 Point to be Specified in CHnDAR

42.3.3 RGB/YCbCr Format Conversion

Conversions are possible between RGB formats and between YCbCr formats (YCbCr4:2:0 and YCbCr4:2:2). For available RGB formats, see table 42.3. However, no format conversion is possible between RGB and YCbCr.

42.3.4 Simple Magnification

Setting the MX and MY bits in CTRL to 1 allows enlarged output in the X and Y directions respectively.

Simple magnification method is used for the magnification processing where the pixels in the source image are copied in the X and Y directions and are then output.

Figure 42.10 is a schematic diagram of simple magnification of pixels.

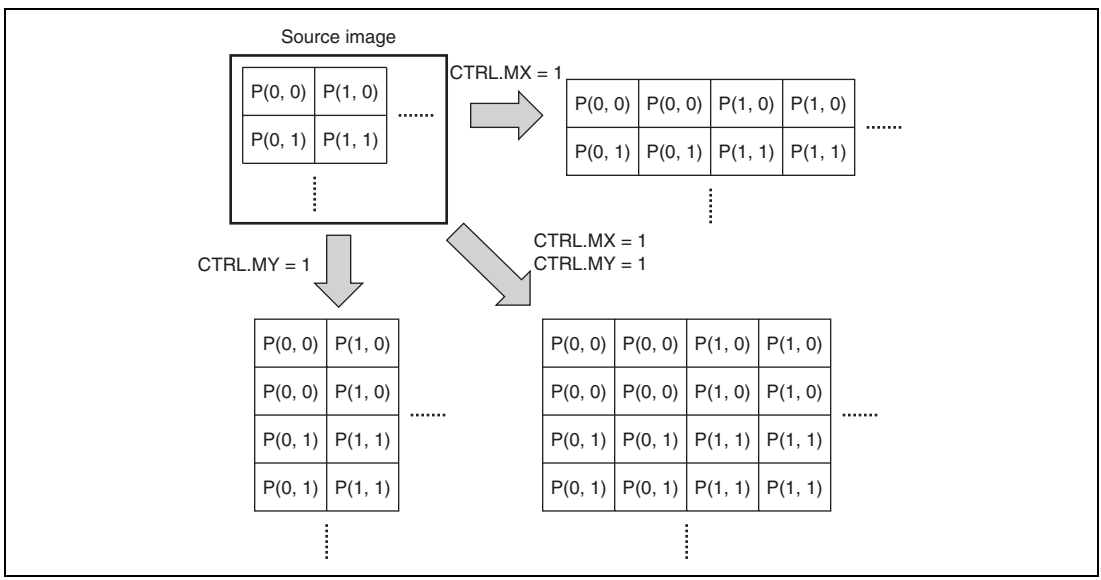


Figure 42.10 Enlarged Output of Pixels

42.4 Transfer Flow

Figure 42.11 shows a flowchart of transfer using the 2D-DMAC.

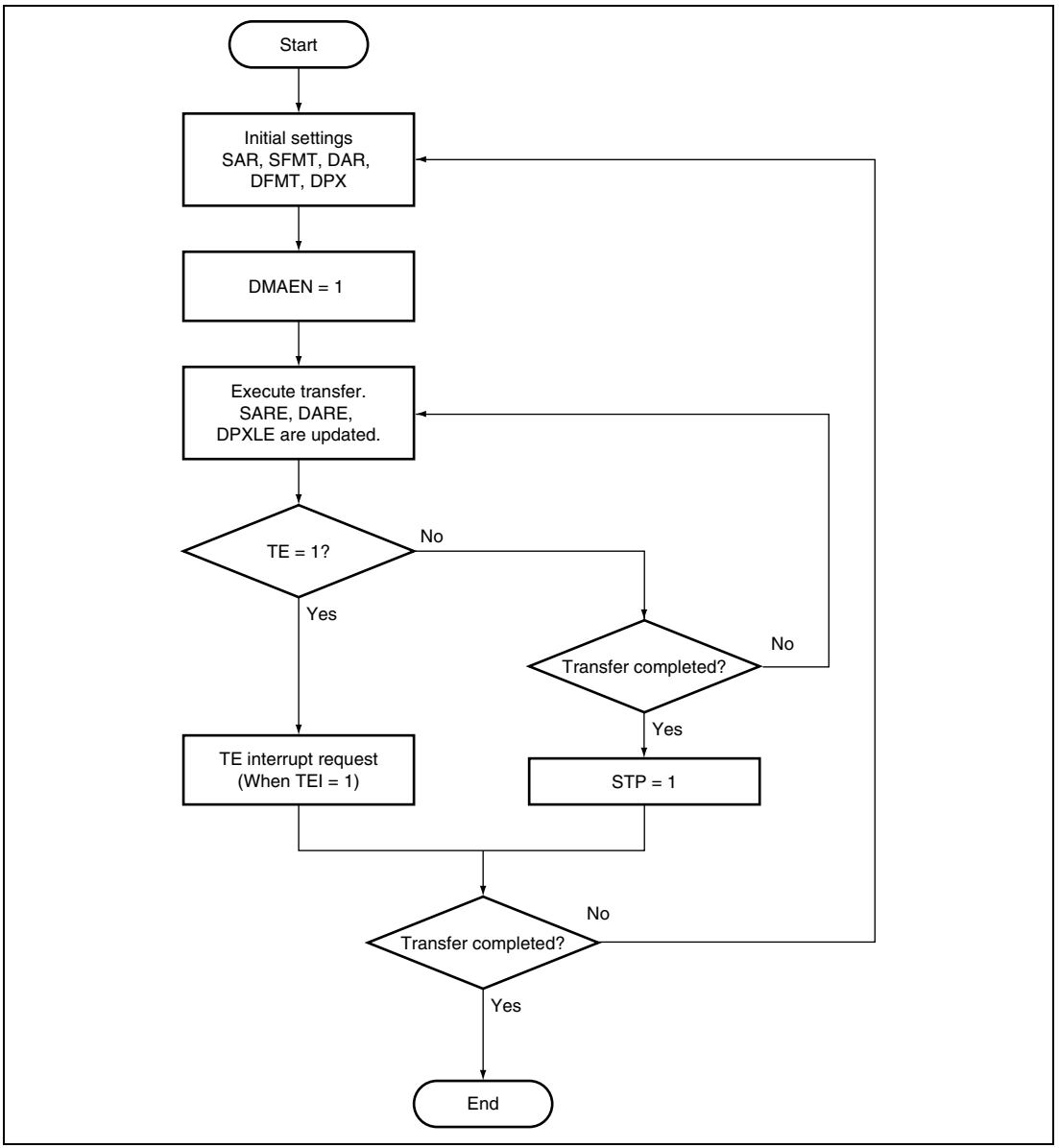


Figure 42.11 Flowchart of Transfer Using 2D-DMAC

Section 43 TS Interface (TSIF)

The transport stream interface (TSIF) is a module for receiving the MPEG2 transport stream (TS) used in one-segment broadcasting implemented as part of the digital terrestrial broadcasting services. The TSIF extracts packet data and controls PCR, which are required to decode the system layer of the MPEG2 standard.

43.1 Features

The TSIF has the following features.

- Serial data input
- Support for TS data transfer by DMA auto request
- Acquisition of TS packets
  - Filters 38 kinds of PIDs (Packet ID) in total (The PID values of PAT and CAT packets are fixed. For PCR, video, and audio packets, the PID values are predefined.)
  - Supports all valid packet receive mode (Null packet is deleted).
  - Supports all packet receive mode including Null packet.
  - Supports duplicate packet delete mode.
  - The endian type at the TS packet data reading can be set.
  - Supports the time stamp function at the TS packet data acquisition.
- TS data analysis
- Detects random access indicator.
- Detects discontinuity indicator.
- Detects video start code and short header.
- Extraction of PCR information
- Support for system clock generation

[Legend]

MPEG: Moving picture expert group
TS: Transport stream
PID: Packet ID
PAT: Program association table
CAT: Conditional access table
PCR: Program clock reference
ES: Elementary stream

Figure 43.1 shows a block diagram of the TSIF.

The signals to transfer or control TS data are input as an input signal, and the TS packet data filtered by this module will be output as an output signal.

The serially input TS data is converted into 8-bit parallel data and the header of the TS packet is detected by the TS synchronous detection circuit. The TS filter circuit then determines and filters the PID of the TS packet according to the predefined PID table and stores the TS packet in the buffer for TS packets. Following these processes, only predefined TS packets are stored in the buffer and transferred to memories via a bus interface.

The analysis circuit of a TS header is a block that analyses the header of a TS packet, acquires the header information, and generates a trigger signal sent to other blocks. The ES data search circuit searches the start code and short header of an elementary stream (ES) contained in a TS packet. This result can be used as supplementary data to control the start timing of image decoding through the upper-level software that controls image decoding.

Based on the PCR information extracted from the TS packet, the PCR control unit outputs information needed for system clock control.

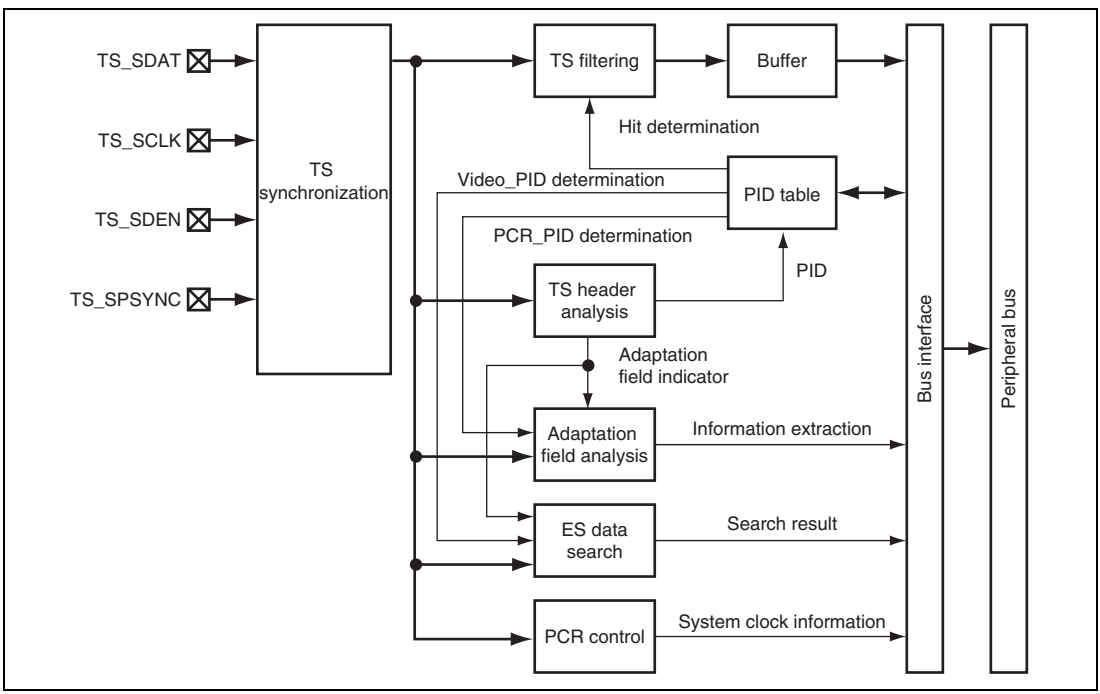


Figure 43.1 TSIF Block Diagram

43.2 Input/Output Pins

Table 43.1 shows the pin configuration.

Table 43.1 Pin Configuration

Pin Name	Function	I/O	Description
TS_SDAT	TS serial data	Input	Serial input pin of TS packet data Polarity inversion is enabled by register setting.
TS_SCK	TS serial clock	Input	Serial input clock pin Polarity inversion is enabled by register setting. The initial value is synchronized with the rising edge.
TS_SDEN	TS data enable	Input	Serial input enable signal pin Polarity inversion and On/Off setting are enabled by register setting. The initial value is On and enabled after the TS_SDEN signal is driven high.
TS_SPSYNC	TS data synchronization	Input	Byte boundary signal pin Polarity inversion is enabled by register setting. The initial value is set on a byte boundary at the rising edge.

43.3 Register Descriptions

Table 43.2 shows the TSIF register configuration. Table 43.3 shows the register states in each operating mode.

Table 43.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
TSIF control register	TSCTLR	R/W	H'A4C8 0000	32
TSIF PID data register	TSPIDR	R/W	H'A4C8 0004	32
TSIF command register	TSCMDR	R/W	H'A4C8 0008	32
TSIF interrupt status register	TSSTR	R/W	H'A4C8 000C	32
TSIF TS data register	TSTSDR	R	H'A4C8 0010	32
TSIF buffer clear register	TSBUFCLRR	R/W	H'A4C8 0014	32
TSIF interrupt enable register	TSINTER	R/W	H'A4C8 0018	32
TSIF PSCALE register	TSPSCALER	R/W	H'A4C8 0020	32
TSIF PSCALE_R register	TSPSCALERR	R/W	H'A4C8 0024	32
TSIF PCRADC mode register	TSPCRADCMDR	R/W	H'A4C8 0028	32
TSIF PCRADC register	TSPCRADCR	R/W	H'A4C8 002C	32
TSIF TR_PCRADC register	TSTRPCRADCR	R/W	H'A4C8 0030	32
TSIF D_PCRADC register	TSDPCRADCR	R/W	H'A4C8 0034	32
TSIF free-running counter	TSFRC	R/W	H'A4C8 0040	32

Table 43.3 Register States in Each Operating Mode

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
TSCTLR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSPIDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSCMDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSSTR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSTSDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSBUFCLRR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSINTER	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSPSCALER	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSPSCALERR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSPCRADCMDR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSPCRADCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSTRPCRADCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSDPCRADCR	Initialized	Initialized	Retained	Retained	Initialized	Retained
TSFRC	Initialized	Initialized	Retained	Retained	Initialized	Retained

43.3.1 TSIF Control Register (TSCTLR)

TSCTLR is a register for controlling the TSIF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	PLN GTH	TSD ATP	TSC LKP	TSV LDP	—	PSY CP	—	—	SDE NE	—	—	—	TFA	DPD MD	DREQ MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EN	PCR MD	—	—	—	—	FRC ADD	FRC STR	FRCPSC[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	PLNGTH	0	R/W	Byte Configuration Set of TS Packet Data 0: TS packet is 204-byte configuration. 1: TS packet is 188-byte configuration.
29	TSDATP	0	R/W	Input Polarity Set of TS Packet Data 0: TS data input is positive polarity. 1: TS data input is negative polarity (used after polarity inversion).
28	TSCLKP	0	R/W	Input Polarity Set of TS Clock 0: TS data clock input is positive polarity. (TS data is received at the rising edge.) 1: TS data clock input is negative polarity. (TS data is received at the falling edge.)
27	TSVLDP	0	R/W	Input Polarity Set of TS Packet Data Enable Signal 0: The TS_SDEN signal from the TS decoder is positive polarity (enabled at high). 1: The TS_SDEN signal from the TS decoder is negative polarity (enabled at low).
26	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
25	PSYCP	0	R/W	Input Polarity Set of TS Packet Data Sync Signal 0: TS packet sync signal TS_SPSYNC is positive polarity (sync byte at high). 1: TS packet sync signal TS_SPSYNC is negative polarity (sync byte at low).
24, 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22	SDENE	0	R/W	TS Packet Data Enable Signal Enabled/Disabled 0: The TS_SDEN signal from the TS decoder is enabled. 1: The TS_SDEN signal from the TS decoder is disabled.
21 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18	TFA	0	R/W	Software Reset Signal in the TSIF This bit automatically enters the reset state on startup. Release the reset by writing 1. The TSIF internal registers other than TSSTR are not initialized with this bit. When a reset is issued, confirm 0 first by reading this bit and release the reset. Note: Input TS_SCK by 8 or more cycles and set the TFA bit to 1. 0: The TSIF is reset internally (except internal registers). 1: Reset is released.

Bit	Bit Name	Initial Value	R/W	Description
17	DPDMD	0	R/W	Duplicate Data of Consecutive TS Packets (Duplicate Packets) Delete When the PID of the previous consecutive packet matches continuity_counter, the data is deleted. However, when adaptation_field_control = x0 (x: don't care), the data is not deleted. 0: Duplicate data valid mode Even when the PID of the consecutive packet and continuity_counter match, the data is not deleted. 1: Duplicate data delete mode When the PID of the previous consecutive packet and continuity_counter match, the data is deleted. Note that this is valid only when adaptation_field_control = x1 (x: don't care).
16	DREQMD	0	R/W	After TS data transfer of one packet (188 or 192 bytes) by DMAC, selects whether the PEC bit in TSBUFCLRR is set to 1 automatically. 0: DMA interrupt transfer mode After TS data transfer of one packet (188 or 192 bytes), the PEC bit is not set to 1. 1: DMA auto-transfer mode After TS data transfer of one packet (188 or 192 bytes), the PEC bit is set to 1 and the internal buffer is cleared.
15	EN	0	R/W	Endian Type Set at TS Packet Data Reading from TSTSDR 0: Big endian 1: Little endian
14	PCRMD	0	R/W	Time Indication Set in TSPCRADC and TSTRPCRADC 0: Time is indicated in a unit of 90 kHz. 1: Time is indicated in a unit of 45 kHz.
13 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9	FRCADD	0	R/W	TSFRC Value (Time Stamp) Addition at PID Matching 0: The TSFRC value (time stamp) is not added to TSTSDR. (TSTSDR is 188-byte TS packet data.) 1: The TSFRC value (time stamp) is added to the 189th to 192nd byte of TSTSDR. (TSTSDR is 188-byte TS packet data + 4-byte TSFRC.)
8	FRCSTR	0	R/W	TSFRC Operation Set 0: TSFRC stops counting. 1: TSFRC counts up.
7 to 0	FRCPSC [7:0]	H'00	R/W	Frequency Dividing Clock of TSFRC When set to the clock dividing Bϕ by n, set FRCPSC to (n – 1). Example: When set to 1-MHz clock dividing Bϕ (50 MHz) by 50, set FRCPSC to 49 (H'31).

The table below shows the conditions where the TFA bit is reflected in the TSIF.

The TFA bit is reflected in the TSIF depending on the condition of the DREQMD bit with the following timing.

DREQMD Bit	TFA Bit	Reflection Timing
0	0 → 1	Reflected accordingly by register setting
	1 → 0	Reflected accordingly by register setting
1	0 → 1	Reflected accordingly by register setting
	1 → 0	Timing when the PEC bit is automatically set to 1

43.3.2 TSIF PID Data Register (TSPIDR)

TSPIDR is a register used to set the PID value that sets the section table of the PID filter. The settings to PIDD can be reflected by writing to TSCMDR after this register is written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PIDD[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 16	PIDD[12:0]	H'0000	R/W	Byte Configuration Set of TS Packet Data These bits set the PID value that sets the section table of the PID filter. The setting of this register can be reflected by writing to TSCMDR after setting PIDD.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

43.3.3 TSIF Command Register (TSCMDR)

TSCMDR is a register that sets the value set by TSPIDR to the PID table.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PT[5]	—	—	ALLPIDMD	PIDMD	TFE	—	PPS	PT[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	PT[5]	0	R/W	Section Selection for PID Table Setting (see Table 43.4) This bit is only valid when bits PIDMD and TFE are cleared to 0.
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	ALLPIDMD	0	R/W	PID Filter Mode Set ALLPIDMD PIDMD:
24	PIDMD	0	R/W	0 0: A value set in the PID table is valid (PID filter mode). 0 1: All PIDs other than Null packet (PID = H'1FFF) are valid (all valid packet receive mode). 1 —: All PIDs including Null packet (PID = H'1FFF) are valid (all packet receive mode).

Bit	Bit Name	Initial Value	R/W	Description
23	TFE	0	R/W	Content of the PID Table Cleared 0: — 1: The content of the PID table is cleared.
22	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
21	PPS	0	R/W	Select of PID Table Set Section (see Table 43.4)
20 to 16	PT[4:0]	00000	R/W	These bits are valid only when the PIDMD bit and the TFE bit are 0.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Write a value to be set in the section table of the PID filter for TSPIDR while the PIDMD bit is 0. Then, writing one of the following values in Table 43.4 sets a PID value in the specified select section.

In addition to the PID value set above, PIDs for PAT (fixed to H'0000) and CAT (fixed to H'0001) are set.

Table 43.4 Relation among PPS, PT Setting, and PID Values

PPS	PT	Select Section
1	XXXXXX	PCR_PID
0	000000	Video
0	000001	Audio
0	000010	PID_1
0	000011	PID_2
0	000100	PID_3
0	000101	PID_4
0	000110	PID_5
0	000111	PID_6
0	001000	PID_7
0	001001	PID_8
0	001010	PID_9
0	001011	PID_10
0	001100	PID_11
0	001101	PID_12
0	001110	PID_13
0	001111	PID_14
0	010000	PID_15
0	010001	PID_16
0	010010	PID_17
0	010011	PID_18
0	010100	PID_19
0	010101	PID_20
0	010110	PID_21
0	010111	PID_22
0	011000	PID_23
0	011001	PID_24
0	011010	PID_25
0	011011	PID_26
0	011100	PID_27

PPS	PT	Select Section
0	011101	PID_28
0	011110	PID_29
0	011111	PID_30
0	100000	PID_31
0	100001	PID_32
0	100010	PID_33

Note: X: Don't care

43.3.4 TSIF Interrupt Status Register (TSSTR)

TSSTR is a register that indicates the internal status of the TSIF. Each bit in TSSTR can be cleared by writing 0. Bits other than the SYNCF bit are set in TSSTR when packet reception has been completed. After a bit is set to 1, it is cleared to 0 only when 0 is written to it; be sure to clear it before the next packet reception is completed. Each bit in TSSTR (except the SYNCF bit) is initialized by the TFA bit in TSCTLR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PID[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADCF	PIDF	STOF	RANDF	DISCF	SYNCF	OFEF	—	VSCF	VSHF	TSIF INTF
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	—	0	0	0
R/W:	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R/(W)*	R	R/(W)*	R/(W)*	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28 to 16	PID[12:0]	H'0000	R/(W)*	These bits indicate the PID of the TS packet input immediately before.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	ADCF	0	R/(W)*	PCRADC Update Flag [Setting condition] Indicates that PCRADC was updated (the PCR packet was input). [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.
9	PIDF	0	R/(W)*	Relevant PID Packet Reception Flag [Setting condition] Indicates that the set PID packet was received in the PID filter mode (ALLPIDMD = 0, PIDMD = 0). [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.

Bit	Bit Name	Initial Value	R/W	Description
8	STOF	0	R/(W)*	Packet Reception End Flag when Interrupt Generating [Setting condition] Indicates that the status of bits was updated (except this bit) by the completion of the next packet reception with the interrupt generated. [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.
7	RANDF	0	R/(W)*	random_access_indicator Flag [Setting condition] Indicates that random_access_indicator of the TS packet input immediately before is 1. [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.
6	DISCF	0	R/(W)*	PCR Discontinuity Flag [Setting condition] Indicates that discontinuity_indicator of the PCR packet input immediately before shows the PCR discontinuous state. [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.
5	SYNCF	1	R	Internal Synchronous/Asynchronous Mode State caused by Detection of the TS Packet Sync Signal When the sync signal (H'47) is detected three times consecutively in asynchronous mode, the TSIF enters the synchronous mode. When the sync signal is not detected four times consecutively in synchronous mode, it enters the asynchronous mode. The write value should always be 0. 0: Indicates that the TSIF is in synchronous mode (the TS packet is synchronized). 1: Indicates that the TSIF is in asynchronous mode (the TS packet is not synchronized).
4	OFEF	0	R/(W)*	TSIF Internal Buffer Overflow [Setting condition] Indicates that an overflow error occurred in the internal buffer. [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.

Bit	Bit Name	Initial Value	R/W	Description
3	—	Undefined	R	Reserved The write value should always be 0.
2	VSCF	0	R/(W)*	Video Packet Start Code Detection Flag [Setting condition] Indicates that the TS packet input immediately before is a video packet and its payload contains a start code (B'0000 0000 0000 0000 0000 0001 xxxx xxxx). [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.
1	VSHF	0	R/(W)*	Video Packet Short Header Detection Flag [Setting condition] Indicates that the TS packet input immediately before is a video packet and its payload contains a short header (B'0000 0000 0000 0000 1000 00xx xxxx xxxx). [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.
0	TSIFINTF	0	R/(W)*	TSIF Transfer Request Flag Indicates that the internal buffer contains one packet (188 bytes) of TS data. This bit is automatically cleared by reading 32 bytes of TSTSDR. [Setting condition] Indicates that the internal buffer contains one packet of TS data. [Clearing condition] 0 is written or the condition that TFA = 0 is reflected.

Note * The only value that can be written is 0.

43.3.5 TSIF TS Data Register (TSTSDR)

TSTSDR is a register to read TS packet data extracted by the set PID.

The endian type is set by the EN bit in TSCTLR.

In addition, when the FRCADD bit in TSCTLR is set to 1, the TSFRC value (time stamp) is added to the 189th to 192nd byte of TSTSDR at PID matching and can be used as the packet management data.

After reading one packet (188 bytes) of TS data from this register, the PEC bit in TSBUFCLRR should be set to 1 (unless it is in DMA auto-transfer mode).

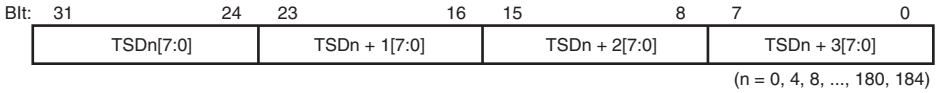
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSD[31:16]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSD[15:0]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

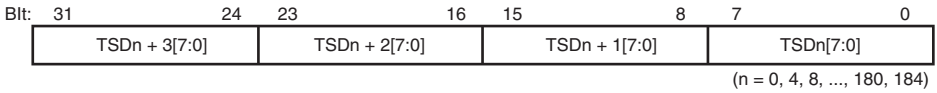
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TSD [31:0]	Undefined	R	When the FRCADD bit in TSCTLR = 0 — 1st to 188th bytes: TS packet data extracted by the set PID When the FRCADD bit in TSCTLR = 1 • 1st to 188th bytes: TS packet data extracted by the set PID • 189th to 192nd bytes: TSFRC value* at PID matching

- TS packet data format

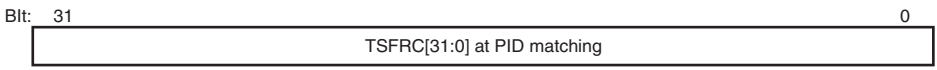
[Big endian] When the EN bit in TSCTLR = 0



[Little endian] When the EN bit in TSCTLR = 1



Note: * Reading the TSFRC value in the 189th to 192nd byte of TSTSDR at PID matching
As TSFRC is the value of 32-bit counter, it is always in a unit of longword.



43.3.6 TSIF Buffer Clear Register (TSBUFCLRR)

TSBUFCLRR is a register that clears the internal buffer of the TSIF after the completion of the TS packet data reading.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PEC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PEC	0	R/W	Completion of TS Packet Data Reading This bit should be set to 1 immediately after the completion of the TS packet data reading. When one packet of TS data is discarded because of a transfer error or such, this bit should be set to 1. This bit is automatically cleared to 0 after it has been set to 1. Do not access this register while the DREQMD bit in TSCTLR is set to 1 or before the completion of the TS packet data reading. 1: Clears the internal buffer after the TS packet is transferred (automatically returns to 0 after 1 is set). When one packet of data transferred is discarded because of a transfer error or such, 1 should always be set. [Setting condition] 1 is written. [Clearing condition] This bit is automatically cleared.

43.3.7 TSIF Interrupt Enable Register (TSINTER)

TSINTER is a register that controls an interrupt request from the TSIF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADCE	PIDE	STOE	RANDE	DISCE	SYNCE	OFEE	—	VSCE	VSHE	TSIF INTE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	ADCE	0	R/W	0: Disable an interrupt when PCRADC is updated (PCR packet is input). 1: Enables an interrupt when PCRADC is updated (PCR packet is input).
9	PIDE	0	R/W	0: Disable an interrupt when the set PID packet is received in PID filter mode. 1: Enables an interrupt when the set PID packet is received in PID filter mode.
8	STOE	0	R/W	0: Disable an interrupt when the status is updated (interrupt time is over) by the completion of the next packet reception with the interrupt generated. 1: Enables an interrupt when the status is updated (interrupt time is over) by the completion of the next packet reception with the interrupt generated.
7	RANDE	0	R/W	0: Disable an interrupt when random_access_indicator of the TS packet input immediately before is 1. 1: Enables an interrupt when random_access_indicator of the TS packet input immediately before is 1.

Bit	Bit Name	Initial Value	R/W	Description
6	DISCE	0	R/W	0: Disable an interrupt when discontinuity_indicator of the PCR packet input immediately before indicates PCR discontinuity. 1: Enables an interrupt when discontinuity_indicator of the PCR packet input immediately before indicates PCR discontinuity.
5	SYNCE	0	R/W	0: Disable an interrupt when the TSIF is in asynchronous mode. 1: Enables an interrupt when the TSIF is in asynchronous mode.
4	OFEE	0	R/W	0: Disable an interrupt when an overflow error occurs in the internal buffer. 1: Enables an interrupt when an overflow error occurs in the internal buffer.
3	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	VSCE	0	R/W	0: Disable an interrupt when the start code of a video packet is detected in the TS packet input immediately before. 1: Enables an interrupt when the start code of a video packet is detected in the TS packet input immediately before.
1	VSHE	0	R/W	0: Disable an interrupt when the short header of a video packet is detected in the TS packet input immediately before. 1: Enables an interrupt when the short header of a video packet is detected in the TS packet input immediately before.
0	TSIFINTE	0	R/W	0: Disable an interrupt when the internal buffer contains one packet (188 bytes) of TS data. 1: Enables an interrupt when the internal buffer contains one packet (188 bytes) of TS data.

43.3.8 TSIF PSCALE Register (TSPSCALER)

TSPSCALER is a register that sets counter values 1 and 2 of the internal prescaler. When EXT_MODE = B'0, set the value of the PSCALE2 bit so that the value is equivalent to the value of the PSCALE1 bit plus 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PSCALE2[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PSCALE1[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	PSCALE2 [9:0]	H'000	R/W	These bits set counter value 2 of the internal prescaler.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	PSCALE1 [9:0]	H'000	R/W	These bits set counter value 1 of the internal prescaler.

43.3.9 TSIF PSCALE_R Register (TSPSCALERR)

TSPSCALERR is a register that sets the ratio of counter values 1 and 2 of the internal prescaler.

When EXT_MODE = B'0, set the values of the PSCALE_R1 bit and the PSCALE_R2 bit so that the sum of these two bits is 9.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PSCALE_R2[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PSCALE_R1[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	PSCALE_R2[3:0]	H'0	R/W	These bits set the ratio of counter value 2 of the internal prescaler.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	PSCALE_R1[3:0]	H'0	R/W	These bits set the ratio of counter value 1 of the internal prescaler.

After setting TSPSCALERR while TSPSCALER is H'0000 0000, set TSPSCALER.

When EXT_MODE = B'0, set TSPSCALER and TSPSCALERR so that the following equation is satisfied.

$$\left(\frac{\text{PSCALE\_R1} + \text{PSCALE\_R2}}{\text{PSCALE1} \times \text{PSCALE\_R1} + \text{PSCALE2} \times \text{PSCALE\_R2}} \right) \times B\phi = 90\ \text{kHz}$$

PSCALE_R1 + PSCALE_R2 = 9
PSCALE2 = PSCALE1 + 1

The examples below show register settings of TSPSCALER and TSPSCALERR corresponding to each operating frequency of the TSIF. When EXT_MODE = B'1, operating frequencies listed in Table 43.5 are only supported.

Table 43.5 Setting Example of TSPSCALER and TSPSCALERR Corresponding to Each Operating Frequency

- EXT_MODE = B'0

Operating Frequency [MHz]	PSCALE2	PSCALE1	PSCALE_R2	PSCALE_R1
10.0	H'070	H'06F	H'1	H'8
12.0	H'086	H'085	H'3	H'6
12.6	H'08C	H'08B	H'9	H'0
12.8	H'08F	H'08E	H'2	H'7
14.4	H'0A1	H'0A0	H'0	H'9
16.0	H'0B2	H'0B1	H'7	H'2
19.2	H'0D6	H'0D5	H'3	H'6
20.0	H'0DF	H'0DE	H'2	H'7
27.0	H'12D	H'12C	H'0	H'9
33.0	H'16F	H'16E	H'6	H'3
48.0	H'216	H'215	H'3	H'6
50.0	H'22C	H'22B	H'5	H'4
54.0	H'259	H'258	H'0	H'9
66.0	H'2DE	H'2DD	H'3	H'6

- EXT_MODE = B'1

Operating Frequency [MHz]	PSCALE2	PSCALE1	PSCALE_R2	PSCALE_R1
13.5	H'000	H'001	H'0	H'1
27.0	H'000	H'002	H'0	H'1
40.5	H'000	H'003	H'0	H'1
54.0	H'000	H'004	H'0	H'1

Note: The bus clock (Bϕ) is used for the operating frequency.

43.3.10 TSIF PCRADC Mode Register (TSPCRADCMDR)

TSPCRADCMDR is a register that sets the PCRADC mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	EXT_MODE	TR_SEL[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	EXT_MODE	0	R/W	Sets whether PCR_extension is included in the calculation of D_PCRADC. The EXT_MODE bit can be set to 1 only when this block operates at (13.5 × n) MHz (n = 1, 2, 3, 4). 0: PCR_extension is not included in the D_PCRADC calculation. 1: PCR_extension is included in the D_PCRADC calculation.
4 to 0	TR_SEL[4:0]	H'00	R/W	These bits select a trigger signal for PCRADC to be input to the TSIF. Do not set two or more bits to 1. 00000: A trigger signal is not selected. 10000: Setting prohibited 01000: An interrupt signal from DMAC0 channel 3 is selected as a trigger. 00100: An interrupt signal from DMAC0 channel 2 is selected as a trigger. 00010: An interrupt signal from DMAC0 channel 1 is selected as a trigger. 00001: An interrupt signal from DMAC0 channel 0 is selected as a trigger. Other than above: Setting prohibited

43.3.11 TSIF PCRADC Register (TSPCRADCR)

TSPCRADCR is a register that indicates the PCR value incremented by the internal clock (indicates time in a unit of 90 kHz).

Counting starts when a PCR packet is input after a reset is canceled. After that, the counter value is corrected against the PCR value in the stream every time a PCR packet is input. This register is updated even in packet asynchronous state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PCRADC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PCRADC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	PCRADC [31:0]	H'0000 0000	R	These bits indicate the PCR value incremented by the internal clock. When the PCRMD bit in TSCTLR = 0: These bits indicate time in a unit of 90 kHz. When the PCRMD bit in TSCTLR = 1: These bits indicate time in a unit of 45 kHz.

43.3.12 TSIF TR_PCRADC Register (TSTRPCRADCR)

TSTRPCRADCR indicates the value of PCRADC read by a trigger signal for PCRADC selected by the TR_SEL bit (indicates time in a unit of 90 kHz).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TR_PCRADC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TR_PCRADC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TR_PCRADC [31:0]	H'0000 0000	R	These bits indicate the PCRADC value read by a trigger signal for PCRADC selected by the TR_SEL bit. When the PCRMD bit in TSCTLR = 0: These bits indicate time in a unit of 90 kHz. When the PCRMD bit in TSCTLR = 1: These bits indicate time in a unit of 45 kHz.

43.3.13 TSIF D_PCRADC Register (TSDPCRADCR)

TSDPCRADCR indicates the value of the stream PCR minus TSPCRADC in a 2's complement value. This register value becomes valid when the second PCR packet is input after a reset is canceled.

When the value of the stream PCR minus TSPCRADC is greater than 32767 or smaller than –32768, bits D_PCRADC[31:16] indicate H'8000. (D_PCRADC is not affected by the PCRMD bit setting in TSCTLR. D_PCRADC[31:16] and D_PCRADC[15:0] always indicate a clock difference in a unit of 90 kHz and 27 MHz, respectively.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	D_PCRADC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D_PCRADC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	D_PCRADC [31:0]	H'0000 0000	R	These bits indicate the value of the stream PCR minus TSPCRADC.

43.3.14 TSIF Free-Running Counter (TSFRC)

TSFRC is a 32-bit free-running counter that increments by the frequency dividing clock, set by bits FRCPSC[7:0] in TSCTLR. TSFRC starts counting after the FRCSTR bit in TSCTLR is set to 1.

TSFRC is initialized to H'0000 0000 by a power-on reset, a manual reset, or U-standby. TSFRC is not initialized when the TFA bit in TSCTLR = 0.

The TSFRC value (time stamp) is added to the 189th to 192nd byte of TSTSDR at PID matching when the FRCADD bit in TSCTLR = 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSFRC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSFRC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TSFRC [31:0]	H'0000 0000	R/W	These bits indicate the value of free-running counter.

43.4 Operation

43.4.1 TS Data Protocol

The TS packet supported by this LSI consists of 188-byte execution data including the sync byte and 16-byte Reed-Solomon code (parity data) for an error correction, with configuration of 204 bytes in total (Figure 43.2). The TSIF supports TS data formats of both 188 bytes (without parity) and 204 bytes (with parity).

The data input pins consist of the following.

- Serial data (TS_SDAT) input
- Serial data clock (TS_SCK) input
- Synchronous (TS_SPSYNC) input
- Enable (TS_SDEN) input

Depending on the specifications of the LSI to be connected, the polarity of these pins can be switched by software (Figure 43.3).

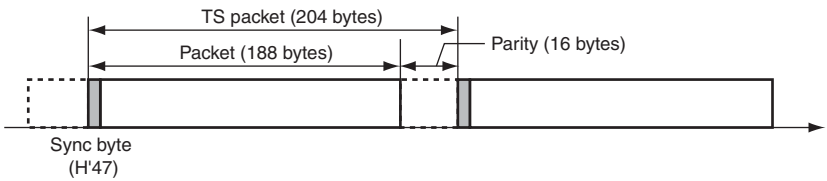


Figure 43.2 MPEG2-TS Packet Configuration

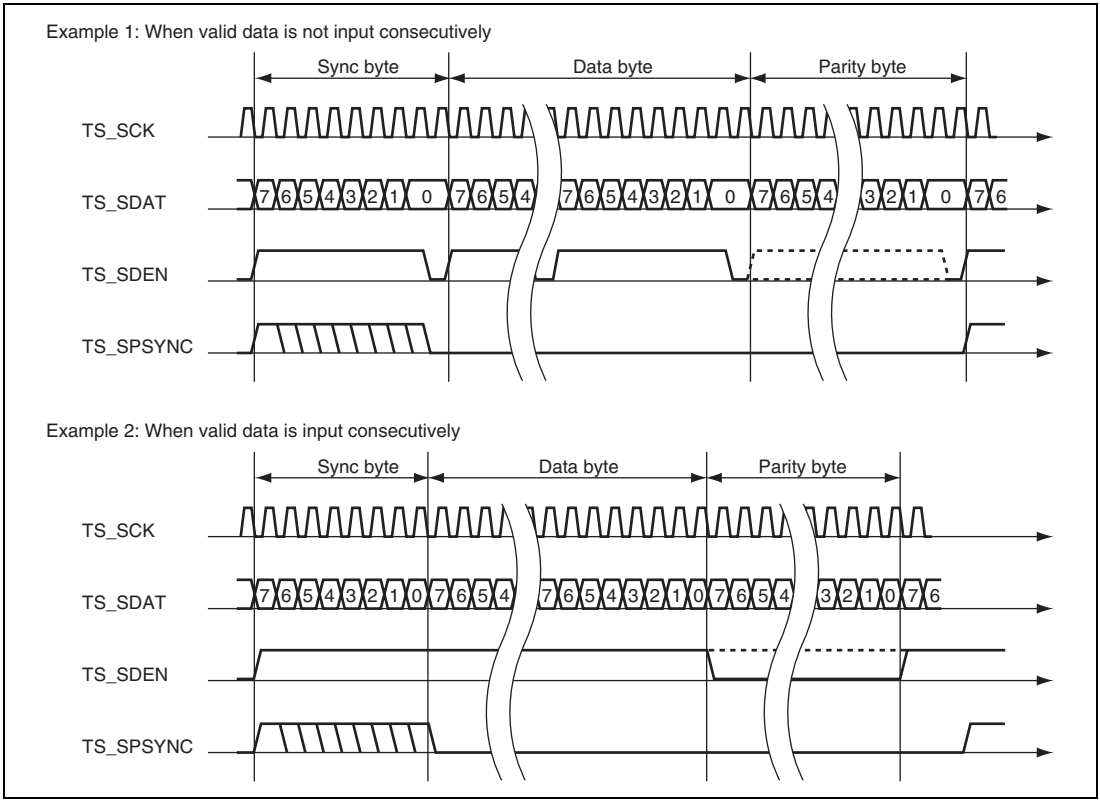


Figure 43.3 Timing of TS Data Serial Input

43.4.2 Synchronous Processing Unit

The function of synchronous processing is to synchronize packets for carrying out the subsequent PID filtering and such.

Though TS packet data is constantly sent from a terrestrial station, the receiving terminal starts receiving the data regardless of the timing of the sender. This causes the receiving terminal to receive an incomplete packet. The synchronous processing is essential to identifying the start of TS packet data that are not played from the start and accurately process various data in the packet.

Moreover, as the signal quality of input data may deteriorate according to the transmission status, the TS packet data received via a tuner, OFDM demodulator, and the like may contain errors. The synchronous processing includes protection against such errors.

When the receiving terminal is started up, it searches for the sync byte (H'47). If the sync byte is detected, the receiving terminal checks whether or not the sync byte will appear again after one packet cycle (188 bytes or 204 bytes). If the sync byte appears twice consecutively, synchronization is confirmed. Once synchronized, even when the sync byte does not appear at a fixed position, an error is not generated immediately and the regular processing is continued.

When the sync byte has not appeared 4 times consecutively, the receiving terminal determines that synchronization is lost. Once synchronization is lost, the process for synchronization starts in the same way as when the receiving terminal is started up.

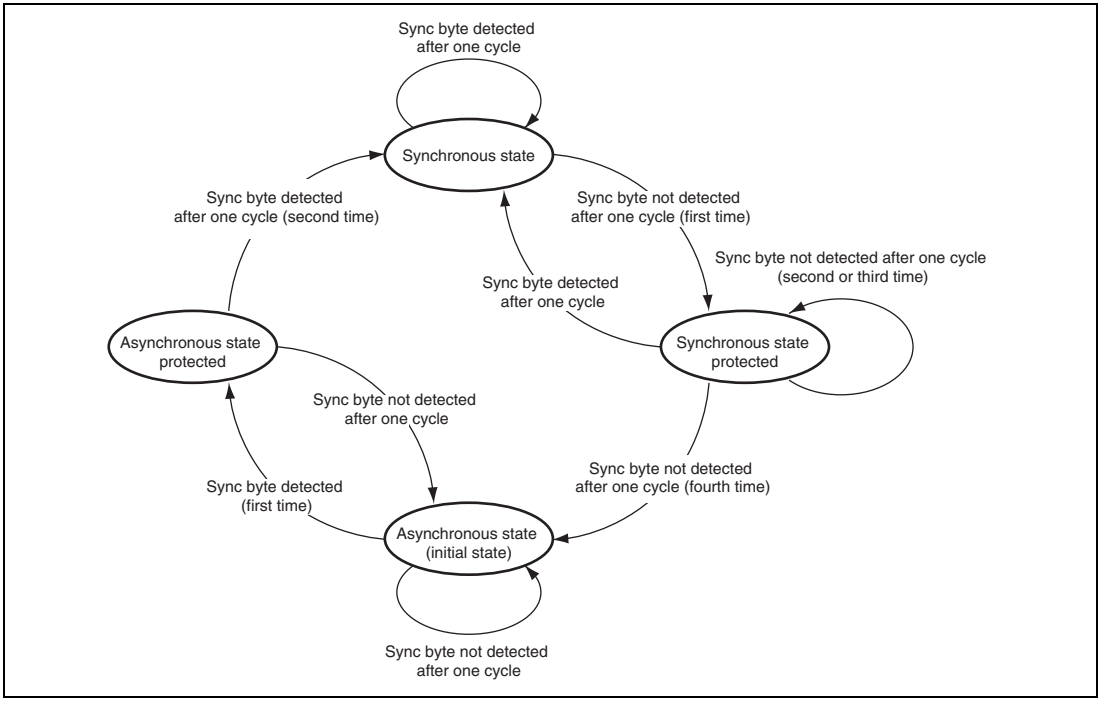


Figure 43.4 Detection State Configuration of Sync Byte

43.4.3 PID Filter Unit

PID filtering checks the PIDs of TS packets and if they match the predefined PID table values, the packets are stored in the buffer.

A total of 26 values can be registered as PID values. However, the values for PAT (H'0000) and CAT (H'0001) are fixed and not changeable. The PID values for the PCR packet, video packet, and audio packet should be specified in the dedicated sections, respectively. Another 21 PID values can be specified in any of the sections.

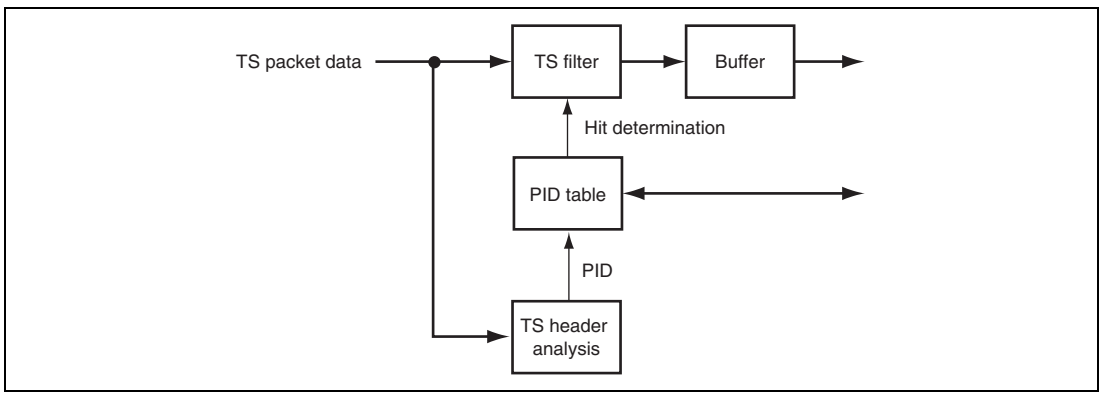


Figure 43.5 Block Diagram of PID Filtering

Table 43.6 PID Table

No.	Section	PID Value (12 Bits)	Description
1	Video	Any setting	Dedicated for video packet
2	Audio	Any setting	Dedicated for audio packet
3	PID1	Any setting	For any packet
4	PID2	Any setting	For any packet
5	PID3	Any setting	For any packet
6	PID4	Any setting	For any packet
7	PID5	Any setting	For any packet
8	PID6	Any setting	For any packet
9	PID7	Any setting	For any packet
10	PID8	Any setting	For any packet
11	PID9	Any setting	For any packet
12	PID10	Any setting	For any packet
13	PID11	Any setting	For any packet
14	PID12	Any setting	For any packet
15	PID13	Any setting	For any packet
16	PID14	Any setting	For any packet
17	PID15	Any setting	For any packet
18	PID16	Any setting	For any packet
19	PID17	Any setting	For any packet
20	PID18	Any setting	For any packet
21	PID19	Any setting	For any packet
22	PID20	Any setting	For any packet
23	PID21	Any setting	For any packet
24	PID22	Any setting	For any packet
25	PID23	Any setting	For any packet
26	PID24	Any setting	For any packet
27	PID25	Any setting	For any packet
28	PID26	Any setting	For any packet
29	PID27	Any setting	For any packet
30	PID28	Any setting	For any packet

No.	Section	PID Value (12 Bits)	Description
31	PID29	Any setting	For any packet
32	PID30	Any setting	For any packet
33	PID31	Any setting	For any packet
34	PID32	Any setting	For any packet
35	PID33	Any setting	For any packet
36	PAT	Fixed to H'0000	Dedicated for PAT packet
37	CAT	Fixed to H'0001	Dedicated for CAT packet
38	PCR	Any setting	Dedicated for PCR packet

43.4.4 Bitstream Search Unit

The function of the bitstream search unit is to check the contents of the PCR and video packets to reduce the load of software. The functions listed below are available.

1. PCR discontinuity check

   PCR discontinuity (discontinuity_indicator) is automatically detected and reflected in the status register. This is to reduce the delay in software processing if a PCR packet is input. PCR is a reference clock sent from a base station (broadcasting station) and may be switched to another PCR because of program changes and so on.
2. PCR_flag detection

   The PCR_flag is detected from the adaptation_field of the TS packet and reflected in the status register.
3. random_access_indicator detection

   The random_access_indicator is detected from the adaptation_field of the TS packet and reflected in the status register.
4. Duplicate packet delete

   The PID and continuity_counter are checked from consecutive TS packets to prevent duplicate TS data from being stored in the buffer.
5. Start code check

   When a start code (H'0000 01xx) is contained in a video stream, it is reflected in the status register. The start code marks the boundary in image data of the MPEG standard and is applied at the start of a sequence, image, and such.
6. Short header check

   When a short header (B'0000 0000 0000 0000 1000 00xx xxxx xxxx) is contained in a video stream, it is reflected in the status register.

These processes are performed in the adaptation field analysis unit and the ES data search unit shown in Figure 43.6.

At the time of TS header analysis for PID filtering, the presence of an adaptation field is analyzed. When hit information of a PCR packet or video packet in the PID table is input, the necessary process at each block is initiated.

Each result is reflected in the status register.

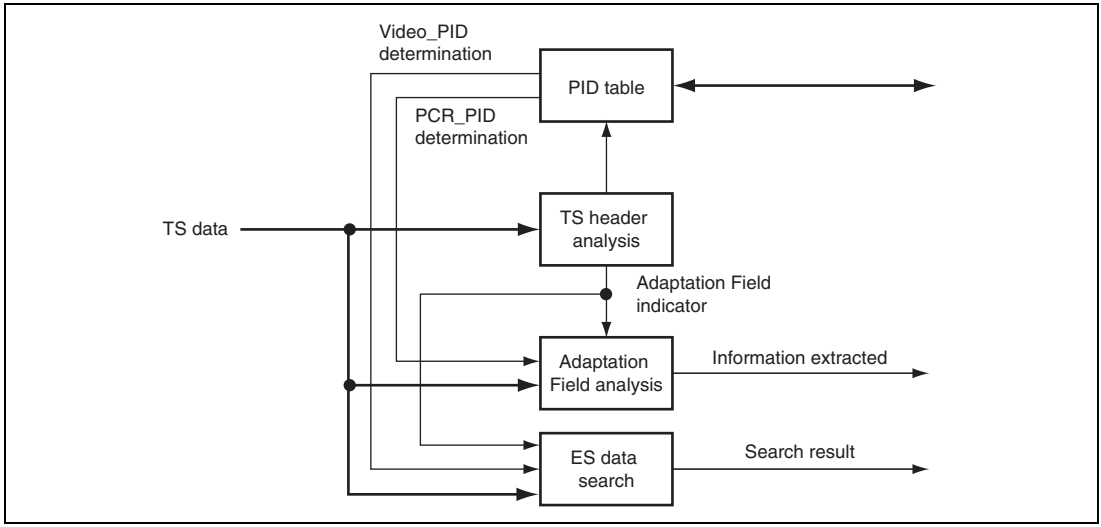


Figure 43.6 Block Diagram of Stream Search Unit

43.4.5 Process Flow

Figure 43.7 shows an outline of the process flow in the TSIF module.

At power-on, the TSIF is in the reset state and does not generate any signals until the initialization is completed. After a TS packet ID (PID) to be extracted by the TSIF is set in the PID table and other necessary settings have been completed, software reset is released and the TSIF is started up while a desired interrupt is allowed at the same time. After this, the TSIF waits until the necessary TS packet is extracted.

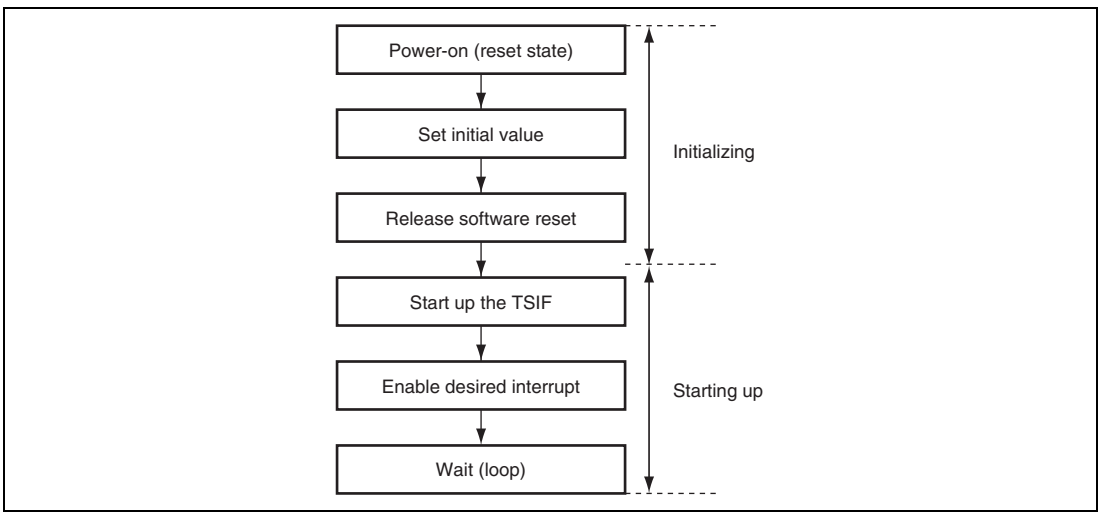


Figure 43.7 TSIF Process Flow

Two types of TS packet transfer are available as described below.

(1) DMA Interrupt Transfer Mode (Figure 43.8)

The interrupt routine transfers the packet data stored in the buffer to a queue created in memory. Multiple queues must be created depending on the values of PIDs (e.g., video or audio). When an interrupt is generated, the TSSTR register is read first. If a status error is detected during reading, and the packet data is going to be discarded, the PEC bit in the TSBUFCLRR register should be set. The value of the PID bit in the TSSTR register should then be checked.

For example, if the packet is a video packet, it will be transferred to a video packet queue by DMA and so on. After 188-byte data or 192-byte data (TS data + TSFRC) is transferred, the PEC bit must be set to clear the internal buffer.

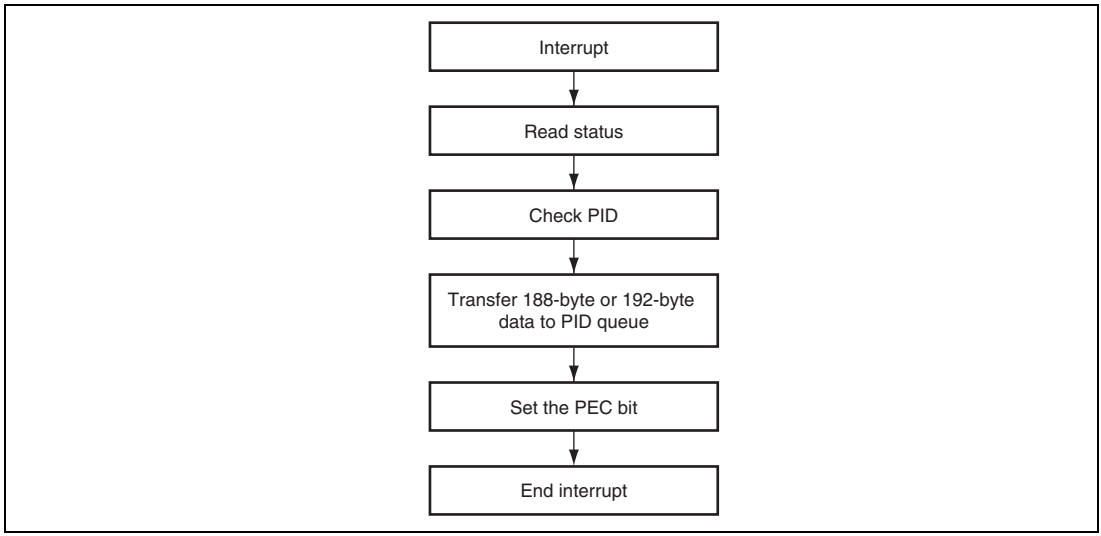


Figure 43.8 TSIF Interrupt Process Flow

(2) DMA Auto-Transfer Mode

To use this mode, the necessary setting should be made in the DMAC block before starting up the TSIF. When one packet (188 bytes) or one packet + the TSFRC value (192 bytes in total) of TS data is input, the TSIF automatically issues a transfer request signal to the DMA and starts DMA transfer. Using this mode reduces the frequency of interrupts.

Section 44 Sound Processing Unit 2 (SPU2)

This section is covered by a non-disclosure agreement. Please contact a Renesas Electronics sales representative for details.

The sound processing unit 2 (SPU2) is a dedicated unit for sound and audio processing, which consists of two DSPs that are specially designed to process audio CODECs such as AAC or MP3 and to perform pre- and post-sound processing and two bus bridges (SPU–DMAC) each of which has a DMAC that can be started by the DSP and system CPU.

The SPU2 is enhanced to support direct access from the system CPU to DSP I/O peripheral modules and access from the DSP to SPU2 external spaces in order to perform pre- and post-processing as well as audio CODEC processing by the DSP.

When the functions of SPU2 are to be used, set bit 7 in the physical address space control register (PASCR) to 1. For details of the PASCR, refer to section 7, Memory Management Unit (MMU). Furthermore, use frequency control register A (FRQCRA) to set the ratio between the frequencies of Iϕ and Sϕ as follows. For details of the FRQCRA register, refer to section 17, Clock Pulse Generator (CPG).

Iϕ : Sϕ = N : 1 N = 1, 2, 3 (Setting where N = 4 or more are prohibited)

44.1 Features

Table 44.1 shows the SPU2 functions and features.

Table 44.1 SPU2 Functions and Features

Component	Item	Description
DMAC	Number of channels	• 3 channels
	Transfer data length	• SHwy: 1 longword (4 bytes), 8 longwords (32 bytes) • DSP-P RAM: 32 bits/32 bits × 8 bursts • DSP-X/Y RAM: 24 bits 16 bits × 16/24 bits x 8 bursts (aligned mode) 24 bits × 10/11 bursts (seamless mode)
	Max. transfer count	• 65,535 (16 bits)
	Address mode	• Dual address mode
	Addressing	• Upper 9 bits fixed (base address) • Fixed/23-bit (8-Mbyte) increment
	Transfer request	• Auto request
	Priority order	• Fixed (Ch0>Ch1>Ch2)
	Interrupt request	• Interrupt requests can be generated to the SH and DSP at the end of data transfer. Both of interrupt sources and interrupt requests can be masked.
	Data swap	• DMA transfer data can be swapped. (Byte swap, word swap, byte-word swap)
	• A DMAC that supports all the above functions is integrated with a DSP. • The SPU2 has two sets of DSPs and DMACs. • With more than one DSP, transfers between the DSP-X/Y and memory are complete within the SPU without using the SHwy.	
Bus bridge	DSP interface	• Supports DMA transfers between the external memory and the DSP. • Provides direct access to the DSP memory from the CPU in target through mode. • Provides access to the external space (such as the CPU registers) of the SPU2 from the DSP in bus-through mode. • Provides access to the SPU2 registers from the DSP.
	IO interface	• Arbitrates access to the SPU2 registers and external peripheral modules from the DSP and the CPU. • Can be connected with peripheral modules.

Section 45 FIFO-Buffered Serial Interface (FSI)

The FIFO-buffered serial interface (FSI) is a sound I/O interface that supports both analog and digital connections. It has an output function that outputs 24-bit data output from the audio processing circuit with 3 lines: LR clock, bit clock, and serial clock lines and an input function that converts input 3-line data to 24-bit data. It supports continuous I/O operation by managing FIFO states and generating interrupts.

This module is a peripheral circuit of the SPU2 (sound processing unit 2) audio processing circuit.

45.1 Features

The functions and features are listed in table 45.1.

Table 45.1 List of FSI Functions

Item	Contents	Details
Operating frequency	Sampling frequency (fs)	8kHz to 96kHz (max.)
	Audio clock	Clock from which the LR and bit clocks are derived (64 fs, 128 fs, 256 fs, 512 fs) 512 kHz to 25 MHz (max.) (64 × 8 kHz to 512 × 96kHz)
	FSI operating frequency	4 MHz to 83.4 MHz
DSP/IO	Data bus width	24 bits
Supported I/O format	Monaural (1 channel)	MONO, MONO Delay
	Stereo (2 channels)	I2S, PCM
	Multi-channel (1 to 8 channels)	TDM, TDM Delay

Item	Contents	Details
Output interface	3-line serial output (× 2: ports A and B)	Master operation (generates the LR and bit clocks from the audio clocks: 64 fs, 128 fs, 256 fs, 512 fs) LR clock (fs): 8 kHz to 9 6kHz Bit clock: 32 bit/fs, 64 bit/fs, 128 bit/fs, 256 bit/fs, 512 bit/fs (monaural) 64 bit/fs, 128 bit/fs, 256 bit/fs, 512 bit/fs (stereo) 256 bit/fs, 512 bit/fs (multi-channel)
		Slave operation (can operate with the external LR and bit clocks) LR clock (fs): 8 kHz to 96 kHz Bit clock: 32 bit/fs, 64 bit/fs, 128 bit/fs, 256 bit/fs, 512 bit/fs (monaural) 64 bit/fs, 128 bit/fs, 256 bit/fs, 512 bit/fs (stereo) 256 bit/fs, 512 bit/fs (multi-channel)
Input interface	3-line serial input (× 2: ports A and B)	Master operation Supports the same LR and bit clocks as the output interface.
		Slave operation Supports the same LR and bit clocks as the output interface.
FIFO	FIFO status	State, number of stored sample data, overflow, underflow
	FIFO clear	Clears the address pointer.
	Interrupt source condition select	Output operation: (1) Empty. (2) More than half of data areas are free. (3) Data areas for more than 1 sample are free. Input operation: (1) Full. (2) More than half of data areas are occupied. (3) Data areas for more than 1 sample are occupied.
	FIFO word counts	256 words

Item	Contents	Details
Interrupt	Interrupt control	Can generate interrupt requests externally. Supports masks for interrupt sources and interrupt signals.
MUTE	MUTE mode	Supports normal MUTE and zero cross MUTE.

Figure 45.1 shows a block diagram of the FSI.

Serial port B has the same structure and interface as serial port A.

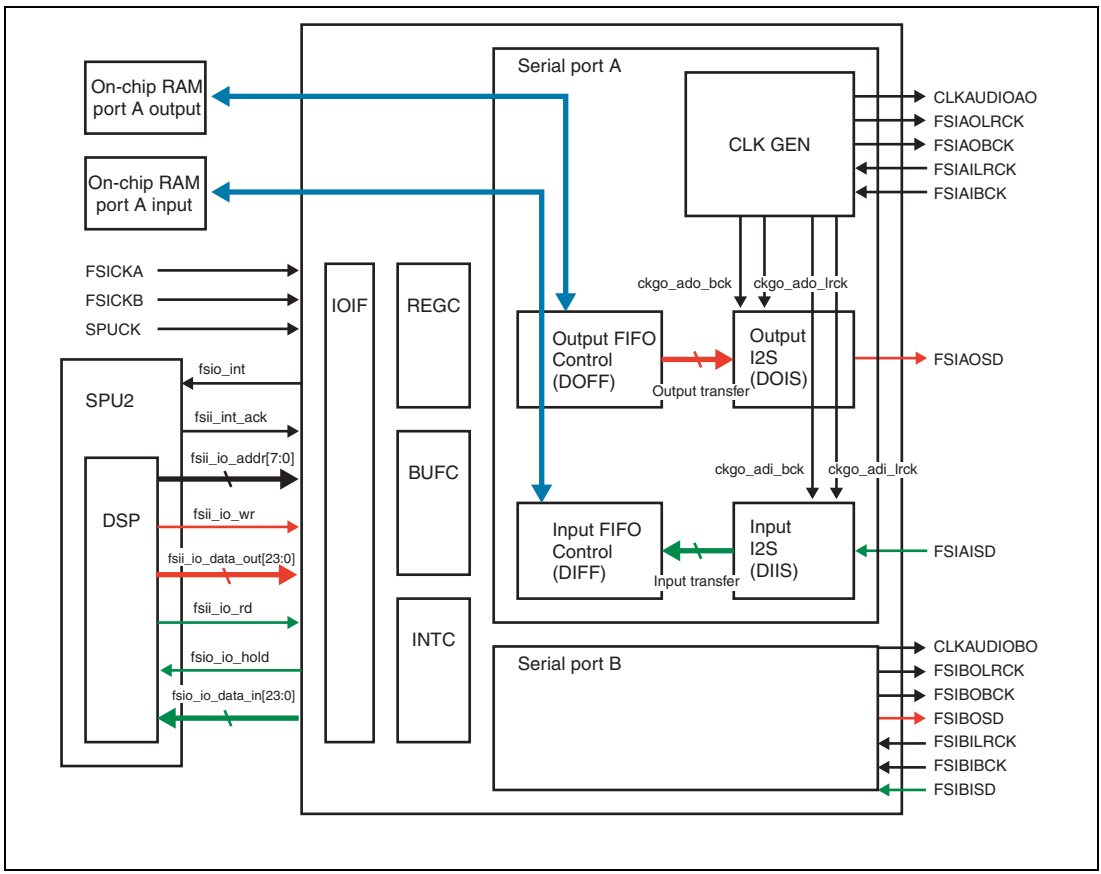


Figure 45.1 Block Diagram

The followings are the brief descriptions of each block.

(1) Serial Port A

Serial port A is a module that consists of 5 modules: CLK GEN, data output FIFO control, data input FIFO control, data output I2S, and data input I2S, and performs I/O operations for audio data

(a) CLK GEN

CLK GEN selects the master clock generated internally or the slave clock supplied external and outputs it to the internal modules (DOIS, DIIS).

(b) Data Output FIFO Control (DOFF)

A FIFO to which data from the DSP is written is called output FIFO. The DOFF controls access to the DP-RAM. The FIFO states are reflected on the FIFO status register (A_DOFF_ST).

The output FIFO control register (A_DOFF_CTL) can be used to select the condition for generating the FIFO interrupt source (A_DOFF_CTL.IHC).

The FIFO clear (A_DOFF_CTL.CLR) initializes the access pointer to the on-chip RAM and the error history.

(c) Data Output I2S (DOIS)

The DOIS controls a counter based on the rising and falling edges of the LR clock generated from the CLK GEN to request data to the DOFF.

It also determines the channel number associated with data and stores data into the corresponding channel buffer. It reads data at the next LR clock and shifts it to the left to output serial data.

(d) Data Input FIFO Control (DIFF)

A FIFO to which external data is written is called input FIFO. The DIFF controls access to the DP-RAM. The FIFO states are reflected on the FIFO status register (A_DIFF_ST).

The input FIFO control register (A_DIFF_CTL) can be used to select the condition for generating the FIFO interrupt source (A_DIFF_CTL.IHC).

The FIFO clear (A_DIFF_CTL.CLR) initializes the access pointer to the on-chip RAM and the error history.

(e) Data Input I2S (DIIS)

The DIIS controls a counter based on the rising and falling edges of the LR clock generated from the CLK GEN to store the parallel converted signal to each channel buffer. The stored data are concatenated the channel number and written to input FIFOs.

(2) Serial Port B

Serial port B has the same structure and interface as serial port A and operates independently.

(3) Other Management Modules

These modules control registers for major functions.

(a) INTC

This module controls interrupt signals. It aggregates 4 interrupt sources generated internally and outputs as one interrupt signal externally. Interrupt events are triggered based on the FIFO capacity. The condition can be changed.

Interrupt source and interrupt signal masks can be used to control how to handle those triggered events.

(b) BUFC

The BUFC block controls data reads/writes between the DSP and FIFOs. To read input data continuously, data are prefetched from FIFOs. Data are output with 4 waits on the initial read, but the following reads can be performed continuously with one wait by controlling the buffer pointer and reading data stored in buffers.

(c) REGC

This module manages registers that are not used by INTC and BUFC. It is used to manage format, clock, and MUTE settings.

(d) IOIF

This module selects read data of INTC, BUFC, and REGC.

Table 45.2 summarizes the number of wait cycles for the registers accessible from the DSP-IO.

Table 45.2 Wait Cycles for FSI Register Read/Write

Register	Read	Write
A_DIDT	Initial read (4 waits)	Prohibited
B_DIDT	2nd and following reads (1 wait)	
Other registers	1 wait	No wait

45.2 Input/Output Pins

Table 45.3 shows the FSI pin configurations.

Table 45.3 Pin Configurations

FSI Block	Pin Name	Function	Input/Output	Description
Port A	FSIAOLRCK	Port A sound output L/R clock	Output	Sound output L/R clock pin (master)
	FSIAOBCK	Port A sound output bit clock	Output	Sound output bit clock pin (master)
	FSIAOSD	Port A sound output serial data	Output	Sound output serial data pin
	FSIAILRCK	Port A sound input L/R clock	Input	Sound input L/R clock pin (slave)
	FSIAIBCK	Port A sound input bit clock	Input	Sound input bit clock pin (slave)
	FSIAISD	Port A sound input serial data	Input	Sound input serial data pin
	FSICKA	Port A source clock input	Input	Source clock input pin for port A (input to the CPG module)
	CLKAUDIOAO	Port A audio clock output	Output	Audio clock output pin for port A
Port B	FSIBOLRCK	Port B sound output L/R clock	Output	Sound output L/R clock pin (master)
	FSIBOBCK	Port B sound output bit clock	Output	Sound output bit clock pin (master)
	FSIBOSD	Port B sound output Serial data	Output	Sound output serial data pin
	FSIBILRCK	Port B sound input L/R clock	Input	Sound input L/R clock pin (slave)
	FSIBIBCK	Port B sound input bit clock	Input	Sound input bit clock pin (slave)
	FSIBISD	Port B sound input Serial data	Input	Sound input serial data pin
	FSICKB	Port B source clock input	Input	Source clock input pin for port B (input to the CPG module)
	CLKAUDIOBO	Port B audio clock output	Output	Audio clock output pin for port B

45.3 Register Descriptions

Table 45.4 shows the FSI register configuration.

Table 45.4 Register Configuration

Type	Register Name	Abbreviation	R/W (Common to CPU/DSP)	Access Size		Physical Address	DSP Address
				CPU	DSP		
Port A	Port A output serial format register	A_DO_FMT	R/W	32	24	H'FE3C0000	H'00
	Port A output FIFO control register	A_DOFF_CTL	R/W	32	24	H'FE3C0004	H'01
	Port A output FIFO status register	A_DOFF_ST	R/W	32	24	H'FE3C0008	H'02
	Port A input serial format register	A_DI_FMT	R/W	32	24	H'FE3C000C	H'03
	Port A input FIFO control register	A_DIFF_CTL	R/W	32	24	H'FE3C0010	H'04
	Port A input FIFO status register	A_DIFF_ST	R/W	32	24	H'FE3C0014	H'05
	Port A clock set 1 register	A_CKG1	R/W	32	24	H'FE3C0018	H'06
	Port A clock set 2 register	A_CKG2	R/W	32	24	H'FE3C001C	H'07
	Port A read data register	A_DIDT	R	32	24	H'FE3C0020	H'08
	Port A write data register	A_DODT	W	32	24	H'FE3C0024	H'09
	Port A MUTE state register	A_MUTE_ST	R	32	24	H'FE3C0028	H'0A
Port B	Port B output serial format register	B_DO_FMT	R/W	32	24	H'FE3C0040	H'10
	Port B output FIFO control register	B_DOFF_CTL	R/W	32	24	H'FE3C0044	H'11
	Port B output FIFO status register	B_DOFF_ST	R/W	32	24	H'FE3C0048	H'12
	Port B input serial format register	B_DI_FMT	R/W	32	24	H'FE3C004C	H'13
	Port B input FIFO control register	B_DIFF_CTL	R/W	32	24	H'FE3C0050	H'14
	Port B input FIFO status register	B_DIFF_ST	R/W	32	24	H'FE3C0054	H'15
	Port B clock set 1 register	B_CKG1	R/W	32	24	H'FE3C0058	H'16
	Port B clock set 2 register	B_CKG2	R/W	32	24	H'FE3C005C	H'17
	Port B read data register	B_DIDT	R	32	24	H'FE3C0060	H'18
	Port B write data register	B_DODT	W	32	24	H'FE3C0064	H'19
	Port B MUTE state register	B_MUTE_ST	R	32	24	H'FE3C0068	H'1A
Interrupt	Interrupt state register	INT_ST	R/W	32	24	H'FE3C0200	H'80
	Interrupt source mask register	IEMSK	R/W	32	24	H'FE3C0204	H'81
	Interrupt signal mask register	IMSK	R/W	32	24	H'FE3C0208	H'82
MUTE	MUTE set register	MUTE	R/W	32	24	H'FE3C020C	H'83

Type	Register Name	Abbreviation	R/W (Common to CPU/DSP)	Access Size		Physical Address	DSP Address
				CPU	DSP		
Reset	Clock reset register	CLK_RST	R/W	32	24	H'FE3C0210	H'84
	Software reset register	SOFT_RST	R/W	32	24	H'FE3C0214	H'85
On-chip RAM capacity	FIFO size register	FIFO_SZ	R	32	24	H'FE3C0218	H'86

Notes: 1. The port A and port B registers have the same functions and settings.
2. Initial values of bits 31 to 24 in each register are all 0 when accessing the CPU.

Table 45.5 Register States in Each Operating Mode

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep
A_DO_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DOFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DOFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DI_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DIFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DIFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_CKG1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_CKG2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DIDT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_DODT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
A_MUTE_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DO_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DOFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DOFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DI_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DIFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DIFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_CKG1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_CKG2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DIDT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_DODT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
B_MUTE_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
INT_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
IEMSK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
IMSK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
CLK_RST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
SOFT_RST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained
FIFO_SZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained

45.3.1 Port A/B Output Serial Format Register (A_DO_FMT/B_DO_FMT)

A_DO_FMT/B_DO_FMT is a 24-bit readable/writable register. It selects the format from MONO, Mono delay, PCM, I2S, TDM, and TDM_Delay that is used in output operations on serial ports A/B.

When TDM and TDM_Delay modes are selected, it also specifies the number of channels (1 to 8 channels).

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	FMT[2:0]			—	NCH[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 4	FMT[2:0]	011	R/W	Format Select 000: MONO (1 channel) 001: MONO Delay (1 channel) 010: PCM (2 channels) 011: I2S (2 channels) 100: TDM (1 to 8 channels) 101: TDM Delay (1 to 8 channels) 110: Setting prohibited 111: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	NCH[2:0]	000	R/W	These bits specify the number of channels in TDM and TDM_Delay modes. When selecting the modes other than TDM and TDM_Delay, specify B'000. 000: 1 channel 001: 2 channels 010: 3 channels 011: 4 channels 100: 5 channels 101: 6 channels 110: 7 channels 111: 8 channels

45.3.2 Port A/B Output FIFO Control Register (A_DOFF_CTL/B_DOFFCTL)

A_DOFF_CTL and B_DOFF_CTL are 24-bit readable/writable registers that select the conditions for generating interrupts from serial ports A and B. An FIFO clear bit also initializes the access pointer to the on-chip RAM and the history of overflow and underflow errors.

- Volumes of stored data that leave more than half of the data area free:

  1 channel: 127 or fewer samples (127 or fewer words)
  2 channels: 63 or fewer samples (126 or fewer words)
  3 to 4 channels: 31 or fewer samples (3 channels: 93 or fewer words, 4 channels: 124 or fewer words)
  5 to 8 channels: 15 or fewer samples (5 channels: 75 or fewer words, 6 channels: 90 or fewer words, 7 channels: 105 or fewer words, 8 channels: 120 or fewer words)

Bit:	23	22	21	20	19	18	17	16
	—	—	IHC[1:0]		—	—	—	—
Initial value:	0	0	0	1	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	IHC[1:0]	01	R/W	Interrupt Source Condition Select 00: Empty 01: More than half of the data area is free. 10: Data area for more than 1 sample is free. 11: Setting prohibited (full state)
19 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CLR	0	R/(W)*	FIFO Clear This bit initializes the access pointer and the error history. This bit is always read as 0. 0: When only the interrupt source condition is changed, the write value should be 0. 1: Clears FIFOs.

Note: * If 1 is written to the CLR bit, it is asserted for 1 pulse and cleared to 0.

45.3.3 Port A/B Output FIFO Status Register (A_DOFF_ST/B_DOFF_ST)

A_DOFF_ST/B_DOFF_ST is a 24-bit readable/writable register. It can be used to read the output FIFO states for serial ports A/B, the number of stored sample data, and the history of overflow and underflow errors.

The OF and UF bits clear the history when they are written to 0.

For clearing procedures for OF and UF bits, see section 45.6, FIFO Overflow/Underflow Specifications.

Bit:	23	22	21	20	19	18	17	16
	—	—	ST[1:0]		—	—	—	SZ[8]
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SZ[7:0]								—	—	—	OF	—	—	—	UF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	R	R	R	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	ST[1:0]	00	R	FIFO State 00: Empty state. 01: Less than half of data areas are occupied. 10: More than half of data areas are occupied. 11: Full state.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
16 to 8	SZ[8:0]	H'000	R	Number of Sample Data These bits indicate the number of sample data stored in FIFOs. The maximum value varies according to with the number of channels selected for the format. The maximum numbers of sample data for all numbers of channels are shown (in the case of 256 words). If more data than the maximum number in the parenthesis are read, an underflow occurs. FIFOs are used in 4-channel units in 3-channel mode and in 8-channel units in 5- to 7-channel mode, meaning that more FIFOs than the required size of DP-RAM are used. The SPU2/DSP should take into account the maximum number of writes for these extra FIFOs. 1 channel: 256 (256 × 1 = 256) 2 channels: 128 (128 × 2 = 256) 3 channels: 64 (64 × 3 = 192) 4 channels: 64 (64 × 4 = 256) 5 channels: 32 (32 × 5 = 160) 6 channels: 32 (32 × 6 = 192) 7 channels: 32 (32 × 7 = 224) 8 channels: 32 (32 × 8 = 256)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OF	0	R/(W)*	FIFO Error History 0: No error has occurred. 1: Overflow has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	UF	0	R/(W)*	FIFO Error History 0: No error has occurred. 1: Underflow has occurred.

Note: * The FIFO error history is cleared by writing 0 to this bit and retained by writing 1 to it.

45.3.4 Port A/B Input Serial Format Register (A_DI_FMT/B_DI_FMT)

A_DI_FMT/B_DI_FMT is a 24-bit readable/writable register that selects the format used in I/O operations of serial ports A/B from MONO, Mono delay, PCM, I2S, TDM, and TDM_Delay.

When TDM and TDM_Delay are selected, the number of channels should be specified.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	FMT[2:0]			—	NCH[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 4	FMT[2:0]	011	R/W	Format Select 000: MONO (1 channels 001: MONO Delay (1 channel) 010: PCM (2 channels) 011: I2S (2 channels) 100: TDM (1 to 8 channels) 101: TDM Delay (1 to 8 channels) 110: Setting prohibited 111: Setting prohibited
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	NCH[2:0]	000	R/W	These bits specify the number of channels for the TDM and TDM_Delay formats. When selecting the modes other than TDM and TDM_Delay, specify B'000. 000: 1 channel 001: 2 channels 010: 3 channels 011: 4 channels 100: 5 channels 101: 6 channels 110: 7 channels 111: 8 channels

45.3.5 Port A/B Input FIFO Control Register (A_DIFF_CTL/B_DIFF_CTL)

A_DIFF_CTL and B_DIFF_CTL are 24-bit readable/writable registers that select the conditions for generating interrupts from the input FIFOs for serial ports A and B. An FIFO clear bit also initializes the access pointer to the on-chip RAM. This bit also deletes the history of overflow and underflow errors.

- Volumes of data that have been stored when the data area is at least half full:

1 channel:	128 or more samples (128 or more words)
2 channels:	64 or more samples (128 or more words)
3 to 4 channels:	32 or more samples (3 channels: 96 or more words, 4 channels: 128 or more words)
5 to 8 channels:	16 or more samples (5 channels: 80 or more words, 6 channels: 96 or more words, 7 channels: 112 or more words, 8 channels: 128 or more words)

Bit:	23	22	21	20	19	18	17	16
	—	—	IHC[1:0]		—	—	—	—
Initial value:	0	0	0	1	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	IHC[1:0]	01	R/W	Condition for generating interrupt 00: Full 01: Amount of data is over half the FIFO capacity. 10: Data for more than 1 sample are in the FIFO. 11: Setting prohibited (empty state)
19 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	CLR	0	R/(W)*	FIFO Clear This bit initializes the access pointer and the error history. This bit is always read as 0. 0: When only the interrupt source condition is changed, the write value should be 0. 1: Clears FIFOs.

Note: * If 1 is written to the CLR bit, it is asserted for 1 pulse and cleared to B'0.

45.3.6 Port A/B Input FIFO Status Register (A_DIFF_ST/B_DIFF_ST)

A_DIFF_ST/B_DIFF_ST is a 24-bit readable/writable register. It can be used to read the input FIFO states for serial ports A/B, the number of stored sample data, and the history of overflow and underflow errors.

The OF and UF bits clear the history when they are written to 0.

For clearing procedures for OF and UF bits, see section 45.6, FIFO Overflow/Underflow Specifications.

Bit:	23	22	21	20	19	18	17	16
	—	—	ST[1:0]		—	—	—	SZ[8]
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SZ[7:0]								—	—	—	OF	—	—	—	UF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	ST[1:0]	00	R	FIFO State 00: Empty state. 01: Less than half of data areas are occupied. 10: More than half of data areas are occupied. 11: Full state.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
16 to 8	SZ[8:0]	H'000	R	Number of Sample Data These bits indicate the number of sample data stored in FIFOs. The maximum value varies according to with the number of channels selected for the format. The maximum numbers of sample data for all numbers of channels are shown (in the case of 256 words). If more data than the maximum number in the parenthesis are read, an underflow occurs. FIFOs are used in 4-channel units in 3-channel mode and in 8-channel units in 5- to 7-channel mode, meaning that more FIFOs than the required size of DP-RAM are used. The SPU2/DSP should take into account the maximum number of writes for these extra FIFOs. 1 channel: 256 (256 × 1 = 256) 2 channels: 128 (128 × 2 = 256) 3 channels: 64 (64 × 3 = 192) 4 channels: 64 (64 × 4 = 256) 5 channels: 32 (32 × 5 = 160) 6 channels: 32 (32 × 6 = 192) 7 channels: 32 (32 × 7 = 224) 8 channels: 32 (32 × 8 = 256)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OF	0	R/(W)*	FIFO Error History 0: No error has occurred. 1: Overflow has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	UF	0	R/(W)*	FIFO Error History 0: No error has occurred. 1: Underflow has occurred.

Note: * The FIFO error history is cleared by writing 0 to this bit and retained by writing 1 to it.

45.3.7 Port A/B Clock Set 1 Register (A_CKG1/B_CKG1)

A_CKG1/B_CKG1 is a 24-bit readable/writable register. It generates the master clock used by serial ports A/B and selects master/slave operation of the LR and bit clocks used internally.

The master clock is generated from the CPG audio clock (FSICKA/FSICKB). The block diagram is shown in figure 45.2.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ACKMD[1:0]		—	BPFMD[2:0]			—	—	—	DIMD	—	—	—	DOMD
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	ACKMD[1:0]	00	R/W	These bits specify the audio clock to generate the LR clock during master operation. Select the audio clock (fs) to be input for the sampling frequency (fs). 00: 512 fs 01: 256 fs 10: 128 fs 11: 64 fs Note: When selecting TDM and TDM_Delay, select 512 fs/256 fs. For the output waveforms of the LR clock, refer to section 45.8, Serial Data Formats.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 8	BPFMD[2:0]	001	R/W	These bits set the division ratio of the bit clock during master operation. Some settings are unavailable depending or the format and the input audio clock. Refer to table 45.6. 000: 32 bit/fs 001: 64 bit/fs 010: 128 bit/fs 011: 256 bit/fs 100: 512 bit/fs 101: Setting prohibited 110: Setting prohibited 111: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	DIMD	0	R/W	DIIS Module Master/Slave Select 0: Slave operation 1: Master operation
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DOMD	0	R/W	DOIS Module Master/Slave Select 0: Slave operation 1: Master operation

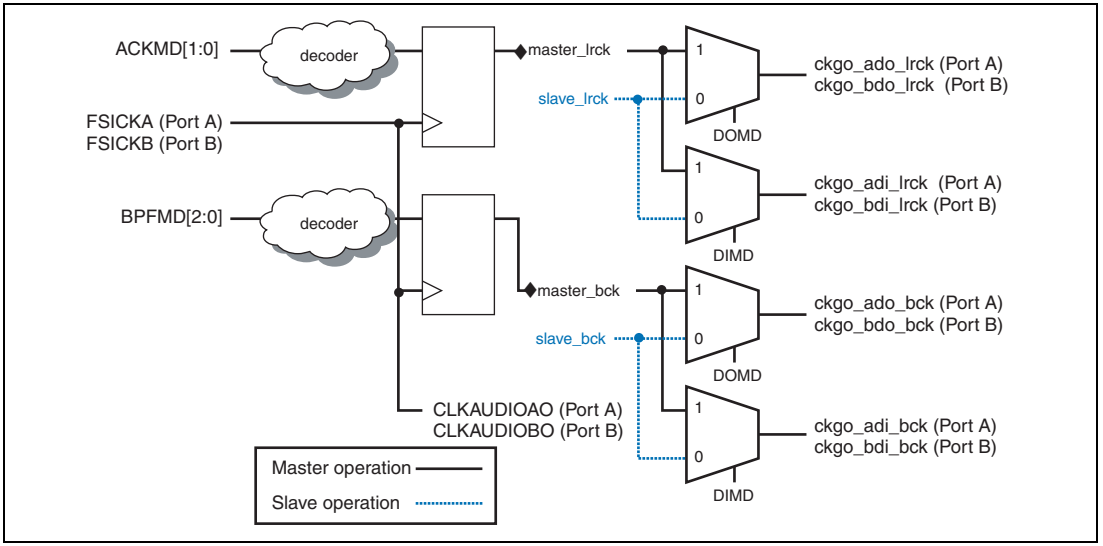


Figure 45.2 Master Clock Generation, I/O Operations, Master/Slave Selection

Decoding is synchronized with audio clock (FSICKA/FSICKB) from the CPG in master operation and the LR clocks (FSIAOLRCK/ FSIBOLRCK) and bit clocks (FSIAOBCK/ FSIBOBCK) are generated. CLKAUDIOAO/ CLKAUDIOBO output FSICKA/FSICKB directly. The possible combinations of CKMD and BPFMD are shown in table 45.6. No bit clock faster than the audio clock can be generated.

For slave operation, the external LR and bit clocks are used. Since the internal modules convert data based on the falling edge of the LR clock that is inverted on the rising edge of the bit clock, configure the inversion setting according to the external interface specification in the A_CKG2 register for slave operation. (In master mode, the inversion setting can be configured according to the specifications of the external DAC or ADC.)

Table 45.6 List of Supported Formats

	MONO, MONO Delay	I2S, PCM	TDM, TDM Delay
Operating Clock	**BPFMD**	**BPFMD**	**BPFMD**
Master ACKMD: 64 fs	64 bit/fs 32 bit/fs	64 bit/fs	(*)
Master ACKMD: 128 fs	128 bit/fs 64 bit/fs 32 bit/fs	128 bit/fs 64 bit/fs	(*)
Master ACKMD: 256 fs	256 bit/fs 128 bit/fs 64 bit/fs 32 bit/fs	256 bit/fs 128 bit/fs 64 bit/fs	256 bit/fs
Master ACKMD: 512 fs	512 bit/fs 256 bit/fs 128 bit/fs 64 bit/fs 32 bit/fs	512 bit/fs 256 bit/fs 128 bit/fs 64 bit/fs	512 bit/fs 256 bit/fs
Slave	512 bit/fs 256 bit/fs 128 bit/fs 64 bit/fs 32 bit/fs	512 bit/fs 256 bit/fs 128 bit/fs 64 bit/fs	512 bit/fs 256 bit/fs

Note: * In TDM/TDM Delay modes, select 256 fs and 512 fs.

45.3.8 Port A/B Clock Set 2 register (A_CKG2/B_CKG2)

A_CKG2/B_CKG2 is a 24-bit readable/writable register that selects inversion of the master and slave clocks for serial ports A/B. They can be inverted according to the external interface specifications.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	LRM	—	—	—	BRM	—	—	—	LRS	—	—	—	BRS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	LRM	0	R/W	Master LR Clock Inversion 0: Clock not inverted 1: Clock inverted
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	BRM	0	R/W	Master Bit Clock Inversion 0: Clock not inverted 1: Clock inverted
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	LRS	0	R/W	Slave LR Clock Inversion 0: Clock not inverted 1: Clock inverted
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	BRS	0	R/W	Slave Bit Clock Inversion 0: Clock not inverted 1: Clock inverted

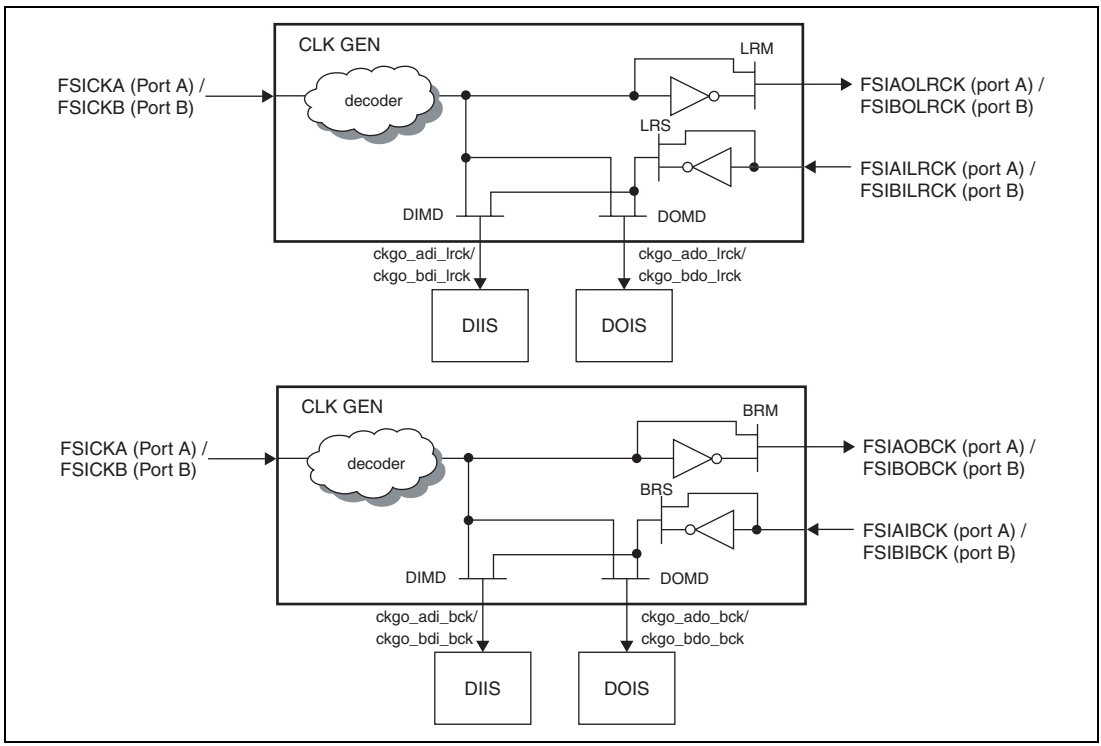


Figure 45.3 Master/Slave Clock Inversion Selector

Only one clock system (master output, bit and LR clocks for slave input pins) can be generated for one serial port. Thus, master-master and slave-slave operations with different formats and different bit and LR clocks cannot be performed on the same serial port.

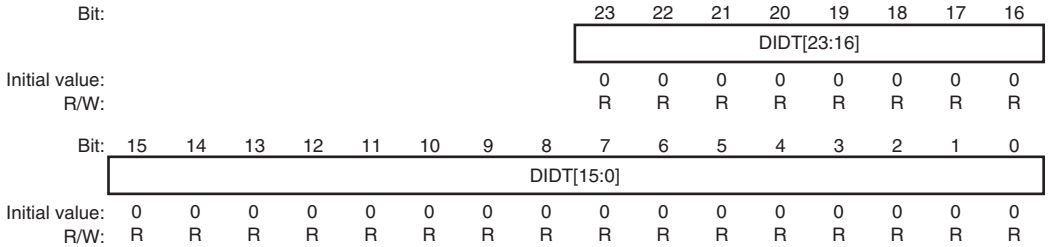
To perform simultaneous input and output operations with different formats, bit clocks, and LR clocks, configure those operations for port A and port B separately.

If input and output in master or slave mode are to be in different formats, generate the LR clock for the output format. For transfer where different formats are not in use, select “slave” (the default value) in selection of master or slave.

45.3.9 Port A/B Read Data Register (A_DIDT/B_DIDT)

A_DIDT/B_DIDT is a 24-bit read-only register that can be used to read data stored in serial ports A/B input FIFOs. If FIFOs are empty, a read causes an underflow error.

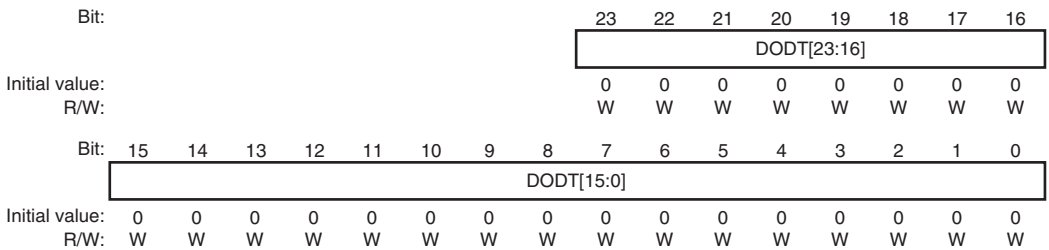
Reading of the read registers (A_DIDT, B_DIDT) is prohibited during a software rest (SOFT_RST.SR) or resetting of the corresponding port (SOFT_RST.SRA, SOFT_RST.SRB).



Bit	Bit Name	Initial Value	R/W	Description
23 to 0	DIDT[23:0]	H'00 0000	R	These bits contain data from input FIFOs. This register is read-only.

45.3.10 Port A/B Write Data Register (A_DODT/B_DODT)

A_DODT/B_DODT is a 24-bit write-only register that can be used to write data to serial ports A/B output FIFOs. If FIFOs are full, a write causes an overflow error.



Bit	Bit Name	Initial Value	R/W	Description
23 to 0	DODT[23:0]	H'00 0000	W	These bits contain data to output FIFOs. This register is write-only.

45.3.11 Port A/B MUTE State (A_MUTE_ST/B_MUTE_ST)

A_MUTE_ST/B_MUTE_ST is a 24-bit read-only register that indicates the MUTE state for each channel in I/O operations on ports A/B.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	O_CH8	O_CH7	O_CH6	O_CH5	O_CH4	O_CH3	O_CH2	O_CH1	I_CH8	I_CH7	I_CH6	I_CH5	I_CH4	I_CH3	I_CH2	I_CH1
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	O_CH8	0	R	Port A Output MUTE State 0: Channel 8 MUTE disabled 1: Channel 8 MUTE on
14	O_CH7	0	R	Port A Output MUTE State 0: Channel 7 MUTE off 1: Channel 7 MUTE on
13	O_CH6	0	R	Port A Output MUTE State 0: Channel 6 MUTE off 1: Channel 6 MUTE on
12	O_CH5	0	R	Port A Output MUTE State 0: Channel 5 MUTE off 1: Channel 5 MUTE on
11	O_CH4	0	R	Port A Output MUTE State 0: Channel 4 MUTE off 1: Channel 4 MUTE on
10	O_CH3	0	R	Port A Output MUTE State 0: Channel 3 MUTE off 1: Channel 3 MUTE on

Bit	Bit Name	Initial Value	R/W	Description
9	O_CH2	0	R	Port A Output MUTE State 0: Channel 2 MUTE off 1: Channel 2 MUTE on
8	O_CH1	0	R	Port A Output MUTE State 0: Channel 1 MUTE off 1: Channel 1 MUTE on
7	I_CH8	0	R	Port A Input MUTE State 0: Channel 8 MUTE off 1: Channel 8 MUTE on
6	I_CH7	0	R	Port A Input MUTE State 0: Channel 7 MUTE off 1: Channel 7 MUTE on
5	I_CH6	0	R	Port A Input MUTE State 0: Channel 6 MUTE off 1: Channel 6 MUTE on
4	I_CH5	0	R	Port A Input MUTE State 0: Channel 5 MUTE off 1: Channel 5 MUTE on
3	I_CH4	0	R	Port A Input MUTE State 0: Channel 4 MUTE off 1: Channel 4 MUTE on
2	I_CH3	0	R	Port A Input MUTE State 0: Channel 3 MUTE off 1: Channel 3 MUTE on
1	I_CH2	0	R	Port A Input MUTE State 0: Channel 2 MUTE off 1: Channel 2 MUTE on
0	I_CH1	0	R	Port A Input MUTE State 0: Channel 1 MUTE off 1: Channel 1 MUTE on

45.3.12 Interrupt State Register (INT_ST)

INT_ST is a 24-bit readable/writable register that retains 4 interrupt sources separately. Only writing 0 to clear is accepted. Writing 1 does not change the bit value. To clear a specific interrupt source, write 0 to the corresponding bit and 1 to the other bits.

The interrupt state register is set if the specified interrupt condition matches the FIFO state while the corresponding interrupt mask is cleared. If an interrupt source is cleared at the same time as the interrupt source occurs, the interrupt condition takes precedence.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IIB	—	—	—	IOB	—	—	—	IIA	—	—	—	IOA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/(W)*	R	R	R	R/(W)*	R	R	R	R/(W)*	R	R	R	R/(W)*

Bit	Bit Name	Initial Value	R/W	Description
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	IIB	0	R/(W)*	Port B Input FIFO Interrupt 0: Interrupt is cleared. 1: Interrupt source is retained.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	IOB	0	R/(W)*	Port B Output FIFO Interrupt 0: Interrupt is cleared. 1: Interrupt source is retained.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	IIA	0	R/(W)*	Port A Input FIFO Interrupt 0: Interrupt is cleared. 1: Interrupt source is retained.

Bit	Bit Name	Initial Value	R/W	Description
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	IOA	0	R/(W)*	Port A Output FIFO Interrupt 0: Interrupt is cleared. 1: Interrupt source is retained.

Note: * Writing 0 to this bit clears the interrupt source. Writing 1 does not change the bit value.

45.3.13 Interrupt Source Mask Set Register (IEMSK)

IEMASK is a 24-bit readable/writable register that masks the interrupt source for each FIFO. If this register masks an interrupt source, the interrupt source is not enabled even if the IMASK register below unmasks the interrupt signal. Thus, the interrupt does not occur.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MIRB	—	—	—	MORB	—	—	—	MIRA	—	—	—	MORA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	MIRB	0	R/W	Port B Input FIFO Interrupt Request Mask 0: Port A input FIFO interrupt request disabled 1: Port A input FIFO interrupt request enabled
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	MORB	0	R/W	Port B Output FIFO Interrupt Request Mask 0: Port B output FIFO interrupt request disabled 1: Port B output FIFO interrupt request enabled

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	MIRA	0	R/W	Port A Input FIFO Interrupt Request Mask 0: Port A input FIFO interrupt request disabled 1: Port A input FIFO interrupt request enabled
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MORA	0	R/W	Port A Output FIFO Interrupt Request Mask 0: Port A output FIFO interrupt request disabled 1: Port A output FIFO interrupt request enabled

45.3.14 Interrupt Signal Mask Set Register (IMSK)

IMASK is a 24-bit readable/writable register that masks the interrupt signal for each FIFO to the DSP. If an interrupt source for a FIFO whose interrupt signal is unmasked, the interrupt request signal occurs.

Figure 45.4 shows the relationship between the interrupt state (INT_ST.IOA), interrupt mask (IEMSK.MORA), and signal mask (IMSK.MRA) for an interrupt source as an example. The same relationship applies to other interrupt sources.

For the connections for all interrupt sources, refer to section 44.3.5, Interrupt Interface Specifications.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MIB	—	—	—	MOB	—	—	—	MIA	—	—	—	MOA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	MIB	0	R/W	Port B Input FIFO Interrupt Signal Mask 0: Masks the interrupt signal. 1: Enables the interrupt signal.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	MOB	0	R/W	Port B Output FIFO Interrupt Signal Mask 0: Masks the interrupt signal. 1: Enables the interrupt signal.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	MIA	0	R/W	Port A Input FIFO Interrupt Signal Mask 0: Masks the interrupt signal. 1: Enables the interrupt signal.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MOA	0	R/W	Port A Output FIFO Interrupt Signal Mask 0: Masks the interrupt signal. 1: Enables the interrupt signal.

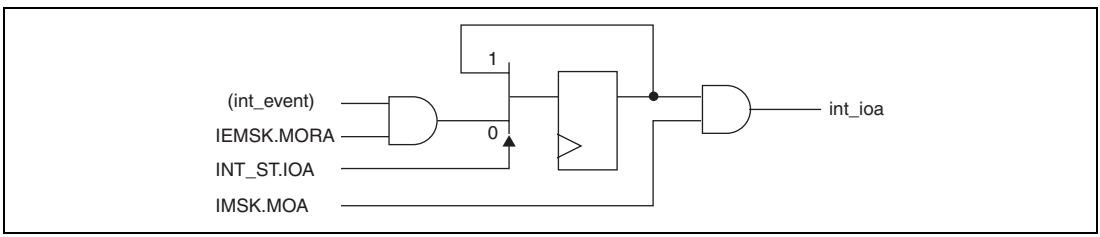


Figure 45.4 Port A Output FIFO Interrupt Request Signal

45.3.15 MUTE Set Register (MUTE)

MUTE is a 24-bit readable/writable register that enables the MUTE function of the inputs/outputs for ports A/B. Set MUTE on/off after selecting MUTE mode. For the mute timing specifications, refer to section 45.5, MUTE Specifications.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TSIB	TYIB	—	—	TSOB	TYOB	—	—	TSIA	TYIA	—	—	TSOA	TYOA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	TSIB	0	R/W	Port B Input MUTE 0: MUTE off 1: MUTE on
12	TYIB	0	R/W	Port B Input MUTE Mode Set 0: Normal MUTE 1: Zero cross MUTE
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	TSOB	0	R/W	Port B Output MUTE 0: MUTE off 1: MUTE on
8	TYOB	0	R/W	Port B Output MUTE Mode Set 0: Normal MUTE 1: Zero cross MUTE
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5	TSIA	0	R/W	Port A Input MUTE 0: MUTE off 1: MUTE on
4	TYIA	0	R/W	Port A Input MUTE Mode Set 0: Normal MUTE 1: Zero cross MUTE
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TSOA	0	R/W	Port A Output MUTE 0: MUTE off 1: MUTE on
0	TYOA	0	R/W	Port A Output MUTE Mode Set 0: Normal MUTE 1: Zero cross MUTE

45.3.16 Clock Reset Set Register (CLK_RST)

CLK_RST is a 24-bit readable/writable register. During reset, it masks the LR and bit clocks that are output from the clock-generating block. (unless the division ratio of the bit clock relative to fs is 1:1, in which case the bit clock is not masked).) Clear the reset after setting formats and clocks by follow the operation flow.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CRB	—	—	—	CRA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CRB*	0	R/W	Port B Clock Reset 0: Issues a reset. 1: Clear the reset.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CRA*	0	R/W	Port A Clock Reset 0: Issues a reset. 1: Clears the reset.

Note: * This bit should be asserted for longer than the audio clock period so that the counter synchronized with it is reset.

45.3.17 Software Reset Set Register (SOFT_RST)

SOFT_RST is a 24-bit readable/writable register that initializes the internal registers.

Reading of the read registers (A_DIDT, B_DIDT) is prohibited during a software rest (SOFT_RST.SR) or resetting of the corresponding port (SOFT_RST.SRA, SOFT_RST.SRB).

Writing during a software reset (SOFT_RST.SR) has no effect, and reading of registers other than read registers returns 0.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SRB	—	—	—	SRA	—	—	—	IR	—	—	—	SR
Initial value:	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	1
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	SRB*	1	R/W	Port B Software Reset This bit performs a port B FIFO clear. 0: Issues a reset. 1: Clears the reset.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SRA*	1	R/W	Port A Software Reset This bit performs a port A FIFO clear. 0: Issues a reset. 1: Clears the reset.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	IR	1	R/W	Interrupt Reset This bit negates the interrupt signal without receiving an acknowledge. 0: Issues a reset. 1: Clears the reset.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SR	1	R/W	Software Reset This bit initializes all registers. 0: Issues a reset. 1: Clears the reset.

Note: * The SRA and SRB bits should be asserted for longer than the bit clocks for DOIS and DIIS of each port.

45.3.18 FIFO Size Register (FIFO_SZ)

FIFO_SZ is a 24-bit read-only register that indicates the on-chip RAM capacity.

Bit:	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BI[2:0]			—	BO[2:0]			—	AI[2:0]			—	AO[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
23 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	BI[2:0]	000	R	Port B Input FIFO Capacity These bits indicate the on-chip RAM capacity. 000: 256 words 001: Reserved (512 words) 010: Reserved (1024 words) 011: Reserved (2048 words) 111: Out-of-spec capacity is set (error) Note: No value other than listed above is output.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	BO[2:0]	000	R	Port B Output FIFO Capacity These bits indicate the on-chip RAM capacity. 000: 256 words 001: Reserved (512 words) 010: Reserved (1024 words) 011: Reserved (2048 words) 111: Out-of-spec capacity is set (error) Note: No value other than listed above is output.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	AI[2:0]	000	R	Port A Input FIFO Capacity These bits indicate the on-chip RAM capacity. 000: 256 words 001: Reserved (512 words) 010: Reserved (1024 words) 011: Reserved (2048 words) 111: Out-of-spec capacity is set (error) Note: No value other than listed above is output.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	AO[2:0]	000	R	Port A Output FIFO Capacity These bits indicate the on-chip RAM capacity. 000: 256 words 001: Reserved (512 words) 010: Reserved (1024 words) 011: Reserved (2048 words) 111: Out-of-spec capacity is set (error) Note: No value other than listed above is output.

45.4 Interface Specifications

45.4.1 FSI and SPU2 Connection Interface Specifications

The FSI uses an 8-bit address space (H'00 to H'FF). The FSI registers can be read and written via the SPU2 DSP/IO bus.

(1) BUFC Specifications

Figure 45.5 shows the data flow from read requests to the A_DIDT and B_DIDT registers to the data output.

To read data stored in input FIFOs continuously, the fsio_io_hold signal is maintained when the A_DIDT or B_DIDT register is read. When a read request is sent to the FSI, the hold signal is not driven low and the read request is kept asserted until data are stored in buffers in the control block. Thus, data requests are generated continuously and the next data to be read are stored in the control block buffers.

Data can be read continuously with one wait like other registers by prefetching data in the control block buffers.

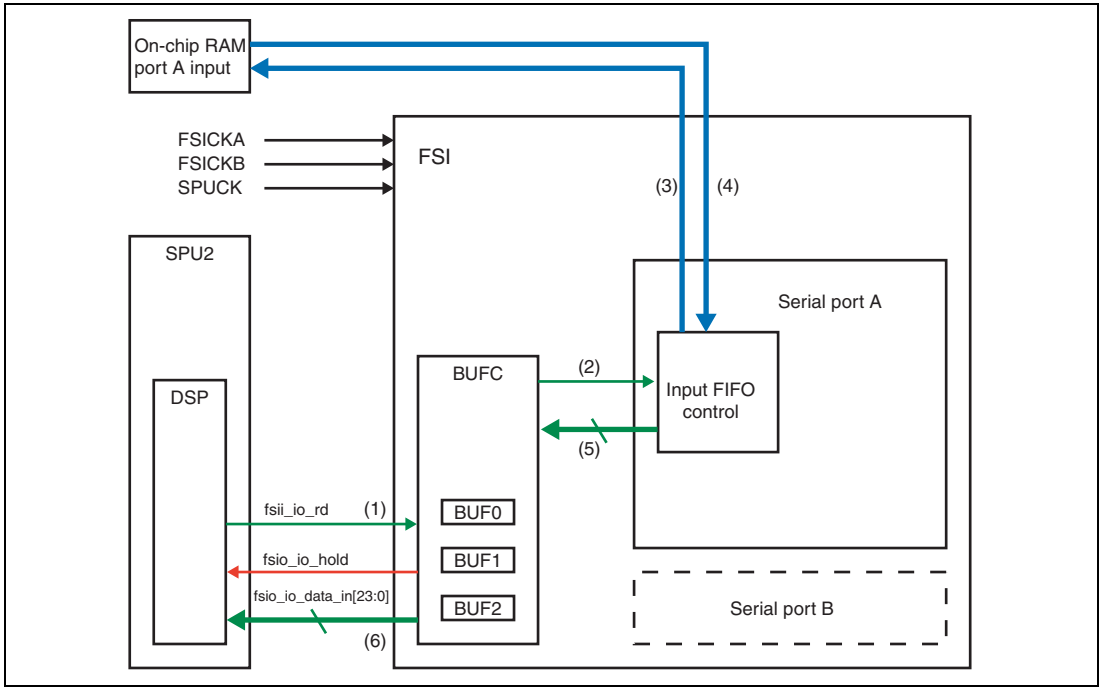


Figure 45.5 Block Diagram of Input FIFO Read Request

45.4.2 Clock Interface Specifications

A block diagram of clock signals is shown in Figure 45.6.

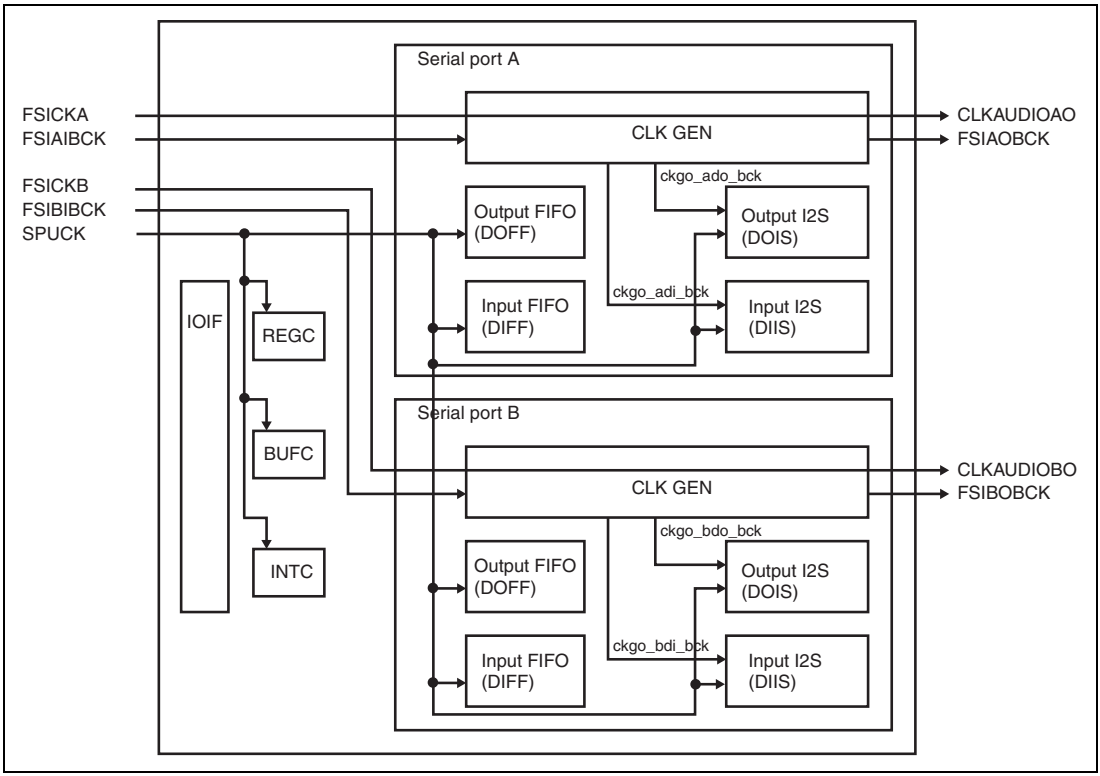


Figure 45.6 Block Diagram of FSI Clock Inputs

45.4.3 Reset Interface Specifications

Table 45.7 lists the FSI resets.

Table 45.7 FSI Rest Type, Initialization Register and Block

			Initializing Module							
Reset Name	**Register**	**Initialization Register**	**INTC**	**BUFC**	**REGC**	**CLK GEN**	**Output FIFO**	**Input FIFO**	**Output I2S**	**Input I2S**
Hardware reset	—	All registers	√							
Software reset	SOFT_RST.SR	All registers except software reset	√							√ (except FFs synchronized with the Bitclk)
Port A reset	SOFT_RST.SRA	Performs a port A FIFO clear.	—	—	—	—		√ (Port A only)		
Port B reset	SOFT_RST.SRB	Performs a port B FIFO clear.	—	—	—	—		√ (Port B only)		
Clock reset A	CLK_RST.CRA	—	—	—	—	√ (Port A)	—	—	—	—
Clock reset B	CLK_RST.CRB	—	—	—	—	√ (Port B)	—	—	—	—
Interrupt reset	SOFT_RST.IR	—	√ (fsio_int negated)	—	—	—	—	—	—	—
Port A output FIFO clear	A_DOFF_CTL.CLR	—	—	—	—	—	√ (Port A)	—	—	—
Port A input FIFO clear	A_DIFF_CTL.CLR	—	—	√ (Port A)	—	—	—	√ (Port A)	—	—
Port B output FIFO clear	B_DOFF_CTL.CLR	—	—	—	—	—	√ (Port B)	—	—	—
Port B input FIFO clear	B_DIFF_CTL.CLR	—	—	√ (Port B)	—	—	—	√ (Port B)	—	—

[Legend] √: Reset, —: Register retained

(1) Hardware Reset

Initializes all registers and modules.

(2) Software Reset

Initializes all registers except the software reset register (SOFT_RST.SR) and all modules. In master operation, the bit clock that is used for asynchronous passing may be fixed due to a clock reset, perform a software reset after resetting ports A/B (see (8) Procedures for Initializing the FSI).

(3) Port A/B Reset

These are the reset signals to initialize the FFs that are synchronized with the bit clock.

Since the bit clock is used for asynchronous passing, this reset requires to wait for longer than 1 bit clock cycle after a register write to perform it.

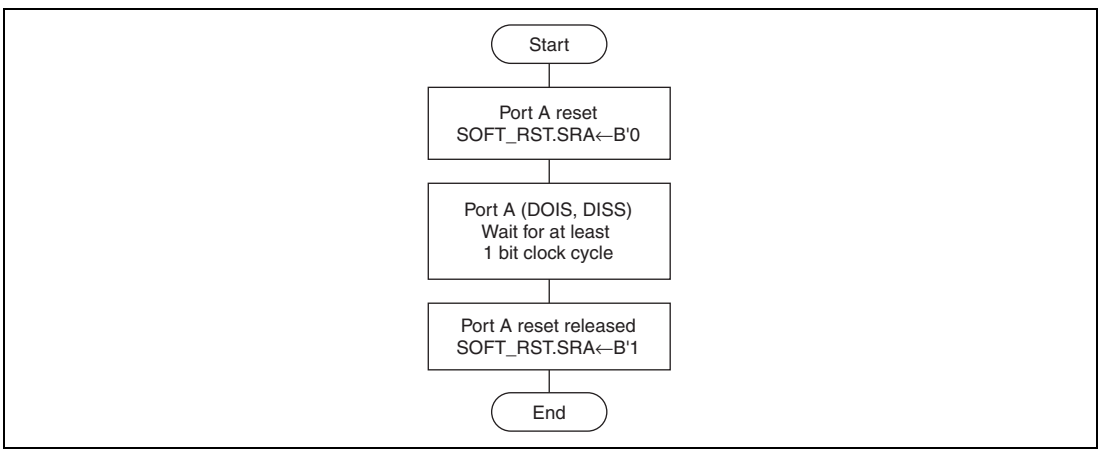


Figure 45.7 Flow of Port A Reset

(4) Clock Reset A/B

These are the reset signals to initialize the FFs that are synchronized with the audio clock.

Since the audio clock is used for asynchronous passing, this reset requires to wait for longer than 1 audio clock cycle after a register write to perform it.

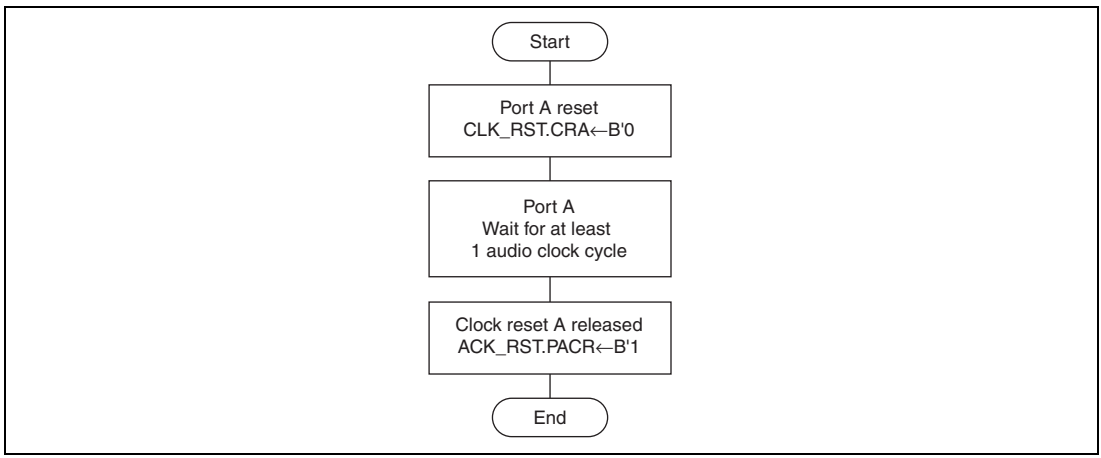


Figure 45.8 Flow of Clock Reset A

(5) Interrupt Reset

Negates the interrupt signal (fsio_int) when the corresponding register is written.

(6) Port A/B Output FIFO Clear

Clears the address pointer of the FIFO control block and the error history.

(7) Port A/B Input FIFO Clear

Clears the address pointer of the FIFO control block, the error history, and buffers for continuous 1-wait reads.

(8) Procedures for Initializing the FSI

The procedures for initializing the FSI are shown in Figure 45.9. There are two types of procedures: one by using a hardware reset and another by using a software reset. To achieve the same effect as a hardware reset, perform a software reset after port A/B reset (initializing the FFs that are synchronized with the bit clock).

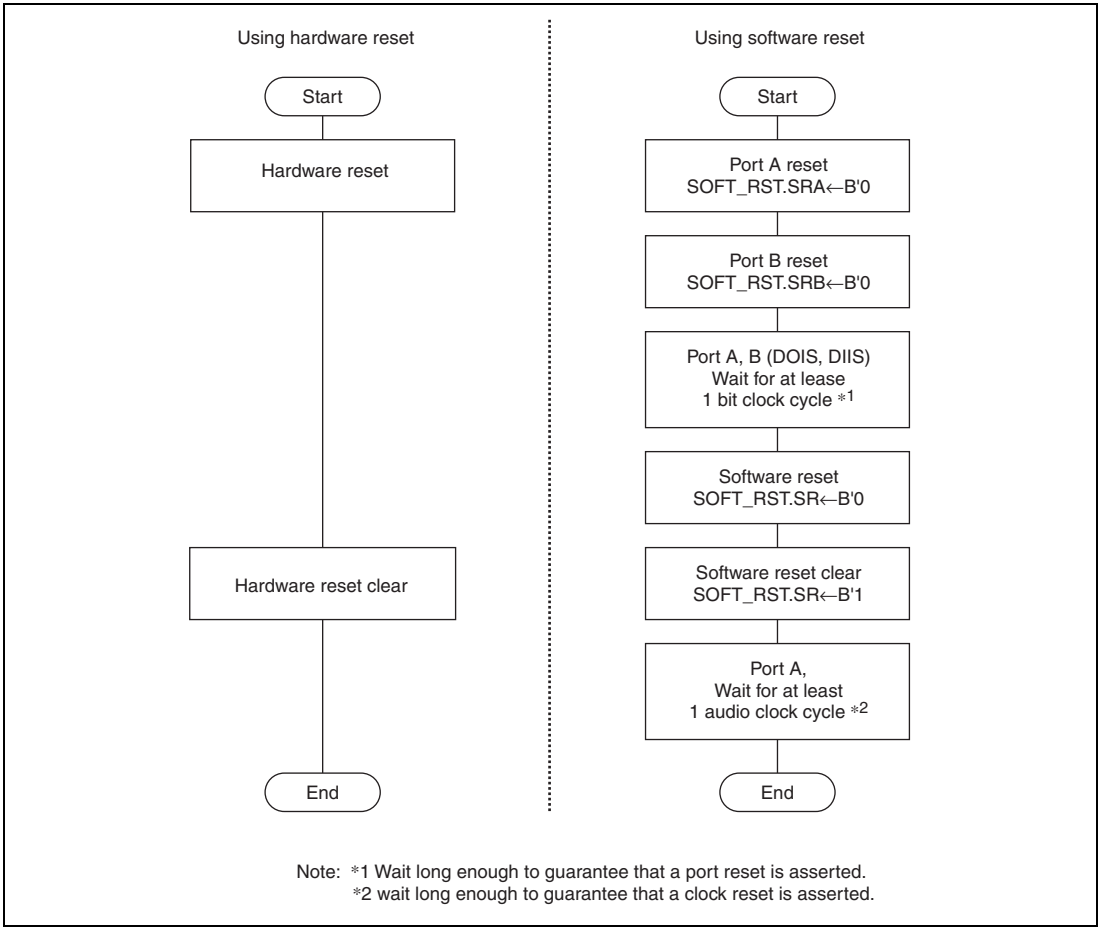


Figure 45.9 Operation Flow of FSI Initial States

45.5 MUTE Specifications

The FSI has normal MUTE and zero cross MUTE functions for input/output operations. The MUTE circuits are implemented for output FIFOs and input FIFOs.

(1) Normal MUTE

The normal MUTE mutes on a per sample data basis. It is turned on (outputs 0) for channel 1 first 2 samples after it is enabled. Similarly, it is turned off for channel 1 first 2 samples after it is disabled.

Figure 45.10 shows the 2-channel audio output and the MUTE signal.

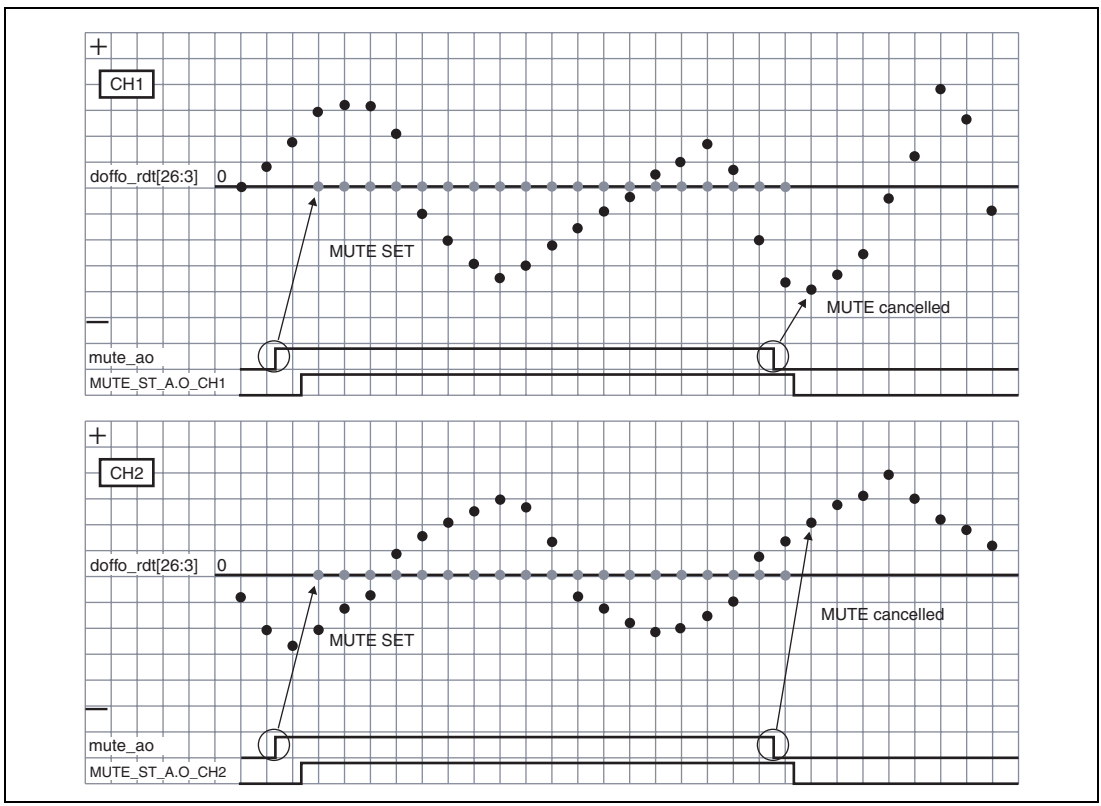


Figure 45.10 Normal MUTE Specification (Output Operation)

When the MUTE signal is asserted in input operation, the conversion module (input I2S) takes data written to input FIFOs and writes 0 to the on-chip RAM.

(2) Zero Cross MUTE

If the zero cross MUTE function is enabled, it compares current data to the previous one on each channel independently.

The conditions for turning on the mute are (1) the previous data is 0, (2) current data is 0, and (3) the sign of current data does not match the one of the previous data.

If any of these conditions ((1) to (3)) is satisfied, the mute is turned on (0 is output) until the zero cross MUTE function is disabled.

The mute (outputting 0) is turned off, just like it is turned on, if any of the conditions ((1) to (3)) is satisfied when current data is compared to the previous one on each channel independently.

Figure 45.11 shows the 2-channel audio output and zero cross MUTE signal.

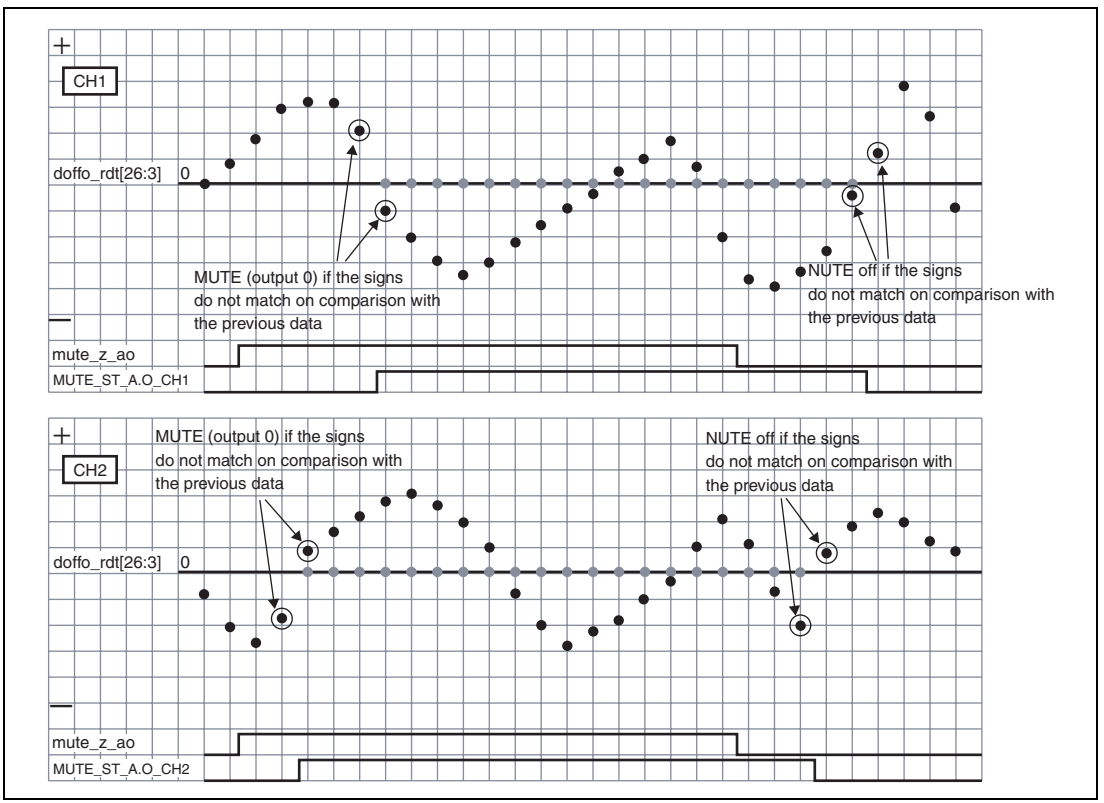


Figure 45.11 Zero Cross MUTE Specification (Output Operation)

Even if the zero cross MUTE function is enabled, the mute (outputting 0) is not turned on unless any of the conditions (1) to (3) is met.

Even if any of the conditions (1) to (3) is met, the mute (outputting 0) is not turned off unless the zero cross MUTE function is disabled.

After the zero cross MUTE function is enabled, it can be switched to the normal MUTE function to turn on/off the mute forcibly.

When the MUTE signal is asserted in input operation, the conversion module (input I2S) compares data written to input FIFOs to determine if the conditions (1) to (3) are met and writes 0 to the on-chip RAM.

45.6 FIFO Overflow/Underflow Specifications

When an underflow or overflow error occurs, the FIFO contains the error related information in a register. This section describes underflows and overflows in the output FIFOs and input FIFOs.

An output FIFO underflow occurs when the output I2S requests data (req) while the FIFO is empty. When it occurs, the output FIFO outputs H'00 0000 and the channel number [2:0]. An overflow occurs when the DSP tries to write more data than the FIFO capacity. Such write operations are ignored. No overflow occurs under management by the DSP.

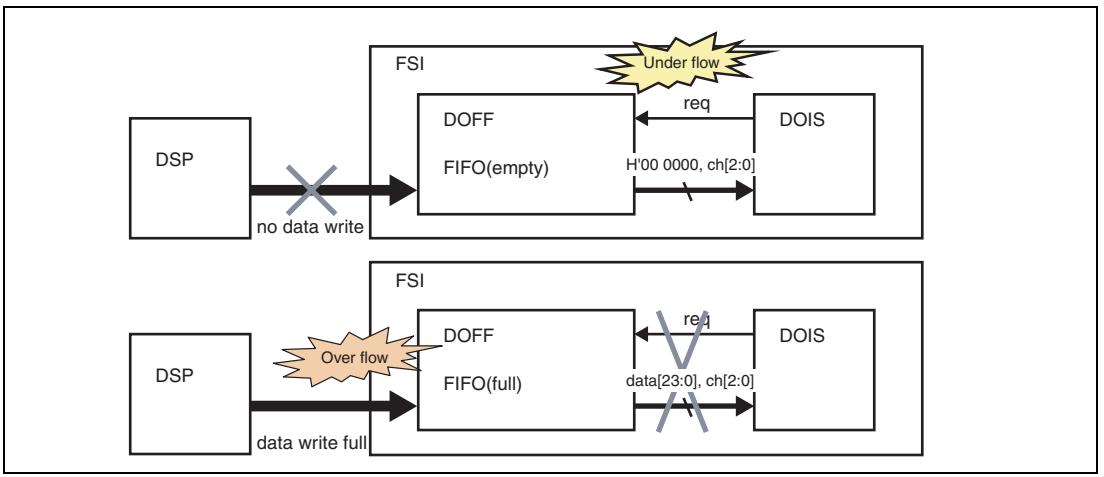


Figure 45.12 Output FIFO Underflow/Overflow

An input FIFO overflow occurs if 3-line serial data is input to the FIFO externally when it is full. The write to the input FIFO that caused an overflow is ignored. An underflow occurs when a read request is issued by the DSP while the input FIFO is empty. When it occurs, the read request returns 0. No underflow occurs under management by the DSP.

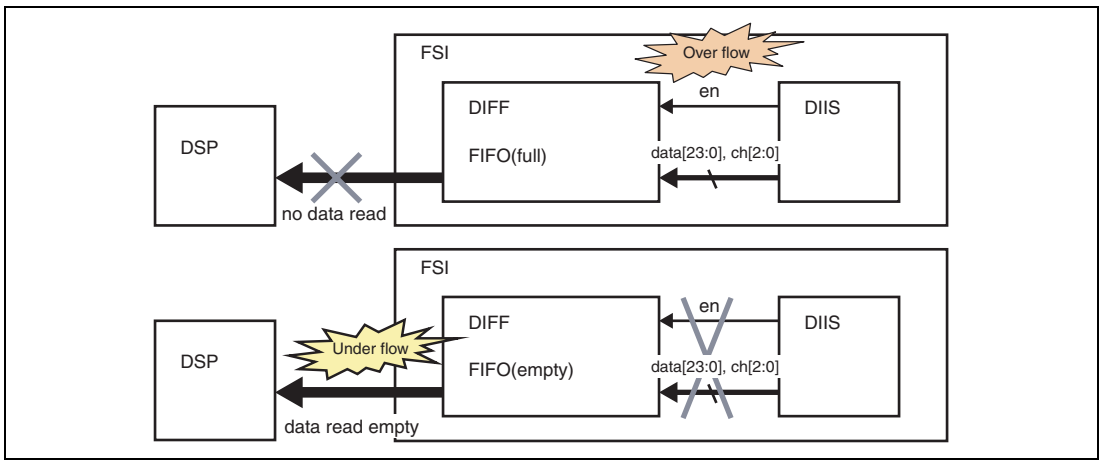


Figure 45.13 Input FIFO Overflow/Underflow

Table 45.8 lists the error flags of the FSI. The FSI has four error flags. The flow of clearing the error flags for the serial output is shown in figure 45.14. Similarly, the flow of clearing the error flags for the serial input is shown in figure 45.15.

Table 45.8 Error Flags

Number	Register	Contents	
(1)	A_DOFF_ST.OF	Serial output	Overflow error
(2)	A_DOFF_ST.UF		Underflow error
(3)	A_DIFF_ST.OF	Serial input	Overflow error
(4)	A_DIFF_ST.UF		Underflow error

(1) Clearing the overflow error flag for serial output

When an overflow occurs during serial output and the corresponding flag is detected, execute application-specific processing before clearing the flag.

(2) Clearing the underflow error flag for serial output

When an underflow occurs during serial output and the corresponding flag is detected, execute application-specific processing and then write the output data to the A_DODT register before clearing the flag.

(3) Clearing the overflow error flag for serial intput

When an overflow occurs during serial intput and the corresponding flag is detected, execute application-specific processing and then read intput data from the A_DIDT register before clearing the flag.

(4) Clearing the underflow error flag for serial intput

When an underflow occurs during serial intput and the corresponding flag is detected, execute application-specific processing before clearing the flag.

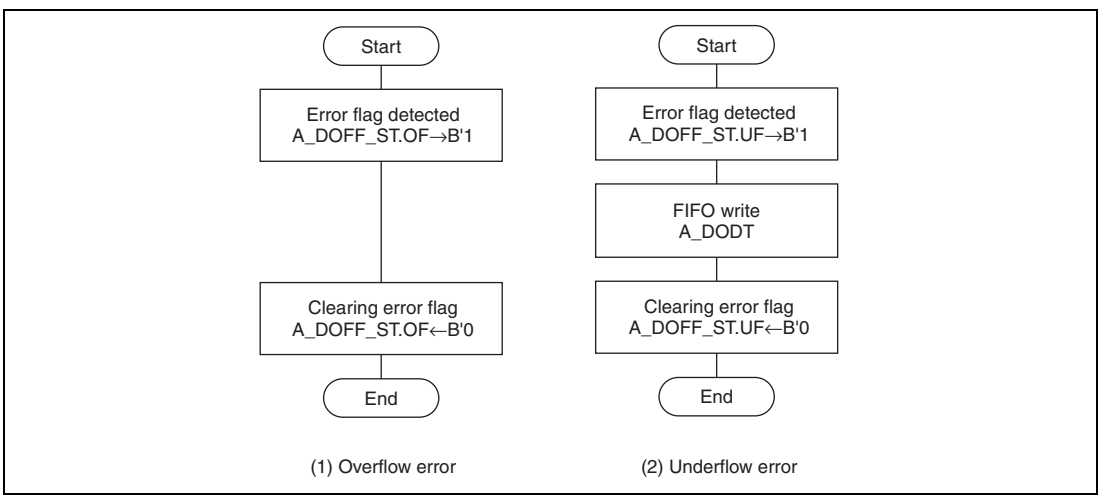


Figure 45.14 Flow of Clearing the Error Flag for Serial Output

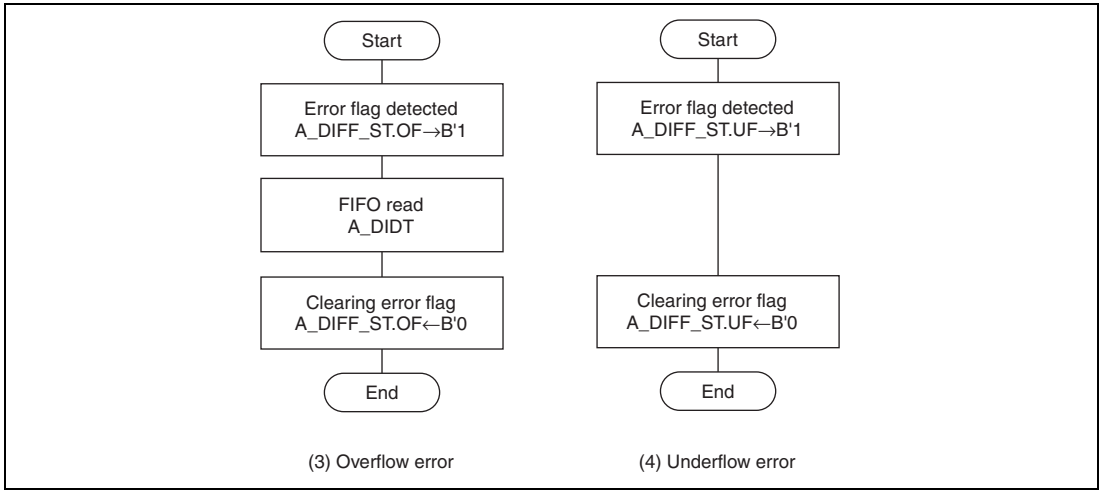


Figure 45.15 Flow of Clearing the Error Flag for Serial Input

45.7 FIFO Clear

The FSI has a FIFO clear function for input and output separately. This function does the following:

(1) Clearing the UF/OF error history

Clears the FIFO status to 0.

(2) Clearing the address pointer of the FIFO control

Initializes the counter that controls the on-chip RAM access. When a FIFO clear is performed, the address pointer returns to the start address and data prior to the clear is overwritten.

(3) Clearing data prefetched by the BUFC block and the buffer pointer

Clears data prefetched by the BUFC block for continuous reads and the buffer pointer.

(4) Masking the FIFO control block

Can be enabled for the request (CH0) corresponding to the second sample after a FIFO clear. This is because the state of the mask is determined by the request (output) and enable (input) after the format and clock settings is fixed.

45.8 Serial Data Formats

All formats for 3-line serial data are shown below.

(1) Slave Clock Inversion

The serial conversion blocks (DOIS and DIIS) operate based on the falling edge of the LR clock (CH0) that is inverted on the rising edge of the bit clock. Although the master clock is generated according to the specification in master operation, the inversion register needs to be set properly if the external LR and bit clocks are input directly in slave operation. Along with serial data formats, the inversion settings for slave operation are shown below for reference.

Table 45.9 Slave Clock Inversion Register Settings

Reference	Number of Channels	Format	Slave LR Clock Register Inversion Setting A(B)_CKG2.LRS	Slave Bit Clock Register Inversion Setting A(B)_CKG2.BRS
Figure 45.16	1 channel	MONO	1 (inverted)	0
	1 channel	MONO Delay	1 (inverted)	0
Figure 45.17	2 channels	PCM (STEREO)	1 (inverted)	1 (inverted)
Figure 45.18	2 channels	I2S (STEREO Delay)	0	1 (inverted)
Figure 45.19	1 to 8 channels	TDM (multi-channel)	1 (inverted)	1 (inverted)
Figure 45.20	1 to 8 channels	TDM (multi-channel) Delay	1 (inverted)	1 (inverted)

(2) Master Clock Inversion

The LR and bit clock signals, which are output in master mode, can be inverted according to the specifications of the DAC and ADC to be connected. In the related figures below, if the bit clock and LR clock signals are assumed to be output in master mode, the clock inversion register is set as shown in table 45.10.

Table 45.10 Master Clock Inversion Register Settings

Reference	Number of Channels	Format	Master LR Clock Register Inversion Setting A(B)_CKG2_RV.LRM	Master Bit Clock Register Inversion Setting A(B)_CKG2.BRM
Figure 45.16	1 channel	MONO	0	0
Figure 45.18	1 channel	MONO Delay	0	0
Figure 45.17	2 channels	PCM (STEREO)	0	1 (inverted)
Figure 45.18	2 channels	I2S (STEREO Delay)	0	1 (inverted)
Figure 45.19	1 to 8 channels	TDM (multi-channel)	0	1 (inverted)
Figure 45.20	1 to 8 channels	TDM (multi-channel) Delay	0	1 (inverted)

The MONO and Mono delay serial data formats are shown below.

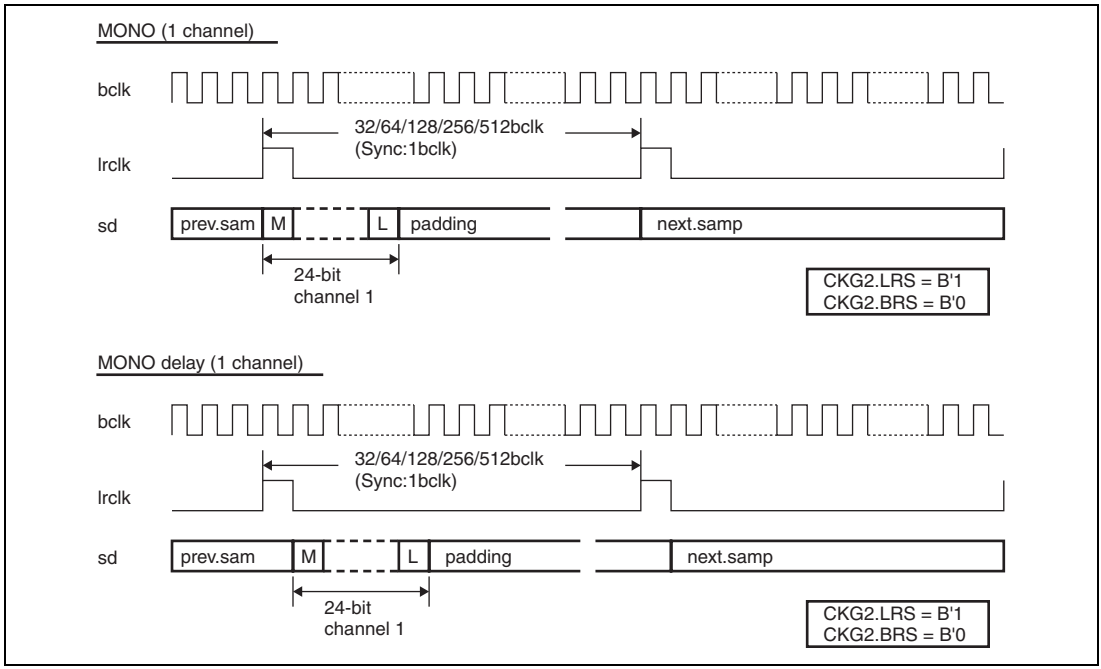


Figure 45.16 MONO and Mono delay Formats

The PCM and I2S serial data formats are shown below.

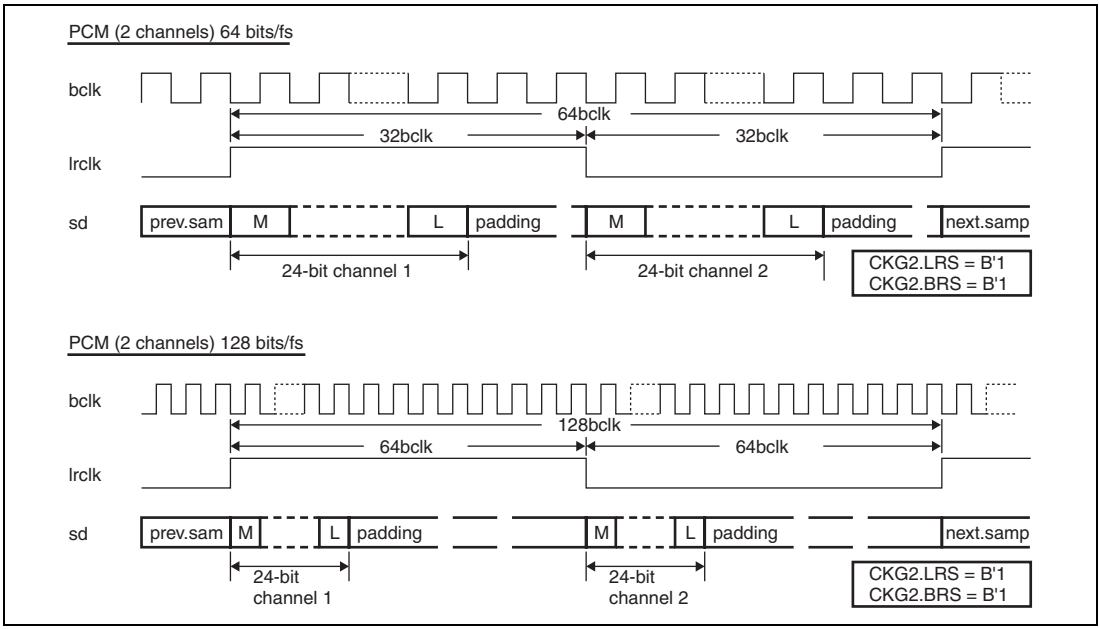


Figure 45.17 PCM Formats

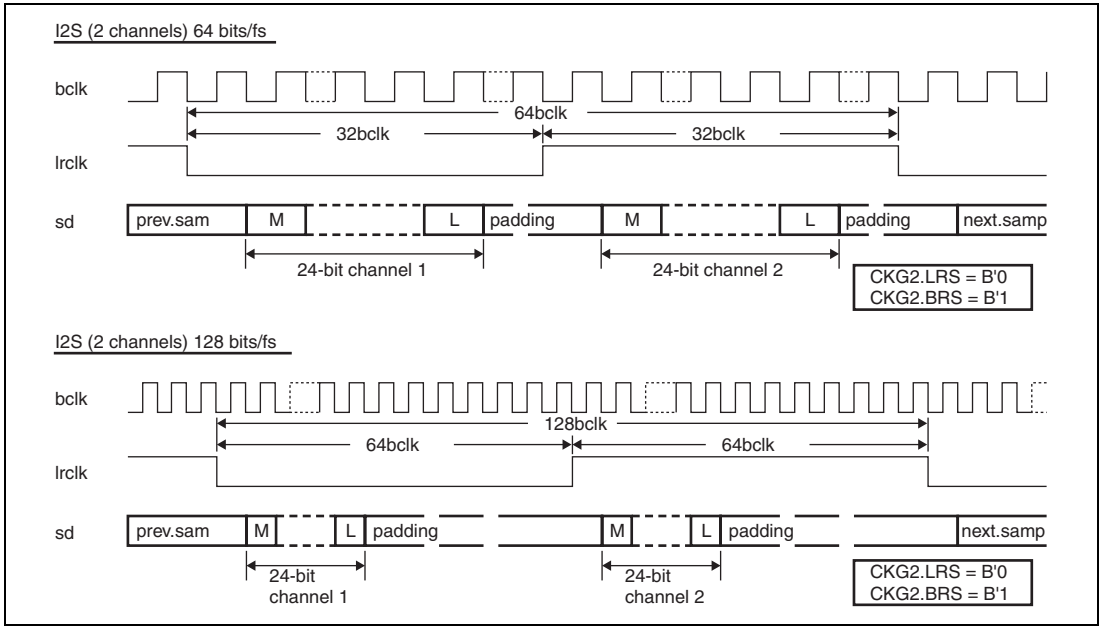


Figure 45.18 I2S Formats

The TDM and TDM_Delay serial data formats are shown below.

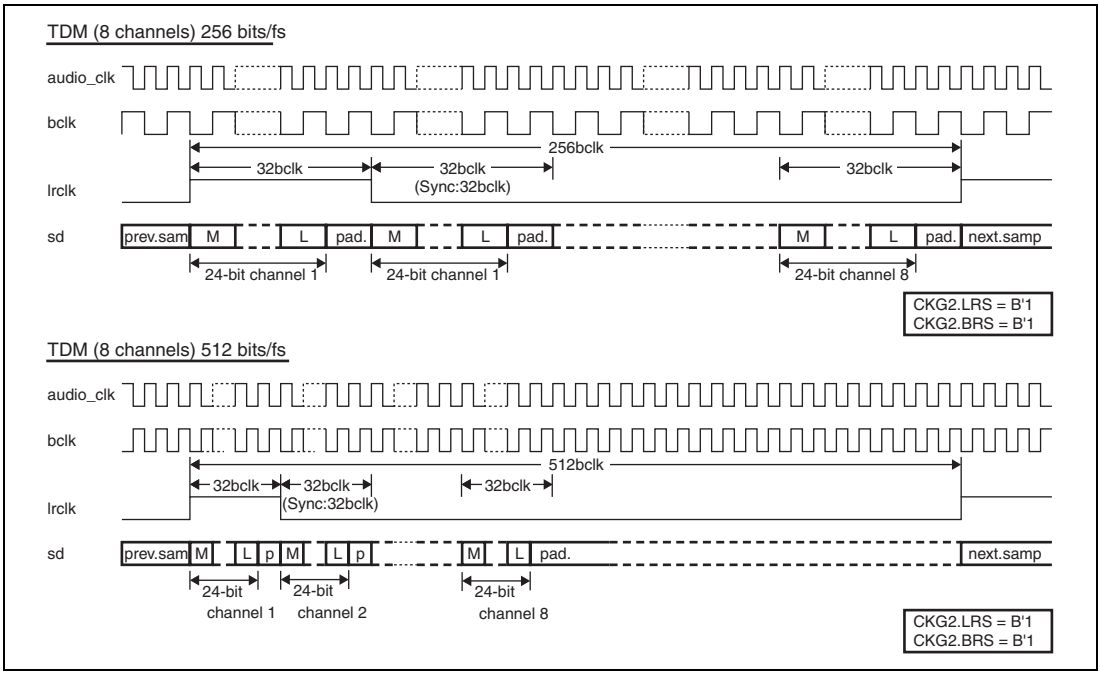


Figure 45.19 TDM Formats

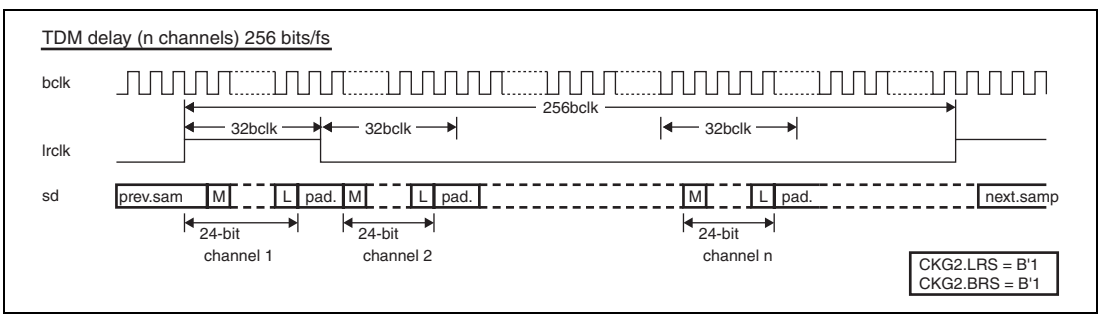


Figure 45.20 TDM Delay Formats

45.9 Operation Flows

A relationship between the entire FSI operation and each operation flow is shown in figure 45.21.

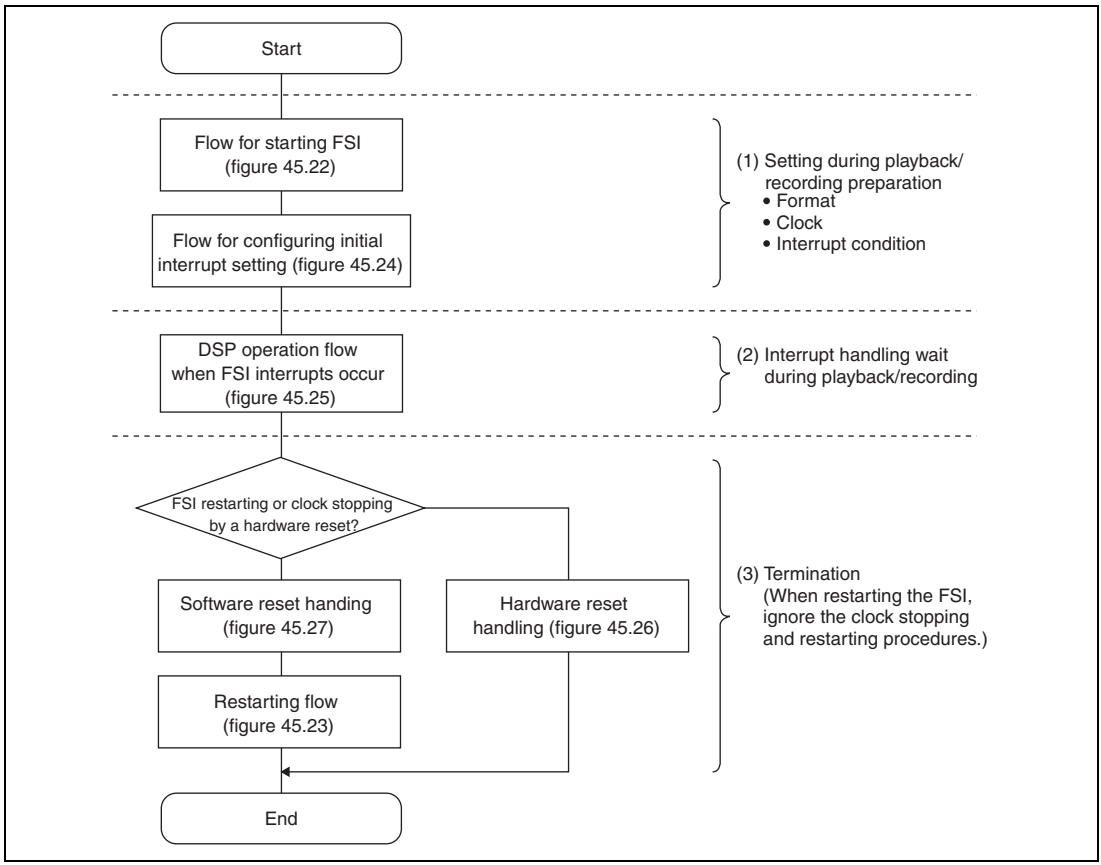


Figure 45.21 Entire FSI Operation and Each Operation Flow

45.9.1 Procedure for Starting the FSI

A flow chart of the procedure for starting the FSI for serial port A is shown in Figure 45.22. This flow chart is applicable for serial port B.

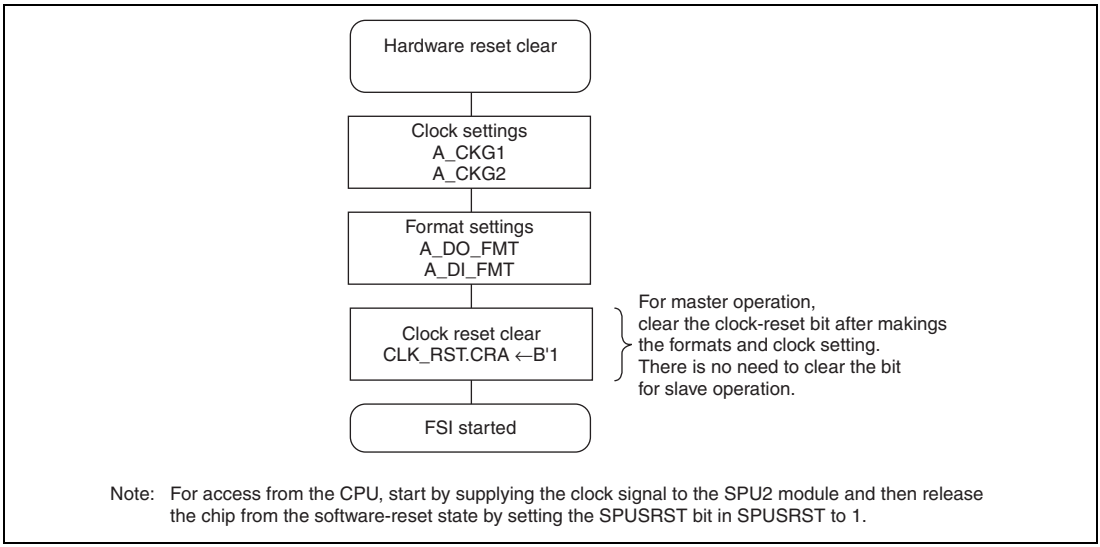


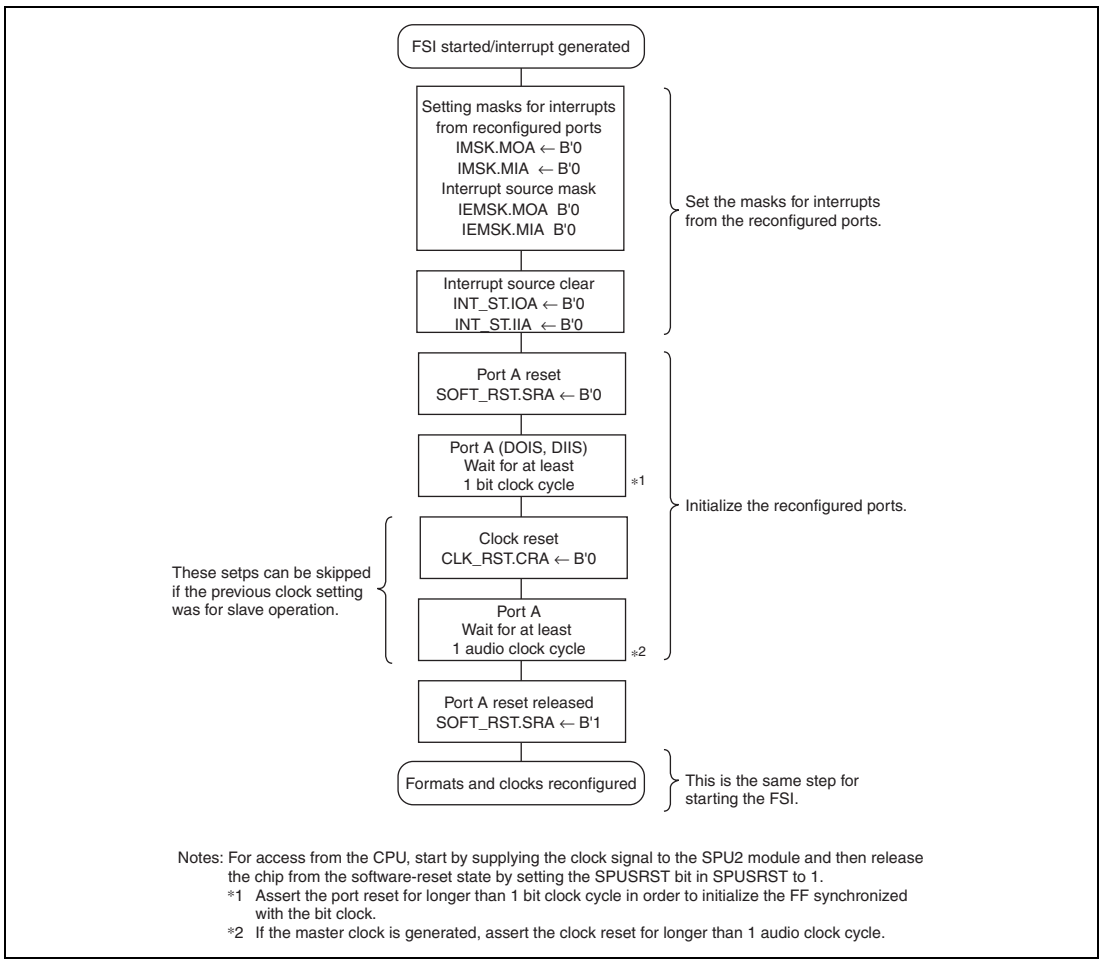
Figure 45.22 Flow for Starting FSI (Port A)

Use the steps above to start the FSI. The configured bit and LR clocks are input to the conversion block (DOIS and DIIS modules), and data requests generated and data writes are started in the FIFO control block (DOFF and DIFF modules).

45.9.2 Procedure for Restarting the FSI

A flow chart for restarting the FSI for serial port A is shown below. This flow chart is applicable for serial port B. To reconfigure transfer format and clock setting, follow the steps in this flow.

The flow above is not necessary when a hardware reset is used for re-setting.



Notes: For access from the CPU, start by supplying the clock signal to the SPU2 module and then release the chip from the software-reset state by setting the SPUSRST bit in SPUSRST to 1.
*1 Assert the port reset for longer than 1 bit clock cycle in order to initialize the FF synchronized with the bit clock.
*2 If the master clock is generated, assert the clock reset for longer than 1 audio clock cycle.

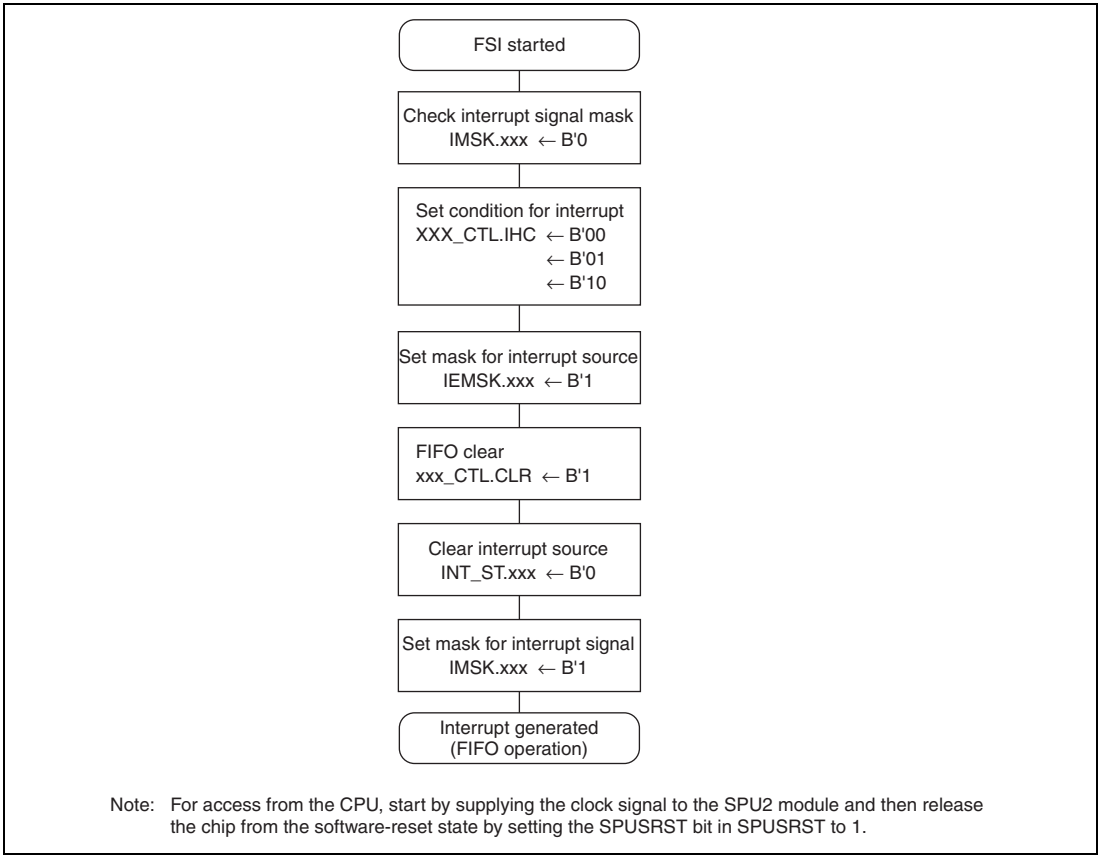
Figure 45.23 Flow for Restarting FSI (Port A)

Perform a port reset to initialize the port after setting the masks for the reconfigured port.

After initializing the port, return to the steps in section 45.9.1, Procedure for Starting the FSI to reconfigure the format and clock settings.

45.9.3 Procedure for Configuring Initial Interrupt Setting

A flow for configuring the initial interrupt setting after the FSI is started is shown in Figure 45.24.



Note: For access from the CPU, start by supplying the clock signal to the SPU2 module and then release the chip from the software-reset state by setting the SPUSRST bit in SPUSRST to 1.

Figure 45.24 Flow for Configuring Initial Interrupt Setting

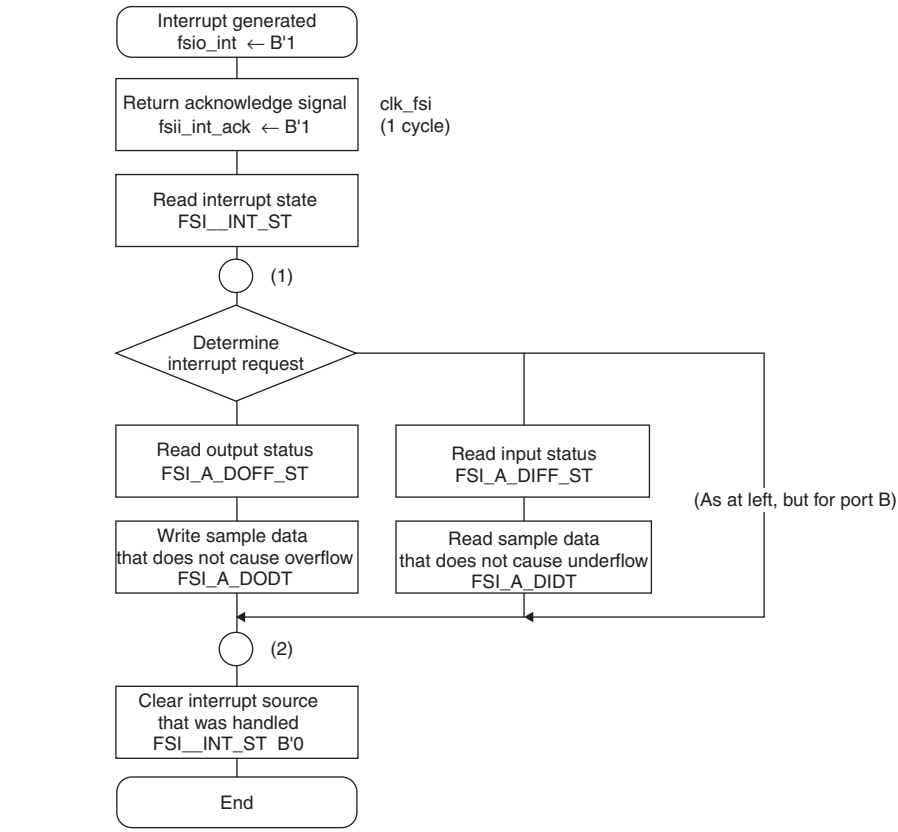
Interrupt settings can be configured for ports A/B and input/output (4 modes) independently. "xxx" in the figure indicates any signal to be transferred.

After checking if the mask for the signal to be transferred is set, select the interrupt condition and set the interrupt source mask, then perform a FIFO clear. Clear the interrupt source that may be generated and retained before the FIFO clear, clear the signal mask.

An interrupt signal occurs based on the FIFO capacity after a FIFO clear. Use this flow to configure interrupts if audio data are used for input and output operation initially.

45.9.4 Example of SPU2/DSP Operation Flow when FSI Interrupts Occur

Figure 45.25 shows an example of the operation flow that the SPU2/DSP follows to transfer audio data continuously after it accepts an interrupt.



Note: For access from the CPU, start by supplying the clock signal to the SPU2 module and then release the chip from the software-reset state by setting the SPUSRST bit in SPUSRST to 1.

Figure 45.25 Example of SPU2/DSP Operation Flow when FSI Interrupts Occur

The DSP that accepts the interrupt signal returns an acknowledge signal for one cycle to the FSI. Then, it reads the interrupt state to determine which FIFO caused the interrupt and handles the interrupt adequately. When it is handled, the DSP clears the interrupt source and exits the flow. Clearing the interrupt source allows the FSI to assert a new interrupt signal.

If processing for multiple interrupt sources is to proceed in response to a single interrupt, return from point (2) to point (1) in the figure the required number of times, and then clear the source bits by a single register-write operation.

Although they will only not be generated if the flow of DSP operations is correct, use a register-write operation to clear the source bit in the case of interrupt generation by an empty source. If the source bit is not cleared after processing in response to the interrupt, the interrupt signal will not be generated in response to the corresponding condition.

45.9.5 Procedures for Stopping and Restarting Clocks

Figures 45.26 and 45.27 show the procedures for stopping and restarting clocks.

Before stopping all the FSI clocks, apply a hardware reset. After restarting the clocks, follow the procedure for starting the FSI shown in figure 45.22.

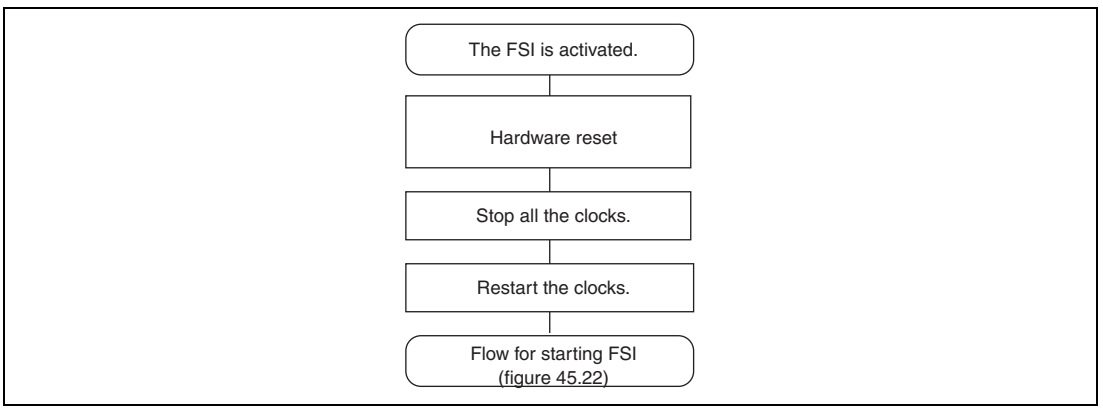


Figure 45.26 Procedures for Stopping and Restarting All Clocks (Hardware Reset)

When stopping only the clock for transferring one port, follow the FSI restarting procedure shown in figure 45.23 to reset the pertinent transfer and stop the clocks. When restarting the transfer, restart the stopped clocks and only set the items for the terminated transfer according to the flow for starting FSI shown in figure 45.22.

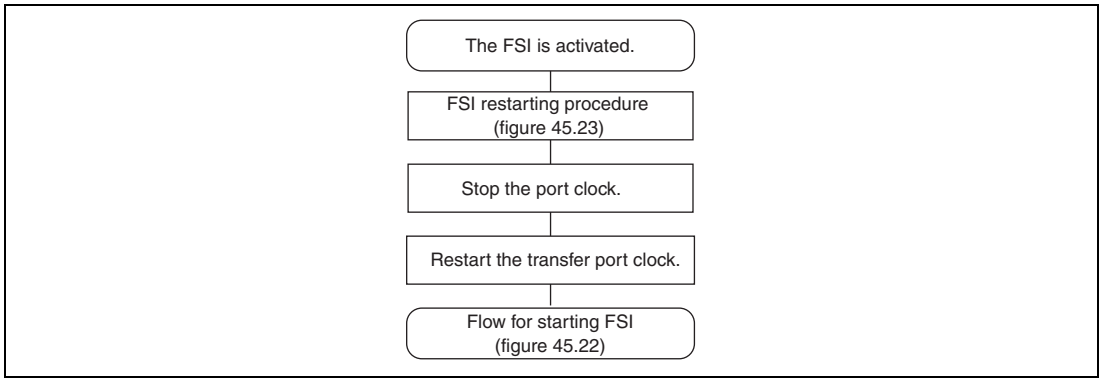


Figure 45.27 Procedures for Stopping and Restarting All Clocks (Software Reset)

When stopping all clocks without applying hardware reset, follow the flow for restarting FSI shown in figure 45.23 to stop. After having restarted the clocks, follow the flow for restarting FSI shown in figure 45.22.

45.10 Restrictions

Table 45.11 shows the restrictions that apply to the use of the FSI.

Table 45.11 Usage Restrictions

		Restrictions
No.	**Item**	**Description**
1	Clock setting for master operation	In TDM and TDM Delay modes, the LR clock cannot be generated from 64 fs and 128 fs of audio clock, so they should not be set.
2		The bit clock setting (BPFMD register) is restricted by the input audio clock and format (any bit clock faster than the audio clock cannot be generated).
3	Timing for generating interrupt signal	An interrupt signal is generated 2 cycles (clk_fsi) after the FIFO status matches the interrupt source condition (match event) while all masks are cleared.
4	Worst case overflow/underflow	In the worst case such as system clock: 4MHz, fs: 96kHz (bit_clk = 50MHz), TDM (8 channels), overflow and underflow errors occur about 10 system clock cycles after the output interrupt (FIFO empty) and input interrupt (FIFO full) signals are generated. To enable continuous operation in such a worst case, select the half of the FIFO capacity as the level for generating interrupt. Additionally, the system clock should be nearly equal to the bit clock for stable continuous operation. In this case, the number of cycles from the FIFO full state to generating an overflow is extended from 10 cycles to about 100 cycles by using the system clock that is ten times faster (40MHz) and to about 700 cycles by selecting the half of the FIFO capacity as the level for generating interrupt. Consequently, it takes about 70000 cycles (product of the two cycle counts) until an overflow occurs by doing the both.
5	FIFO status read	Since the FIFO status is read with 1 wait, the result indicates the status that is 1 cycle (clk_fsi) old.
6	Simultaneous input and output operations	Simultaneous operation such as master-master or slave-slave operation is not allowed on the same serial port with different formats and LR/bit clocks (one serial port has only one I/O system)
7	Port reset	When performing an internal reset of the DOIS and DIIS modules due to the change in formats or clocks, the reset should be asserted longer than the input bit clock.

Restrictions

No.	Item	Description
8	Clock reset	Since the counter for generating the master clock is synchronized with the audio clock, a reset should be asserted longer than the audio clock to reset the counter for reconfiguration.
9	Master output in the same phase	When ports A and B are in master operation at the same time, the clock signals being in the same phase cannot be guaranteed.
10	Access restrictions during software rest	Reading of the read registers (A_DIDT, B_DIDT) is prohibited during a software rest (SOFT_RST.SR) or resetting of the corresponding port (SOFT_RST.SRA, SOFT_RST.SRB).

Section 46 ATAPI Interface (ATAPI)

46.1 General Description

The ATAPI interface provides both the ATA and ATAPI physical interfaces. This device also supports both the ATA task and ATAPI packet commands.

46.2 Features

- Supporting primary channel
- Supporting master/slave
- Supporting 3.3 V I/O interface
- Supporting PIO modes 0 to 4, the multiword DMA modes 0 to 2, and the Ultra DMA modes 0 to 4
- Supporting descriptor mode

46.3 External Interface

Signal	(ATAPI Specification)	Function	I/O
IDED[15:0]	(DD(15:0))	Bidirectional data bus	I/O
IDEA[2:0]	(DA(2:0))	Address bus	Output
$\overline{\text{IODACK}}$	(DMACK#)	Primary channel DMA acknowledge (active low)	Output
IODREQ	(DMARQ)	Primary channel DMA request (active high)	Input
$\overline{\text{IDECS}}$[1:0]	(CS0#, CS1#)	Primary channel chip select (active low)	Output
$\overline{\text{IDEIOWR}}$	(DIOW#, STOP)	Primary channel disk write (active low)	Output
$\overline{\text{IDEIORD}}$	(DIOR#, HDMARDY#, HSTROBE)	Primary channel disk read (active low)	Output
IDEIORDY	(IORDY, DDMARDY#, DSTROBE)	Primary channel ready signal (active high)	Input
IDEINT	(INTRQ)	Primary channel interrupt request* (active high)	Input
$\overline{\text{IDERST}}$	(RESET#)	Primary channel ATAPI device reset (active low)	Output
DIRECTION	—	External level shifter direction signal (0 when writing to the device)	Output
$\overline{\text{EXBUF\_ENB}}$	—	External level shifter enable signal (active low)	Output

Note: * The ATAPI interface treats the interrupt signal from the ATAPI device as a level-triggered input.

46.4 Block Diagram

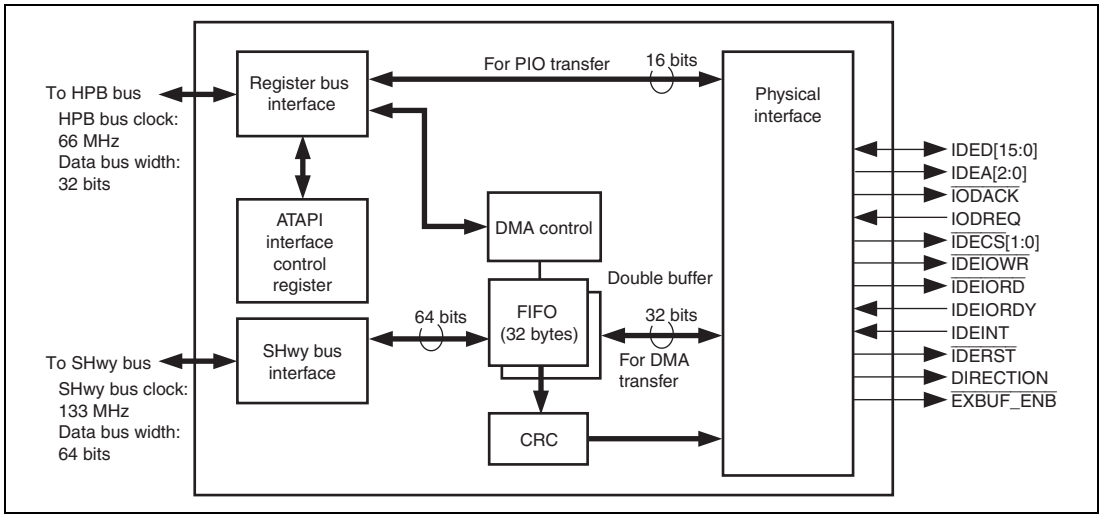


Figure 46.1 ATAPI Block Diagram

46.5 Register Description

The following register set is allocated to the SH register map space.

46.5.1 ATAPI Interface Registers

Table 46.1 ATA Task File Register Map

Address	Read Register	Write Register	Pin Address ($\overline{\text{IDECS}}$[1:0], IDEA[2:0]) H: High Level L: Low Level @3.3 V I/O	Access Size*[1] (Available Bit Size)	Register Location
H'A4DA2100	Data	Data	HL-LLL/HH-XXX (X: Don't care)	32 (16)*[2]	Drive
H'A4DA2104	Error	Function	HL-LLH	32 (8)*[3]	Drive
H'A4DA2108	Sector count	Sector count	HL-LHL	32 (8)*[3]	Drive
H'A4DA210C	Sector number	Sector number	HL-LHH	32 (8)*[3]	Drive
H'A4DA2110	Cylinder low	Cylinder low	HL-HLL	32 (8)*[3]	Drive
H'A4DA2114	Cylinder high	Cylinder high	HL-HLH	32 (8)*[3]	Drive
H'A4DA2118	Device/head	Device/head	HL-HHL	32 (8)*[3]	Drive
H'A4DA211C	Status	Command	HL-HHH	32 (8)*[3]	Drive
H'A4DA2120	Eliminated	Eliminated	LH-LLL	32 (8)*[2]	External*[4]
H'A4DA2124	Eliminated	Eliminated	LH-LLH	32 (8)*[2]	External*[4]
H'A4DA2128	Eliminated	Eliminated	LH-LHL	32 (8)*[2]	External*[4]
H'A4DA212C	Eliminated	Eliminated	LH-LHH	32 (8)*[2]	External*[4]
H'A4DA2130	Eliminated	Eliminated	LH-HLL	32 (8)*[2]	External*[4]
H'A4DA2134	Eliminated	Eliminated	LH-HLH	32 (8)*[2]	External*[4]
H'A4DA2138	Alternate status	Device control	LH-HHL	32 (8)*[3]	Drive
H'A4DA213C	Eliminated	Eliminated	LH-HHH	32 (8)*[3]	External*[4]

Notes: These resisters are allocated to the ATAPI or ATA device, and are not allocated to this module.

1. These registers must be accessed in longwords (32 bits) by the CPU. Byte or word accesses are prohibited.
2. Bits 15 to 0 of the data bus are used.
3. Bits 7 to 0 of the data bus are used.
4. External access is valid when the RADRE bit in the ATAPI control registers is 1.

Table 46.2 ATAPI Packet Command Task File Register Map

Address	Read Register	Write Register	Pin Address ($\overline{\text{IDECS}}$[1:0], IDEA[2:0])	Access Size*1 (Available Bit Size)	Register Location
H'A4DA2100	Data	Data	HL-LLL	32 (16)*2	Drive
H'A4DA2104	Error	Function	HL-LLH	32 (8)*3	Drive
H'A4DA2108	Interrupt source	—	HL-LHL	32 (8)*3	Drive
H'A4DA210C	—	—	HL-LHH	32 (8)*3	Drive
H'A4DA2110	Byte Count Low	Byte Count Low	HL-HLL	32 (8)*3	Drive
H'A4DA2114	Byte Count High	Byte Count High	HL-HLH	32 (8)*3	Drive
H'A4DA2118	Device select	Device select	HL-HHL	32 (8)*3	Drive
H'A4DA211C	Status	Command	HL-HHH	32 (8)*3	Drive
H'A4DA2120	Eliminated	Eliminated	LH-LLL	32 (8)*2	External*4
H'A4DA2124	Eliminated	Eliminated	LH-LLH	32 (8)*2	External*4
H'A4DA2128	Eliminated	Eliminated	LH-LHL	32 (8)*2	External*4
H'A4DA212C	Eliminated	Eliminated	LH-LHH	32 (8)*2	External*4
H'A4DA2130	Eliminated	Eliminated	LH-HLL	32 (8)*2	External*4
H'A4DA2134	Eliminated	Eliminated	LH-HLH	32 (8)*2	External*4
H'A4DA2138	Alternate Status	Device Control	LH-HHL	32 (8)*3	Drive
H'A4DA213C	Eliminated	Eliminated	LH-HHH	32 (8)*3	External*4

Notes: These resisters are allocated to the ATAPI or ATA device, and are not allocated to this module.

1. These registers must be accessed in longwords (32 bits) by the CPU. Byte or word accesses are prohibited.
2. Bits 15 to 0 of the data bus are used.
3. Bits 7 to 0 of the data bus are used.
4. External access is valid when the RADRE bit in the ATAPI control register is 1.

Table 46.3 ATAPI Interface Control Register Map

Address	Register Name	Abbreviation	Access Type	Register Access Size*
H'A4DA2000 to H'A4DA20FC	Reserved		R	32
H'A4DA2140 to H'A4DA217C	Reserved		R	32
H'A4DA2180	ATAPI control	ATAPI_CONTROL1	R/W	32
H'A4DA2184	ATAPI status	ATAPI_STATUS	R/W	32
H'A4DA2188	Interrupt enable	ATAPI_INT_ENABLE	R/W	32
H'A4DA218C to H'A4DA2194	Reserved		R	32
H'A4DA2198	Descriptor table base address	ATAPI_DTB_ADR	R/W	32
H'A4DA219C	DMA start address	ATAPI_DMA_START_ADR	R/W	32
H'A4DA21A0	DMA transfer count	ATAPI_DMA_TRANS_CNT	R/W	32
H'A4DA21A4	ATAPI control 2	ATAPI_CONTROL2	R/W	32
H'A4DA21A8	Reserved		R	32
H'A4DA21AC	Reserved		R	32
H'A4DA21B0	ATAPI signal status	ATAPI_SIG_ST	R	32
H'A4DA21BC	Byte swap	ATAPI_BYTE_SWAP	R/W	32
H'A4DA21C0	PIO timing 1	ATAPI_PIO_TIMING1	R/W	32
H'A4DA21C4	PIO timing 2	ATAPI_PIO_TIMING2	R/W	32
H'A4DA21C8	Multiword DMA timing	ATAPI_MULTI_TIMING	R/W	32
H'A4DA21CC	Ultra DMA timing	ATAPI_ULTRA_TIMING	R/W	32
H'A4DA21D0 to H'A4DA21FC	Reserved		R	32

Notes: These resisters are allocated to this module.

* These registers must be accessed in longwords (32 bits) by the CPU. Byte or word accesses are prohibited.

46.5.2 ATAPI I/F Control Register Map

- Legends for register description:
  Initial value: Register value after reset
  —: Undefined value
  R/W: Readable and writable bit. The write value can be read.
  R/WC0: Readable and writable bit. When 0 is written, the bit is initialized. When 1 is written, it is ignored.
  R: Read only register, only 0 should be written.
  —/W: Write only bit. The read value is undefined.

All control/status registers are active high.

(1) ATAPI Control Register (ATAPI_CONTROL1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RADRE	DTA 32M	—	DTCD	—	RESET	M/S	—	UDM AEN	DESE	R/W	STOP	START
Initial value:	—	—	—	0	0	0	0	—	0	0	1	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R	R/W	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	—	R	Reserved
12	RADRE	0	R/W	Enables access to reserved bit in the ATA task file register map. 1: Access to reserved bit is enabled. 0: Access to reserved bit is disabled.
11	DTA32M	0	R/W	Enables bits 31 to 29 of the descriptor DMA start address in descriptor table operation mode. This causes the termination flag to be changed from bit 32 to bit 0. 1: Enabled bits of the descriptor DMA start address are bits 31 to 2. 0: Enabled bits of the descriptor DMA start address are bits 28 to 2
10	—	0	R	This bit is fixed to 0 and cannot be set to 1.

Bit	Bit Name	Initial Value	R/W	Description
9	DTCD	0	R/W	DTCD controls the operating mode for device termination occurring when the Ultra DMA operates. No abnormal termination occurs if the specified number of transfers has not been reached after the reception of device termination. Transfer will restart after waiting a DMARQ from the next device. Some of the existing ATA devices are those handling the pause the same way as device termination. If, therefore, the specified number of transfers has not been reached after the reception of device termination, no abnormal termination occurs and it is necessary to restart transfer after waiting a DMARQ from the next device. This operating mode is called the "device termination continuation mode." 1: Suppressing the device termination continuation mode 0: Device termination continuation mode
8	—	0	R	This bit is fixed to 0 and cannot be set to 1
7	RESET	0	R/W	RESET controls an ATAPI device reset. If this bit is set to 1 then the ATAPI reset signal is asserted. $\overline{\text{IDERST}}$ is active low signal. When this bit is set to 1 then $\overline{\text{IDERST}}$ is low. When this bit is cleared to 0 then $\overline{\text{IDERST}}$ is high.
6	M/S	0	R/W	M/S selects an ATAPI device master or slave. 1: ATAPI device functions as a master 0: ATAPI device functions as a slave
5	—	1	R	This bit is fixed to 1 and cannot be cleared to 0.
4	UDMAEN	0	R/W	UDMAEN is Ultra DMA enable. When Ultra DMA is used, set this bit to 1. Clear it to 0 when using the multiword DMA mode.
3	DESE	0	R/W	DESE controls the descriptor table operation mode. 1: Descriptor functions enabled 0: Descriptor functions disabled
2	R/W	0	R/W	R/W controls FIFO read/write. 1: FIFO read (data-in operation at DMA transfer) 0: FIFO write (data-out operation at DMA transfer) Set this bit to 1 when reading data from the ATAPI device. Clear it to 0 when writing data to the ATAPI device.

Bit	Bit Name	Initial Value	R/W	Description
1	STOP	0	R/W	This bit forces termination of DMA transfer. When writing 0: Ignored 1: Forcibly terminates data transfer When reading 0: Forced termination command is not issued. 1: Forced termination of data transfer command is issued. This bit is cleared to 0 when the next DMA starts. To power down (ATAPI internal clock turns off) while DMA is active (ACT = 1), DMA transfer should be forcibly terminated. Note: Transfer cannot always be resumed from the address at which DMA transfer has been forcibly terminated.
0	START	0	R/W	This bit initiates DMA transfer. If this bit set to 1 then the DMA transfer is started. When cleared to 0, this bit is ignored. When writing 0: Ignored 1: Starts DMA transfer When reading 0: DMA transfer is not active 1: DMA transfer is in busy state Note: Access to the task file register is prohibited while DMA is active.

(2) ATAPI Status Register (ATAPI_STATUS)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SW ERR	IFERR	DN END	DEV TRM	DEV INT	TOUT	ERR	NEND	ACT
Initial value:	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC	R/WC	R/WC	R/WC	R	R/WC	R/WC	R/WC	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	—	R	Reserved
8	SWERR	0	R/WC0	Software error bit. Setting this bit to 1 indicates that the task file register has been accessed while DMA is active. Such access is prohibited. For example, this bit is set to 1 if PIO transfer is performed during the transfer of the Ultra DMA or multiword DMA. In this case, no output is sent to the outside of the LSI so that access is ignored. This bit is reset by writing 0.
7	IFERR	0	R/WC0	IFERR Indicates that an ATAPI interface protocol error has been detected in the following cases: (IODREQ = 1) or (IDEIORDY = 0) when the ULTRA DMA data-in burst is in the host termination. IDEIORDY = 0 when the ULTRA DMA data-out burst is in the device termination. IDEIORDY = 0 when the ULTRA DMA data-out burst is initiated. (IODREQ = 1) or (IDEIORDY = 0) when the ULTRA DMA data-out burst is in the host termination. This bit is reset by writing 0.
6	DNEND	0	R/WC0	DNEND indicates that all DMAs have been successfully terminated in the descriptor mode. This bit is reset by writing 0.
5	DEVTRM	0	R/WC0	DEVTRM is set to 1 when the ATAPI device is terminated in the Ultra DMA mode before the number of DMA transfer bytes reach the value set in this ATAPI module. This bit is reset by writing 0.

Bit	Bit Name	Initial Value	R/W	Description
4	DEVINT	0	R	DEVINT indicates ATAPI device interrupt IDEINT status. This bit is read only. Since this bit does not hold its status in this module, if IDEINT becomes 0, this bit will also become 0. ATAPI interface treats the interrupt signal from the ATAPI device as a level-triggered input. According to the ATAPI standard, IDEINT will be negated by the ATAPI device within 400 ns of the negation of IDEIORD that reads the Status register to clear a pending interrupt.
3	TOUT	0	R/WC0	TOUT indicates that an IORDY timeout is detected. Timeout is detected if no response is returned (the IDEIORDY pin is at the Low level.) in 225 cycles or longer of Pixel Bus clock cycle time. This bit is reset by writing 0.
2	ERR	0	R/WC0	ERR is set to 1 when a DMA abort is detected. ERR=1 occurs if: 1. The host brings DMA transfer to a forced stop. 2. DTCD=1 and ACT=0 because of device termination This bit is reset by writing 0.
1	NEND	0	R/WC0	NEND indicates that DMA has been successfully terminated. This bit is reset by writing 0.
0	ACT	0	R	ACT indicates that DMA is active. To power down (ATAPI internal clock turns off) while DMA is active (ACT = 1), DMA transfer should be forcibly terminated (STOP = 1).

(3) Interrupt Enable Register (ATAPI_INT_ENABLE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ISW ERR	IIFERR	IDN END	IDEV TRM	IDEV INT	ITOUT	IERR	INEND	IACT
Initial value:	—	—	—	—	—	—	—	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	—	R	Reserved
8	iSWERR	0	R/W	SWERR interrupt enable
7	iIFERR	0	R/W	FERR interrupt enable
6	iDNEND	0	R/W	DNEND interrupt enable
5	iDEVTRM	0	R/W	DEVTRM interrupt enable
4	iDEVINT	0	R/W	DEVINT interrupt enable
3	iTOUT	0	R/W	TOUT interrupt enable.
2	iERR	0	R/W	ERR interrupt enable
1	iNEND	0	R/W	NEND interrupt enable
0	iACT	0	R/W	ACT interrupt enable bit. Since ACT is cleared automatically when a DMA transfer is completed, interrupt processing should be completed during assertion.

Note: When 1 is written to a bit, the interrupt signal corresponding to the bit in the ATAPI status register is enabled.

(4) PIO Timing Register 1 (ATAPI_PIO_TIMING1)

Set the number of machine cycles to the following bits before access to the ATAPI device.

The machine cycle is a base bus clock.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	pSDCT						
Initial value:	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	pMDCT						
Initial value:	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	—	R	Reserved
22 to 16	pSDCT	All 0	R/W	pSDCT sets the cycle time of the Slave ATAPI device.
15 to 7	—	—	R	Reserved
6 to 0	pMDCT	All 0	R/W	pMDCT sets the cycle time of the Master ATAPI device.

(5) PIO Timing Register 2 (ATAPI_PIO_TIMING2)

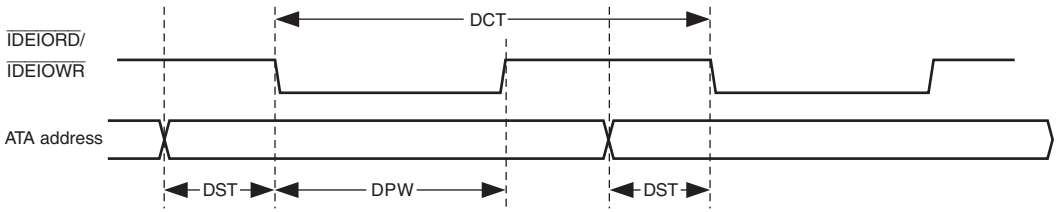
Set the number of machine cycles to the following bits before access to the ATAPI device.

The machine cycle is a base bus clock.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	pSDPW						—	—	—	—	pSDST			
Initial value:	—	—	0	0	0	0	0	0	—	—	—	—	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	pMDPW						—	—	—	—	pMDST			
Initial value:	—	—	0	0	0	0	0	0	—	—	—	—	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29 to 24	pSDPW	All 0	R/W	These bits set the IDEIORD/IDEIOWR pulse width of the slave ATAPI device.
23 to 20	—	—	R	Reserved
19 to 16	pSDST	All 0	R/W	These bits set the address setup time with respect to IDEIORD/IDEIOWR for the slave ATAPI device in the PIO mode.
15, 14	—	—	R	Reserved
13 to 8	pMDPW	All 0	R/W	These bits set the IDEIORD/IDEIOWR pulse width of the master ATAPI device.
7 to 4	—	—	R	Reserved
3 to 0	pMDST	All 0	R/W	These bits set the address setup time with respect to IDEIORD/IDEIOWR for the master ATAPI device in the PIO mode.



DCT: Cycle setting
DPW: Setting the low pulse width for IDEIORD/IDEIOWR
DST: Setting the setup time for the address and IDEIORD/IDEIOWR
Note: The prefix pS means slave and the prefix pM means master. DCT, DST, and DPW are set by multiplying the setting of each register by the cycle of the base clock.

Figure 46.2 PIO Timing Register

PIO timing register value table 1 (ATAPI_PIO_TIMING1) (Master / Slave)

Base Clock	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4
133 MHz	H'0051	H'0035	H'002D	H'0019	H'0011

PIO timing register value table 2 (ATAPI_PIO_TIMING2) (Master / Slave)

Base Clock	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4
133 MHz	H'280B	H'2808	H'2805	H'0C05	H'0B05

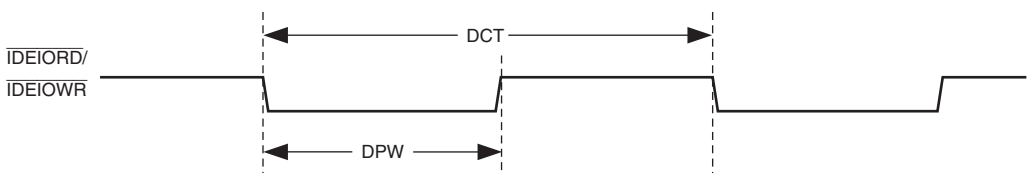
(6) Multiword DMA Timing Register (ATAPI_MULTI_TIMING)

Set the number of machine cycles to the following bits in this register before access to the ATAPI device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	mSDCT							—	—	mSDPW					
Initial value:	—	0	0	0	0	0	0	0	—	—	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	mMDCT							—	—	mMDPW					
Initial value:	—	0	0	0	0	0	0	0	—	—	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	—	R	Reserved
30 to 24	mSDCT	All 0	R/W	mSDCT sets the cycle time of the slave ATAPI device.
23, 22	—	—	R	Reserved
21 to 16	mSDPW	All 0	R/W	mSDPW sets the IDEIORD/IDEIOWR pulse width of the slave ATAPI device.
15	—	—	R	Reserved
14 to 8	mMDCT	All 0	R/W	mMDCT sets the cycle time of the master ATAPI device.
7, 6	—	—	R	Reserved
5 to 0	mMDPW	All 0	R/W	mMDPW sets the IDEIORD/IDEIOWR pulse width of the master ATAPI device.



DCT: Cycle setting
DPW: Setting the low pulse width for IDEIORD/IDEIOWR
Note: The prefix nS means slave and the prefix mM means master. DCT and DPW are set by multiplying the setting of each register by the cycle of the base clock.

Figure 46.3 Multiword DMA Timing Register

Multiword DMA timing register value table

Base Clock	Mode 0	Mode 1	Mode 2
133 MHz	H'411E	H'150C	H'110B

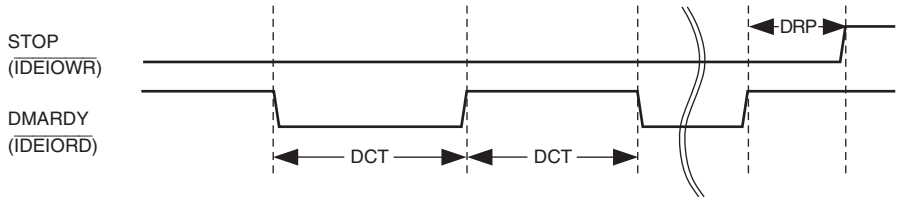
(7) Ultra DMA Timing Register (ATAPI_ULTRA_TIMING)

Set the number of machine cycles to the following bits in this register before access to the ATAPI device.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	uSDCT					—	—	uSDRP					
Initial value:	—	—	—	0	0	0	0	0	—	—	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	uMDCT					—	—	uMDRP					
Initial value:	—	—	—	0	0	0	0	0	—	—	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	—	R	Reserved
28 to 24	uSDCT	—	R/W	uSDCT sets the cycle time of the slave ATAPI device.
23, 22	—	—	R	Reserved
21 to 16	uSDRP	All 0	R/W	uSDRP sets the time from negating DMARDY (Not IDEIORDY) until suspension by the slave ATAPI device.
15 to 13	—	—	R	Reserved
12 to 8	uMDCT	All 0	R/W	uMDCT sets the cycle time of the master ATAPI device.
7, 6	—	—	R	Reserved
5 to 0	uMDRP	All 0	R/W	uMDRP sets the time from negating DMARDY (Not IDEIORDY) until suspension by the master ATAPI device.



DCT: Cycle setting
DRP: Setting the period from negating the DMARDY (IDEIORD) signal until issuing the STOP (IDEIOWR) signal; used for data-in burst
Note: The prefix uS means slave and the prefix uM means master. DCT and DRP are set by multiplying the setting of each register by the cycle of the base clock.

Figure 46.4 Ultra DMA Timing Register

Ultra DMA timing register value table

Base Clock	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4
133 MHz	H'1017	H'0B12	H'090F	H'070F	H'050F

Note: The maximum transfer rates for each mode are calculated from T2CYCTYPmin.
With T2CYCTYP(min) and 4-byte information, the maximum transfer rates in Mbyte/s can be obtained by (1/T2CYCTYPmin) * 4.

(8) Descriptor Table Base Address Register (ATAPI_DTB_ADR)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTBA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTBA[15:2]														—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	DTBA [31:2]	All 0	R/W	These bits indicate the descriptor table base address. Bits 31 to 0 are used to set the descriptor table base address on a byte basis. Bits 1 and 0 are ignored because it is necessary to secure the boundary of 32-bit addresses for the descriptor table.
1, 0	—	—	R	Reserved

Note: This address will not change and will retain its setting even after the DMA becomes active.

(9) Descriptor Table

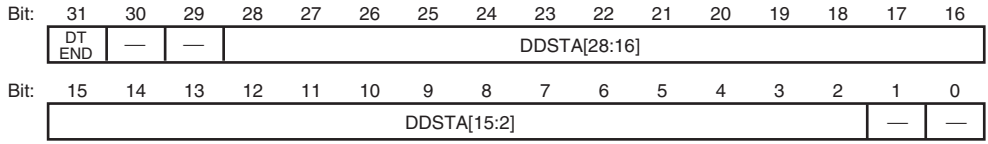
The descriptor table consists of the termination flag, the descriptor DMA start address (DDSTA), and the descriptor DMA transfer count (DDTRC).

(a) When Bits 28 to 2 of Descriptor DMA Start Address are Valid (DTA32M = 0, Initial value)

Descriptor Table Map in Memory

Address	Data Description
DTBA	The first termination flag (bit 31 = 0) and DDSTA
DTBA + 4	The first DDTRC
DTBA + 8	The second termination flag (bit 31 = 0) and DDSTA
DTBA + 12	The second DDTRC
:	:
DTBA + 8 × (n − 1)	The n-th termination flag (bit 31 = 1) and DDSTA
DTBA + 8 × (n − 1) + 4	The n-th DDTRC

(10) Termination Flag and Descriptor DMA Start Address



Bit	Bit Name	Description
31	DTEND	DTEND controls the termination of a descriptor DMA operation. 1: Terminating the descriptor DMA operation (When DTEND is 1, the system recognizes that the descriptor table is the last one.) 0: Validating the descriptor table (When DTEND is 0, the system reads the DMA transfer count, transfers the DMA, and reads the next descriptor table.)
30 to 29	—	Reserved
28 to 2	DDSTA[28:2]	DDSTA shows the DMA start address in descriptor operation. Bits 28 to 0 are used to set the descriptor table start address on a byte basis. Bits 1 and 0 are ignored because it is necessary to secure the boundary of 32-bit addresses in the DMA start address.
1, 0	—	Reserved

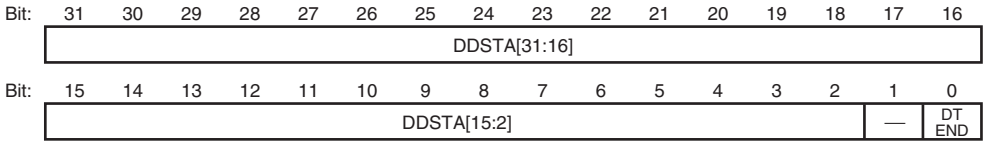
The valid flag and descriptor DMA start address should be set in the descriptor table base address + "m" in the memory, where m is multiple of 2 (such as 0, 2, 4…).

When Bits 31 to 2 of Descriptor DMA Start Address are Valid (DTS32M = 1)

Descriptor Table Map in Memory

Address	Data Description
DTBA	The first termination flag (bit 0 = 0) and DDSTA
DTBA + 4	The first DDTRC
DTBA + 8	The second termination flag (bit 0 = 0) and DDSTA
DTBA + 12	The second DDTRC
:	:
DTBA + 8 × (n – 1)	The n-th termination flag (bit 0 = 1) and DDSTA
DTBA + 8 × (n – 1) + 4	The n-th DDTRC

(11) Termination Flag and Descriptor DMA Start Address

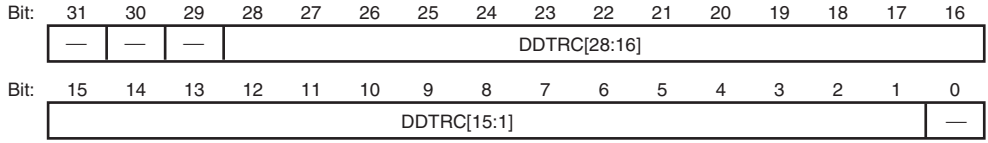


Bit	Bit Name	Description
31 to 2	DDSTA[31:2]	DDSTA shows the DMA start address in descriptor operation. Bits 31 to 0 are used to set the descriptor table start address on a byte basis. Bits 1 and 0 are ignored because it is necessary to secure the boundary of 32-bit addresses in the DMA start address.
1	—	Reserved
0	DTEND	DTEND controls the termination of a descriptor DMA operation. 1: Terminates the descriptor DMA operation (When DTEND is 1, the last descriptor table is detected.) 0: Validates the descriptor table (When DTEND is 0, the DMA transfer count is read, DMA transfer is performed, and the next descriptor table is read.)

The valid flag and descriptor DMA start address should be set in the descriptor table base address + "m" in the memory, where m is multiple of 2 (such as 0, 2, 4, …).

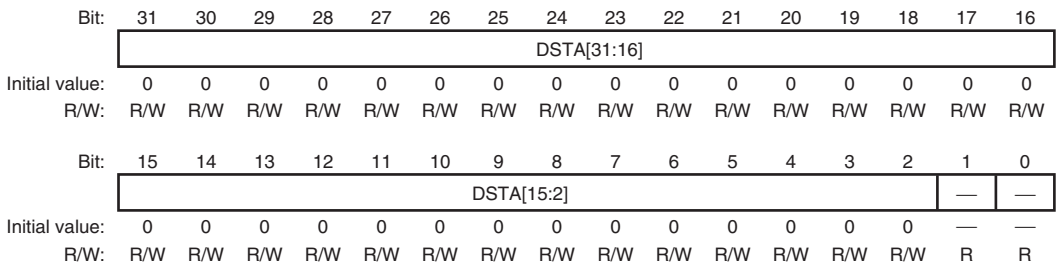
(12) Descriptor DMA Transfer Count

The descriptor DMA transfer count should be set in the descriptor table base address + "m" in the memory, where the value of m is any multiple number of 2 plus 1 (such as 1, 3, 5, …).



Bit	Bit Name	Description
31 to 29	—	Reserved
28 to 1	DDTRC[28:1]	These bits set the DMA transfer count during descriptor operation. Bits 28 to 0 are used to set the DMA transfer count on a byte basis. Bit 0 is ignored because the ATAPI data bus is handled on a 16-bit basis (on a word basis).
0	—	Reserved

(13) DMA Start Address Register (ATAPI_DMA_START_ADR)



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	DSTA[31:2]	All 0	R/W	DSTA sets a DMA start address that indicates the data transfer start address in the memory. Bits 31 to 0 are used to specify the DMA start address in byte. Bits 1 and 0 are ignored because it is necessary to secure the boundary of 32-bit addresses in the DMA start address.
1, 0	—	—	R	Reserved

Note: This address does not change and the set value is retained even after the DMA becomes active.

(14) DMA Transfer Count Register (ATAPI_DMA_TRANS_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	DTRC[28:16]												
Initial value:	—	—	—	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTRC[15:1]															—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	—	R	Reserved
28 to 1	DTRC [28:1]	All 0	R/W	DTRC sets the DMA transfer count. Bits 28 to 0 are used to set the DMA transfer count on a byte basis. Bit 0 is ignored because the ATAPI data bus is handled on a 16-bit basis (on a word basis).
0	—	—	R	Reserved

Note: This count value does not change and the set value is retained even after the DMA becomes active.

(15) ATAPI Control 2 Register (ATAPI_CONTROL2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	LWORD SWAP	WORD SWAP	IFEN
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	—	R	Reserved
2	LWORDSWAP	0	R/W	LWORDSWAP controls the swapping of the upper 32-bit data and the lower 32-bit data on a 2-longword basis (64-bit data) on the SHwy bus. 0: Longword swap is not executed. The 64-bit data on the SHwy bus appears in a big endian format. 1: Longword swap is executed between the ATAPI interface and the SHwy bus interface. Note that word swap is only available on data transfer when ATAPI Control Register[0] = 1: DMA mode start.
1	WORDSWAP	0	R/W	WORDSWAP controls the swapping of the upper 16-bit data and the lower 16-bit data on a longword basis when the 32-bit data bus is enabled in the SHwy bus. 0: Word swap is not executed. 32-bit Data on the SHwy bus appears in a big endian format. 1: Word swap is executed between the ATAPI interface and the SHwy bus interface. Note that word swap is only available on data transfer when ATAPI Control Register[0] = 1: DMA mode start.
0	IFEN	0	R/W	IFEN controls an ATAPI interface enable. 0: ATAPI interface disabled 1: ATAPI interface enabled Note: At the value of 0, ATAPI interface I/O pins function as input, and output pins goes high impedance.

(16) ATAPI Signal Status Register (ATAPI_SIG_ST)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DDMA RDY	DM ARQ
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	—	R	Reserved
1	DDMARDY	—	R	DDMARDY indicates ATAPIDDMARDY (inverted IDEIORDY) signal status.
0	DMARQ	—	R	DMARQ indicates ATAPIDMARQ (IDEDREQ) signal status.

(17) Byte swap Register (ATAPI_BYTE_SWAP)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BYTE SWAP
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	—	R	Reserved
0	BYTESWAP	0	R/W	BYTESWAP controls the swapping of the upper 8 bit data and the lower 8 bit data in the ATAPI interface. 1: Byte swap is executed between the ATAPI interface and SHwy bus. Note that byte swap is only available on data transfer when ATAPI Control Register[0] = 1: DMA mode start.

(18) ATAPI Data Bus Alignment

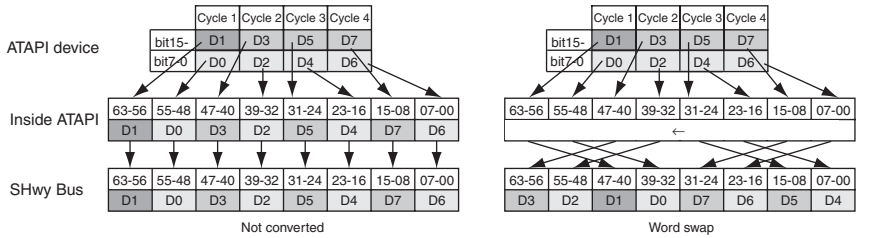
- Data bus alignment on the HPB bus
  There is no difference between big endian and little endian settings.
  Physical bus width is fixed to 32 bits.

Access \ Bus		32-bit bus				16-bit bus				8-bit bus			
		31	16	8	0	31	16	8	0	31	16	8	0
Size Byte	Address 4n 4n+1 4n+2 4n+3	Not specified				Not specified				Not specified			
Word	4n 4n+2	Not specified				Not specified				Not specified			
Longword	4n	B3	B2	B1	B0	Not specified				Not specified			

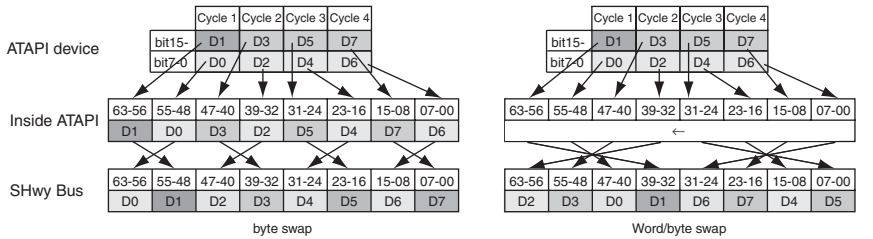
B3: 31 to 24, B2: 23 to 16, B1: 15 to 8, and B0: 7 to 0

- Data bus alignment on the SHwy bus
  Bus width is fixed to 64 bits.

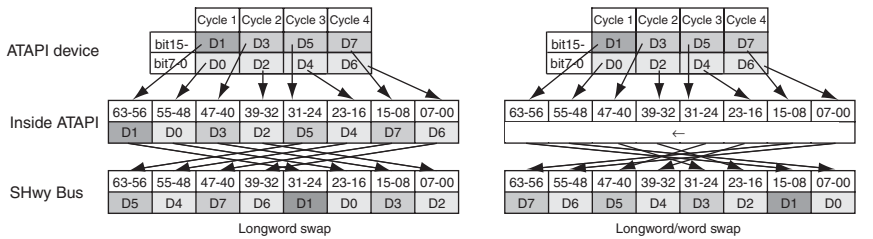
Data direction: ATAPI device → Inside ATAPI → SHwy-bus



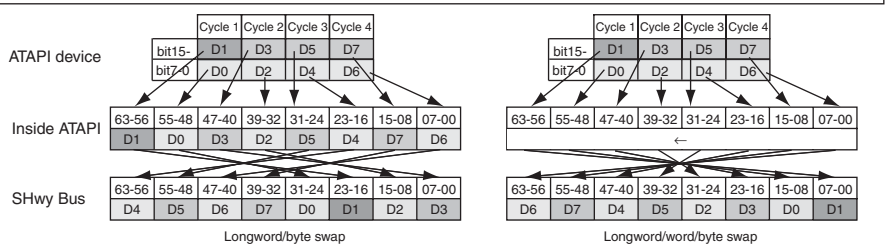
LONGWORDSWAPbit:	←0	←0
WORDSWAPbit:	←0	←1
BYTESWAPbit:	←0	←0



LONGWORDSWAPbit:	←0	←0
WORDSWAPbit:	←0	←1
BYTESWAPbit:	←1	←1



LONGWORDSWAPbit:	←1	←1
WORDSWAPbit:	←0	←1
BYTESWAPbit:	←0	←0



LONGWORDSWAPbit:	←1	←1
WORDSWAPbit:	←0	←1
BYTESWAPbit:	←1	←1

LONGWORDSWAPbit:	ATAPI Control 2 Reg (bit2)
WORDSWAPbit:	ATAPI Control 2 Reg (bit1)
BYTESWAPbit:	Byte Swap Reg (bit0)

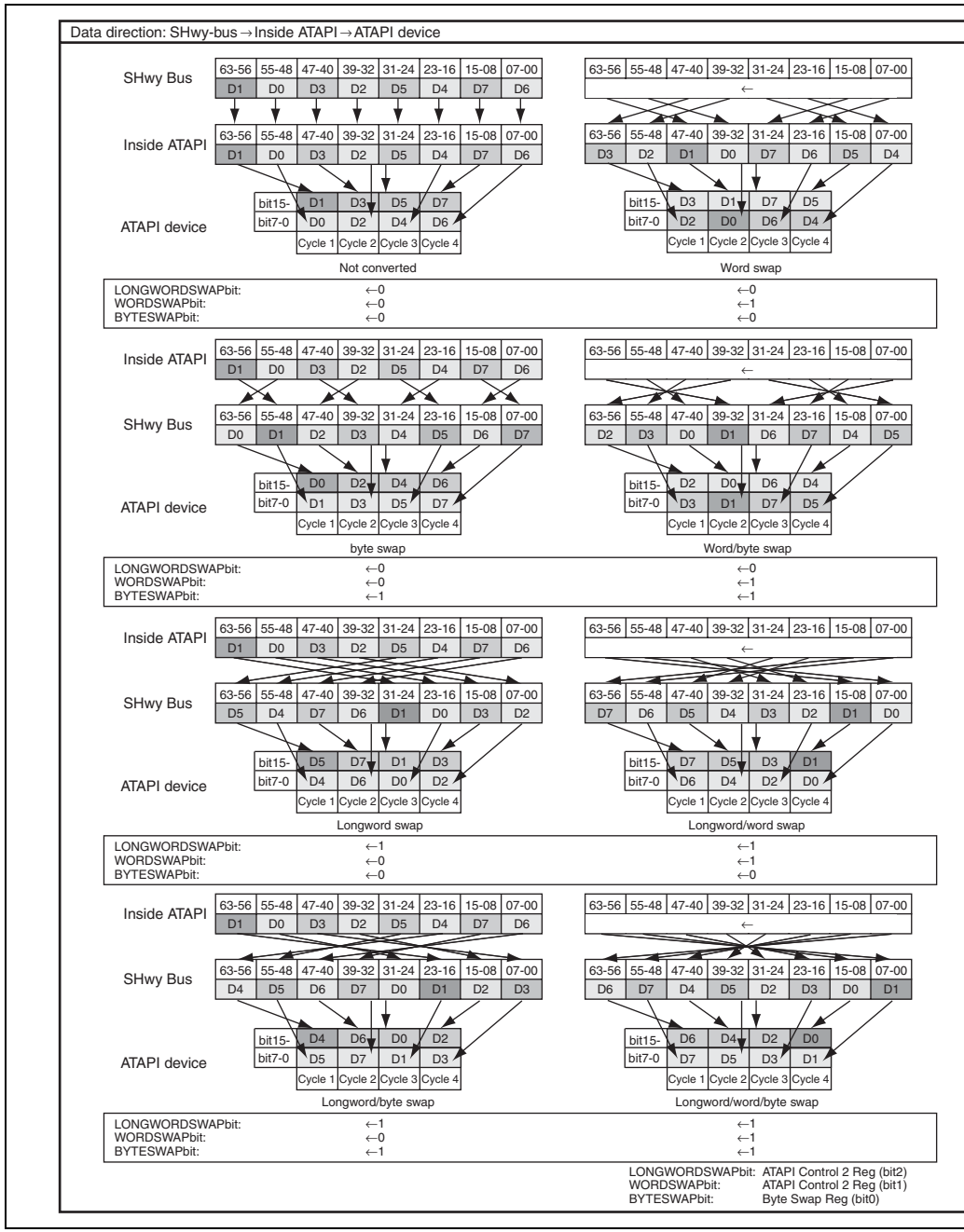


Figure 46.5 ATAPI Data Bus Alignment

46.6 Functional Description

This module supports a primary channel as a host. The master/slave configuration is also supported as defined in the ATAPI interface specification. This module supports the 3.3-V I/O interface.

The ATA task file register and ATAPI packet command task file register are allocated to the SH register map space. Thus, to access these registers by the SH, the registers in the DVDROM drive and others should be addressed using DCS[1:0] and DSA[2:0] pins.

46.6.1 Data Transfer Modes

ATAPI interface control register supports the PIO transfer, multiword DMA transfer, and Ultra DMA transfer mode. It initiates transfer modes and sets a specific ATAPI interface timing which is different in each mode.

PIO modes 0 to 4, multiword DMA modes 0 to 2 and Ultra DMA mode 0 to 4 are supported.

For both multiword DMA and Ultra DMA data transfers, the SHwy bus can be used while the HPB bus can only be used for the PIO transfer.

Table 46.4 Data Transfer Modes

	Data Transfer Mode		
		DMA Data Transfer between ATA Device and SHwy Bus	
Internal Operation and Internal Register	**PIO Data Transfer**	**Multiword DMA**	**Ultra DMA**
FIFO operation	Bypass*	Used	Used
UDMAEN bit in control register	Don't care	0	1
START/STOP bit in control register	Not used	Used	Used

Note: * The CPU accesses the ATA device in PIO mode. For DMA transfer via the SHwy bus, data is transferred between the ATAPI device and the memory.

46.6.2 Descriptor Function

A DMA transfer for which continuous memory areas are specified can be used in this module. Set individual DMA start addresses and DMA transfer counts in the descriptor table.

46.6.3 External Level Shifter Control Function

This module provides the direction signal DIRECTION that is 0 for writing to the device and the active low enable signal EXBUF_ENB for the external level shifter, and operates as shown in figures 46.6 through 46.11 for PIO, multiword DMA, and ultra DMA mode operations.

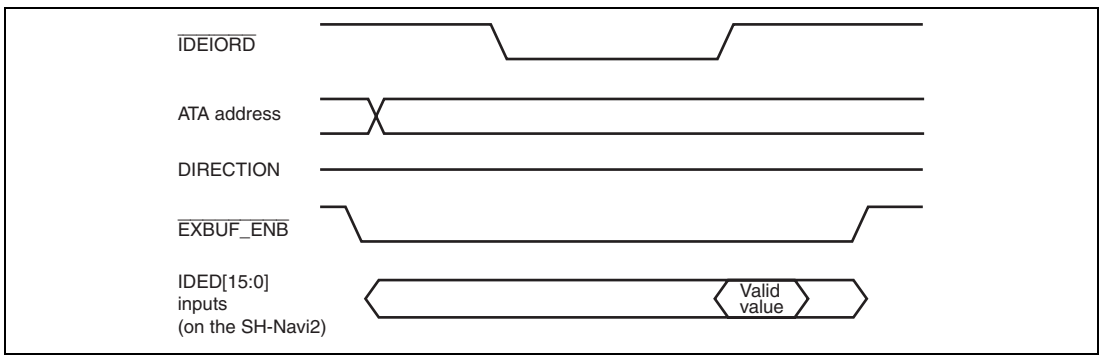


Figure 46.6 PIO Read Data Transfer

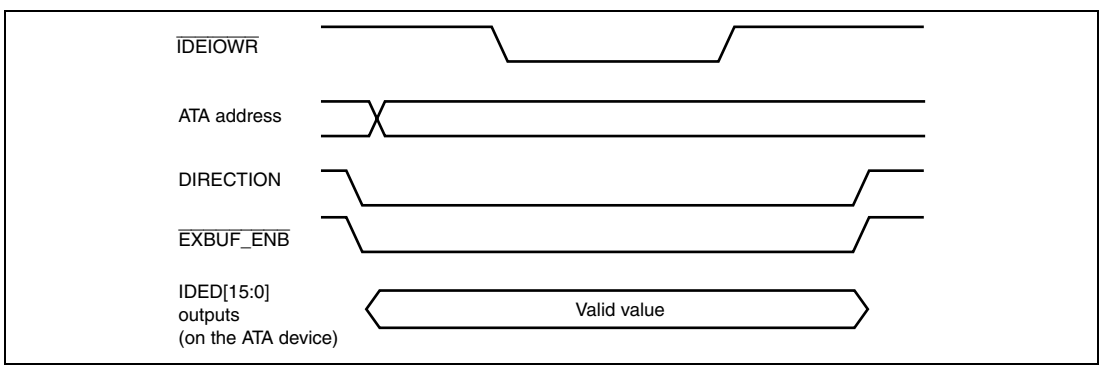


Figure 46.7 PIO Write Data Transfer

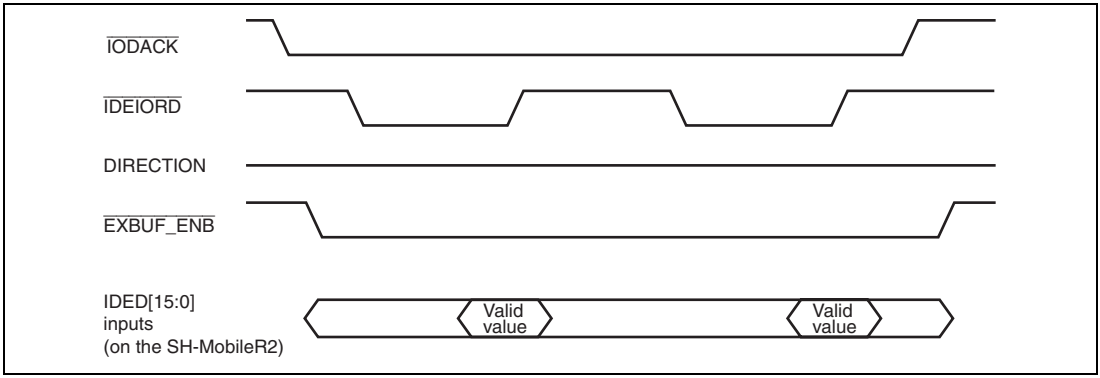


Figure 46.8 Multiword DMA Data-in Burst

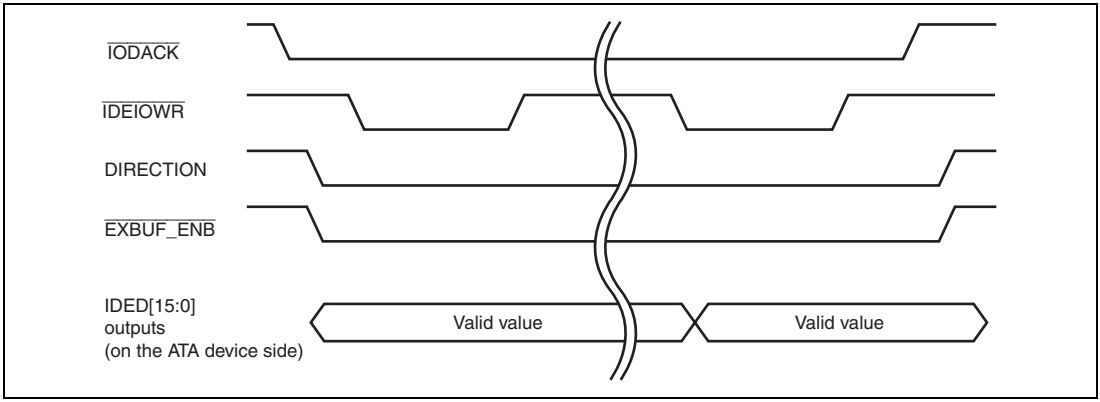


Figure 46.9 Multiword DMA Data-out Burst

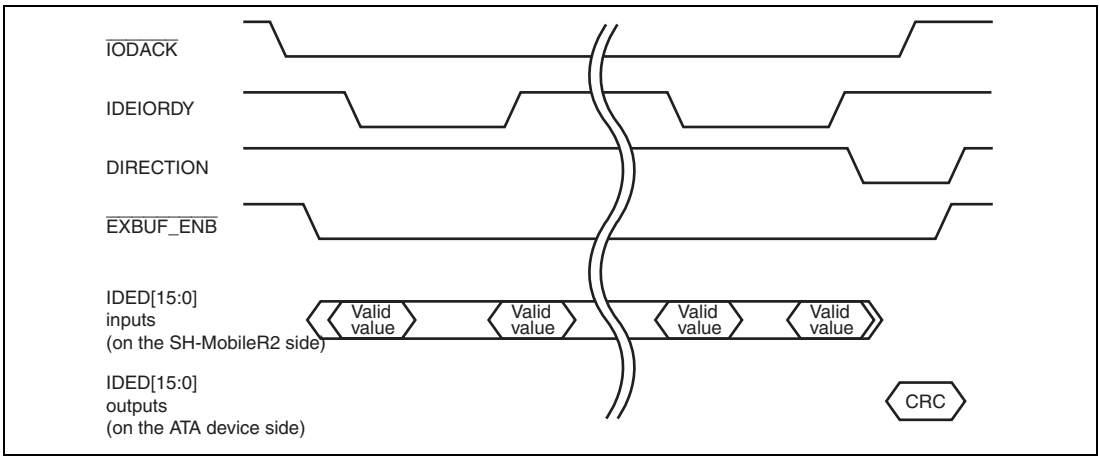


Figure 46.10 Ultra DMA Data-in Burst

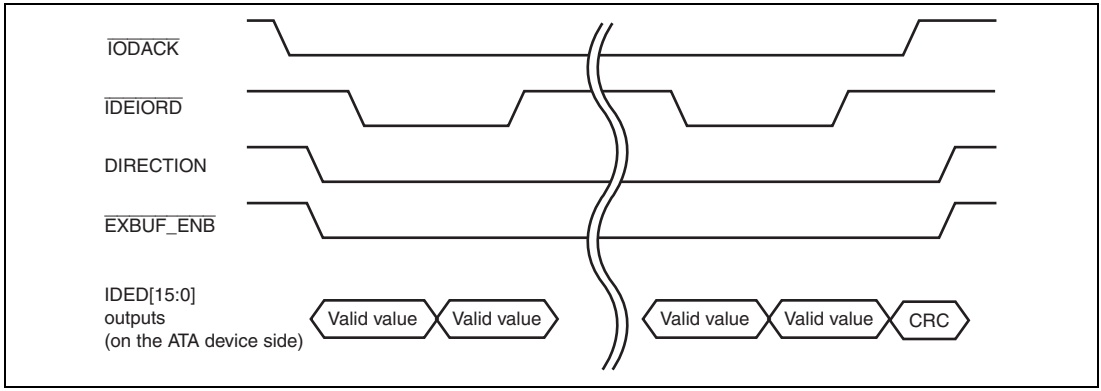


Figure 46.11 Ultra DMA Data-out Burst

46.7 Operating Procedure

46.7.1 Initialization

(1) Setting of Interface Enable Bit

Set the IFEN bit of the ATAPI control 2 register to 1.

(2) Setting of Timing Registers

Write appropriate values to the following registers.

For details, refer to register descriptions.

- PIO timing register
- Multiword DMA timing register
- Ultra DMA timing register

46.7.2 Procedure in PIO Transfer Mode

- Case Not Using FIFO

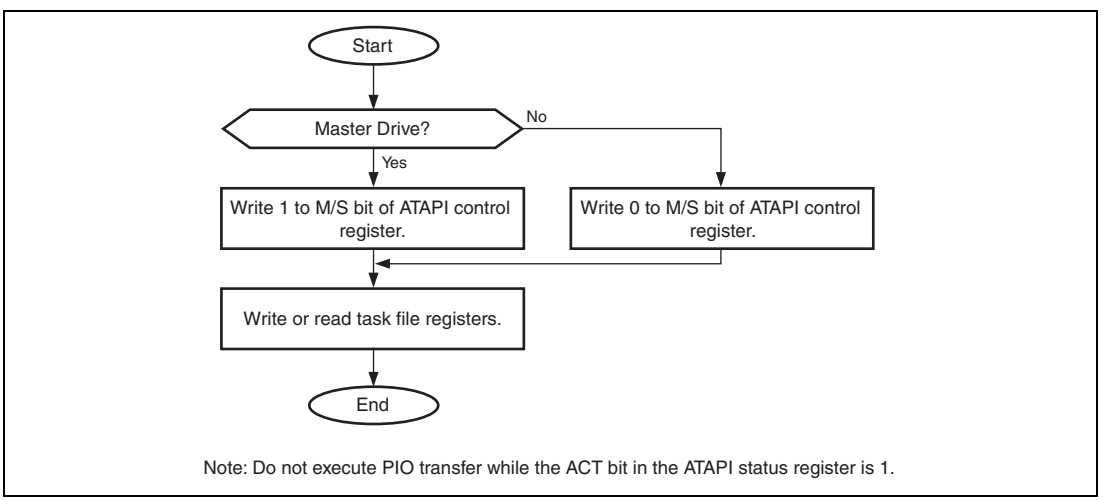


Figure 46.12 Procedure in PIO Transfer Mode

46.7.3 Procedure in Multiword DMA Transfer Mode

- Transfer to and from memory via the SHwy bus by polling

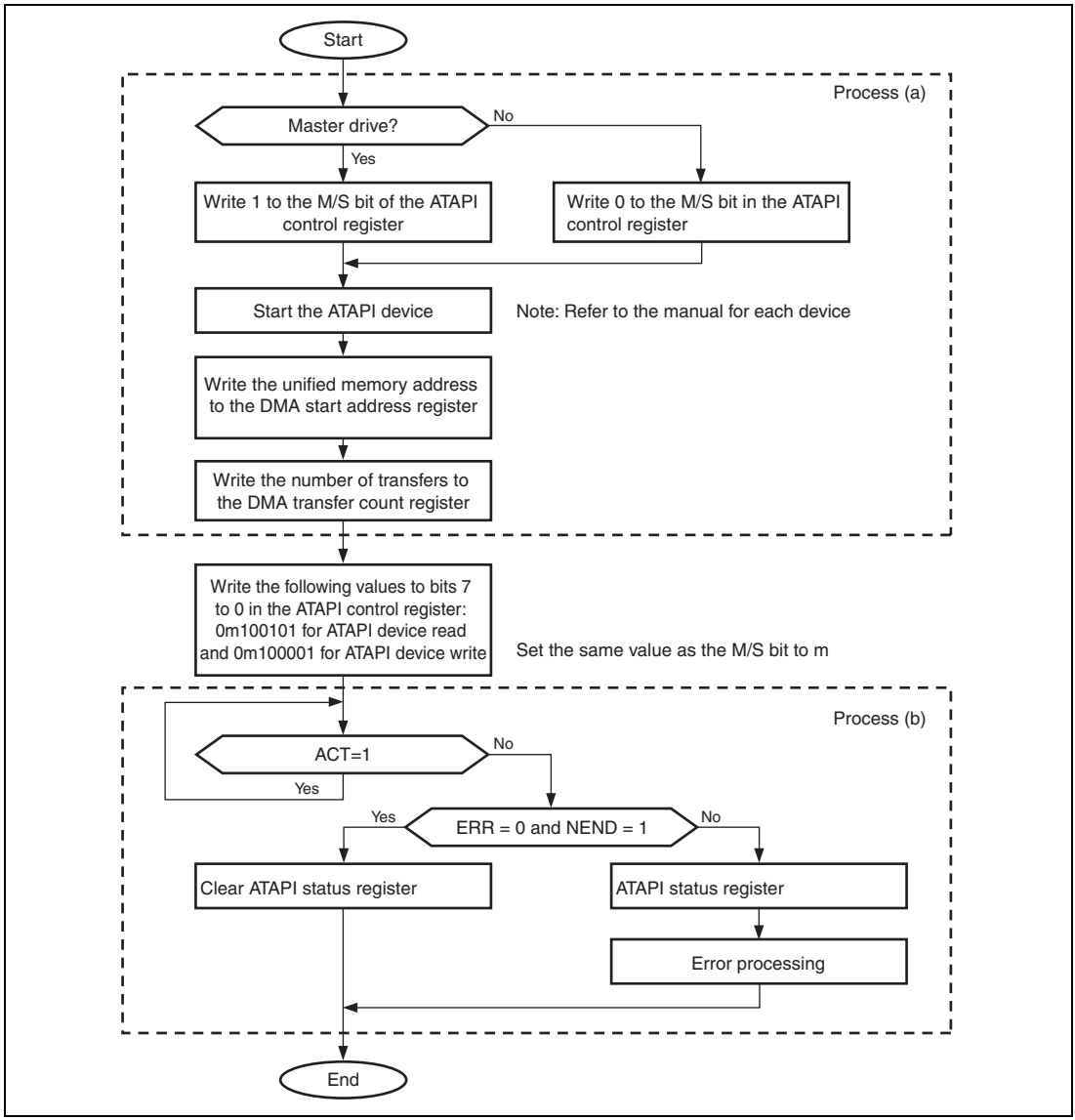


Figure 46.13 Transfer to and from Memory via SHwy Bus by Polling

- Transfer to and from memory via the SHwy bus by interrupt

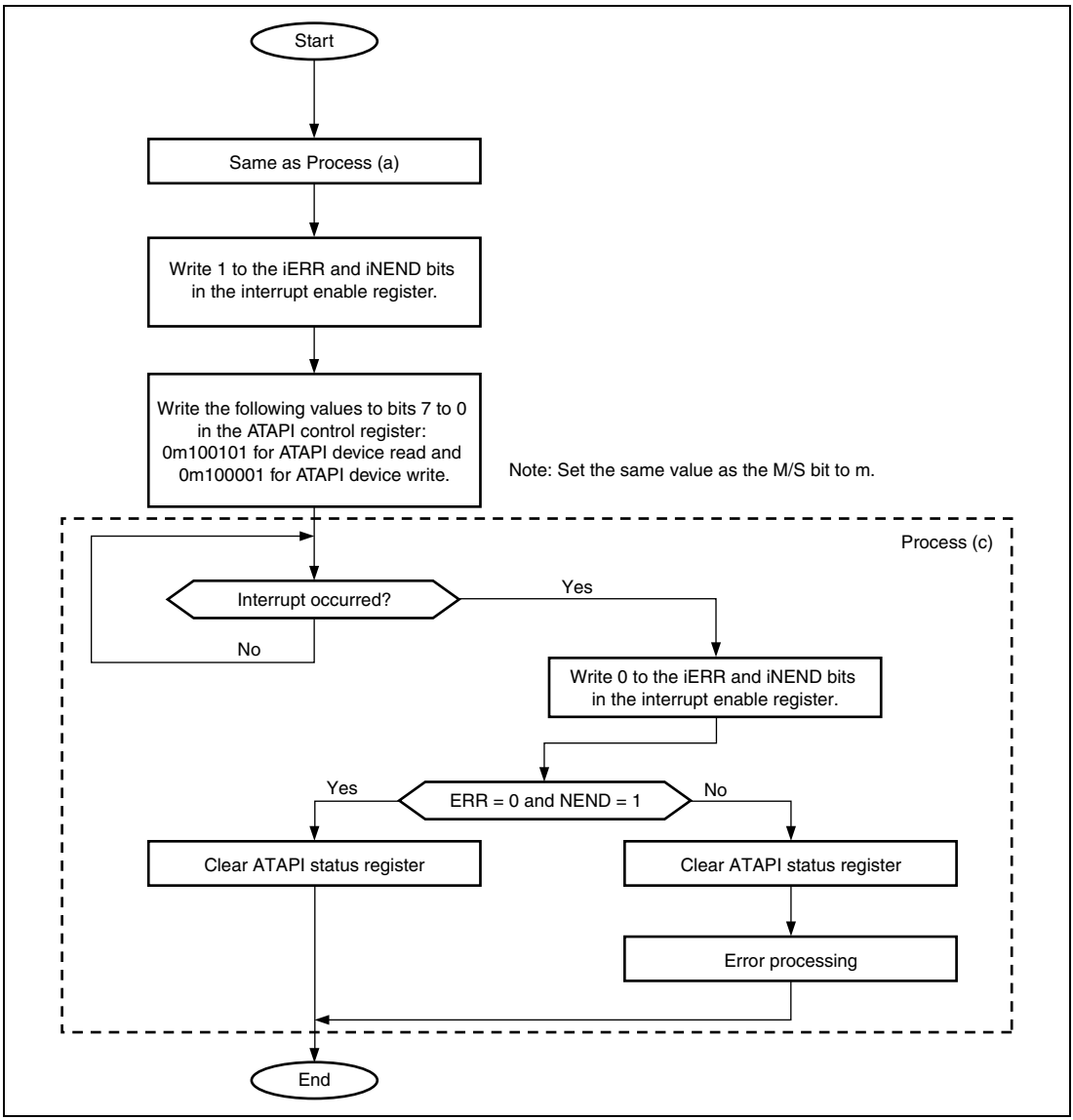


Figure 46.14 Transfer to and from Memory via SHwy Bus by Interrupt

46.7.4 Procedure in Ultra DMA Transfer Mode

- Transfer to and from memory via the SHwy bus by polling

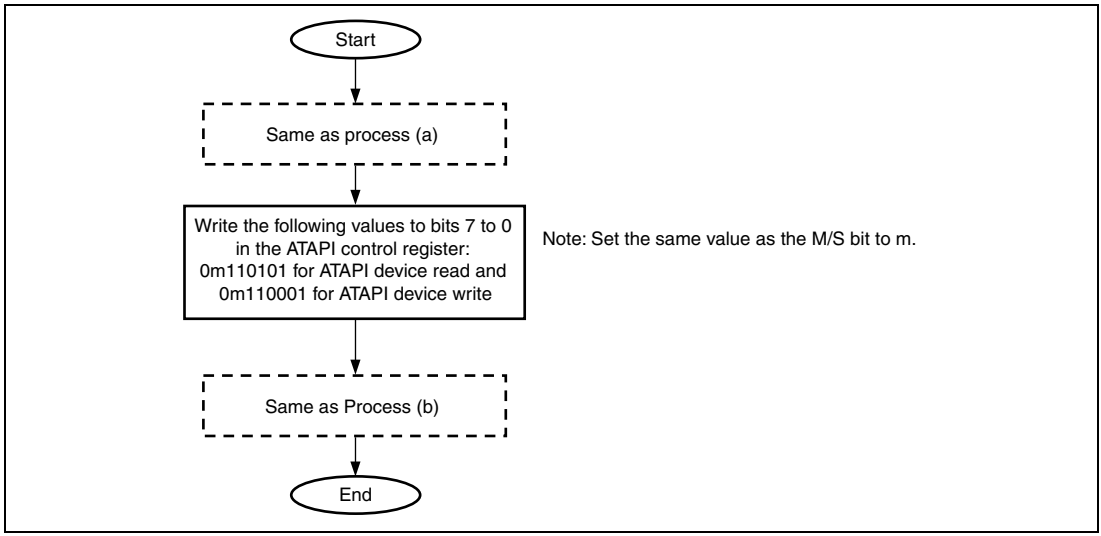


Figure 46.15 Transfer to and from Memory via SHwy Bus by Polling

- Transfer to and from memory via the SHwy bus by interrupt

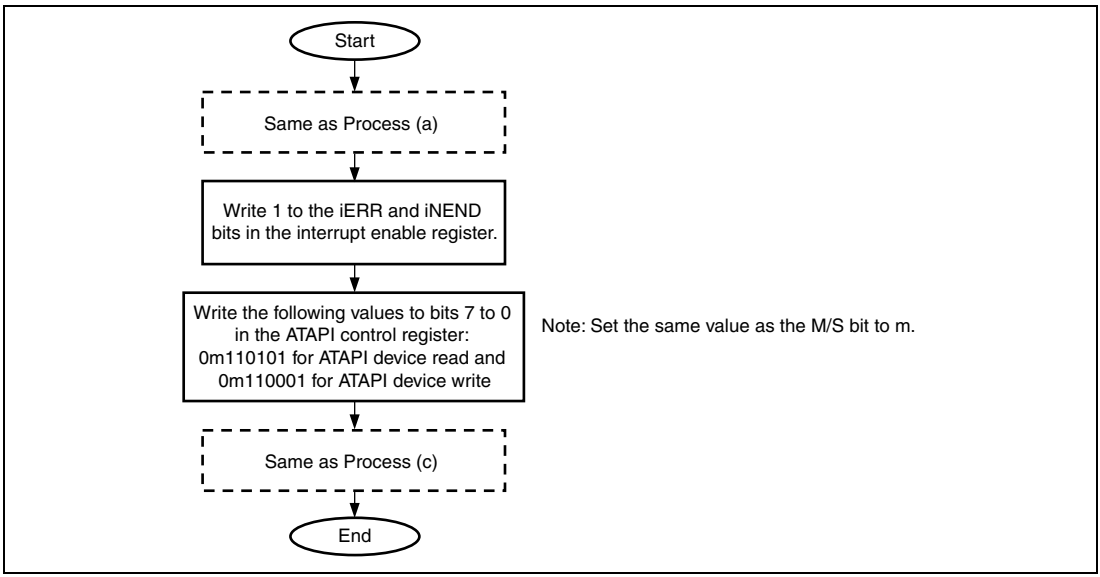


Figure 46.16 Transfer to and from Memory via SHwy Bus by Interrupt

46.7.5 Procedure in Hardware Reset for ATAPI Device

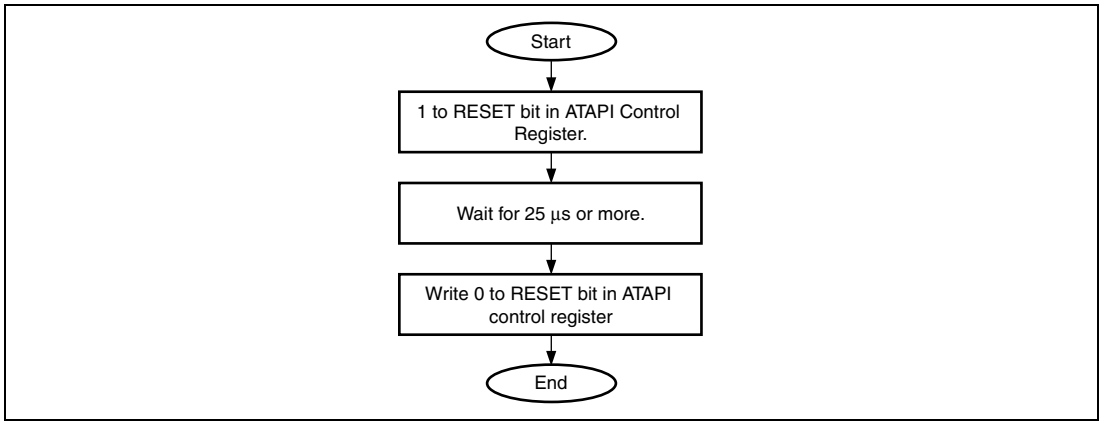


Figure 46.17 Procedure in Hardware Reset for ATAPI Device

Section 47 Ethernet Controller Memory Access Controller (E-MAC)

47.1 Overview

This LSI has an on-chip memory access controller (E-MAC) conforming to the Ethernet or the IEEE802.3 MAC (Media Access Control) layer standard. The E-MAC is configured with the Ethernet controller (EtherC) and the Ethernet Direct Memory Access Controller (E-DMAC). Connecting a physical-layer LSI (PHY-LSI) complying with this standard enables the Ethernet controller (EtherC) to perform transmission and reception of Ethernet/IEEE802.3 frames. And also this LSI has one MAC layer interface port.

This LSI has an on-chip direct memory access controller (E-DMAC) directly connected to the Ethernet controller (EtherC), which enable the high-speed access to the memory with using the E-DMAC.

The E-DMAC controls the most part of the buffer management by using descriptors. This reduces the load on the CPU, thus enabling efficient data transmission and reception.

This LSI can connect to PHY-LSI through RMII (Reduced Media Independent Interface) by MII-RMII conversion circuit.

47.1.1 Features

- The EtherC has the following features:
- Transmission and reception of Ethernet/IEEE802.3 frames
- Supports 10/100 Mbps receive/transfer
- Supports full-duplex and half-duplex modes
- Conforms to IEEE802.3u standard MII (Media Independent Interface)
- Flow control conforming to IEEE802.3x

- The E-DMAC has the following features:
- The load on the CPU is reduced by means of a descriptor management system
- Transmit/receive frame status information is indicated in descriptors
- Achieves efficient system bus utilization through the use of DMA block transfer (32-byte units)
- Supports single-frame/multi-buffer operation
- Improves software performance by padding insertion in receive data.

MII-RMII conversion circuit has the following features:

- Supports RMII Rev.1.2

47.1.2 Logical Configuration

Figure 47.1 shows a block diagram of E-MAC and Table 47.1 lists the functions of each block.

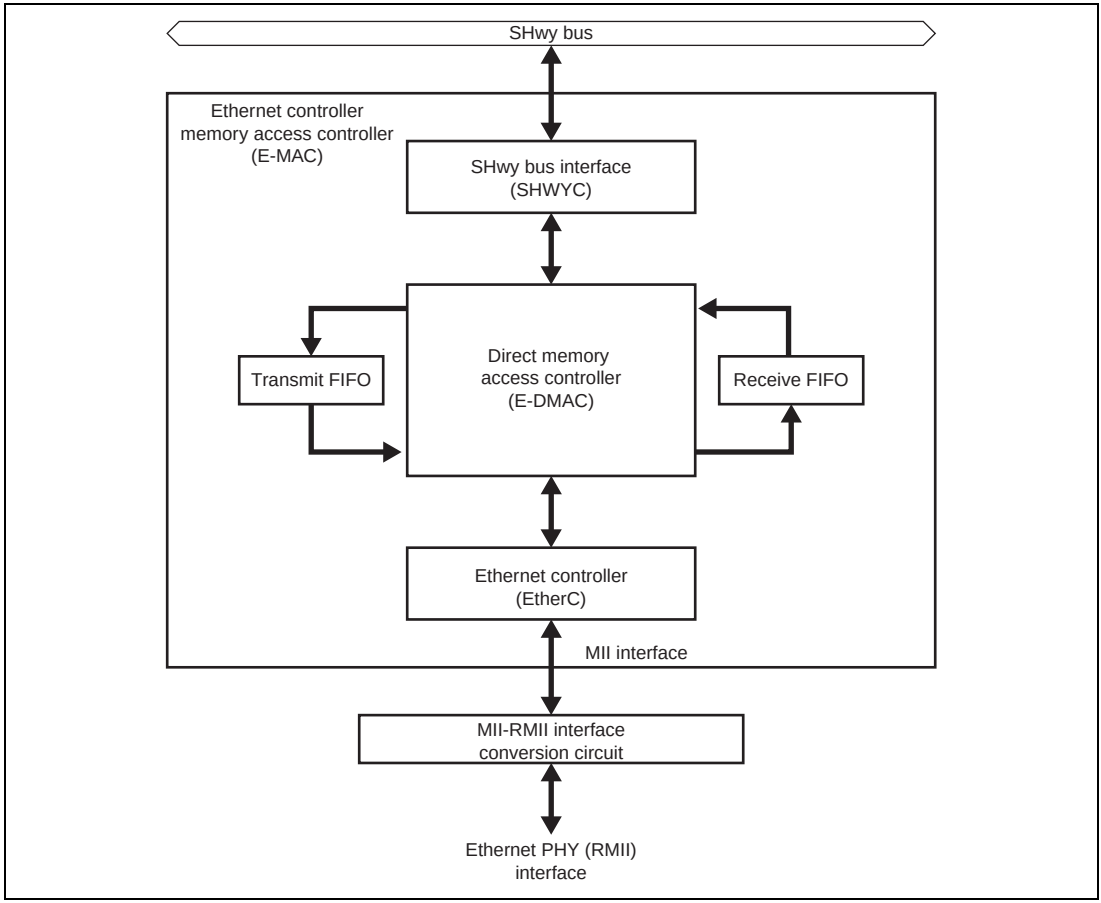


Figure 47.1 Block Diagram of Ethernet MAC Controller

Table 47.1 Functions of Each Block

Block	Function
SHwy Interface Converter (SHWYC)	• Interface converter for DMA-IF of the initiator port and the direct memory access controller.
Direct Memory Access Controller (E-DMAC)	• DMATransfer the transmit or receive data controlled by Ethernet between the transmit/receive buffer on memory and the on-chip FIFO. For details, see section 47.5, E-DMAC Operation.
Ethernet Controller (EtherC)	• Conforming to the IEEE802.3uMAC layer standard. • Transmission and reception of frames • Supports 10/100 Mbps receive/transfer • Conforms to IEEE802.3u standard MII (Media Independent Interface) • Flow control (IEEE802.3x/Back pressure) For details, see section 47.6, EtherC Operation.
Transmit/Receive FIFO (TFIFO/RFIFO)	• Transmit/receive FIFO for the direct memory access controller. Capacity of Transmission FIFO: 2Kbytes Capacity of Reception FIFO: 2Kbytes

47.2 Input/Output Pins

Table 47.2 lists the external pin configuration. Note that the pin names described in this section are abbreviated (without describing Ether_).

Table 47.2 Pin Configuration

Name	Polality	I/O	Function	Number of pins
RMII_REF_CLK	P	I	RMII reference clock	1
RMII_TXD[1:0]	P	O	Transmit data	2
RMII_TX_EN	P	O	Transmit data enable	1
RMII_RX_ER	P	I	Receive error	1
RMII_CRS_DV	P	I	Carrier Sense/Receive Data Valid	1
RMII_RXD[1:0]	P	I	Receive data	2
MDC	P	O	Management data clock	1
MDIO	P	I/O	Management data	1
LNKSTA	P	I	Link signal of PHY output status	1

47.3 Address Map

Table 47.3 lists the address map within the Ethernet MAC controller.

Table 47.3 Address Map

Address (Hex)	Capacity (Byte)	Register
A4600000 to A46000FF	256	E-DMAC register
A4600100 to A46001FF	256	EtherC register
A4600200 to A46002FF	256	(Reserved)
A4600300 to A46003FF	256	(Reserved)
A4600400 to A46004FF	256	(Reserved)
A4600500 to A46005FF	256	(Reserved)
A4600600 to A46006FF	256	(Reserved)
A4600700 to A46007FF	256	(Reserved)
A4600800 to A460FFFF	63744	(Reserved)

47.4 Register Descriptions

47.4.1 Notes on Accessing Registers

When higher-level software access the registers defined in this section, the following limitations and notes should be implemented:

The register bits that are not defined explicitly should be written to 0 in writing and be treated as undefined in reading.

The limitations above should also be applied to the undefined bits of registers for which no restrictions on writing are specified in their properties.

In general, each register is 32-bit long and should be accessed in 32-bit units. Therefore, they do not support partial write and read operations.

47.4.2 Register Descriptions

Table 47.4 shows the configuration of registers of the E-DMAC. Table 47.5 shows the configuration of registers of the EtherC.

Table 47.4 E-DMAC Register Configuration

Register Name	Abbreviation	Function	Address
E-DMAC mode register	EDMR	Specifies the E-DMAC operation mode	H'A460 0000
E-DMAC transmit request register	EDTRR	Calls for transmission	H'A460 0008
E-DMAC receive request register	EDRRR	Calls for reception	H'A460 0010
Transmit descriptor list start address register	TDLAR	Specifies the transmit fetch descriptor address	H'A460 0018
Receive descriptor list start address register	RDLAR	Specifies the receive fetch descriptor address	H'A460 0020
EtherC/E-DMAC status register	EESR	Indicates the E-DMAC status	H'A460 0028
EtherC/E-DMAC status interrupt permission register	EESIPR	Specifies the E-DMAC interrupt mask	H'A460 0030
Transmit/receive status copy enable register	TRSCER	Specifies the E-DMAC error mask	H'A460 0038
Receive missed-frame counter register	RMFCR	Count the missed frame numbers	H'A460 0040

Register Name	Abbreviation	Function	Address
Transmit FIFO threshold register	TFTR	Specifies the transmit FIFO threshold value	H'A460 0048
FIFO depth register	FDR	Specifies the external FIFO capacity	H'A460 0050
Receiving method control register	RMCR	Specifies the resetting method for receiving execution	H'A460 0058
		(Reserved)	H'A460 0060
Transmit FIFO Undercount Counter	TFUCR	Transmit FIFO Underrun Counter	H'A460 0064
Receive FIFO Overflow Counter	RFOCR	Receive FIFO Overflow Counter	H'A460 0068
		(Reserved)	H'A460 006C
Flow control start FIFO threshold setting register	FCFTR	Specifies the receive FIFO sending out busy signal threshold value	H'A460 0070
		(Reserved)	H'A460 0074
		(Reserved)	H'A460 0078
Transmit Interrupt Setting Register	TRIMD	Specifies the transmit interrupt mode	H'A460 007C
		(Reserved)	H'A460 0080
		(Reserved)	H'A460 00C8
		(Reserved)	H'A460 00CC
		(Reserved)	H'A460 00D0
		(Reserved)	H'A460 00D4
		(Reserved)	H'A460 00D8
		(Reserved)	H'A460 00DC
		(Reserved)	H'A460 00E0

Table 47.5 EtherC Register Configuration

Register Name	Abbreviation	Function	Address
EtherC mode register	ECMR	Specifies the EtherC operation mode	H"A460 0100
Receive frame length upper limit register	RFLR	Specifies the frame length check value	H'A460 0108
EtherC status register	ECSR	Indicates the status	H"A460 0110
		(Reserved)	H'A460 0118
PHY interface register	PIR	MII control	H"A460 0120
PHY status register	PSR	Indicates the PHY status	H'A460 0128
Random number generation counter upper limit value	RDMLR	Specifies the random number generation counter upper limit value	H"A460 0140
IPG register	IPGR	Specifies the IPG counter	H'A460 0150
Automatic PAUSE frame register	APR	Specifies the Auto PAUSE parameter	H"A460 0154
Manual PAUSE frame register	MPR	Specifies the Manual PAUSE parameter	H'A460 0158
Receive PAUSE frame counter	RFCF	Receive PAUSE frame counter	H"A460 0160
Automatic PAUSE frame retransmit counter register	TPAUSER	Specifies the PAUSE frame retransmit counter setting	H'A460 0164
PAUSE frame retransmit counter register	TPAUSECR	PAUSE frame transmit counter	H"A460 0168
		(Reserved)	H'A460 016C
MAC address high register	MAHR	MAC address (high)	H"A460 01C0
MAC address low register	MALR	MAC address (low)	H'A460 01C8
Transmit retry over counter register	TROCR	Transmit retry over counter	H"A460 01D0
Delayed collision detect counter register	CDCR	Delayed collision detect counter	H'A460 01D4
Lost carrier counter register	LCCR	Lost carrier counter	H"A460 01D8
Carrier not detect counter register	CNDCR	Carrier not detect counter	H'A460 01DC
CRC error frame receive counter register	CEFCR	CRC error frame receive counter	H'A460 01E4
Frame receive error counter register	FRECR	Frame receive error counter	H'A460 01E8
Too-short frame receive counter register	TSFRCR	Too-short frame receive counter	H"A460 01EC

Register Name	Abbreviation	Function	Address
Too-long frame receive counter register	TLFRCR	Too-long frame receive counter	H'A460 01F0
Residual-bit frame receive counter register	RFCR	Residual-bit frame receive counter	H"A460 01F4
Multicast address frame receive counter register	MAFCR	Multicast address frame receive counter	H'A460 01F8

47.4.3 E-DMAC Register

(1) E-DMAC Mode Register (EDMR)

EDMR is a 32-bit readable/writable register that specifies E-DMAC operating mode. This register should usually be set at initialization after a reset. If the EtherC and E-DMAC are initialized with this register during data transmission, abnormal data may be transmitted on the line. It is prohibited to modify the operating mode while transmission or reception function is enabled. Before modifying the operating mode, the EtherC and E-DMAC should be initialized by setting the software reset bit (SWR). Note that it takes 64 cycles of internal bus clock for the EtherC and E-DMAC to be completely initialized. Therefore, the registers in the EtherC or E-DMAC should be accessed after that.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	DE	DL[1:0]		—	—	—	SWR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	DE	0	R/W	DMA Data Endian Mode 0: Change DMA data endian 1: Not change DMA data endian The setting does not apply to the transmit and receive descriptor or register.
5, 4	DL[1:0]	00	R/W	Transmit/Receive Descriptor Length 00: 16 bytes (Initial value) 01: 32 bytes 10: 64 bytes 11: Setting is prohibited.

Bit	Bit Name	Initial Value	R/W	Description
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SWR	0	R/W	Software Reset [Writing] 0: Disabled 1: E-DMAC and EtherC are reset. TDLAR, RDLAR, RMFCR, TFUCR, and RFOCR are not reset.

(2) E-DMAC Transmit Request Register (EDTRR)

EDTRR is a 32-bit readable/writable register that issues transmit directives to the E-DMAC. After having transmitted one frame, the E-DMAC reads the next descriptor. If the transmit descriptor valid bit in this descriptor is set (valid), the E-DMAC continues transmission. Otherwise, the E-DMAC clears the TR bit and stops the transmit DMAC operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TR	0	R/W	Transmit Request 0: Transmission-halted state. Writing 0 does not stop transmission. Termination of transmission is controlled by the TACT bit of the transmit descriptor. 1: Transmit DMA operation being performed by the E-DMAC. After writing 1 to this bit, the E-DMAC starts reading a transmit descriptor.

(3) E-DMAC Receive Request Register (EDRRR)

EDRRR is a 32-bit readable/writable register that issues receive directives to the E-DMAC. After writing 1 to the RR bit in this register, the E-DMAC reads the receive descriptor. If the RACT bit of this receive descriptor is set to 1 (valid), the E-DMAC starts receive DMA transfer. When DMA transfer based on the first receive descriptor is completed, the E-DMAC reads the next receive descriptor. If the RACT bit of that receive descriptor is set to 1 (valid), the E-DMAC continues receive DMA operation. If the RACT bit of the receive descriptor is cleared to 0 (invalid), the E-DMAC clears the RR bit and stops receive DMAC operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RR	0	R/W	Receive Request 0: Receiving function is disabled After the E-DMAC has completed the reception and write-back to the receive descriptor has been confirmed, disable the receiving function using this register. 1: Receive descriptor is read, and the E-DMAC is ready to receive E-DMAC access the receive descriptor, and the E-DMAC is ready to receive. After the E-DMAC has completed the reception, this bit performs different. It depends on RMCR setting.

(4) Transmit Descriptor List Start Address Register (TDLAR)

TDLAR is a 32-bit readable/writable register that specifies the start address of the transmit descriptor list. Descriptors have a boundary configuration in accordance with the descriptor length indicated by the DL bits in EDMR. This register must not be modified during transmission. Modifications to this register should only be made in the transmission-halted state specified by bits TR[1:0] (= 00) in the E-DMAC transmit request register (EDTRR).

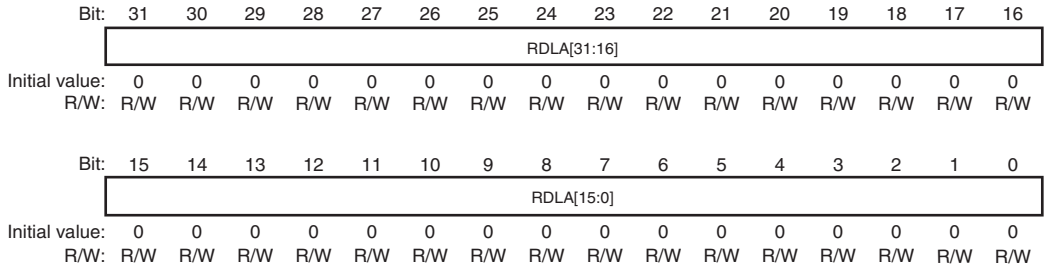
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TDLA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TDLA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDLA[31:0]	All 0	R/W	Transmit Descriptor Start Address The lower bits are set according to the specified descriptor length. 16-byte boundary: TDLA[3:0] = 0000 32-byte boundary: TDLA[4:0] = 00000 64-byte boundary: TDLA[5:0] = 000000 Note: When executed by software reset, the bits are not reset.

(5) Receive Descriptor List Start Address Register (RDLAR)

RDLAR is a 32-bit readable/writable register that specifies the start address of the receive descriptor list. Descriptors have a boundary configuration in accordance with the descriptor length indicated by the DL bits in EDMR. This register must not be modified during reception. Modifications to this register should only be made while reception is disabled by the RR bit (= 0) in the E-DMAC receive request register (EDRRR).



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDLA[31:0]	All 0	R/W	Receive Descriptor Start Address The lower bits are set according to the specified descriptor length. 16-byte boundary: RDLA[3:0] = 0000 32-byte boundary: RDLA[4:0] = 00000 64-byte boundary: RDLA[5:0] = 000000 Note: When executed by software reset, the bits are not reset.

(6) E-MAC/E-DMAC Status Register (EESR)

EESR is a 32-bit readable/writable register that shows communications status information on the E-DMAC in combination with the E-MAC. The information in this register is reported in the form of interrupt sources. Individual bits are cleared by writing 1 (however, bit 22 (ECI) is a read-only bit that is not cleared by writing 1) and are not affected by writing 0. Each interrupt source can also be masked by means of the corresponding bit in the E-MAC/E-DMAC status interrupt permission register (EESIPR).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	TWB	—	—	—	TABT	RABT	RFCOF	DMAER	ECI	TC	TDE	TFUF	FR	RDE	RFOF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CND	DLC	CD	TRO	RMAF	—	—	RRF	RTLF	RTSF	PRE	CERF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	TWB	0	R/W	Write-Back Complete Indicates that write-back from the E-DMAC to the corresponding descriptor after frame transmission has completed. This operation is enabled only when the TIS bit in TRIMD is set to 1. 0: Write-back has not completed, or no transmission directive 1: Write-back has completed
29 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	TABT	0	R/W	Transmit Abort Detect Indicates that the EtherC aborts transmitting a frame because of failures during frame transmission. 0: Frame transmission has not been aborted or no transmission directive 1: Frame transmission has been aborted

Bit	Bit Name	Initial Value	R/W	Description
25	RABT	0	R/W	Receive Abort Detect Indicates that the EtherC aborts receiving a frame because of failures during frame reception. 0: Frame reception has not been aborted or no reception directive 1: Frame reception has been aborted
24	RFCOF	0	R/W	Receive Frame Counter Overflow Indicates that the frame counter in the receive FIFO has overflowed. 0: Receive frame counter has not overflowed 1: Receive frame counter has overflowed
23	DMAER	0	R/W	DMA Transfer Error Indicates that the E-DMAC detects DMA transfer error. 0: DMA transfer error is not detected (normal operation) 1: IDMA transfer error is detected Note: When a DMA transfer error is detected, the E-DMAC halts transmitting/receiving. To resume the operation, execute a software reset with the SWR bit in EDMR.
22	ECI	0	R	EtherC Status Register Source This bit is a read-only bit. When the source of an ECSR interrupt is cleared, this bit is also cleared. 0: EtherC status interrupt source has not been detected 1: EtherC status interrupt source has been detected
21	TC	0	R/W	Frame Transmit Complete Indicates that all the data specified by the transmit descriptor has been transmitted from the EtherC. This bit is set to 1, assuming the completion of transmission, when transmission of one frame is completed in single-frame/single-buffer operation or when the last data of a frame has been transmitted and the transmit descriptor valid bit (TACT) of the next descriptor is not set in for the processing of multi-buffer frame. After frame transmission, the E-DMAC writes the transmission status back to the relevant descriptor. 0: Transfer not complete, or no transfer directive 1: Transfer complete

Bit	Bit Name	Initial Value	R/W	Description
20	TDE	0	R/W	Transmit Descriptor Empty Indicates that the transmit descriptor valid bit (TACT) of a transmit descriptor read by the E-DMAC is not set if the previous descriptor does not represent the end of a frame in multi-buffer frame processing based on single-frame/multi-descriptor operation. As a result, an incomplete frame may be sent. 0: Transmit descriptor active bit TACT = 1 detected 1: Transmit descriptor active bit TACT = 0 detected
19	TFUF	0	R/W	Transmit FIFO Underflow Indicates that an underflow has occurred in the transmit FIFO during frame transmission. Incomplete data is sent onto the line. 0: Underflow has not occurred 1: Underflow has occurred
18	FR	0	R/W	Frame Reception Indicates that a frame has been received and the receive descriptor has been updated. This bit is set to 1 each time a frame is received. 0: Frame has not been received 1: Frame has been received
17	RDE	0	R/W	Receive Descriptor Empty When receive descriptor empty (RDE = 1) occurs, reception can be resumed by setting the RACT bit (cleared to 0) of the receive descriptor to 1 and then restarting the receive operation. 0: Receive descriptor active bit RACT = 1 detected 1: Receive descriptor active bit RACT = 0 detected
16	RFOF	0	R/W	Receive FIFO Overflow Indicates that the receive FIFO has overflowed during frame reception. 0: Overflow has not occurred 1: Overflow has occurred
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
11	CND	0	R/W	Carrier Not Detect Indicates the carrier detection status during preamble transmission. 0: A carrier is detected when transmission starts 1: A carrier is not detected
10	DLC	0	R/W	Detect Loss of Carrier Indicates that loss of the carrier has been detected during frame transmission. 0: Loss of carrier has not been detected 1: Loss of carrier has been detected
9	CD	0	R/W	Delayed Collision Detect Indicates that a delayed collision has been detected during frame transmission. 0: Delayed collision has not been detected 1: Delayed collision has been detected
8	TRO	0	R/W	Transmit Retry Over Indicates that a retry-over condition has occurred during frame transmission. Total 16 transmission retries including 15 retries based on the back-off algorithm have failed after the E-MAC transmission starts. 0: Transmit retry-over condition not detected 1: Transmit retry-over condition detected
7	RMAF	0	R/W	Receive Multicast Address Frame 0: Multicast address frame has not been received 1: Multicast address frame has been received
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RRF	0	R/W	Receive Residual-Bit Frame 0: Residual-bit frame has not been received 1: Residual-bit frame has been received

Bit	Bit Name	Initial Value	R/W	Description
3	RTLF	0	R/W	Receive Too-Long Frame Indicates that a frame whose byte size exceeds the upper limit for the receive frame length set by RFLR in EtherC has been received. 0: Too-long frame has not been received 1: Too-long frame has been received
2	RTSF	0	R/W	Receive Too-Short Frame Indicates that a frame of fewer than 64 bytes has been received. 0: Too-short frame has not been received 1: Too-short frame has been received
1	PRE	0	R/W	PHY-LSI Receive Error 0: PHY-LSI receive error has not been detected 1: PHY-LSI receive error has been detected
0	CERF	0	R/W	CRC Error on Received Frame 0: CRC error has not been detected 1: CRC error has been detected

(7) E-MAC/E-DMAC Status Interrupt Permission Register (EESIPR)

EESIPR is a 32-bit readable/writable register that enables interrupts corresponding to individual bits in the E-MAC/E-DMAC status register (EESR). An interrupt is enabled by writing 1 to the corresponding bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	TWB IP	—	—	—	TABT IP	RABT IP	RFCOF IP	DMAER IP	ECI IP	TC IP	TDE IP	TFUF IP	FR IP	RDE IP	RFOF IP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CND IP	DLC IP	CD IP	TRO IP	RMAF IP	—	—	RRF IP	RTLF IP	RTSF IP	PRE IP	CERF IP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	TWBIP	0	R/W	Write-Back Complete Interrupt Enable 0: Write-back complete interrupt is disabled 1: Write-back complete interrupt is enabled
29 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	TABTIP	0	R/W	Transmit Abort Detect Interrupt Enable 0: Transmit abort detect interrupt is disabled 1: Transmit abort detect interrupt is enabled
25	RABTIP	0	R/W	Receive Abort Detect Interrupt Enable 0: Receive abort detect interrupt is disabled 1: Receive abort detect interrupt is enabled
24	RFCOFIP	0	R/W	Receive Frame Counter Overflow Interrupt Enable 0: Receive frame counter overflow interrupt is disabled 1: Receive frame counter overflow interrupt is enabled
23	DMAERIP	0	R/W	DMA Transfer Error Interrupt Enable 0: DMA transfer error interrupt is disabled 1: DMA transfer error interrupt is enabled

Bit	Bit Name	Initial Value	R/W	Description
22	ECIIP	0	R/W	EtherC Status Register Source Interrupt Enable 0: EtherC status interrupt is disabled 1: EtherC status interrupt is enabled
21	TCIP	0	R/W	Frame Transmission Complete Interrupt Enable 0: Frame transmission complete interrupt is disabled 1: Frame transmission complete interrupt is enabled
20	TDEIP	0	R/W	Transmit Descriptor Empty Interrupt Enable 0: Transmit descriptor empty interrupt is disabled 1: Transmit descriptor empty interrupt is enabled
19	TFUFIP	0	R/W	Transmit FIFO Underflow Interrupt Enable 0: Underflow interrupt is disabled 1: Underflow interrupt is enabled
18	FRIP	0	R/W	Frame Reception Interrupt Enable 0: Frame reception interrupt is disabled 1: Frame reception interrupt is enabled
17	RDEIP	0	R/W	Receive Descriptor Empty Interrupt Enable 0: Receive descriptor empty interrupt is disabled 1: Receive descriptor empty interrupt is enabled
16	RFOFIP	0	R/W	Receive FIFO Overflow Interrupt Enable 0: Overflow interrupt is disabled 1: Overflow interrupt is enabled
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	CNDIP	0	R/W	Carrier Not Detect Interrupt Enable 0: Carrier not detect interrupt is disabled 1: Carrier not detect interrupt is enabled
10	DLCIP	0	R/W	Detect Loss of Carrier Interrupt Enable 0: Detect loss of carrier interrupt is disabled 1: Detect loss of carrier interrupt is enabled
9	CDIP	0	R/W	Delayed Collision Detect Interrupt Enable 0: Delayed collision detect interrupt is disabled 1: Delayed collision detect interrupt is enabled

Bit	Bit Name	Initial Value	R/W	Description
8	TROIP	0	R/W	Transmit Retry Over Interrupt Enable 0: Transmit retry over interrupt is disabled 1: Transmit retry over interrupt is enabled
7	RMAFIP	0	R/W	Receive Multicast Address Frame Interrupt Enable 0: Receive multicast address frame interrupt is disabled 1: Receive multicast address frame interrupt is enabled
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RRFIP	0	R/W	Receive Residual-Bit Frame Interrupt Enable 0: Receive residual-bit frame interrupt is disabled 1: Receive residual-bit frame interrupt is enabled
3	RTLFIP	0	R/W	Receive Too-Long Frame Interrupt Enable 0: Receive too-long frame interrupt is disabled 1: Receive too-long frame interrupt is enabled
2	RTSFIP	0	R/W	Receive Too-Short Frame Interrupt Enable 0: Receive too-short frame interrupt is disabled 1: Receive too-short frame interrupt is enabled
1	PREIP	0	R/W	PHY-LSI Receive Error Interrupt Enable 0: PHY-LSI receive error interrupt is disabled 1: PHY-LSI receive error interrupt is enabled
0	CERFIP	0	R/W	CRC Error on Received Frame Interrupt Enable 0: CRC error interrupt is disabled 1: CRC error interrupt is enabled

(8) Transmit/Receive Status Copy Enable Register (TRSCER)

TRSCER specifies whether the information for the transmit and receive state reported by bits in the E-MAC/E-DMAC status register (EESR) is to be reflected in the TFS25 to TFS0 or RFS26 to RFS0 bits of the corresponding descriptor. The bits in this register correspond to bits 11 to 0 in EESR. When a bit is cleared to 0, the transmit status (bits 11 to 8 in EESR) is reflected in the TFS3 to TFS0 bits of the transmit descriptor, and the receive status (bits 7 to 0 in EESR) is reflected in the RFS7 to RFS0 bits of the receive descriptor. When a bit is set to 1, the occurrence of the corresponding source is not reflected in the descriptor. After this LSI is reset, all bits are cleared to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

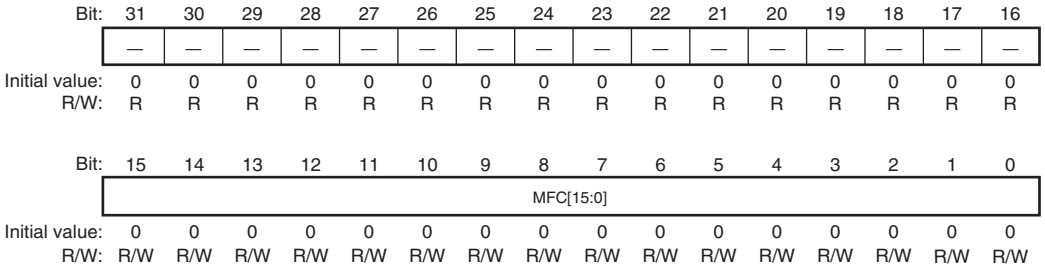
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CND CE	DLC CE	CD CE	TRO CE	RMAF CE	—	—	RRF CE	RTLF CE	RTSF CE	PRE CE	CERF CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	CNDCE	0	R/W	CND Bit Copy Directive 0: Reflects the CND bit status in the TFS bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFS bit of the transmit descriptor
10	DLCCE	0	R/W	DLC Bit Copy Directive 0: Reflects the DLC bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
9	CDCE	0	R/W	CD Bit Copy Directive 0: Reflects the CD bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor

Bit	Bit Name	Initial Value	R/W	Description
8	TROCE	0	R/W	TRO Bit Copy Directive 0: Reflects the TRO bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
7	RMAFCE	0	R/W	RMAF Bit Copy Directive 0: Reflects the RMAF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RRFCE	0	R/W	RRF Bit Copy Directive 0: Reflects the RRF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
3	RTLFCE	0	R/W	RTLF Bit Copy Directive 0: Reflects the RTLF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
2	RTSFCE	0	R/W	RTSF Bit Copy Directive 0: Reflects the RTSF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
1	PRECE	0	R/W	PRE Bit Copy Directive 0: Reflects the PRE bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
0	CERFCE	0	R/W	CERF Bit Copy Directive 0: Reflects the CERF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor

(9) Receive Missed-Frame Counter Register (RMFCR)

RMFCR is a 16-bit counter that indicates the number of frames that could not be saved in the receive buffer and so were discarded during reception. When the receive FIFO overflows, the receive frames in the FIFO are discarded. The number of frames discarded at this time is counted. When the value in this register reaches H'FFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MFC [15:0]	All 0	R/W	Missed-Frame Counter These bits indicate the number of frames that are discarded and not transferred to the receive buffer during reception. Note: When executed by software reset, the bits are not reset.

(10) Transmit FIFO Threshold Register (TFTR)

TFTR is a 32-bit readable/writable register that specifies the transmit FIFO threshold at which the first transmission is started. The actual threshold is 4 times the set value. The EtherC starts transmission when the amount of data in the transmit FIFO exceeds the number of bytes specified by this register, when the transmit FIFO is full, or when one frame of data write is performed. When setting this register, do so in the transmission-halt state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TFT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	TFT[10:0]	All 0	R/W	Transmit FIFO Threshold A value and smaller than the FIFO size specified by FDR must be set as the transmit FIFO threshold. H'000: Store and forward modes H'001 to H'00C: Setting prohibited H'00D: 52 bytes H'00E: 56 bytes : : H'01F: 124 bytes H'020: 128 bytes : : H'03F: 252 bytes H'040: 256 bytes : : H'07F: 508 bytes H'080: 512 bytes : : H'0FF: 1,020 bytes H'100: 1,024 bytes : : H'1FF: 2,044 bytes H'200: 2,048 bytes H'201 to H'7FF: Setting prohibited

Notes: 1. When starting transmission before one frame of data write has completed, take care no underflow occurs.

2. Operation cannot be guaranteed when the value set in this register is greater than the transmit FIFO size.

(11) FIFO Depth Register (FDR)

FDR is a 32-bit readable/writable register that specifies the sizes of the transmit and receive FIFOs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	TFD[4:0]					—	—	—	RFD[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	TFD[4:0]	B'00000	R/W	Transmit FIFO Size Specifies the size of the transmit FIFO. The setting must not be changed after transmission/reception has started. 00000: 256 bytes 00001: 512 bytes 00010: 768 bytes 00011: 1024 bytes 00100: 1280 bytes 00101: 1536 bytes 00110: 1792 bytes 00111: 2048 bytes Other than above: Setting prohibited Note: When EDTRR.TR= 1, write to these bits are invalid.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	RFD[4:0]	B'00000	R/W	Receive FIFO Size Specifies the size of the receive FIFO. The setting must not be changed after transmission/reception has started. 00000: 256 bytes 00001: 512 bytes 00010: 768 bytes 00011: 1024 bytes 00100: 1280 bytes 00101: 1536 bytes 00110: 1792 bytes 00111: 2048 bytes Other than above: Setting prohibited Note: When EDTRR.TR= 1, write to these bits are invalid.

Note: Operation cannot be guaranteed when the value set in this register is greater than the transmit FIFO size.

(12) Receiving Method Control Register (RMCR)

RMCR is a 32-bit readable/writable register that specifies the control method for the RE bit in ECMR while a frame is received. This register must be set during the receiving-halted state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RNC	RNR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	RNC	0	R/W	Receive Start Bit Non-Reset Mode 0: nop 1: Allows the software to reset the receive start bit (RR) in EDRRR. In this case, even when the RACT bit in the fetched descriptor is 0 (receive descriptor empty), the receive start bit (RR) in EDRRR is not automatically reset and the receive descriptor is continuously fetched to continue DMA transfers of the receive frames.

Bit	Bit Name	Initial Value	R/W	Description
0	RNR	0	R/W	Receive Start Bit Reset 0: Allows the hardware to reset the receive start bit (RR) in EDRRR automatically upon completion of reception of one frame. Control is possible for each frame. To receive the subsequent receive frame, the receive start bit in EDRRR needs to be set again. 1: Allows the higher-level software to control the receive start bit (RR) in EDRRR. Once the receive start bit (RR) in EDRRR is set to 1, the hardware continues to fetch the receive descriptor and receive frames automatically until the RR bit in EDRRR is cleared to 0. In other words, continuous reception of multiple frames are possible. It is recommended to set this bit to 1 when continuous reception is used. However, when a receive descriptor empty is detected, the hardware clears the RR bit in EDRRR automatically.

(13) Transmit FIFO Underrun Counter (TFUCR)

TFUCR is a register that indicates the count of underruns having occurred in the transmit FIFO. The count value is cleared to 0 by writing any value to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UNDER[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	UNDER [15:0]	All 0	R/W	Transmit FIFO Underflow Count Indicates the count of underflows having occurred in the transmit FIFO. The counter stops when the count value reaches H'FFFF.

(14) Receive FIFO Overflow Counter (RFOCR)

RFOCR is a register that indicates the count of overflows having occurred in the receive FIFO. The count value is cleared to 0 by writing any value to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVER[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	OVER[15:0]	All 0	R/W	Receive FIFO Overflow Count Indicates the count of overflows having occurred in the receive FIFO. The counter stops when the count value reaches H'FFFF.

(15) Flow Control Start FIFO Threshold Setting Register (FCFTR)

FCFTR is a 32-bit readable/writable register that sets the flow control of the EtherC (sets the threshold of automatic PAUSE output). The threshold can be set in terms of the data size in the receive FIFO (RFDO[2:0]) and the number of receive frames (RFFO[2:0]). Flow control is turned on when either of the data size in the receive FIFO or the number of receive frames is determined as the threshold value.

If the same receive FIFO size as set by the FIFO depth register (FDR) is set when flow control is to be turned on according to the RFDO setting condition, flow control is turned on with (FIFO data size − 64) bytes. For instance, when the RFD bits in FDR = 1 and the RFDO bits in this register = 1, flow control is turned on when (2,048 − 64) bytes of data is stored in the receive FIFO. The value set in the RFDO bits in this register should be equal to or less than the value set in the RFD bits in FDR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFFO[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RFDO[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	RFFO[2:0]	111	R/W	Receive Frame Count Overflow BSY Output Threshold 000: When two receive frames have been stored in the receive FIFO. 001: When four receive frames have been stored in the receive FIFO. 010: When six receive frames have been stored in the receive FIFO. : 110: When 14 receive frames have been stored in the receive FIFO. 111: When 16 receive frames have been stored in the receive FIFO.

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	RFDO[2:0]	111	R/W	Receive FIFO Overflow BSY Output Threshold 000: When (256 − 32) bytes of data is stored in the receive FIFO. 001: When (512 − 32) bytes of data is stored in the receive FIFO. : 110: When (1792 − 32) bytes of data is stored in the receive FIFO. 111: When (2048 − 32) bytes of data is stored in the receive FIFO Note: When set the same value as FDR.RFD, output busy with (FIFO data size − 64) bytes.

(16) Transmit Interrupt Setting Register (TRIMD)

TRIMD is a 32-bit readable/writable register that specifies whether to notify write-back completion of each frame during transmission by means of the TWB bit in EESR or the interrupt

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TIM	—	—	—	TIS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TIM	0	R/W	Transmit Interrupt Mode 0: Per-transmit-frame mode An interrupt is notified upon write-back completion of each frame. 1: Interrupt mode An interrupt is notified upon write-back completion of the transmit descriptor with the TWBI bit set to 1.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TIS	0	R/W	Transmit Interrupt Setting 0: An interrupt is not notified in the mode selected by the TIM bit. When this bit is 0, the TIM bit setting is invalid. 1: An interrupt is notified by setting the TWB bit in EESR to 1 in the mode selected by the TIM bit.

47.4.4 EtherC Register

(1) EtherC Mode Register (ECMR)

ECMR is a 32-bit readable/writable register that specifies the operating mode of the EtherC. The settings in this register are normally made in the initialization process following a reset.

The operating mode setting must not be changed while the transmitting and receiving functions are enabled. To switch the operating mode, return the EtherC and E-DMAC to their initial states by means of the SWR bit in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	TPC	ZPF	PFR	RXF	TXF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PRCEF	—	—	—	—	—	RE	TE	RTM	ILB	—	DM	PRM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	TPC	0	R/W	PAUSE Frame Transmission 0: PAUSE frame is not transmitted in a PAUSE period 1: PAUSE frame is transmitted even in a PAUSE period

Bit	Bit Name	Initial Value	R/W	Description
19	ZPF	0	R/W	PAUSE Frame Usage with TIME = 0 Enable 0: Control of a PAUSE frame whose TIME parameter value is 0 is disabled. The next frame is not transmitted until the time specified by the Timer value has elapsed. If a PAUSE frame whose time specified by the Timer value is 0 is received, that PAUSE frame is discarded. 1: Control of a PAUSE frame whose TIME parameter value is 0 is enabled. When the data size in the receive FIFO becomes smaller than the FCFTR setting before the time specified by the Timer value elapses, an automatic PAUSE frame with a Timer value of 0 is transmitted. On receiving a PAUSE frame with a Timer value of 0, the transmission wait state is canceled.
18	PFR	0	R/W	PAUSE Frame Receive Mode 0: PAUSE frame is not transferred to E-DMAC 1: PAUSE frame is transferred to E-DMAC
17	RXF	0	R/W	Operating Mode for Receiving Port Flow Control 0: PAUSE frame detection is disabled 1: Flow control for the receiving port is enabled
16	TXF	0	R/W	Operating Mode for Transmitting Port Flow Control 0: Flow control for the transmitting port is disabled (Automatic PAUSE frame is not transmitted) 1: Flow control for the transmitting port is enabled (Automatic PAUSE frame is transmitted as required)
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	PRCEF	0	R/W	CRC Error Frame Reception Enable 0: A frame with a CRC error is received as a frame with an error 1: A frame with a CRC error is received as a frame without an error
11 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
6	RE	0	R/W	Reception EnableIf a switch is made from receiving function enabled (RE = 1) to disabled (RE = 0) while a frame is being received, the receiving function will be enabled until reception of the corresponding frame is completed. 0: Receiving function is disabled 1: Receiving function is enabled
5	TE	0	R/W	Transmission EnableIf a switch is made from transmitting function enabled (TE = 1) to disabled (TE = 0) while a frame is being transmitted, the transmitting function will be enabled until transmission of the corresponding frame is completed. 0: Transmitting function is disabled 1: Transmitting function is enabled
4	RTM	0	R/W	
3	ILB	0	R/W	Internal Loop Back Mode Specifies loopback mode in the EtherC. 0: Normal data transmission/reception is performed 1: Data loopback is performed inside the MAC in the EtherC when DM = 1
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	DM	0	R/W	Duplex ModeSpecifies the EtherC transfer method. 0: Half-duplex transfer is specified Setting EtherC internal back mode (ILB= 1) is prohibited. 1: Full-duplex transfer is specified
0	PRM	0	R/W	Promiscuous ModeSetting this bit enables all Ethernet frames to be received. All Ethernet frames means all receivable frames, irrespective of differences or enabled/disabled status (destination address, broadcast address, multicast bit, etc.). 0: EtherC performs normal operation 1: EtherC performs promiscuous mode operation When MPDE bit is set, EtherC performs nomal operation regardless of this bit setting.

(2) Receive Frame Length Register (RFLR)

RFLR is a 32-bit readable/writable register that specifies the maximum frame length (in bytes) that can be received by this LSI. The settings in this register must not be changed while the receiving function is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RFL[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	RFL[11:0]	All 0	R/W	Receive Frame Length The frame data described here refers to all fields from the destination address up to the CRC data. Frame contents from the destination address up to the data are actually transferred to memory. CRC data is not included in the transfer. When data that exceeds the specified value is received, the part of data that exceeds the specified value is discarded. H'000 to H'5EE: 1,518 bytes H'5EF: 1,519 bytes H'5F0: 1,520 bytes : : H'7FF: 2,047 bytes H'800 to H'FFF: 2048 bytes

(3) EtherC Status Register (ECSR)

ECSR is a 32-bit readable/writable register that indicates the status in the EtherC. This status can be notified to the CPU by interrupts. When 1 is written to the PFROI, LCHNG, MPD, and ICD bits, the corresponding flags can be cleared. Writing 0 does not affect the flag. For bits that generate interrupts, the interrupt can be enabled or disabled by the corresponding bit in ECSIPR.

The interrupts generated due to this status register are indicated in ECI bit in EESR of the E-DMAC. Every status factor generates the interrupts to INTC2.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PSRTO	—	LCHNG	—	ICD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PSRTO	0	R/W	PAUSE Frame Retransmit Retry Over Indicates whether the retransmit count for retransmitting a PAUSE frame when flow control is enabled has exceeded the retransmit upper-limit set in the automatic PAUSE frame retransmit count register (TPAUSER). 0: PAUSE frame retransmit count has not exceeded the upper limit 1: PAUSE frame retransmit count has exceeded the upper limit
3	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2	LCHNG	0	R/W	Link Signal Change Indicates that the LNKSTA signal input from the PHY-LSI has changed from high to low or low to high. To check the current Link state, refer to the LMON bit in the PHY status register (PSR). 0: Change in the LNKSTA signal has not been detected 1: Change in the LNKSTA signal has been detected (high to low or low to high)
1	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ICD	0	R/W	Illegal Carrier Detection Indicates that the PHY-LSI has detected an illegal carrier on the line. More specifically, this bit is set when the signals transmitted from the PHY-LSI to this LSI through the RX-DV, RX-ER and MII-RXD3 to 0 pins are 0,1, and 1110. If a change in the signal input from the PHY-LSI occurs in a period shorter than the software recognition period, the correct information may not be obtained. Refer to the timing specification for the PHY-LSI used. 0: PHY-LSI has not detected an illegal carrier on the line 1: PHY-LSI has detected an illegal carrier on the line

(4) EtherC Interrupt Permission Register (ECSIPR)

ECSIPR is a 32-bit readable/writable register that enables or disables the interrupt sources indicated by ECSR. Each bit can disable or enable interrupts corresponding to the bits in ECSR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PSRTOIP	—	LCHNGIP	—	ICDIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PSRTOIP	0	R/W	PAUSE Frame Retransmit Retry Over Interrupt Enable 0: Interrupt notification by the PSRTO bit is disabled 1: Interrupt notification by the PSRTO bit is enabled
3	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	LCHNGIP	0	R/W	LINK Signal Change Interrupt Enable 0: Interrupt notification by the LCHNG bit is disabled 1: Interrupt notification by the LCHNG bit is enabled
1	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ICDIP	0	R/W	Illegal Carrier Detect Interrupt Enable 0: Interrupt notification by the ICD bit is disabled 1: Interrupt notification by the ICD bit is enabled

(5) PHY Interface Register (PIR)

PIR is a 32-bit readable/writable register that provides a means of accessing the PHY-LSI internal registers via the MII.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MDI	MDO	MMD	MDC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MDI	Undefined	R	MII Management Data-In Indicates the level of the MDIO pin.
2	MDO	0	R/W	MII Management Data-Out Outputs the value set in this bit from the MDIO pin when the MMD bit is 1.
1	MMD	0	R/W	MII Management ModeSpecifies the data read/write direction with respect to the MII.0: Read direction is specified1: Write direction is specified
0	MDC	0	R/W	MII Management Data ClockOutputs the value set in this bit from the MDC pin and supplies the MII with the management data clock.

(6) PHY Status Register (PSR)

PSR is a read-only register that can read interface signals from the PHY-LSI.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LMON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LMON	Undefined	R	LNKSTA Pin Status The Link status can be read by connecting the Link signal output from the PHY-LSI to the LNKSTA pin. For the polarity, refer to the specifications of the PHY-LSI to be connected.

(7) Random Number Generation Counter Upper Limit Setting Register (RDMLR)

RDMLR is used to set the upper limit for the counter used in the random number generation block.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	RMD[19:16]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 0	RMD[19:0]	All 0	R/W	Upper Limit for Counter Used in Random Number Generation Block H'00000: Set value in normal operation H'00001to H'FFFFE: Upper limit for the counter

Note: The operation of the random number generation block in the feLic depends on the setting in this register. Accordingly, special attention should be paid when setting a value other than 0.

(8) IPG Register (IPGR)

IPGR sets the IPG (Inter Packet Gap). This register must not be changed while the transmitting and receiving functions of the EtherC mode register (ECMR) are enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	IPG[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	IPG[4:0]	H'14	R/W	Inter Packet Gap Sets the IPG value every 4-bit time. H'00: 16-bit time H'01: 20-bit time : : H'14: 96-bit time (Default) : : H'1F: 140-bit time

(9) Automatic PAUSE Frame Register (APR)

APR is used to set the TIME parameter value of an automatic PAUSE frame. When an automatic PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	AP[15:0]	All 0	R/W	Automatic PAUSE These bits set the TIME parameter value of an automatic PAUSE frame. One bit is equivalent to 512 bit-time.

(10) Manual PAUSE Frame Register (MPR)

MPR is used to set the TIME parameter value of a manual PAUSE frame. When a manual PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MP[15:0]															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	Undefined	W	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MP[15:0]	Undefined	W	Manual PAUSE These bits set the TIME parameter value of a manual PAUSE frame. One bit is equivalent to 512 bit-time. Read value is undefined.

(11) PAUSE Frame Receive Counter Register (RFCF)

RFCF is an 8-bit counter that indicates the number of times a PAUSE frame is received.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RPAUSE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	RPAUSE[7:0]	All 0	R	PAUSE Frame Receive Count

(12) Automatic PAUSE Frame Retransmit Count Register (TPAUSER)

TPAUSER is used to set the upper limit for the number of times to retransmit an automatic PAUSE frame. The settings in this register must not be changed while the transmitting function is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPAUSE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TPAUSE[15:0]	All 0	R/W	Upper Limit for Automatic PAUSE Frame Retransmission H'0000: Retransmit count is unlimited H'0001: Retransmit count is 1 : : H'FFFF: Retransmit count is 65,535

(13) PAUSE Frame Retransmit Counter Register (TPAUSECR)

PFTCR is a 16-bit counter that indicates the number of times a PAUSE frame is retransmitted.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXP[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	TXP[7:0]	All 0	R	PAUSE Frame Retransmit Count

(14) MAC Address High Register (MAHR)

MAHR is a 32-bit readable/writable register that specifies the upper 32 bits of the 48-bit MAC address. The settings in this register are normally made in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Return the EtherC and E-DMAC to their initial states by means of the SWR bit in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MA[47:32]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MA[47:16]	All 0	R/W	MAC Address Bits 47 to 16 These bits are used to set the upper 32 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'01234567 in this register.

(15) MAC Address Low Register (MALR)

MALR is a 32-bit readable/writable register that specifies the lower 16 bits of the 48-bit MAC address. The settings in this register are normally made in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Return the EtherC and E-DMAC to their initial states by means of the SWR bit in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MA[15:0]	All 0	R/W	MAC Address Bits 15 to 0 These bits are used to set the lower 16 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'89AB in this register.

(16) Transmit Retry Over Counter Register (TROCR)

TROCR is a 32-bit counter that indicates the number of frames that were unable to be transmitted in 16 transmission attempts including the retransfer. When 16 transmission attempts have failed, this register is incremented by 1. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TROC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TROC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TROC[31:0]	All 0	R/W	Transmit Retry Over Count These bits indicate the number of frames that were unable to be transmitted in 16 transmission attempts including the retransfer.

(17) Delayed Collision Detect Counter Register (CDCR)

CDCR is a 32-bit counter that indicates the number of all delayed collisions that occurred on the line after the start of data transmission. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	COSDC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COSDC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	COSDC[31:0]	All 0	R/W	Delayed Collision Detect Count These bits indicate the number of all delayed collisions after the start of data transmission.

(18) Lost Carrier Counter Register (LCCR)

LCCR is a 32-bit counter that indicates the number of times the carrier was lost during data transmission. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LCC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LCC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	LCC[31:0]	All 0	R/W	Lost Carrier Count These bits indicate the number of times the carrier was lost during data transmission.

(19) Carrier Not Detect Counter Register (CNDCR)

CNDCR is a 32-bit counter that indicates the number of times carrier could not be detected while the preamble is being sent. When the value in this register reaches H'FFFFFFFF, the counter stops incrementing. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNDC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNDC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CNDC[31:0]	All 0	R/W	Carrier Not Detect Count These bits indicate the number of times carrier was not detected.

(20) CRC Error Frame Receive Counter Register (CEFCR)

CEFCR is a 32-bit counter that indicates the number of times a frame with a CRC error was received. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CEFC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CEFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CEFC[31:0]	All 0	R/W	CRC Error Frame Count These bits indicate the number of CRC error frames received.

(21) Frame Receive Error Counter Register (FRECR)

FRECR is a 32-bit counter that indicates the number of frames for which a receive error was generated by the RX-ER pin input from the PHY-LSI. FRECR is incremented each time the RX-ER pin becomes active. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FREC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FREC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FREC[31:0]	All 0	R/W	Frame Receive Error Count These bits indicate the number of errors during frame reception.

(22) Too-Short Frame Receive Counter Register (TSFRCR)

TSFRCR is a 32-bit counter that indicates the number of frames received with a length fewer than 64 bytes. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSFC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TSFC[31:0]	All 0	R/W	Too-Short Frame Receive Count These bits indicate the number of frames received with a length of less than 64 bytes.

(23) Too-Long Frame Receive Counter Register (TLFRCR)

TLFRCR is a 32-bit counter that indicates the number of frames received with a length exceeding the value specified by the receive frame length register (RFLR). When the value in this register reaches H'FFFFFFFF, count-up is halted. This register is not incremented when a frame containing residual bits is received. In this case, the reception of the frame is indicated in the residual-bit frame receive counter register (RFCR). The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TLFC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TLFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TLFC[31:0]	All 0	R/W	Too-Long Frame Receive Count These bits indicate the number of frames received with a length exceeding the value in RFLR.

(24) Residual-Bit Frame Receive Counter Register (RFCR)

RFCR is a 32-bit counter that indicates the number of frames received containing residual bits (less than an 8-bit unit). When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RFC[31:0]	All 0	R/W	Residual-Bit Frame Receive Count These bits indicate the number of frames received containing residual bits.

(25) Multicast Address Frame Receive Counter Register (MAFCR)

MAFCR is a 32-bit counter that indicates the number of frames received with a specified multicast address. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter value is cleared to 0 by a write to this register with any value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MAFC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MAFC[31:0]	All 0	R/W	Multicast Address Frame Count These bits indicate the number of multicast frames received.

47.5 E-DMAC Operation

The E-DMAC, connected to the EtherC, allows efficient transfer of transmit/receive data between the EtherC and memory (buffers) without CPU intervention. The E-DMAC automatically reads the control information referred to as descriptors. The descriptors corresponding to each buffer hold buffer pointers and other information. The E-DMAC reads transmit data from the transmit buffer and writes receive data to the receive buffer according to the control information. By arranging such multiple descriptors continuously (i.e., making a descriptor list), continuous transmission or reception is possible.

47.5.1 Descriptor Lists and Data Buffers

By the communication program, a transmit descriptor list and a receive descriptor list should be created in memory space prior to transmission and reception. The start addresses of these lists should be set to the transmit descriptor list start address register and receive descriptor list start address register.

The start addresses of the descriptor lists should be placed on the address boundaries in accordance with the descriptor length specified by the E-DMAC mode register (EDMR). Here, the start address of the transmit buffer can be placed on a longword, word, or byte boundary.

(1) Transmit Descriptor

Figure 47.2 shows the relationship between a transmit descriptor and a transmit buffer. The descriptor can relate one transmit frame to one transmit buffer (single-frame/single-buffer operation) or multiple transmit buffers (single-frame/multi-buffer operation).

When the transmit buffer length (TBL) is to be set to 1 to 16 bytes, the buffer address needs to be placed on a 32-byte boundary. When the transmit buffer length (TBL) is set to 0 byte, operation cannot be guaranteed.

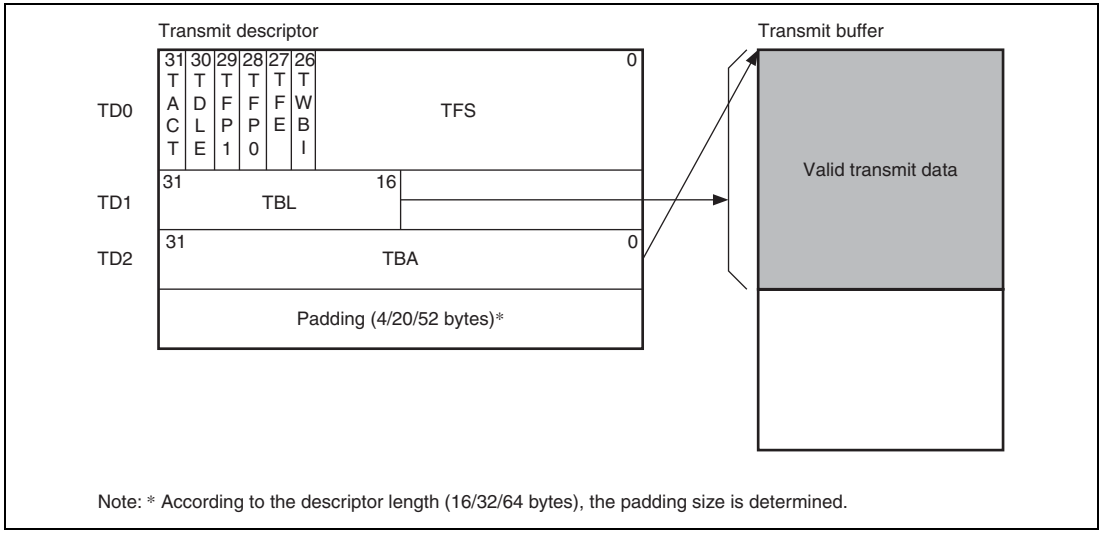


Figure 47.2 Relationship between Transmit Descriptor and Transmit Buffer

(a) Transmit Descriptor 0 (TD0)

TD0 indicates the transmit frame status informing frame transmission status.

(The underlined bits are subject to write-back in the table below.)

Bit	Bit Name	Initial Value	R/W	Description
31	TACT	0	R/W	Transmit Descriptor Valid Indicates that the corresponding descriptor is valid. This bit is set to 1 by software. This bit is cleared to 0 by hardware when a transmit frame has been completely transferred or when transmission has been aborted due to some cause.
30	TDLE	0	R/W	Transmit Descriptor Ring End When set to 1, this bit indicates that the corresponding descriptor is the last one of the descriptor ring.
29 28	TFP1 TFP0	0 0	R/W R/W	Transmit Frame Positions 1 and 0 These bits relate the transmit buffer to the transmit frame. The settings of these bits and the TBL bits should be logically correct in the consecutive descriptors. 00: Transmission of the frame of the transmit buffer specified by this descriptor is continued. (The frame is incomplete.) 01: The transmit buffer specified by this descriptor contains the end of the frame (The frame is complete.) 10: The transmit buffer specified by this descriptor is the start of the frame (The frame is incomplete.) 11: The contents in the transmit buffer specified by this descriptor correspond to one frame (single-frame/single-buffer).
27	TFE	0	R/W	Transmit Frame Error When set to 1, this bit indicates that an error is indicated by any of the TFS bits. (By so setting TRSCER, it is possible to prevent this bit from being set by an event indicated by TFS7 to TFS0. It is impossible, however, if an event indicated by TFS7 to TFS0 also causes TFS8 to be set.) 1: Frame transmission has been aborted.
26	TWBI	0	R/W	Write-Back Completion Interrupt Notification (This bit is valid when TRIMD is set so.) 0: nop 1: An interrupt is generated upon completion of write-back to this descriptor.

Bit	Bit Name	Initial Value	R/W	Description
25 to 0	TFS	All 0	R/W	Transmit Frame Status TFS25 toTFS9 [Reserved (The write value should always be 0.)]: TFS8 [Detect Transmit Abort]; When set to 1, this bit indicates that the abort signal is set to 1 during frame transmission. (causing TFE to be set) TFS7 to TFS4 [Reserved (The write value should always be 0.)]; TFS3 [Detect of No Carrier (corresponding to the CND bit in EESR)]; TFS2 [Detect Loss of Carrier (corresponding to the DLC bit in EESR)]; TFS1 [Detect of Delayed Collision during Transmission (corresponding to the CD bit in EESR)]; TFS0 [Transmit Retry Over (corresponding to the TRO bit in EESR)]: When set to 1, these bits indicate that TFS8 to TFS1 have been set to 1 during frame transmission. (Although TFE is normally set when these bits are set to 1, it can be prevented from being set by so setting TRSCER.)

(b) Transmit Descriptor 1 (TD1)

TD1 indicates the length of the transmit buffer.

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TBL	All 0	R/W	Transmit Buffer Length Indicates the length of the relevant transmit buffer in terms of valid bytes.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

(c) Transmit Descriptor 2 (TD2)

TD2 indicates the start address of the relevant transmit buffer.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TBA	All 0	R/W	Transmit Buffer Address Indicates the start address of the transmit buffer.

(2) Receive Descriptor

Figure 47.3 shows the relationship between a receive descriptor and a receive buffer. The receive buffer address is to be placed on a 32-byte boundary.

When the receive buffer length (RBL) is set to 0 byte, operation specified by the descriptor cannot be guaranteed.

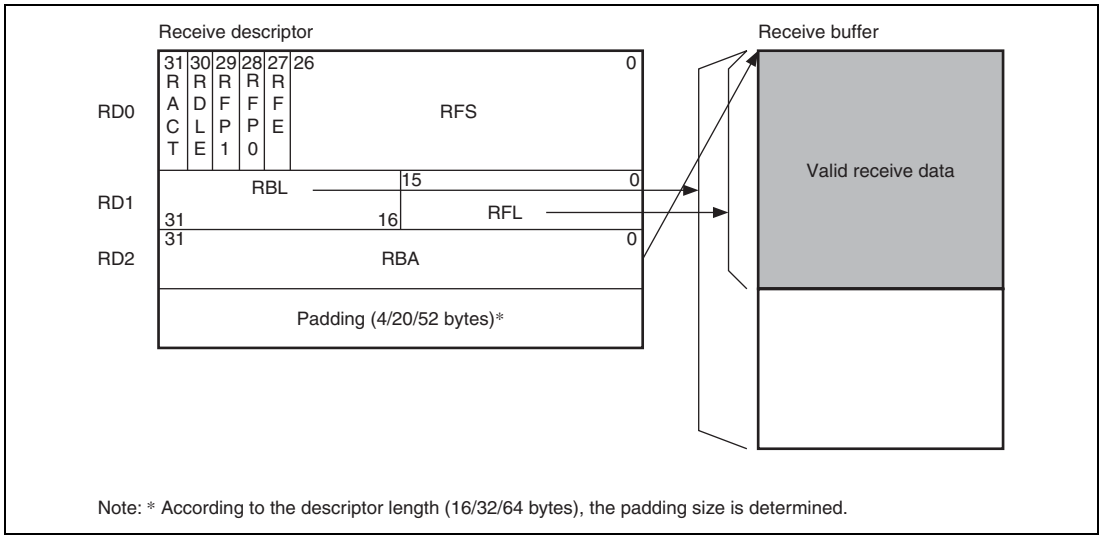


Figure 47.3 Relationship between Receive Descriptor and Receive Buffer

(a) Receive Descriptor 0 (RD0)

RD0 indicates the receive frame status informing frame reception status.

(The underlined bits are subject to write-back in the table below.)

Bit	Bit Name	Initial Value	R/W	Description
31	RACT	0	R/W	Receive Descriptor Valid Indicates that the corresponding descriptor is valid. This bit is set to 1 by software. This bit is cleared to 0 by hardware when an entire receive frame has been completely transferred to the buffer address specified by RD2 or when the receive buffer becomes full.
30	RDLE	0	R/W	Receive Descriptor Ring End When set to 1, this bit indicates that the corresponding descriptor is the last one of the descriptor ring.
29, 28	RFP[1:0]	00	R/W	Receive Frame Positions 1 and 0 These bits relate the receive buffer to the receive frame. The settings of these bits and the TBL bits should be logically correct in the consecutive descriptors. 00: Reception of the frame of the receive buffer specified by this descriptor is continued. (The frame is incomplete.) 01: The receive buffer specified by this descriptor contains the end of the frame (The frame is complete.) 10: The receive buffer specified by this descriptor is the start of the frame (The frame is incomplete.) 11: The contents in the receive buffer specified by this descriptor correspond to one frame (single-frame/single-buffer).
27	RFE	0	R/W	Receive Frame Error When set to 1, this bit indicates that an error is indicated by any of the RFS bits. (By so setting TRSCER, it is possible to prevent this bit from being set by an event indicated by RFS7 to RFS0. It is impossible, however, if an event indicated by RFS7 to RFS0 also causes RFS8 to be set.)

Bit	Bit Name	Initial Value	R/W	Description
26 to 0	RFS	All 0	R/W	Receive Frame Status RF26 to RF10 [Reserved (The write value should always be 0.)]; RFS9 [Receive FIFO Overflow (corresponding to the RFOF bit in EESR)]: When set to 1, this bit indicates that a receive FIFO overflow has occurred terminating the frame halfway and that the frame has been written back. (causing RFE to be set) TFS8 [Detect Receive Abort]; When set to 1, this bit indicates that the abort signal is set to 1 during frame transmission. (causing RFE to be set) RFS7 [Multicast address frame received (corresponding to the RMAF bit in EESR)]; RFS6 and RFS5 [Reserved (The write value should always be 0.)]; RFS4 [Residual-bit frame receive error (corresponding to the RRF bit in EESR)]; RFS3 [Long frame receive error (corresponding to the RTLF bit in EESR)]; RFS2 [Short frame receive error (corresponding to the RTSF bit in EESR)]; RFS1 [PHY-LSI receive error (corresponding to the PRE bit in EESR)]; RFS0 [CRC error detected in receive frame (corresponding to the CERF bit in EESR)]: When set to 1, these bits indicate that RFS8 to RFS1 have been set to 1 during frame reception. (Although RFE is normally set when these bits are set to 1, it can be prevented from being set by so settingTRSCER.)

(b) Receive Descriptor 1 (RD1)

RD1 indicates the length of the receive buffer.

(The underlined bits are subject to write-back in the table below.)

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RBL	All 0	R/W	Receive Buffer Length Indicates the length of the relevant receive buffer in terms of bytes. The buffer length should be set as n in 32 × n, which represents the buffer size.
15 to 0	RFL	All 0	R	Receive Data Length Indicates the length of (number of bytes in) a receive frame stored in the buffer The number of bytes for padding insertion specified by RPADIR are excluded. These bits are written back to the descriptor containing the end of a frame.

(c) Receive Descriptor 2 (RD2)

RD2 indicates the start address of the relevant receive buffer.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RBA	All 0	R/W	Receive Buffer Address Indicates the start address of the receive buffer. The buffer address should be set on a 32-byte boundary.

47.5.2 Transmission

When the transmit request bit (TR) in the E-DMAC transmit request register (EDTRR) is set while the transmission function is enabled, the E-DMAC reads the descriptor following the previously used descriptor from the transmit descriptor list (or the descriptor indicated by the transmit descriptor start address register (TDLAR) at the initial start time). If the TACT bit of the read descriptor is set to 1 (valid), the E-DMAC sequentially reads transmit frame data from the transmit buffer start address specified by TD2 for transfer to the EtherC. The EtherC creates a transmit frame and starts transmission to the MII. After DMA transfer of data equivalent to the buffer length specified in the descriptor, the following processing is carried out according to the TFP value.

1. TFP = 00 or 10 (frame continuation):

   Descriptor write-back (writing 0 to the TACT bit) is performed after DMA transfer.
2. TFP = 01 or 11 (frame end):

   Descriptor write-back (writing 0 to the TACT bit and writing status) is performed after completion of frame transmission.

As long as the TACT bit of a read descriptor is set to 1 (valid), the reading of E-DMAC descriptors and the transmission of frames continue. When a descriptor with the TACT bit cleared to 0 (invalid) is read, the E-DMAC clears the TR bit in EDTRR to 0 and completes transmit processing.

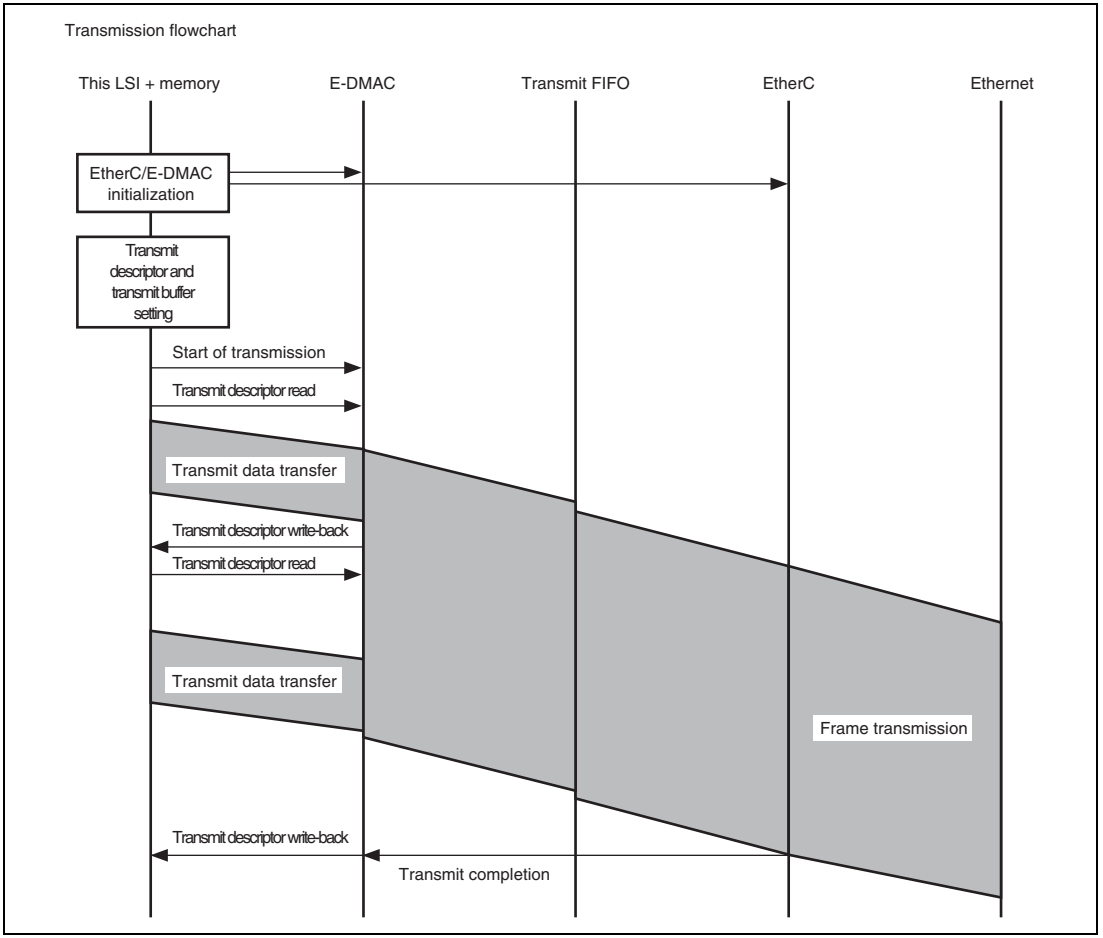


Figure 47.4 Sample Transmission Flowchart (Single-Frame/Two-Descriptor)

47.5.3 Reception

When the CPU sets the receive request bit (RR) in the E-DMAC receive request register (EDRRR) while the receive function is enabled, the E-DMAC reads the descriptor following the previously used descriptor from the receive descriptor list (or the descriptor indicated by the receive descriptor start address register (RDLAR) at the initial start time) then enters the receive standby state. When the EtherC receives a frame for this LSI (with an address enabled for reception by this LSI), the EtherC stores the receive data in the receive FIFO. Upon receiving the frame for own station while the RACT bit is set to 1 (valid), the E-DMAC transfers the frame to the receive buffer specified by RD2. If the data length of a received frame is longer than the buffer length specified by RD1, the E-DMAC performs a write-back operation to the descriptor (with RFP set to 10 or 00) when the buffer becomes full, then reads the next descriptor. The E-DMAC then continues to transfer data to the receive buffer specified by the new RD2. When frame reception is completed, or if frame reception is suspended because of a certain kind of error, the E-DMAC performs write-back to the relevant descriptor (with RFP set to 11 or 01), and then ends the receive processing. The E-DMAC then reads the next descriptor and enters the receive standby state again.

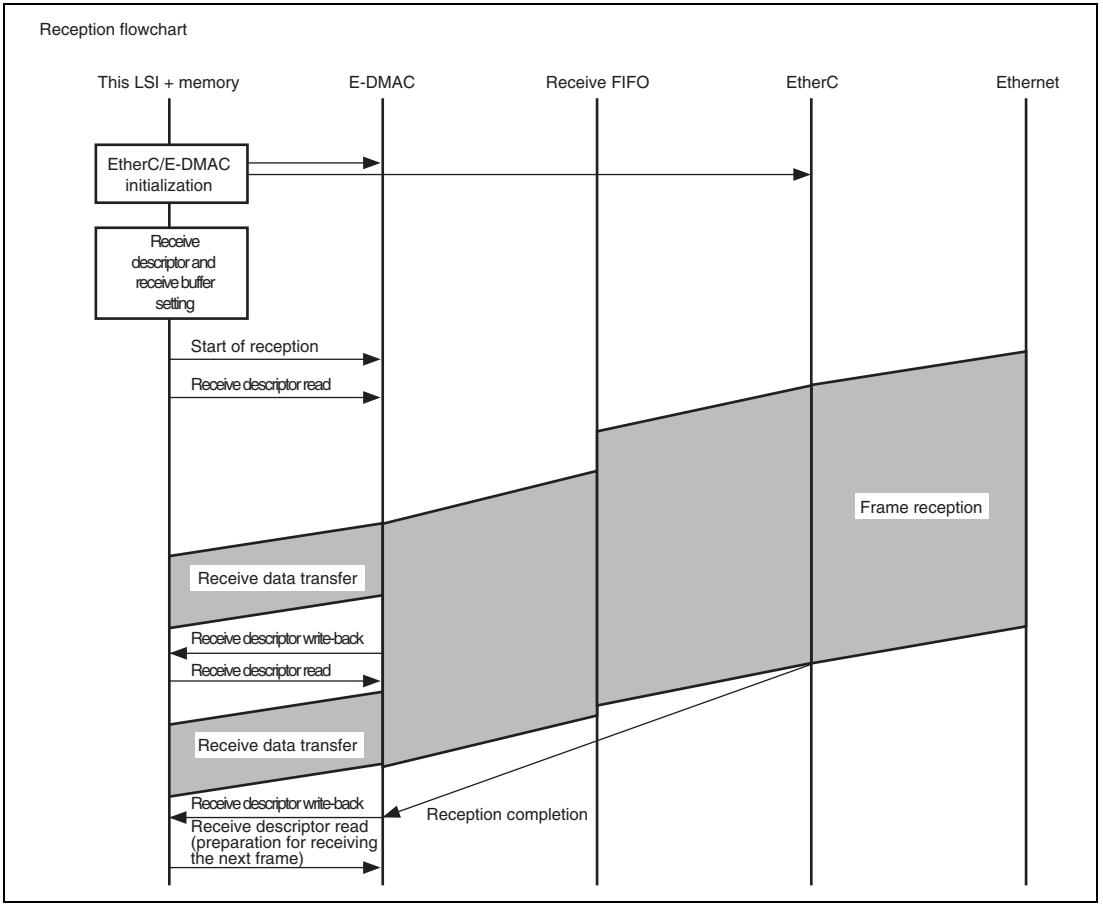


Figure 47.5 Sample Reception Flowchart (Single-Frame/Two-Descriptor)

47.5.4 Receive Processing in Case of Multi-Buffer Frame

If an error occurs during reception in the case of a multi-buffer frame where a receive frame is divided for storage in multiple buffers, the E-DMAC performs the processing shown in Figure 47.6.

In the figure, the invalid receive descriptors (with the RACT bit cleared to 0) represent the normal reception of data to be stored in buffers, and the valid receive descriptors (with the RACT bit set to 1) represent unreceived buffers. If a frame receive error occurs with a descriptor shown in the figure, the status is written back to the corresponding descriptor.

If error interrupts are enabled in the EtherC/E-DMAC status interrupt permission register (EESIPR), an interrupt is generated immediately after the write-back. If there is a new frame receive request, reception is continued from the buffer after that in which the error occurred.

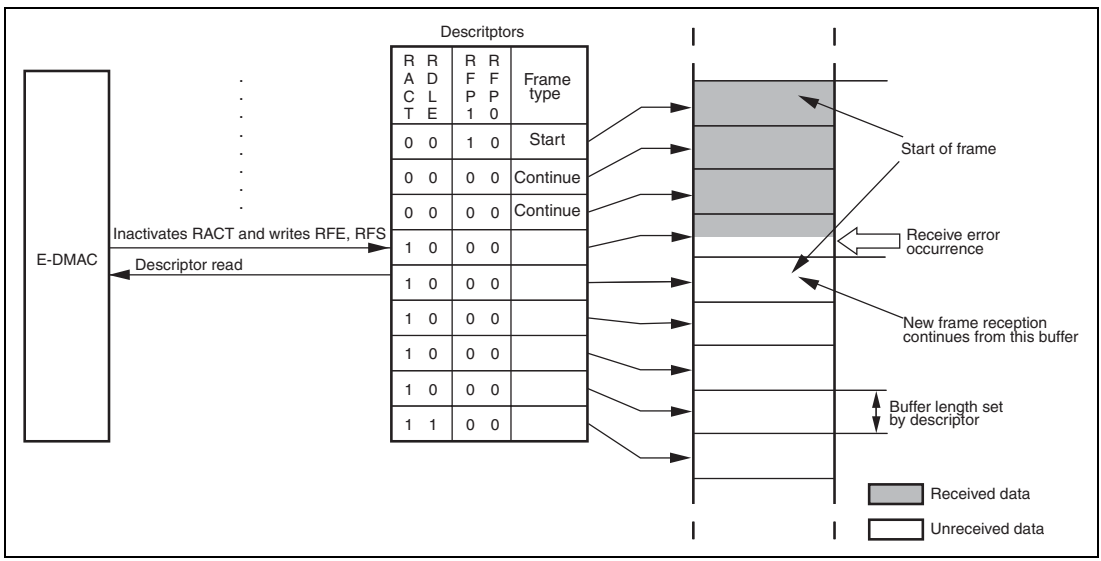


Figure 47.6 E-DMAC Operation after Receive Error

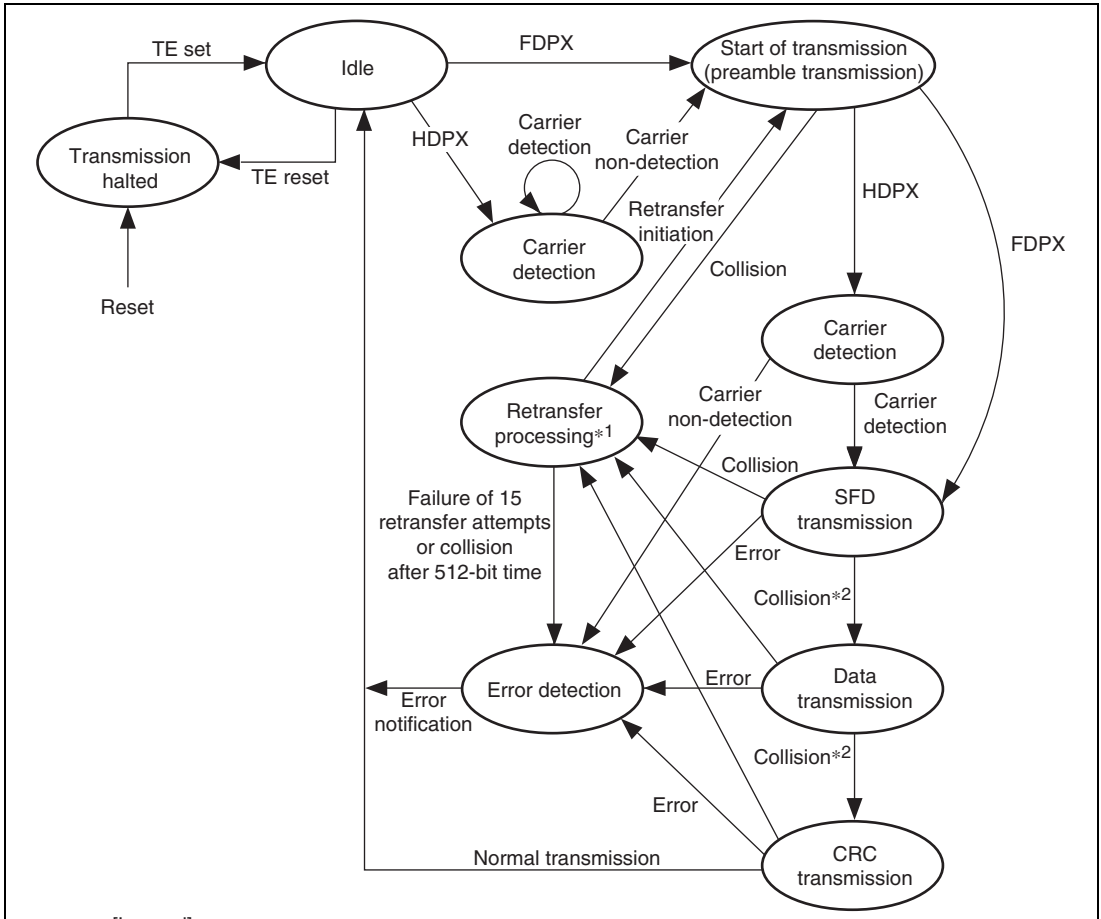
47.6 EtherC Operation

The following outlines the operations of the Ethernet controller (EtherC).

The Ethernet controller (EtherC) supports flow control functions conforming to IEEE802.3x, and transmission/reception of PAUSE frames used for the control is possible.

47.6.1 Transmission

The EtherC transmitter assembles the transmit data on the frame and outputs to MII when there is a transmit request from the E-DMAC. The data transmitted via the MII is transmitted to the lines by PHY-LSI. Figure 47.7 shows the status change of the EtherC transmitter.



[Legend]
FDPX: Full-duplex
HDPX: Half-duplex
SFD: Start Frame Delimiter

Notes:
1. Transmission retry processing includes both jam transmission that depends on collision detection and the adjustment of transmission intervals based on the back-off algorithm.
2. Transmission is retried only when data of 512 bits or less (including the preamble and SFD)is transmitted. When a collision is detected during the transmission of data greater than 512 bits, only jam is transmitted and transmission based on the back-off algorithm is not retried.

Figure 47.7 EtherC Transmitter State Transitions

1. When the transmit enable (TE) bit is set, the transmitter enters the transmit idle state.
2. When a transmit request is issued by the transmit E-DMAC, the EtherC sends the preamble after a transmission delay equivalent to the frame interval time. If full-duplex transfer is selected, which does not require carrier detection, the preamble is sent as soon as a transmit request is issued by the E-DMAC.
3. The transmitter sends the SFD, data, and CRC sequentially. At the end of transmission, the transmit E-DMAC generates a transmission complete interrupt (TC). If a collision or the carrier-not-detected state occurs during data transmission, these are reported as interrupt sources.
4. After waiting for the frame interval time, the transmitter enters the idle state, and if there is more transmit data, continues transmitting.

47.6.2 Reception

The EtherC receiver separates the frame from the MII into preamble, SFD, data and CRC, and the fields from DA (destination address) to the CRC data are transferred to the receive E-DMAC. Figure 47.8 shows the state transitions of the EtherC receiver.

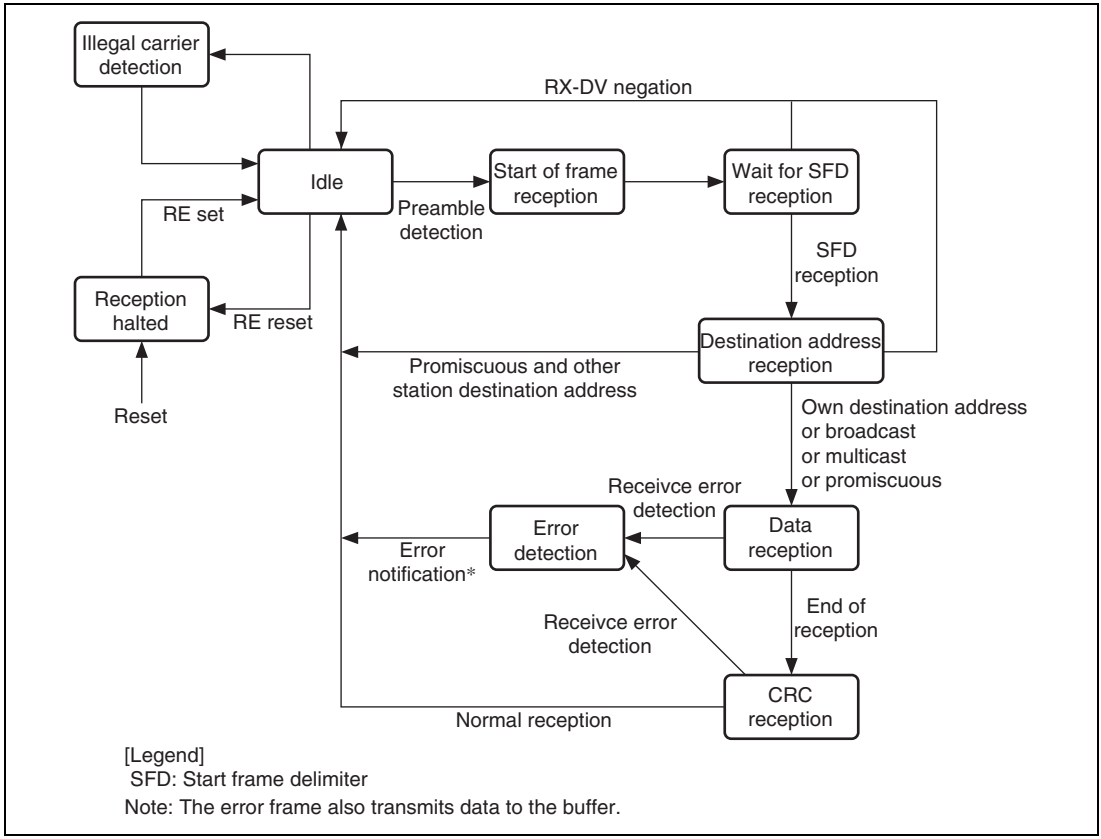


Figure 47.8 EtherC Receiver State Transmissions

1. When the receive enable (RE) bit is set, the receiver enters the receive idle state.
2. When an SFD (start frame delimiter) is detected after a receive packet preamble, the receiver starts receive processing. Discards a frame with an invalid pattern.
3. In normal mode, if the destination address matches the receiver's own address, or if broadcast or multicast transmission or promiscuous mode is specified, the receiver starts data reception.
4. Following data reception from the MII, the receiver carries out a CRC check. The result is indicated as a status bit in the descriptor after the frame data has been written to memory. Reports an error status in the case of an abnormality.
5. After one frame has been received, if the receive enable bit is set (RE = 1) in the EtherC mode register, the receiver prepares to receive the next frame.

47.6.3 MII Frame Timing

Each MII Frame timing is shown in Figure 47.9.

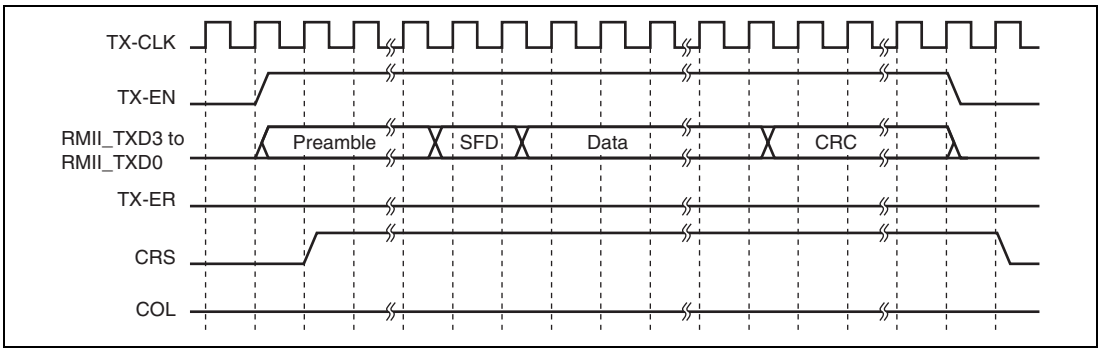


Figure 47.9 (1) MII Frame Transmit Timing (Normal Transmission)

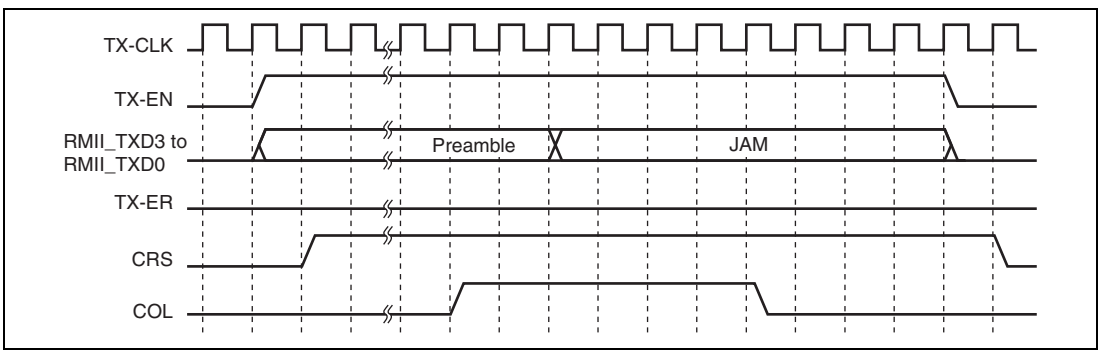


Figure 47.9 (2) MII Frame Transmit Timing (Collision)

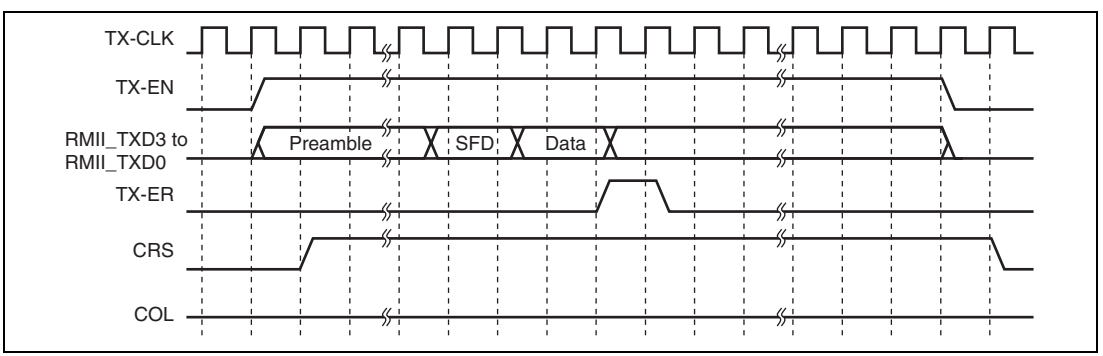


Figure 47.9 (3) MII Frame Transmit Timing (Transmit Error)

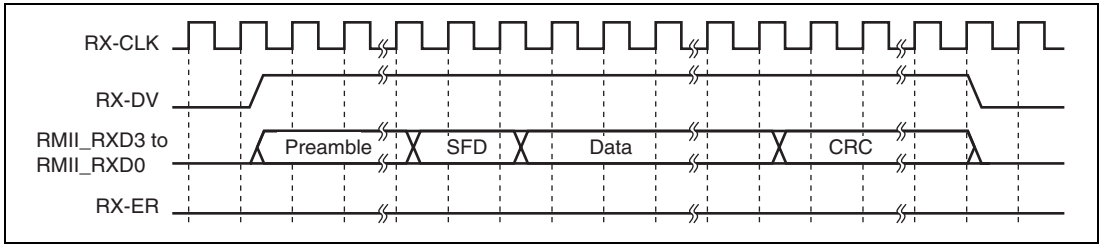


Figure 47.9 (4) MII Frame Receive Timing (Normal Reception)

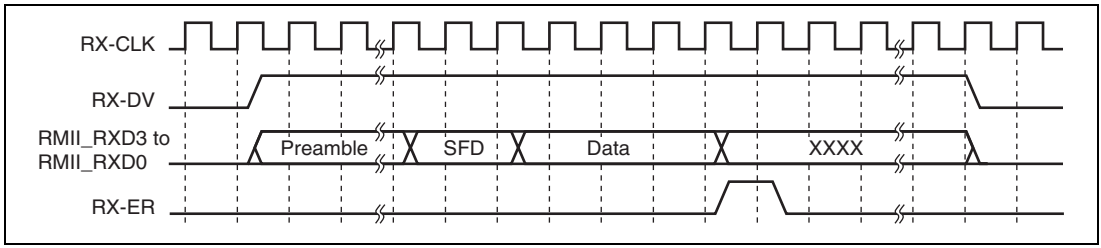


Figure 47.9 (5) MII Frame Receive Timing (Reception Error (1): Receive Error Notification)

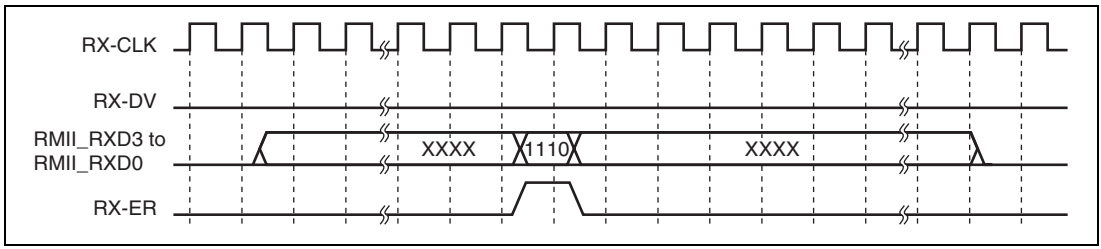


Figure 47.9 (6) MII Fame Receive Timing (Reception Error (2): Carrier Error Notification)

47.6.4 Accessing MII Registers

MII registers in the PHY-LSI are accessed via this LSI's PHY interface register (PIR). Connection is made as a serial interface in accordance with the MII frame format specified in IEEE802.3u.

(1) MII Management Frame Format

The format of an MII management frame is shown in Figure 47.10. To access an MII register, a management frame is implemented by the program in accordance with the procedures shown in MII Register Access Procedure.

Access Type	MII Management Frame							
Item	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
Number of bits	32	2	2	5	5	2	16	
Read	1..1	01	10	00001	RRRRR	Z0	D..D	
Write	1..1	01	01	00001	RRRRR	10	D..D	X

[Legend]
PRE: 32 consecutive 1s
ST: Write of 01 indicating start of frame
OP: Write of code indicating access type
PHYAD: Write of 0001 if the PHY-LSI address is 1 (sequential write starting with the MSB).
This bit changes depending on the PHY-LSI address.
REGAD: Write of 0001 if the register address is 1 (sequential write starting with the MSB).
This bit changes depending on the PHY-LSI register address.
TA: Time for switching data transmission source on MII interface
(a) Write: 10 written
(b) Read: Bus release (notation: Z0) performed
DATA: 16-bit data. Sequential write or read from MSB
(a) Write: 16-bit data write
(b) Read: 16-bit data read
IDLE: Wait time until next MII management format input
(a) Write: Independent bus release (notation: X) performed
(b) Read: Bus already released in TA; control unnecessary

Figure 47.10 MII Management Frame Format

(2) MII Register Access Procedure

The program accesses MII registers via the PHY interface register (PIR). Access is implemented by a combination of 1-bit-unit data write, 1-bit-unit data read, bus release, and independent bus release. Figure 47.11 shows the MII register access timing. The timing will differ depending on the PHY-LSI type.

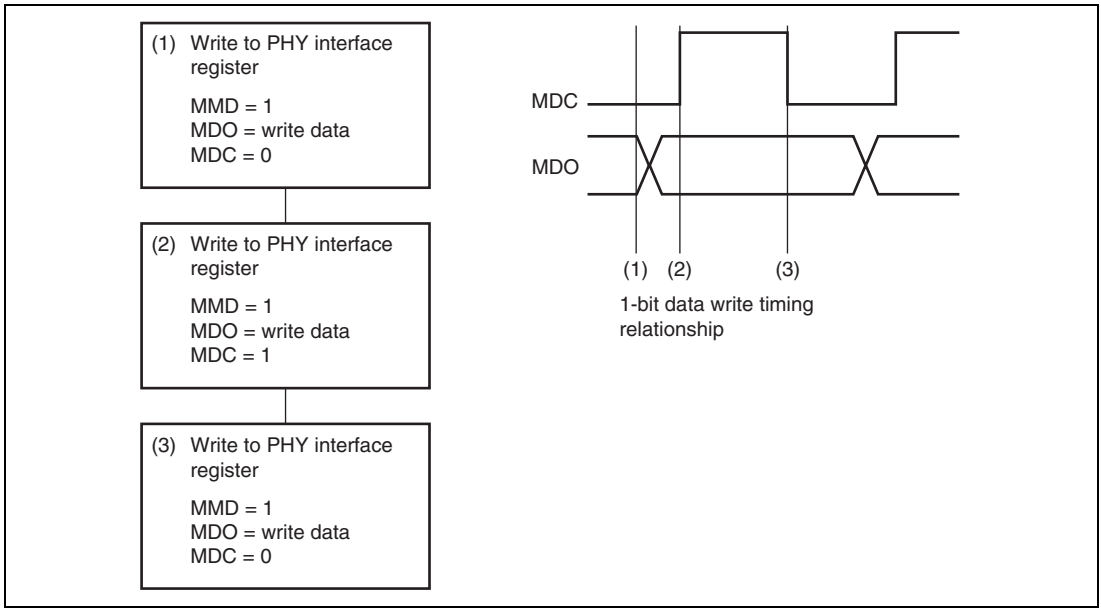


Figure 47.11 (1) 1-Bit Data Write Flowchart

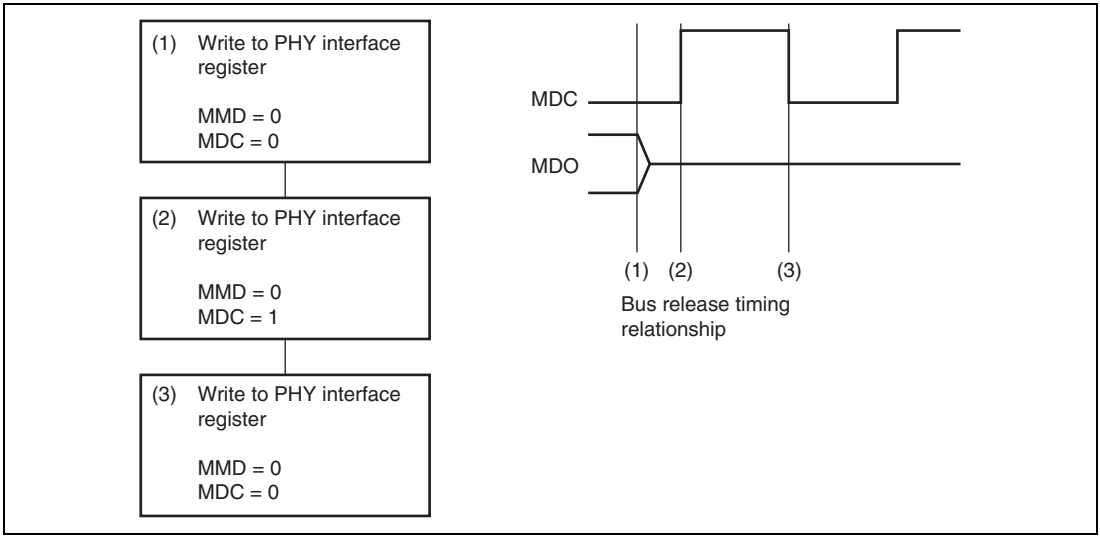


Figure 47.11 (2) Bus Release Flowchart (TA in Read in Figure 47.10)

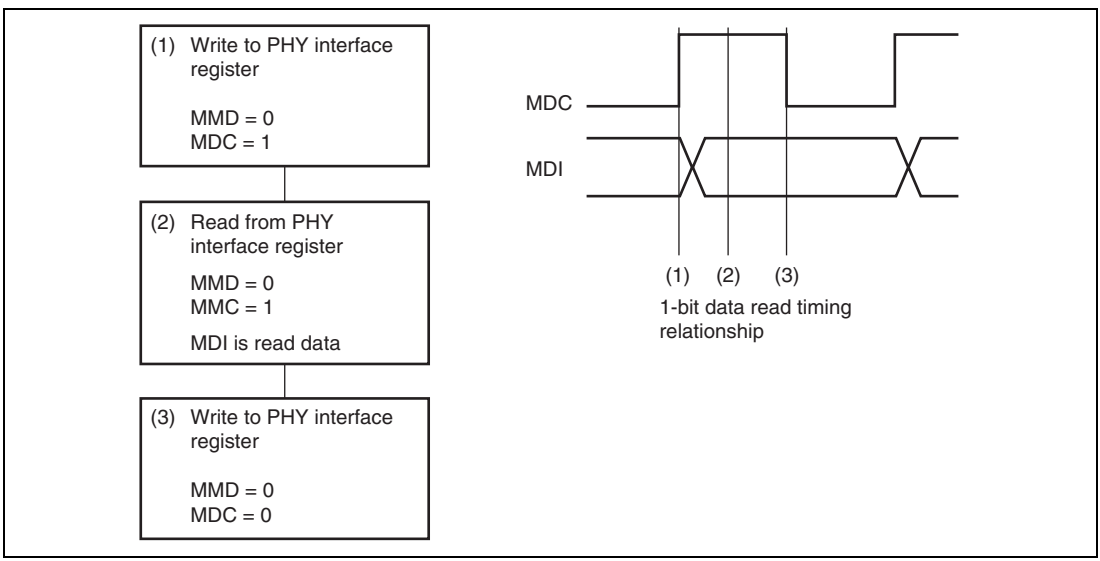


Figure 47.11 (3) 1-Bit Data Read Flowchart

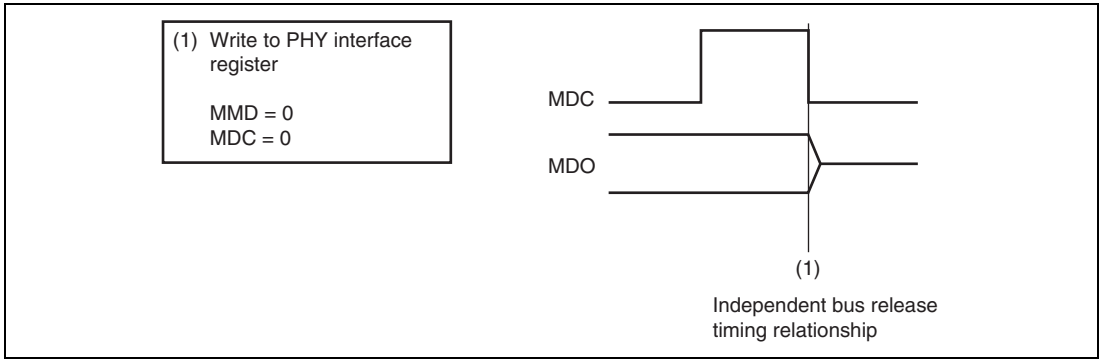


Figure 47.11 (4) Independent Bus Release Flowchart (IDLE in Write in Figure 47.10)

47.6.5 Operation by IPG Setting

The EtherC has a function to change the non-transmission period IPG (Inter Packet Gap) between transmit frames. By changing the set values of the IPG setting register (IPGR), the transmission efficiency can be raised and lowered from the standard value. IPG settings are prescribed in IEEE802.3 standards. When changing settings, adequately check that the respective devices can operate smoothly on the same network.

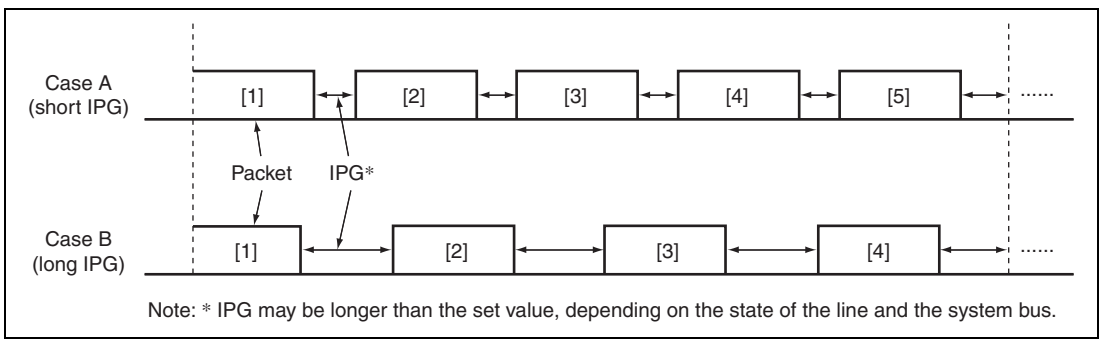


Figure 47.12 Changing IPG and Transmission Efficiency

47.6.6 Flow Control

The EtherC supports flow control functions conforming to IEEE802.3x for full-duplex operation. The flow control can be applied to both receive and transmit operations. When transmitting PAUSE frames, flow control can be performed by the following two procedures :

(1) Automatic PAUSE Frame Transmission

For receive frames, PAUSE frames are automatically transmitted when the number of data written to the receive FIFO reaches the value set in FCFTR. The TIME parameter included in the PAUSE frame is set by APR. The automatic PAUSE frame transmission is repeated until the number of data in the receive FIFO becomes less than the value set in FCFTR as the receive data is read from the FIFO. Using TPAUSER, the upper limit of retransmission counts of the PAUSE frames can also be set. In this case, PAUSE frame transmission is repeated until the number of receive FIFO data becomes less than the FCFTR value, or the number of transmits reaches the value set by TPAUSER.

The automatic PAUSE frame transmission is enabled when the TXF bit in ECMR is 1.

(2) Manual PAUSE Frame Transmission

PAUSE frames are transmitted by directives from the software. When writing the Timer value to MPR, manual PAUSE frame transmission is started. With this method, PAUSE frame transmission is carried out only once.

(3) PAUSE Frame Reception

The next frame is not transmitted until the time indicated by the Timer value elapses after receiving a PAUSE frame. However, the transmission of the current frame is continued. A received PAUSE frame is valid only when the RXF bit in ECMR is set to 1. The number of times of PAUSE frame receptions is counted.

47.7 Connection with PHY-LSI

47.7.1 RMII Frame Timing

Timing of frames for the RMII is shown in figures 47.13 and 47.14.

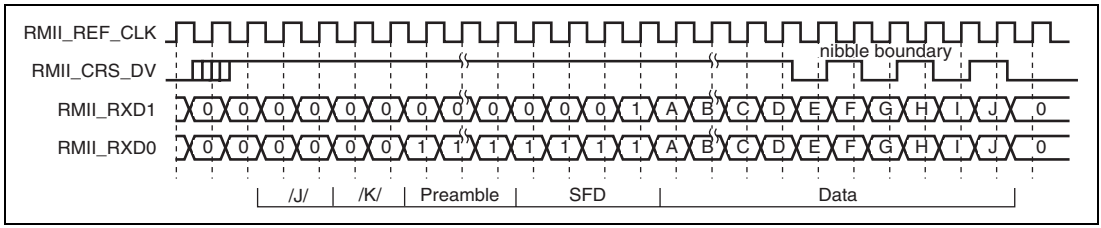


Figure 47.13 Timing of RMII Frame Reception (100 Mbps in Normal Reception)

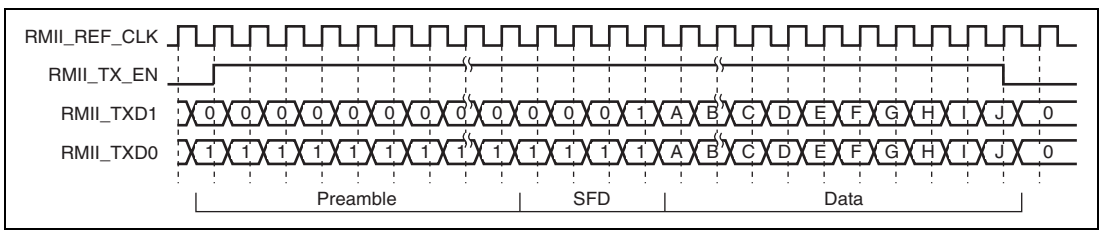


Figure 47.14 Timing of RMII Frame Transmission (100 Mbps in Normal Transmission)

47.7.2 MII-RMII Conversion

This LSI supports an RMII. The RMII includes an MII-RMII conversion circuit to convert signals to and from the MII.

(1) Clock

The REF50CK (50 MHz) signal from the RMII is frequency divided to output the ET_TX-CLK and ET_RX-CLK (25 MHz, 2.5 MHz) signals.

(2) Reception

Waveforms received through the RMII are converted for the MII and output (at 10 Mbps or 100 Mbps).

False carrier detection from the RMII is converted for the MII and output.

The RMI_RX-ER signal from the RMII is output to the MII.

Note: False-carrier-detection is not generated until the preamble is detected and reception is terminated. (ET_RX-DV is negated).

(3) Transmission

Waveforms for transmission from the MII are converted for the RMII and output (at 10 Mbps or 100 Mbps).

The collision signal ET_COL is generated by taking the logical "AND" of the CRS and ET_TX-EN signals.

(4) Selection of Full Duplex/Half Duplex

COL assertion is suppressed in full-duplex transfer mode.

Figure 47.15 is a block diagram of the interface conversion circuit.

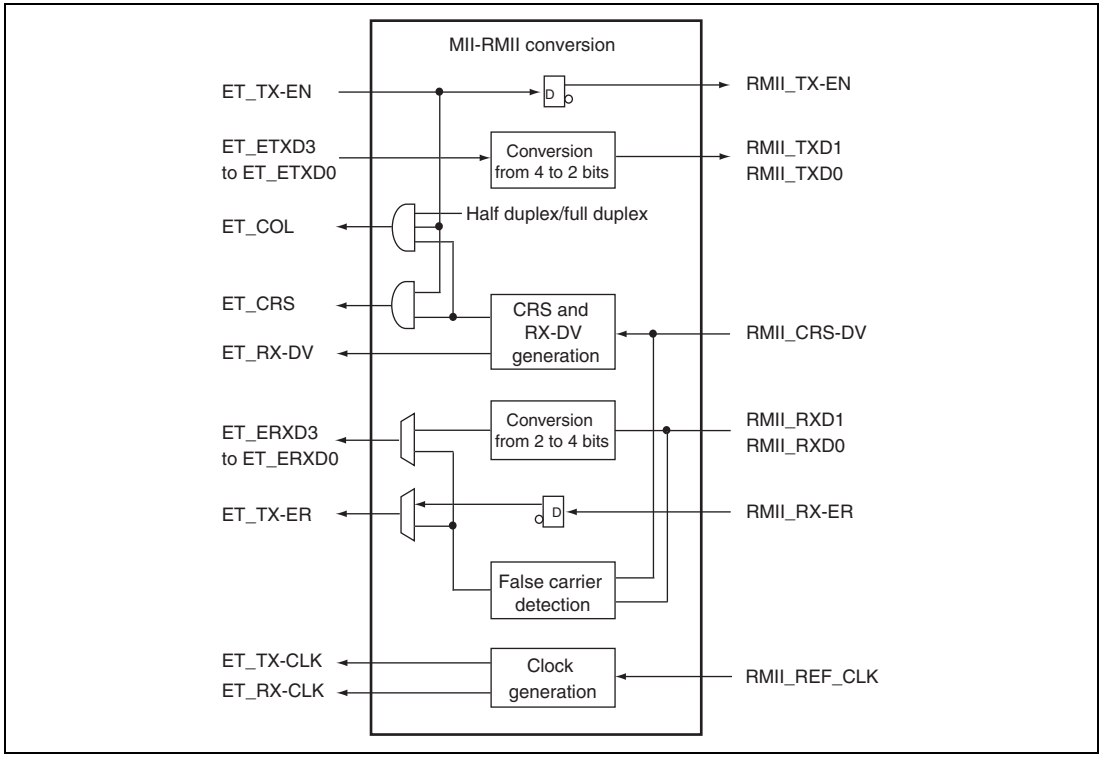


Figure 47.15 Block Diagram of MII-RMII Conversion

47.8 Usage Notes

47.8.1 Pin Multiplexing Set Register

To enable the functions of the Ethernet MAC controller, set the LSI pin multiplexing set register 1 of HPB to Ether. This should be done before the CPU accesses the Ethernet MAC controller.

47.8.2 Soft Reset

The Ethernet MAC controller can be soft reset by setting the SWR bit of EDMR of the E-DMAC to 1.

If a soft reset is performed during a DMA transfer, the transfer is completed, but DMA data cannot be guaranteed.

47.8.3 Standby

When the Ethernet MAC controller is directed to enter standby mode, it completes the ongoing SHwy initiator/target operation before entering standby mode. To wake up the Ethernet MAC controller, a reset and initialization are required.

47.8.4 Backoff Period during Half-Duplex Operation

There is a known problem with the Ethernet MAC controller. The problem occurs only when a conflict with other stations is encountered and the backoff period may be longer than specified. Though the degradation of performance depends on the frequency of conflicts, the more conflicts there are, the more performance degrades.

Section 48 Pin Function Controller (PFC)

48.1 Overview

The pin function controller (PFC) consists of registers to select the functions of the general port and multiplexed pins. Pin functions and I/O directions can be individually selected for every pin regardless of the LSI operating mode.

Table 48.1 lists the multiplexed pins of this LSI. Functions are selectable from a general port, functions 1, 2, and 3 for each pin. For the multiplexed pins with function 1 only, function 1 can be selected by setting the port control register to enable the pin multiplex function. For the multiplexed pins with functions 1, 2, and 3, one of the functions can be selected by setting the port control register and by selecting the function using the pin select register.

The functions in the shaded area in the table are available immediately after a reset. The settings of the I/O buffer Hi-Z control registers have priorities over the setting of the port control register.

Table 48.1 Multiplexed Pins

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTA7 input/output	D23 input/output (BSC)	KEYOUT2 output (KEYSC)	IDED15 input/output (ATAPI)
PTA6 input/output	D22 input/output (BSC)	KEYOUT1 output (KEYSC)	IDED14 input/output (ATAPI)
PTA5 input/output	D21 input/output (BSC)	KEYOUT0 output (KEYSC)	IDED13 input/output (ATAPI)
PTA4 input/output	D20 input/output (BSC)	KEYIN4 input (KEYSC)	IDED12 input/output (ATAPI)
PTA3 input/output	D19 input/output (BSC)	KEYIN3 input (KEYSC)	IDED11 input/output (ATAPI)
PTA2 input/output	D18 input/output (BSC)	KEYIN2 input (KEYSC)	IDED10 input/output (ATAPI)
PTA1 input/output	D17 input/output (BSC)	KEYIN1 input (KEYSC)	IDED9 input/output (ATAPI)
PTA0 input/output	D16 input/output (BSC)	KEYIN0 input (KEYSC)	IDED8 input/output (ATAPI)
PTB7 input/output	D31 input/output (BSC)	TPUTO1 output (TPU)	IDEA1 output (ATAPI)
PTB6 input/output	D30 input/output (BSC)	TPUTO0 output (TPU)	IDEA0 output (ATAPI)
PTB5 input/output	D29 input/output (BSC)	—	IODREQ input (ATAPI)

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTB4 input/output	D28 input/output (BSC)	—	$\overline{\text{IDECS0}}$ output (ATAPI)
PTB3 input/output	D27 input/output (BSC)	—	$\overline{\text{IDECS1}}$ output (ATAPI)
PTB2 input/output	D26 input/output (BSC)	KEYOUT5/IN5 input/output (KEYSC)	$\overline{\text{IDEIORD}}$ output (ATAPI)
PTB1 input/output	D25 input/output (BSC)	KEYOUT4/IN6 input/output (KEYSC)	$\overline{\text{IDEIOWR}}$ output (ATAPI)
PTB0 input/output	D24 input/output (BSC)	KEYOUT3 output (KEYSC)	IDEINT input (ATAPI)
PTC7 input/output	LCDD7 input/output (LCDC)	—	—
PTC6 input/output	LCDD6 input/output (LCDC)	—	—
PTC5 input/output	LCDD5 input/output (LCDC)	—	—
PTC4 input/output	LCDD4 input/output (LCDC)	—	—
PTC3 input/output	LCDD3 input/output (LCDC)	—	—
PTC2 input/output	LCDD2 input/output (LCDC)	—	—
PTC1 input/output	LCDD1 input/output (LCDC)	—	—
PTC0 input/output	LCDD0 input/output (LCDC)	—	—
PTD7 input/output	LCDD15 input/output (LCDC)	—	—
PTD6 input/output	LCDD14 input/output (LCDC)	—	—
PTD5 input/output	LCDD13 input/output (LCDC)	—	—
PTD4 input/output	LCDD12 input/output (LCDC)	—	—
PTD3 input/output	LCDD11 input/output (LCDC)	—	—
PTD2 input/output	LCDD10 input/output (LCDC)	—	—
PTD1 input/output	LCDD9 input/output (LCDC)	—	—
PTD0 input/output	LCDD8 input/output (LCDC)	—	—
PTE7 input/output	FSIMCKB input (FSI)	—	—
PTE6 input/output	FSIMCKA input (FSI)	—	—
PTE5 input/output	LCDD21 input/output (LCDC)	SCIF2_TXD output (SCIF2)	—
PTE4 input/output	LCDD20 input/output (LCDC)	SCIF4_SCK input/output (SCIFA4)	—
PTE3 input/output	LCDD19 input/output (LCDC)	SCIF4_RXD input (SCIFA4)	—
PTE2 input/output	LCDD18 input/output (LCDC)	SCIF4_TXD output (SCIFA4)	—
PTE1 input/output	LCDD17 input/output (LCDC)	—	—
PTE0 input/output	LCDD16 input/output (LCDC)	—	—

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTF7 input/output	LCDVSYN input/output (LCDC)	—	—
PTF6 input/output	LCDDISP/LCDRS output (LCDC)	—	—
PTF5 input/output	LCDHSYN/$\overline{\text{LCDCS}}$ output (LCDC)	—	—
PTF4 input/output	LCDDON output (LCDC)	—	—
PTF3 input/output	LCDDCK/$\overline{\text{LCDWR}}$ output (LCDC)	—	—
PTF2 input/output	LCDVEPWC output (LCDC)	SCIF0_TXD output (SCIF0)	—
PTF1 input/output	LCDD23 input/output (LCDC)	SCIF2_SCK input/output (SCIF2)	—
PTF0 input/output	LCDD22 input/output (LCDC)	SCIF2_RXD input (SCIF2)	—
PTG5 output	AUDCK output (AUD)	—	—
PTG4 output	AUDSYNC output (AUD)	—	—
PTG3 output	AUDATA3 output (AUD)	—	—
PTG2 output	AUDATA2 output (AUD)	—	—
PTG1 output	AUDATA1 output (AUD)	—	—
PTG0 output	AUDATA0 output (AUD)	—	—
PTH7 input/output	VIO0_VD input (VIO)	—	—
PTH6 input/output	VIO0_CLK input (VIO)	—	—
PTH5 input/output	VIO0_D7 input (VIO)	—	—
PTH4 input/output	VIO0_D6 input (VIO)	—	—
PTH3 input/output	VIO0_D5 input (VIO)	—	—
PTH2 input/output	VIO0_D4 input (VIO)	—	—
PTH1 input/output	VIO0_D3 input (VIO)	—	—
PTH0 input/output	VIO0_D2 input (VIO)	—	—
PTJ7 output	PDSTATUS output (System)	—	—
PTJ6 output	STATUS2 output (System)	—	—
PTJ5 output	STATUS0 output (System)	—	—
PTJ3 input/output	A25 output (BSC)	$\overline{\text{BS}}$_ output (BSC)	—
PTJ2 input/output	A24 output (BSC)	—	—
PTJ1 input/output	A23 output (BSC)	—	—

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTJ0 input/output	A22 output (BSC)	—	—
PTK7 input/output	VIO1_D5 input (VIO)	VIO0_D13 input (VIO)	IDED5 input/output (ATAPI)
PTK6 input/output	VIO1_D4 input (VIO)	VIO0_D12 input (VIO)	IDED4 input/output (ATAPI)
PTK5 input/output	VIO1_D3 input (VIO)	VIO0_D11 input (VIO)	IDED3 input/output (ATAPI)
PTK4 input/output	VIO1_D2 input (VIO)	VIO0_D10 input (VIO)	IDED2 input/output (ATAPI)
PTK3 input/output	VIO1_D1 input (VIO)	VIO0_D9 input (VIO)	IDED1 input/output (ATAPI)
PTK2 input/output	VIO1_D0 input (VIO)	VIO0_D8 input (VIO)	IDED0 input/output (ATAPI)
PTK1 input/output	VIO0_FLD input (VIO)	—	—
PTK0 input/output	VIO0_HD input (VIO)	—	—
PTL7 input/output	DV_D5 output (VOU)	SCIF3_SCK input/output (SCIFA3)	RMII_RXD0 input (EtherMAC)
PTL6 input/output	DV_D4 output (VOU)	SCIF3_RXD input (SCIFA3)	RMII_RXD1 input (EtherMAC)
PTL5 input/output	DV_D3 output (VOU)	SCIF3_TXD output (SCIFA3)	RMII_REF_CLK input (EtherMAC)
PTL4 input/output	DV_D2 output (VOU)	SCIF1_SCK input/output (SCIF1)	RMII_TX_EN output (EtherMAC)
PTL3 input/output	DV_D1 output (VOU)	SCIF1_RXD input (SCIF1)	RMII_TXD0 output (EtherMAC)
PTL2 input/output	DV_D0 output (VOU)	SCIF1_TXD output (SCIF1)	RMII_TXD1 output (EtherMAC)
PTL1 input/output	DV_D15 output (VOU)	—	—
PTL0 input/output	DV_D14 output (VOU)	MSIOF0_MCK input/output (MSIOF0)	—
PTM7 input/output	DV_D13 output (VOU)	MSIOF0_TSCK input/output (MSIOF0)	—
PTM6 input/output	DV_D12 output (VOU)	MSIOF0_RXD input (MSIOF0)	—
PTM5 input/output	DV_D11 output (VOU)	MSIOF0_TXD output (MSIOF0)	—

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTM4 input/output	DV_D10 output (VOU)	MSIOF0_TSYNC input/output (MSIOF0)	—
PTM3 input/output	DV_D9 output (VOU)	$\overline{\text{MSIOF0\_SS1}}$/MSIOF0_RSCK input/output (MSIOF0)	—
PTM2 input/output	DV_D8 output (VOU)	$\overline{\text{MSIOF0\_SS2}}$/MSIOF0_RSYNC input/output (MSIOF0)	—
PTM1 input/output	LCDVCPWC output (LCDC)	SCIF0_RXD input (SCIF0)	—
PTM0 input/output	$\overline{\text{LCDRD}}$ output (LCDC)	SCIF0_SCK input/output (SCIF0)	—
PTN7 input/output	VIO0_D1 input (VIO)	—	—
PTN6 input/output	VIO0_D0 input (VIO)	—	—
PTN5 input/output	DV_CLKI input (VOU)	—	—
PTN4 input/output	DV_CLK output (VOU)	SCIF2_SCK input/output (SCIF2)	—
PTN3 input/output	DV_VSYNC output (VOU)	SCIF2_RXD input (SCIF2)	—
PTN2 input/output	DV_HSYNC output (VOU)	SCIF2_TXD output (SCIF2)	—
PTN1 input/output	DV_D7 output (VOU)	$\overline{\text{SCIF3\_CTS}}$ input (SCIFA3)	RMII_RX_ER input (EtherC)
PTN0 input/output	DV_D6 output (VOU)	$\overline{\text{SCIF3\_RTS}}$ output (SCIFA3)	RMII_CRS_DV input (EtherC)
PTQ7 input/output	D7 input/output (BSC)	—	—
PTQ6 input/output	D6 input/output (BSC)	—	—
PTQ5 input/output	D5 input/output (BSC)	—	—
PTQ4 input/output	D4 input/output (BSC)	—	—
PTQ3 input/output	D3 input/output (BSC)	—	—
PTQ2 input/output	D2 input/output (BSC)	—	—
PTQ1 input/output	D1 input/output (BSC)	—	—
PTQ0 input/output	D0 input/output (BSC)	—	—
PTR7 input/output	$\overline{\text{CS6B}}$/$\overline{\text{CE1B}}$ output (BSC)	—	—
PTR6 input/output	$\overline{\text{CS6A}}$/$\overline{\text{CE2B}}$ output (BSC)	—	—
PTR5 input/output	$\overline{\text{CS5B}}$/$\overline{\text{CE1A}}$ output (BSC)	—	—
PTR4 input/output	$\overline{\text{CS5A}}$/$\overline{\text{CE2A}}$ output (BSC)	—	—
PTR3 input	$\overline{\text{IOIS16}}$ input (BSC)	LCDLCLK input (LCDC)	—
PTR2 input	$\overline{\text{WAIT}}$ input (BSC)	—	—

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTR1 input/output	$\overline{\text{WE3}}$/$\overline{\text{ICIOWR}}$ output (BSC)	TPUTO3 output (TPU)	TPUTI3 input (TPU)
PTR0 input/output	$\overline{\text{WE2}}$/$\overline{\text{ICIORD}}$ output (BSC)	TPUTO2 output (TPU)	IDEA2 output (ATAPI)
PTS6 input/output	VIO_CKO output (VIO)	—	—
PTS5 input/output	VIO1_FLD input (VIO)	TPUTI2 input (TPU)	IDEIORDY input (ATAPI)
PTS4 input/output	VIO1_HD input (VIO)	SCIF5_SCK input/output (SCIFA5)	—
PTS3 input/output	VIO1_VD input (VIO)	SCIF5_RXD input (SCIFA5)	—
PTS2 input/output	VIO1_CLK input (VIO)	SCIF5_TXD output (SCIFA5)	—
PTS1 input/output	VIO1_D7 input (VIO)	VIO0_D15 input (VIO)	IDED7input/output (ATAPI)
PTS0 input/output	VIO1_D6 input (VIO)	VIO0_D14 input (VIO)	IDED6input/output (ATAPI)
PTT7 input/output	D15 input/output (BSC)	—	—
PTT6 input/output	D14 input/output (BSC)	—	—
PTT5 input/output	D13 input/output (BSC)	—	—
PTT4 input/output	D12 input/output (BSC)	—	—
PTT3 input/output	D11 input/output (BSC)	—	—
PTT2 input/output	D10 input/output (BSC)	—	—
PTT1 input/output	D9 input/output (BSC)	—	—
PTT0 input/output	D8 input/output (BSC)	—	—
PTU7 input/output	DACK0 output (DMAC)	—	—
PTU6 input/output	DREQ0 input (DMAC)	—	—
PTU5 input/output	FSIOASD output (FSI)	—	—
PTU4 input/output	FSIIABCK input (FSI)	—	—
PTU3 input/output	FSIIALRCK input (FSI)	—	—
PTU2 input/output	FSIOABCK output (FSI)	—	—
PTU1 input/output	FSIOALRCK output (FSI)	—	—
PTU0 input/output	CLKAUDIOAO output (FSI)	—	—
PTV7 input/output	FSIIBSD input (FSI)	$\overline{\text{MSIOF1\_SS2}}$/MSIOF1_RSYNC input/output (MSIOF1)	—
PTV6 input/output	FSIOBSD output (FSI)	$\overline{\text{MSIOF1\_SS1}}$/MSIOF1_RSCK input/output (MSIOF1)	—
PTV5 input/output	FSIIBBCK input (FSI)	MSIOF1_RXD input (MSIOF1)	—

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTV4 input/output	FSIIBLRCK input (FSI)	MSIOF1_TSYNC output (MSIOF1)	—
PTV3 input/output	FSIOBBCK output (FSI)	MSIOF1_TSCK input/output (MSIOF1)	—
PTV2 input/output	FSIOBLRCK output (FSI)	MSIOF1_TXD output (MSIOF1)	—
PTV1 input/output	CLKAUDIOBO output (FSI)	MSIOF1_MCK input (MSIOF1)	—
PTV0 input/output	FSIIASD input (FSI)	—	—
PTW7 input/output	MMC_D7 input/output (MMC)	$\overline{\text{SDHI1CD}}$ input (SDHI1)	$\overline{\text{IODACK}}$ output (ATAPI)
PTW6 input/output	MMC_D6 input/output (MMC)	SDHI1WP input (SDHI1)	$\overline{\text{IDERST}}$ output(ATAPI)
PTW5 input/output	MMC_D5 input/output (MMC)	SDHI1D3 input/output (SDHI1)	$\overline{\text{EXBUF\_ENB}}$ output (ATAPI)
PTW4 input/output	MMC_D4 input/output (MMC)	SDHI1D2 input/output (SDHI1)	DIRECTION output (ATAPI)
PTW3 input/output	MMC_D3 input/output (MMC)	SDHI1D1 input/output (SDHI1)	—
PTW2 input/output	MMC_D2 input/output (MMC)	SDHI1D0 input/output (SDHI1)	—
PTW1 input/output	MMC_D1 input/output (MMC)	SDHI1CMD input/output (SDHI1)	—
PTW0 input/output	MMC_D0 input/output (MMC)	SDHI1CLK output (SDHI1)	—
PTX7 input/output	DACK1 output (DMAC)	IRDA_OUT output (IrDA)	—
PTX6 input/output	DREQ1 input (DMAC)	$\overline{\text{IRDA\_IN}}$ input (IrDA)	—
PTX5 input/output	TS0_SDAT input (TSIF)	—	LNKSTA input (EtherMAC)
PTX4 input/output	TS0_SCK input (TSIF)	—	MDIO input/output (EtherMAC)
PTX3 input/output	TS0_SDEN input (TSIF)	—	MDC output (EtherMAC)
PTX2 input/output	TS0_SPSYNC input (TSIF)	—	—
PTX1 input/output	MMC_CLK output (MMC)	—	—
PTX0 input/output	MMC_CMD input/output (MMC)	—	—
PTY7 input/output	$\overline{\text{SDHI0CD}}$ input (SDHI0)	—	—
PTY6 input/output	SDHI0WP input (SDHI0)	—	—
PTY5 input/output	SDHI0D3 input/output (SDHI0)	—	—
PTY4 input/output	SDHI0D2 input/output (SDHI0)	—	—

General Port	Function 1 (Related Module)	Function 2 (Related Module)	Function 3 (Related Module)
PTY3 input/output	SDHI0D1 input/output (SDHI0)	—	—
PTY2 input/output	SDHI0D0 input/output (SDHI0)	—	—
PTY1 input/output	SDHI0CMD input/output (SDHI0)	—	—
PTY0 input/output	SDHI0CLK output (SDHI0)	—	—
PTZ7 input/output	IRQ7 input (INTC)	$\overline{\text{SCIF3\_CTS}}$ input (SCIFA3)	—
PTZ6 input/output	IRQ6 input (INTC)	$\overline{\text{SCIF3\_RTS}}$ output (SCIFA3)	—
PTZ5 input/output	IRQ5 input (INTC)	SCIF3_SCK input/output (SCIFA3)	—
PTZ4 input/output	IRQ4 input (INTC)	SCIF3_RXD input (SCIFA3)	—
PTZ3 input/output	IRQ3 input (INTC)	SCIF3_TXD output (SCIFA3)	—
PTZ2 input/output	IRQ2 input (INTC)	—	—
PTZ1 input/output	IRQ1 input (INTC)	—	—
PTZ0 input/output	IRQ0 input (INTC)	—	—

48.2 Register Descriptions

Table 48.2 shows the PFC registers. Table 48.3 shows the register states in each operating mode.

Table 48.2 Register Configuration

Register Name	Abbreviation	R/W	Address	Access Size
Port A control register	PACR	R/W	H'A405 0100	16
Port B control register	PBCR	R/W	H'A405 0102	16
Port C control register	PCCR	R/W	H'A405 0104	16
Port D control register	PDCR	R/W	H'A405 0106	16
Port E control register	PECR	R/W	H'A405 0108	16
Port F control register	PFCR	R/W	H'A405 010A	16
Port G control register	PGCR	R/W	H'A405 010C	16
Port H control register	PHCR	R/W	H'A405 010E	16
Port J control register	PJCR	R/W	H'A405 0110	16
Port K control register	PKCR	R/W	H'A405 0112	16
Port L control register	PLCR	R/W	H'A405 0114	16
Port M control register	PMCR	R/W	H'A405 0116	16
Port N control register	PNCR	R/W	H'A405 0118	16
Port Q control register	PQCR	R/W	H'A405 011A	16
Port R control register	PRCR	R/W	H'A405 011C	16
Port S control register	PSCR	R/W	H'A405 011E	16
Port T control register	PTCR	R/W	H'A405 0140	16
Port U control register	PUCR	R/W	H'A405 0142	16
Port V control register	PVCR	R/W	H'A405 0144	16
Port W control register	PWCR	R/W	H'A405 0146	16
Port X control register	PXCR	R/W	H'A405 0148	16
Port Y control register	PYCR	R/W	H'A405 014A	16
Port Z control register	PZCR	R/W	H'A405 014C	16
Port A data register	PADR	R/W	H'A405 0120	8
Port B data register	PBDR	R/W	H'A405 0122	8
Port C data register	PCDR	R/W	H'A405 0124	8
Port D data register	PDDR	R/W	H'A405 0126	8
Port E data register	PEDR	R/W	H'A405 0128	8

Register Name	Abbreviation	R/W	Address	Access Size
Port F data register	PFDR	R/W	H'A405 012A	8
Port G data register	PGDR	R/W	H'A405 012C	8
Port H data register	PHDR	R/W	H'A405 012E	8
Port J data register	PJDR	R/W	H'A405 0130	8
Port K data register	PKDR	R/W	H'A405 0132	8
Port L data register	PLDR	R/W	H'A405 0134	8
Port M data register	PMDR	R/W	H'A405 0136	8
Port N data register	PNDR	R/W	H'A405 0138	8
Port Q data register	PQDR	R/W	H'A405 013A	8
Port R data register	PRDR	R/W	H'A405 013C	8
Port S data register	PSDR	R/W	H'A405 013E	8
Port T data register	PTDR	R/W	H'A405 0160	8
Port U data register	PUDR	R/W	H'A405 0162	8
Port V data register	PVDR	R/W	H'A405 0164	8
Port W data register	PWDR	R/W	H'A405 0166	8
Port X data register	PXDR	R/W	H'A405 0168	8
Port Y data register	PYDR	R/W	H'A405 016A	8
Port Z data register	PZDR	R/W	H'A405 016C	8
Pin select register A	PSELA	R/W	H'A405 014E	16
Pin select register B	PSELB	R/W	H'A405 0150	16
Pin select register C	PSELC	R/W	H'A405 0152	16
Pin select register D	PSELD	R/W	H'A405 0154	16
Pin select register E	PSELE	R/W	H'A405 0156	16
I/O buffer Hi-Z control register A	HIZCRA	R/W	H'A405 0158	16
I/O buffer Hi-Z control register B	HIZCRB	R/W	H'A405 015A	16
I/O buffer Hi-Z control register C	HIZCRC	R/W	H'A405 015C	16
I/O buffer Hi-Z control register D	HIZCRD	R/W	H'A405 015E	16
Module function select register A	MSELCRA	R/W	H'A405 0180	16
Module function select register B	MSELCRB	R/W	H'A405 0182	16
Pull-up control register	PULCR	R/W	H'A405 0184	16
I/O buffer drive control register A	DRVCRA	R/W	H'A405 018A	16
I/O buffer drive control register B	DRVCRB	R/W	H'A405 018C	16
I/O buffer drive control register C	DRVCRC	R/W	H'A405 018E	16

Table 48.3 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
PACR	Initialized	Retained	Retained	—	Initialized	Retained
PBCR	Initialized	Retained	Retained	—	Initialized	Retained
PCCR	Initialized	Retained	Retained	—	Initialized	Retained
PDCR	Initialized	Retained	Retained	—	Initialized	Retained
PECR	Initialized	Retained	Retained	—	Initialized	Retained
PFCR	Initialized	Retained	Retained	—	Initialized	Retained
PGCR	Initialized	Retained	Retained	—	Initialized	Retained
PHCR	Initialized	Retained	Retained	—	Initialized	Retained
PJCR	Initialized	Retained	Retained	—	Initialized	Retained
PKCR	Initialized	Retained	Retained	—	Initialized	Retained
PLCR	Initialized	Retained	Retained	—	Initialized	Retained
PMCR	Initialized	Retained	Retained	—	Initialized	Retained
PNCR	Initialized	Retained	Retained	—	Initialized	Retained
PQCR	Initialized	Retained	Retained	—	Initialized	Retained
PRCR	Initialized	Retained	Retained	—	Initialized	Retained
PSCR	Initialized	Retained	Retained	—	Initialized	Retained
PTCR	Initialized	Retained	Retained	—	Initialized	Retained
PUCR	Initialized	Retained	Retained	—	Initialized	Retained
PVCR	Initialized	Retained	Retained	—	Initialized	Retained
PWCR	Initialized	Retained	Retained	—	Initialized	Retained
PXCR	Initialized	Retained	Retained	—	Initialized	Retained
PYCR	Initialized	Retained	Retained	—	Initialized	Retained
PZCR	Initialized	Retained	Retained	—	Initialized	Retained
PADR	Initialized	Retained	Retained	—	Initialized	Retained
PBDR	Initialized	Retained	Retained	—	Initialized	Retained
PCDR	Initialized	Retained	Retained	—	Initialized	Retained
PDDR	Initialized	Retained	Retained	—	Initialized	Retained
PEDR	Initialized	Retained	Retained	—	Initialized	Retained
PFDR	Initialized	Retained	Retained	—	Initialized	Retained
PGDR	Initialized	Retained	Retained	—	Initialized	Retained

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
PHDR	Initialized	Retained	Retained	—	Initialized	Retained
PJDR	Initialized	Retained	Retained	—	Initialized	Retained
PKDR	Initialized	Retained	Retained	—	Initialized	Retained
PLDR	Initialized	Retained	Retained	—	Initialized	Retained
PMDR	Initialized	Retained	Retained	—	Initialized	Retained
PNDR	Initialized	Retained	Retained	—	Initialized	Retained
PQDR	Initialized	Retained	Retained	—	Initialized	Retained
PRDR	Initialized	Retained	Retained	—	Initialized	Retained
PSDR	Initialized	Retained	Retained	—	Initialized	Retained
PTDR	Initialized	Retained	Retained	—	Initialized	Retained
PUDR	Initialized	Retained	Retained	—	Initialized	Retained
PVDR	Initialized	Retained	Retained	—	Initialized	Retained
PWDR	Initialized	Retained	Retained	—	Initialized	Retained
PYDR	Initialized	Retained	Retained	—	Initialized	Retained
PZDR	Initialized	Retained	Retained	—	Initialized	Retained
PSELA	Initialized	Retained	Retained	—	Initialized	Retained
PSELB	Initialized	Retained	Retained	—	Initialized	Retained
PSELC	Initialized	Retained	Retained	—	Initialized	Retained
PSELD	Initialized	Retained	Retained	—	Initialized	Retained
PSELE	Initialized	Retained	Retained	—	Initialized	Retained
HIZCRA	Initialized	Retained	Retained	—	Initialized	Retained
HIZCRB	Initialized	Retained	Retained	—	Initialized	Retained
HIZCRC	Initialized	Retained	Retained	—	Initialized	Retained
HIZCRD	Initialized	Retained	Retained	—	Initialized	Retained
MSELCRA	Initialized	Retained	Retained	—	Initialized	Retained
MSELCRB	Initialized	Retained	Retained	—	Initialized	Retained
PULCR	Initialized	Retained	Retained	—	Initialized	Retained
DRVCRA	Initialized	Retained	Retained	—	Initialized	Retained
DRVCRB	Initialized	Retained	Retained	—	Initialized	Retained
DRVCRC	Initialized	Retained	Retained	—	Initialized	Retained

48.2.1 Port A Control Register (PACR)

PACR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PA7MD[1:0]		PA6MD[1:0]		PA5MD[1:0]		PA4MD[1:0]		PA3MD[1:0]		PA2MD[1:0]		PA1MD[1:0]		PA0MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PA7MD1	0	R/W	PA7 Mode
14	PA7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PA6MD1	0	R/W	PA6 Mode
12	PA6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PA5MD1	0	R/W	PA5 Mode
10	PA5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PA4MD1	0	R/W	PA4 Mode
8	PA4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PA3MD1	0	R/W	PA3 Mode
6	PA3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PA2MD1	0	R/W	PA2 Mode
4	PA2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PA1MD1	0	R/W	PA1 Mode
2	PA1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PA0MD1	0	R/W	PA0 Mode
0	PA0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.2 Port B Control Register (PBCR)

PBCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PB7MD[1:0]		PB6MD[1:0]		PB5MD[1:0]		PB4MD[1:0]		PB3MD[1:0]		PB2MD[1:0]		PB1MD[1:0]		PB0MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PB7MD1	0	R/W	PB7 Mode
14	PB7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PB6MD1	0	R/W	PB6 Mode
12	PB6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PB5MD1	0	R/W	PB5 Mode
10	PB5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
9	PB4MD1	0	R/W	PB4 Mode
8	PB4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PB3MD1	0	R/W	PB3 Mode
6	PB3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PB2MD1	0	R/W	PB2 Mode
4	PB2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PB1MD1	0	R/W	PB1 Mode
2	PB1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PB0MD1	0	R/W	PB0 Mode
0	PB0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.3 Port C Control Register (PCCR)

PCCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PC7MD[1:0]		PC6MD[1:0]		PC5MD[1:0]		PC4MD[1:0]		PC3MD[1:0]		PC2MD[1:0]		PC1MD[1:0]		PC0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PC7MD1	1	R/W	PC7 Mode
14	PC7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PC6MD1	1	R/W	PC6 Mode
12	PC6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PC5MD1	1	R/W	PC5 Mode
10	PC5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PC4MD1	1	R/W	PC4 Mode
8	PC4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PC3MD1	1	R/W	PC3 Mode
6	PC3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PC2MD1	1	R/W	PC2 Mode
4	PC2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PC1MD1	1	R/W	PC1 Mode
2	PC1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PC0MD1	1	R/W	PC0 Mode
0	PC0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.4 Port D Control Register (PDCR)

PDCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PD7MD[1:0]		PD6MD[1:0]		PD5MD[1:0]		PD4MD[1:0]		PD3MD[1:0]		PD2MD[1:0]		PD1MD[1:0]		PD0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PD7MD1	1	R/W	PD7 Mode
14	PD7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PD6MD1	1	R/W	PD6 Mode
12	PD6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PD5MD1	1	R/W	PD5 Mode
10	PD5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
9	PD4MD1	1	R/W	PD4 Mode
8	PD4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PD3MD1	1	R/W	PD3 Mode
6	PD3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PD2MD1	1	R/W	PD2 Mode
4	PD2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PD1MD1	1	R/W	PD1 Mode
2	PD1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PD0MD1	1	R/W	PD0 Mode
0	PD0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.5 Port E Control Register (PECR)

PECR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PE7MD[1:0]		PE6MD[1:0]		PE5MD[1:0]		PE4MD[1:0]		PE3MD[1:0]		PE2MD[1:0]		PE1MD[1:0]		PE0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PE7MD1	1	R/W	PE7 Mode
14	PE7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				01: Port input (Pull-up MOS: Off)
13	PE6MD1	1	R/W	PE6 Mode
12	PE6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				01: Port input (Pull-up MOS: Off)
11	PE5MD1	1	R/W	PE5 Mode
10	PE5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				01: Port input (Pull-down MOS: Off)
9	PE4MD1	1	R/W	PE4 Mode
8	PE4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				01: Port input (Pull-down MOS: Off)
7	PE3MD1	1	R/W	PE3 Mode
6	PE3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				01: Port input (Pull-down MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PE2MD1	1	R/W	PE2 Mode
4	PE2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 01: Port input (Pull-down MOS: Off)
3	PE1MD1	1	R/W	PE1 Mode
2	PE1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PE0MD1	1	R/W	PE0 Mode
0	PE0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.6 Port F Control Register (PFCR)

PFCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PF7MD[1:0]		PF6MD[1:0]		PF5MD[1:0]		PF4MD[1:0]		PF3MD[1:0]		PF2MD[1:0]		PF1MD[1:0]		PF0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PF7MD1	1	R/W	PF7 Mode
14	PF7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PF6MD1	1	R/W	PF6 Mode
12	PF6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PF5MD1	1	R/W	PF5 Mode
10	PF5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PF4MD1	1	R/W	PF4 Mode
8	PF4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PF3MD1	1	R/W	PF3 Mode
6	PF3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PF2MD1	1	R/W	PF2 Mode
4	PF2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PF1MD1	1	R/W	PF1 Mode
2	PF1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 01: Port input (Pull-down MOS: Off)
1	PF0MD1	1	R/W	PF0 Mode
0	PF0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 01: Port input (Pull-down MOS: Off)

48.2.7 Port G Control Register (PGCR)

PGCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PG5MD[1:0]		PG4MD[1:0]		PG3MD[1:0]		PG2MD[1:0]		PG1MD[1:0]		PG0MD[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	1	0	1	0	1	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	PG5MD1	0	R/W	PG5 Mode
10	PG5MD0	1	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
9	PG4MD1	0	R/W	PG4 Mode
8	PG4MD0	1	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
7	PG3MD1	0	R/W	PG3 Mode
6	PG3MD0	1	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
5	PG2MD1	0	R/W	PG2 Mode
4	PG2MD0	1	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
3	PG1MD1	0	R/W	PG1 Mode
2	PG1MD0	1	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
1	PG0MD1	0	R/W	PG0 Mode
0	PG0MD0	1	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting

48.2.8 Port H Control Register (PHCR)

PHCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH7MD[1:0]		PH6MD[1:0]		PH5MD[1:0]		PH4MD[1:0]		PH3MD[1:0]		PH2MD[1:0]		PH1MD[1:0]		PH0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PH7MD1	1	R/W	PH7 Mode
14	PH7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PH6MD1	1	R/W	PH6 Mode
12	PH6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PH5MD1	1	R/W	PH5 Mode
10	PH5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PH4MD1	1	R/W	PH4 Mode
8	PH4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PH3MD1	1	R/W	PH3 Mode
6	PH3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PH2MD1	1	R/W	PH2 Mode
4	PH2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PH1MD1	1	R/W	PH1 Mode
2	PH1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PH0MD1	1	R/W	PH0 Mode
0	PH0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.9 Port J Control Register (PJCR)

PJCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PJ7MD[1:0]		PJ6MD[1:0]		PJ5MD[1:0]		—	—	PJ3MD[1:0]		PJ2MD[1:0]		PJ1MD[1:0]		PJ0MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PJ7MD1	0	R/W	PJ7 Mode
14	PJ7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
13	PJ6MD1	0	R/W	PJ6 Mode
12	PJ6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
11	PJ5MD1	0	R/W	PJ5 Mode
10	PJ5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 1*: Prohibited setting
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	PJ3MD1	0	R/W	PJ3 Mode
6	PJ3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
5	PJ2MD1	0	R/W	PJ2 Mode
4	PJ2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
3	PJ1MD1	0	R/W	PJ1 Mode
2	PJ1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PJ0MD1	0	R/W	PJ0 Mode
0	PJ0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.10 Port K Control Register (PKCR)

PKCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PK7MD[1:0]		PK6MD[1:0]		PK5MD[1:0]		PK4MD[1:0]		PK3MD[1:0]		PK2MD[1:0]		PK1MD[1:0]		PK0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PK7MD1	1	R/W	PK7 Mode
14	PK7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PK6MD1	1	R/W	PK6 Mode
12	PK6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PK5MD1	1	R/W	PK5 Mode
10	PK5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off
9	PK4MD1	1	R/W	PK4 Mode
8	PK4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PK3MD1	1	R/W	PK3 Mode
6	PK3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PK2MD1	1	R/W	PK2 Mode
4	PK2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PK1MD1	1	R/W	PK1 Mode
2	PK1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PK0MD1	1	R/W	PK0 Mode
0	PK0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.11 Port L Control Register (PLCR)

PLCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PL7MD[1:0]		PL6MD[1:0]		PL5MD[1:0]		PL4MD[1:0]		PL3MD[1:0]		PL2MD[1:0]		PL1MD[1:0]		PL0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PL7MD1	1	R/W	PL7 Mode
14	PL7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PL6MD1	1	R/W	PL6 Mode
12	PL6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PL5MD1	1	R/W	PL5 Mode
10	PL5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PL4MD1	1	R/W	PL4 Mode
8	PL4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PL3MD1	1	R/W	PL3 Mode
6	PL3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PL2MD1	1	R/W	PL2 Mode
4	PL2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PL1MD1	1	R/W	PL1 Mode
2	PL1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PL0MD1	1	R/W	PL0 Mode
0	PL0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.12 Port M Control Register (PMCR)

PMCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PM7MD[1:0]		PM6MD[1:0]		PM5MD[1:0]		PM4MD[1:0]		PM3MD[1:0]		PM2MD[1:0]		PM1MD[1:0]		PM0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PM7MD1	1	R/W	PM7 Mode
14	PM7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PM6MD1	1	R/W	PM6 Mode
12	PM6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PM5MD1	1	R/W	PM5 Mode
10	PM5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PM4MD1	1	R/W	PM4 Mode
8	PM4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PM3MD1	1	R/W	PM3 Mode
6	PM3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PM2MD1	1	R/W	PM2 Mode
4	PM2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
3	PM1MD1	1	R/W	PM1 Mode
2	PM1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PM0MD1	1	R/W	PM0 Mode
0	PM0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.13 Port N Control Register (PNCR)

PNCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PN7MD[1:0]		PN6MD[1:0]		PN5MD[1:0]		PN4MD[1:0]		PN3MD[1:0]		PN2MD[1:0]		PN1MD[1:0]		PN0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PN7MD1	1	R/W	PN7 Mode
14	PN7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PN6MD1	1	R/W	PN6 Mode
12	PN6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PN5MD1	1	R/W	PN5 Mode
10	PN5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
9	PN4MD1	1	R/W	PN4 Mode
8	PN4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PN3MD1	1	R/W	PN3 Mode
6	PN3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PN2MD1	1	R/W	PN2 Mode
4	PN2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PN1MD1	1	R/W	PN1 Mode
2	PN1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PN0MD1	1	R/W	PN0 Mode
0	PN0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.14 Port Q Control Register (PQCR)

PQCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PQ7MD[1:0]		PQ6MD[1:0]		PQ5MD[1:0]		PQ4MD[1:0]		PQ3MD[1:0]		PQ2MD[1:0]		PQ1MD[1:0]		PQ0MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PQ7MD1	0	R/W	PQ7 Mode
14	PQ7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
13	PQ6MD1	0	R/W	PQ6 Mode
12	PQ6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
11	PQ5MD1	0	R/W	PQ5 Mode
10	PQ5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
9	PQ4MD1	0	R/W	PQ4 Mode
8	PQ4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
7	PQ3MD1	0	R/W	PQ3 Mode
6	PQ3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PQ2MD1	0	R/W	PQ2 Mode
4	PQ2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
3	PQ1MD1	0	R/W	PQ1 Mode
2	PQ1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
1	PQ0MD1	0	R/W	PQ0 Mode
0	PQ0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)

48.2.15 Port R Control Register (PRCR)

PRCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PR7MD[1:0]		PR6MD[1:0]		PR5MD[1:0]		PR4MD[1:0]		PR3MD[1:0]		PR2MD[1:0]		PR1MD[1:0]		PR0MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PR7MD1	0	R/W	PR7 Mode
14	PR7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PR6MD1	0	R/W	PR6 Mode
12	PR6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PR5MD1	0	R/W	PR5 Mode
10	PR5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PR4MD1	0	R/W	PR4 Mode
8	PR4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PR3MD1	0	R/W	PR3 Mode
6	PR3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Setting prohibited
				10: Port input (Pull-up MOS: Off)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PR2MD1	0	R/W	PR2 Mode
4	PR2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Setting prohibited
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PR1MD1	0	R/W	PR1 Mode
2	PR1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PR0MD1	0	R/W	PR0 Mode
0	PR0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.16 Port S Control Register (PSCR)

PSCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PS6MD[1:0]		PS5MD[1:0]		PS4MD[1:0]		PS3MD[1:0]		PS2MD[1:0]		PS1MD[1:0]		PS0MD[1:0]	
Initial value:	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	PS6MD1	1	R/W	PS6 Mode
12	PS6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PS5MD1	1	R/W	PS5 Mode
10	PS5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
9	PS4MD1	1	R/W	PS4 Mode
8	PS4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PS3MD1	1	R/W	PS3 Mode
6	PS3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PS2MD1	1	R/W	PS2 Mode
4	PS2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PS1MD1	1	R/W	PS1 Mode
2	PS1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PS0MD1	1	R/W	PS0 Mode
0	PS0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.17 Port T Control Register (PTCR)

PTCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PT7MD[1:0]		PT6MD[1:0]		PT5MD[1:0]		PT4MD[1:0]		PT3MD[1:0]		PT2MD[1:0]		PT1MD[1:0]		PT0MD[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PT7MD1	0	R/W	PT7 Mode
14	PT7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
13	PT6MD1	0	R/W	PT6 Mode
12	PT6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
11	PT5MD1	0	R/W	PT5 Mode
10	PT5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
9	PT4MD1	0	R/W	PT4 Mode
8	PT4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
7	PT3MD1	0	R/W	PT3 Mode
6	PT3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PT2MD1	0	R/W	PT2 Mode
4	PT2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
3	PT1MD1	0	R/W	PT1 Mode
2	PT1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
1	PT0MD1	0	R/W	PT0 Mode
0	PT0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)

48.2.18 Port U Control Register (PUCR)

PUCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PU7MD[1:0]		PU6MD[1:0]		PU5MD[1:0]		PU4MD[1:0]		PU3MD[1:0]		PU2MD[1:0]		PU1MD[1:0]		PU0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PU7MD1	1	R/W	PU7 Mode
14	PU7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PU6MD1	1	R/W	PU6 Mode
12	PU6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PU5MD1	1	R/W	PU5 Mode
10	PU5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PU4MD1	1	R/W	PU4 Mode
8	PU4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PU3MD1	1	R/W	PU3 Mode
6	PU3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PU2MD1	1	R/W	PU2 Mode
4	PU2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PU1MD1	1	R/W	PU1 Mode
2	PU1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PU0MD1	1	R/W	PU0 Mode
0	PU0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.19 Port V Control Register (PVCR)

PVCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PV7MD[1:0]		PV6MD[1:0]		PV5MD[1:0]		PV4MD[1:0]		PV3MD[1:0]		PV2MD[1:0]		PV1MD[1:0]		PV0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PV7MD1	1	R/W	PV7 Mode
14	PV7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PV6MD1	1	R/W	PV6 Mode
12	PV6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PV5MD1	1	R/W	PV5 Mode
10	PV5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
9	PV4MD1	1	R/W	PV4 Mode
8	PV4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
7	PV3MD1	1	R/W	PV3 Mode
6	PV3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PV2MD1	1	R/W	PV2 Mode
4	PV2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PV1MD1	1	R/W	PV1 Mode
2	PV1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PV0MD1	1	R/W	PV0 Mode
0	PV0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.20 Port W Control Register (PWCR)

PWCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PW7MD[1:0]		PW6MD[1:0]		PW5MD[1:0]		PW4MD[1:0]		PW3MD[1:0]		PW2MD[1:0]		PW1MD[1:0]		PW0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PW7MD1	1	R/W	PW7 Mode
14	PW7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PW6MD1	1	R/W	PW6 Mode
12	PW6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PW5MD1	1	R/W	PW5 Mode
10	PW5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
9	PW4MD1	1	R/W	PW4 Mode
8	PW4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PW3MD1	1	R/W	PW3 Mode
6	PW3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PW2MD1	1	R/W	PW2 Mode
4	PW2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PW1MD1	1	R/W	PW1 Mode
2	PW1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PW0MD1	0	R/W	PW0 Mode
0	PW0MD0	1	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.21 Port X Control Register (PXCR)

PXCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PX7MD[1:0]		PX6MD[1:0]		PX5MD[1:0]		PX4MD[1:0]		PX3MD[1:0]		PX2MD[1:0]		PX1MD[1:0]		PX0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PX7MD1	1	R/W	PX7 Mode
14	PX7MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
13	PX6MD1	1	R/W	PX6 Mode
12	PX6MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
11	PX5MD1	1	R/W	PX5 Mode
10	PX5MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
9	PX4MD1	1	R/W	PX4 Mode
8	PX4MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)
7	PX3MD1	1	R/W	PX3 Mode
6	PX3MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-down MOS: On)
				11: Port input (Pull-down MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PX2MD1	1	R/W	PX2 Mode
4	PX2MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-down MOS: On) 11: Port input (Pull-down MOS: Off)
3	PX1MD1	1	R/W	PX1 Mode
2	PX1MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
1	PX0MD1	1	R/W	PX0 Mode
0	PX0MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

48.2.22 Port Y Control Register (PYCR)

PYCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PY7MD[1:0]		PY6MD[1:0]		PY5MD[1:0]		PY4MD[1:0]		PY3MD[1:0]		PY2MD[1:0]		PY1MD[1:0]		PY0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PY7MD1	1	R/W	PY7 Mode
14	PY7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PY6MD1	1	R/W	PY6 Mode
12	PY6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PY5MD1	1	R/W	PY5 Mode
10	PY5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
9	PY4MD1	1	R/W	PY4 Mode
8	PY4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PY3MD1	1	R/W	PY3 Mode
6	PY3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PY2MD1	1	R/W	PY2 Mode
4	PY2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PY1MD1	1	R/W	PY1 Mode
2	PY1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PY0MD1	0	R/W	PY0 Mode
0	PY0MD0	1	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.23 Port Z Control Register (PZCR)

PZCR is a 16-bit readable/writable register that selects the pin function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PZ7MD[1:0]		PZ6MD[1:0]		PZ5MD[1:0]		PZ4MD[1:0]		PZ3MD[1:0]		PZ2MD[1:0]		PZ1MD[1:0]		PZ0MD[1:0]	
Initial value:	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PZ7MD1	1	R/W	PZ7 Mode
14	PZ7MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
13	PZ6MD1	1	R/W	PZ6 Mode
12	PZ6MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
11	PZ5MD1	1	R/W	PZ5 Mode
10	PZ5MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
9	PZ4MD1	1	R/W	PZ4 Mode
8	PZ4MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)
7	PZ3MD1	1	R/W	PZ3 Mode
6	PZ3MD0	0	R/W	00: Other functions (See table 48.1.) 01: Port output 10: Port input (Pull-up MOS: On) 11: Port input (Pull-up MOS: Off)

Bit	Bit Name	Initial Value	R/W	Description
5	PZ2MD1	1	R/W	PZ2 Mode
4	PZ2MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
3	PZ1MD1	1	R/W	PZ1 Mode
2	PZ1MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)
1	PZ0MD1	1	R/W	PZ0 Mode
0	PZ0MD0	0	R/W	00: Other functions (See table 48.1.)
				01: Port output
				10: Port input (Pull-up MOS: On)
				11: Port input (Pull-up MOS: Off)

48.2.24 Port A Data Register (PADR)

PADR is a register that stores input/output data for pins PTA7 to PTA0. The PA7DT to PA0DT bits correspond to the pins PTA7 to PTA0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PA7DT	PA6DT	PA5DT	PA4DT	PA3DT	PA2DT	PA1DT	PA0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PA7DT	0	R/W	PTA7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PA6DT	0	R/W	PTA6	
5	PA5DT	0	R/W	PTA5	
4	PA4DT	0	R/W	PTA4	
3	PA3DT	0	R/W	PTA3	
2	PA2DT	0	R/W	PTA2	
1	PA1DT	0	R/W	PTA1	
0	PA0DT	0	R/W	PTA0	

48.2.25 Port B Data Register (PBDR)

PBDR is a register that stores input/output data for pins PTB7 to PTB0. The PB7DT to PB0DT bits correspond to the pins PTB7 to PTB0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PB7DT	PB6DT	PB5DT	PB4DT	PB3DT	PB2DT	PB1DT	PB0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PB7DT	0	R/W	PTB7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PB6DT	0	R/W	PTB6	
5	PB5DT	0	R/W	PTB5	
4	PB4DT	0	R/W	PTB4	
3	PB3DT	0	R/W	PTB3	
2	PB2DT	0	R/W	PTB2	
1	PB1DT	0	R/W	PTB1	
0	PB0DT	0	R/W	PTB0	

48.2.26 Port C Data Register (PCDR)

PCDR is a register that stores data for pins PTC7 to PTC0. The PC7DT to PC0DT bits correspond to the pins PTC7 to PTC0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PC7DT	PC6DT	PC5DT	PC4DT	PC3DT	PC2DT	PC1DT	PC0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PC7DT	0	R/W	PTC7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PC6DT	0	R/W	PTC6	
5	PC5DT	0	R/W	PTC5	
4	PC4DT	0	R/W	PTC4	
3	PC3DT	0	R/W	PTC3	
2	PC2DT	0	R/W	PTC2	
1	PC1DT	0	R/W	PTC1	
0	PC0DT	0	R/W	PTC0	

48.2.27 Port D Data Register (PDDR)

PDDR is a register that stores input/output data for pins PTD7 to PTD0. The PD7DT to PD0DT bits correspond to the pins PTD7 to PTD0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PD7DT	PD6DT	PD5DT	PD4DT	PD3DT	PD2DT	PD1DT	PD0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PD7DT	0	R/W	PTD7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PD6DT	0	R/W	PTD6	
5	PD5DT	0	R/W	PTD5	
4	PD4DT	0	R/W	PTD4	
3	PD3DT	0	R/W	PTD3	
2	PD2DT	0	R/W	PTD2	
1	PD1DT	0	R/W	PTD1	
0	PD0DT	0	R/W	PTD0	

48.2.28 Port E Data Register (PEDR)

PEDR is a register that stores input/output data for pins PTE7 to PTE0. The PE7DT to PE0DT bits correspond to the pins PTE7 to PTE0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PE7DT	PE6DT	PE5DT	PE4DT	PE3DT	PE2DT	PE1DT	PE0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PE7DT	0	R/W	PTE7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PE6DT	0	R/W	PTE6	
5	PE5DT	0	R/W	PTE5	
4	PE4DT	0	R/W	PTE4	
3	PE3DT	0	R/W	PTE3	
2	PE2DT	0	R/W	PTE2	
1	PE1DT	0	R/W	PTE1	
0	PE0DT	0	R/W	PTE0	

48.2.29 Port F Data Register (PFDR)

PFDR is a register that stores input/output data for pins PTF7 to PTF0. The PF7DT to PF0DT bits correspond to the pins PTF7 to PTF0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PF7DT	PF6DT	PF5DT	PF4DT	PF3DT	PF2DT	PF1DT	PF0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PF7DT	0	R/W	PTF7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PF6DT	0	R/W	PTF6	
5	PF5DT	0	R/W	PTF5	
4	PF4DT	0	R/W	PTF4	
3	PF3DT	0	R/W	PTF3	
2	PF2DT	0	R/W	PTF2	
1	PF1DT	0	R/W	PTF1	
0	PF0DT	0	R/W	PTF0	

48.2.30 Port G Data Register (PGDR)

PGDR is a register that stores input/output data for pins PTG5 to PTG0. The PG5DT to PG0DT bits correspond to the pins PTG5 to PTG0, respectively.

Bit :	7	6	5	4	3	2	1	0
	—	—	PG5DT	PG4DT	PG3DT	PG2DT	PG1DT	PG0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7, 6	—	All 0	R	—	Reserved These bits are always read as 0. The write value should always be 0.
5	PG5DT	0	R/W	PTG5	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
4	PG4DT	0	R/W	PTG4	
3	PG3DT	0	R/W	PTG3	
2	PG2DT	0	R/W	PTG2	
1	PG1DT	0	R/W	PTG1	
0	PG0DT	0	R/W	PTG0	

48.2.31 Port H Data Register (PHDR)

PHDR is a register that stores input/output data for pins PTH7 to PTH0. The PH7DT to PH0DT bits correspond to the pins PTH7 to PTH0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PH7DT	PH6DT	PH5DT	PH4DT	PH3DT	PH2DT	PH1DT	PH0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PH7DT	0	R/W	PTH7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PH6DT	0	R/W	PTH6	
5	PH5DT	0	R/W	PTH5	
4	PH4DT	0	R/W	PTH4	
3	PH3DT	0	R/W	PTH3	
2	PH2DT	0	R/W	PTH2	
1	PH1DT	0	R/W	PTH1	
0	PH0DT	0	R/W	PTH0	

48.2.32 Port J Data Register (PJDR)

PJDR is a register that stores input/output data for pins PTJ7 to PTJ5 and PTJ3 to PTJ0. The PJ7DT to PJ5DT and PJ3DT to PJ0DT bits correspond to the pins PTJ7 to PTJ5 and PTJ3 to PTJ0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PJ7DT	PJ6DT	PJ5DT	—	PJ3DT	PJ2DT	PJ1DT	PJ0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PJ7DT	0	R/W	PTJ7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PJ6DT	0	R/W	PTJ6	
5	PJ5DT	0	R/W	PTJ5	
4	—	0	R	—	Reserved This bit is always read as 0. The write value should always be 0.
3	PJ3DT	0	R/W	PTJ3	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
2	PJ2DT	0	R/W	PTJ2	
1	PJ1DT	0	R/W	PTJ1	
0	PJ0DT	0	R/W	PTJ0	

48.2.33 Port K Data Register (PKDR)

PKDR is a register that stores input/output data for pins PTK7 to PTK0. The PK7DT to PK0DT bits correspond to the pins PTK7 to PTK0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PK7DT	PK6DT	PK5DT	PK4DT	PK3DT	PK2DT	PK1DT	PK0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PK7DT	0	R/W	PTK7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PK6DT	0	R/W	PTK6	
5	PK5DT	0	R/W	PTK5	
4	PK4DT	0	R/W	PTK4	
3	PK3DT	0	R/W	PTK3	
2	PK2DT	0	R/W	PTK2	
1	PK1DT	0	R/W	PTK1	
0	PK0DT	0	R/W	PTK0	

48.2.34 Port L Data Register (PLDR)

PLDR is a register that stores input/output data for pins PTL7 to PTL0. The PL7DT to PL0DT bits correspond to the pins PTL7 to PTL0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PL7DT	PL6DT	PL5DT	PL4DT	PL3DT	PL2DT	PL1DT	PL0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PL7DT	0	R/W	PTL7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PL6DT	0	R/W	PTL6	
5	PL5DT	0	R/W	PTL5	
4	PL4DT	0	R/W	PTL4	
3	PL3DT	0	R/W	PTL3	
2	PL2DT	0	R/W	PTL2	
1	PL1DT	0	R/W	PTL1	
0	PL0DT	0	R/W	PTL0	

48.2.35 Port M Data Register (PMDR)

PMDR is a register that stores input/output data for pins PTM7 to PTM0. The PM7DT to PM0DT bits correspond to the pins PTM7 to PTM0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PM7DT	PM6DT	PM5DT	PM4DT	PM3DT	PM2DT	PM1DT	PM0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PM7DT	0	R/W	PTM7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PM6DT	0	R/W	PTM6	
5	PM5DT	0	R/W	PTM5	
4	PM4DT	0	R/W	PTM4	
3	PM3DT	0	R/W	PTM3	
2	PM2DT	0	R/W	PTM2	
1	PM1DT	0	R/W	PTM1	
0	PM0DT	0	R/W	PTM0	

48.2.36 Port N Data Register (PNDR)

PNDR is a register that stores input/output data for pins PTN7 to PTN0. The PN7DT to PN0DT bits correspond to the pins PTN7 to PTN0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PN7DT	PN6DT	PN5DT	PN4DT	PN3DT	PN2DT	PN1DT	PN0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PN7DT	0	R/W	PTN7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PN6DT	0	R/W	PTN6	
5	PN5DT	0	R/W	PTN5	
4	PN4DT	0	R/W	PTN4	
3	PN3DT	0	R/W	PTN3	
2	PN2DT	0	R/W	PTN2	
1	PN1DT	0	R/W	PTN1	
0	PN0DT	0	R/W	PTN0	

48.2.37 Port Q Data Register (PQDR)

PQDR is a register that stores input/output data for pins PTQ7 to PTQ0. The PQ7DT to PQ0DT bits correspond to the pins PTQ7 to PTQ0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PQ7DT	PQ6DT	PQ5DT	PQ4DT	PQ3DT	PQ2DT	PQ1DT	PQ0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PQ7DT	0	R/W	PTQ7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PQ6DT	0	R/W	PTQ6	
5	PQ5DT	0	R/W	PTQ5	
4	PQ4DT	0	R/W	PTQ4	
3	PQ3DT	0	R/W	PTQ3	
2	PQ2DT	0	R/W	PTQ2	
1	PQ1DT	0	R/W	PTQ1	
0	PQ0DT	0	R/W	PTQ0	

48.2.38 Port R Data Register (PRDR)

PRDR is a register that stores input/output data for pins PTR7 to PTR0. The PR7DT to PR0DT bits correspond to the pins PTR7 to PTR0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PR7DT	PR6DT	PR5DT	PR4DT	PR3DT	PR2DT	PR1DT	PR0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PR7DT	0	R/W	PTR7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PR6DT	0	R/W	PTR6	
5	PR5DT	0	R/W	PTR5	
4	PR4DT	0	R/W	PTR4	
3	PR3DT	0	R	PTR3	
2	PR2DT	0	R/W	PTR2	
1	PR1DT	0	R/W	PTR1	
0	PR0DT	0	R/W	PTR0	

48.2.39 Port S Data Register (PSDR)

PSDR is a register that stores input/output data for pins PTS6 to PTS0. The PS6DT to PS0DT bits correspond to the pins PTS6 to PTS0, respectively.

Bit :	7	6	5	4	3	2	1	0
	—	PS6DT	PS5DT	PS4DT	PS3DT	PS2DT	PS1DT	PS0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	—	0	R	—	Reserved These bits are always read as 0. The write value should always be 0.
6	PS6DT	0	R/W	PTS6	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
5	PS5DT	0	R/W	PTS5	
4	PS4DT	0	R/W	PTS4	
3	PS3DT	0	R/W	PTS3	
2	PS2DT	0	R/W	PTS2	
1	PS1DT	0	R/W	PTS1	
0	PS0DT	0	R/W	PTS0	

48.2.40 Port T Data Register (PTDR)

PTDR is a register that stores input/output data for pins PTT5 to PTT0. The PT5DT to PT0DT bits correspond to the pins PTT5 to PTT0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PT7DT	PT6DT	PT5DT	PT4DT	PT3DT	PT2DT	PT1DT	PT0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PT7DT	0	R/W	PTT7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PT6DT	0	R/W	PTT6	
5	PT5DT	0	R/W	PTT5	
4	PT4DT	0	R/W	PTT4	
3	PT3DT	0	R/W	PTT3	
2	PT2DT	0	R/W	PTT2	
1	PT1DT	0	R/W	PTT1	
0	PT0DT	0	R/W	PTT0	

48.2.41 Port U Data Register (PUDR)

PUDR is a register that stores input/output data for pins PTU7 to PTU0. The PU7DT to PU0DT bits correspond to the pins PTU7 to PTU0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PU7DT	PU6DT	PU5DT	PU4DT	PU3DT	PU2DT	PU1DT	PU0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PU7DT	0	R/W	PTU7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PU6DT	0	R/W	PTU6	
5	PU5DT	0	R/W	PTU5	
4	PU4DT	0	R/W	PTU4	
3	PU3DT	0	R/W	PTU3	
2	PU2DT	0	R/W	PTU2	
1	PU1DT	0	R/W	PTU1	
0	PU0DT	0	R/W	PTU0	

48.2.42 Port V Data Register (PVDR)

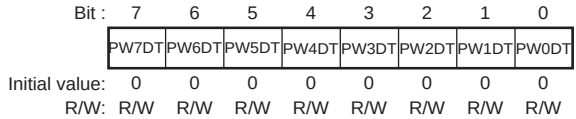
PVDR is a register that stores input/output data for pins PTV7 to PTV0. The PV7DT to PV0DT bits correspond to the pins PTV7 to PTV0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PV7DT	PV6DT	PV5DT	PV4DT	PV3DT	PV2DT	PV1DT	PV0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PV7DT	0	R/W	PTV7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PV6DT	0	R/W	PTV6	
5	PV5DT	0	R/W	PTV5	
4	PV4DT	0	R/W	PTV4	
3	PV3DT	0	R/W	PTV3	
2	PV2DT	0	R/W	PTV2	
1	PV1DT	0	R/W	PTV1	
0	PV0DT	0	R/W	PTV0	

48.2.43 Port W Data Register (PWDR)

PWDR is a register that stores input/output data for pins PTW7 to PTW0. The PW7DT to PW0DT bits correspond to the pins PTW7 to PTW0, respectively.



Bit :	7	6	5	4	3	2	1	0
	PW7DT	PW6DT	PW5DT	PW4DT	PW3DT	PW2DT	PW1DT	PW0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PW7DT	0	R/W	PTW7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PW6DT	0	R/W	PTW6	
5	PW5DT	0	R/W	PTW5	
4	PW4DT	0	R/W	PTW4	
3	PW3DT	0	R/W	PTW3	
2	PW2DT	0	R/W	PTW2	
1	PW1DT	0	R/W	PTW1	
0	PW0DT	0	R/W	PTW0	

48.2.44 Port X Data Register (PXDR)

PXDR is a register that stores input/output data for pins PTX7 to PTX0. The PX7DT to PX0DT bits correspond to the pins PTX7 to PTX0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PX7DT	PX6DT	PX5DT	PX4DT	PX3DT	PX2DT	PX1DT	PX0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PX7DT	0	R/W	PTX7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PX6DT	0	R/W	PTX6	
5	PX5DT	0	R/W	PTX5	
4	PX4DT	0	R/W	PTX4	
3	PX3DT	0	R/W	PTX3	
2	PX2DT	0	R/W	PTX2	
1	PX1DT	0	R/W	PTX1	
0	PX0DT	0	R/W	PTX0	

48.2.45 Port Y Data Register (PYDR)

PYDR is a register that stores input/output data for pins PTY7 to PTY0. The PY7DT to PY0DT bits correspond to the pins PTY7 to PTY0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PY7DT	PY6DT	PY5DT	PY4DT	PY3DT	PY2DT	PY1DT	PY0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PY7DT	0	R/W	PTY7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PY6DT	0	R/W	PTY6	
5	PY5DT	0	R/W	PTY5	
4	PY4DT	0	R/W	PTY4	
3	PY3DT	0	R/W	PTY3	
2	PY2DT	0	R/W	PTY2	
1	PY1DT	0	R/W	PTY1	
0	PY0DT	0	R/W	PTY0	

48.2.46 Port Z Data Register (PZDR)

PZDR is a register that stores input/output data for pins PTZ7 to PTZ0. The PZ7DT to PZ0DT bits correspond to the pins PTZ7 to PTZ0, respectively.

Bit :	7	6	5	4	3	2	1	0
	PZ7DT	PZ6DT	PZ5DT	PZ4DT	PZ3DT	PZ2DT	PZ1DT	PZ0DT
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Port Name	Description
7	PZ7DT	0	R/W	PTZ7	In output state, the value in this register is read. The write value is output from the pin. In input state, the pin level is read. The value is written to this register, but does not affect the pin state. In function other than general port, the value in this register is read. The value is written to this register, but does not affect the pin state.
6	PZ6DT	0	R/W	PTZ6	
5	PZ5DT	0	R/W	PTZ5	
4	PZ4DT	0	R/W	PTZ4	
3	PZ3DT	0	R/W	PTZ3	
2	PZ2DT	0	R/W	PTZ2	
1	PZ1DT	0	R/W	PTZ1	
0	PZ0DT	0	R/W	PTZ0	

48.2.47 Pin Select Register A (PSELA)

PSELA is a 16-bit readable/writable register that selects the functions of pins on which two or more pin multiplex functions are multiplexed.

To use one of the pin multiplex functions of the pins on which two or more pin multiplex functions are multiplexed, set the corresponding bit in PSELA, and then set the corresponding bit in the port control register to pin multiplex function.

Setting example: To use the KEYSC function in the PTA7 to PTA0, BSC and KEYSC pins

1. Write B'01 to the PSA15 and PSA14 bits in PSELA.
2. Set the PAnMD[1:0] (n = 7 to 0) bits in the port A control register (PACR) to B'00 (pin multiplex function).
3. Set the HIZA15 bit in the I/O buffer Hi-Z control register A (HIZCRA) to B'0 (the I/O buffer operates normally).

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSA15	PSA14	PSA13	PSA12	—	PSA10	PSA9	PSA8	PSA7	PSA6	PSA5	—	PSA3	PSA2	PSA1	PSA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PSA15	0	R/W	PTA7 to PTA0, PTB2, PTB1, and PTB0 Multiplex Function
14	PSA14	0	R/W	Select 00: Selects BSC (D26 to D16) 01: Selects KEYSC (KEYOUT2 to KEYOUT0, KEYIN4 to KEYIN0, KEYOUT5/IN5, KEYOUT4/IN6, and KEYOUT3) 10: Selects ATAPI (IDED15 to IDED18, $\overline{\text{IDEIORD}}$, $\overline{\text{IDEIOWR}}$, and IDEINT) 11: Setting prohibited
13	PSA13	0	R/W	PTW3 to PTW0 Multiplex Function Select 0: Selects MMCIF (MMC_D3 to MMC_D0) 1: Selects SDHI1 (SDHI1D3 to SDHI1D0, SDHI1CMD, and SDHI1CLK)

Bit	Bit Name	Initial Value	R/W	Description
12	PSA12	0	R/W	PTX7, PTX6 Multiplex Function Select 0: Selects DMAC (DACK1, DERQ1) 1: Selects IrDA (IRDA_OUT, $\overline{\text{IRDA\_IN}}$)
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	PSA10	0	R/W	PTS4 Multiplex Function Select 00: Selects VIO (VIO1_HD) 01: Selects SCIFA5 (SCIF5_SCK)
9	PSA9	0	R/W	PTS3, PTS2 Multiplex Function Select 0: Selects VIO (VIO1_VD, VIO1_CLK) 1: Selects SCIFA5 (SCIF5_RXD, SCIF5_TXD)
8	PSA8	0	R/W	PTJ3 Multiplex Function Select 0: Selects BSC (A25) 1: Selects BSC ($\overline{\text{BS}}$)
7	PSA7	0	R/W	PTM0 Multiple Function Select 0: Selects LCDC ($\overline{\text{LCDRD}}$) 1: Selects SCIF0 (SCIF0_SCK)
6	PSA6	0	R/W	PTF2, PTM1 Multiplex Function Select 0: Selects LCDC (LCDVEPWC, LCDVCPWC) 1: Selects SCIF0 (SCIF0_TXD, SCIF0_RXD)
5	PSA5	0	R/W	PPTR3 Multiplex Function Select 0: Selects BSC ($\overline{\text{IOIS16}}$) 1: Selects LCDC (LCDLCLK)
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	PSA3	0	R/W	PTE3, PTE2 Multiplex Function Select
2	PSA2	0	R/W	00: Selects LCDC (LCDD19, LCDD18) 01: Selects SCIFA4 (SCIF4_RXD, SCIF4_TXD) 10: Setting prohibited 11: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
1	PSA1	0	R/W	PTR1 Multiplex Function Select
0	PSA0	0	R/W	00: Selects BSC ($\overline{WE3}$/$\overline{ICIOWR}$)
				01: TPU (TPUTO3)
				10: TPU (TPUTI3)
				11: Setting prohibited

48.2.48 Pin Select Register B (PSELB)

PSELB is a 16-bit readable/writable register that selects the functions of pins on which two or more other functions are multiplexed.

To use one of the other functions of the pins on which two or more other functions are multiplexed, set the corresponding bit in PSELB, and then set the corresponding bit in the port control register to other function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PSB14	PSB13	PSB12	PSB11	PSB10	PSB9	PSB8	PSB7	PSB6	PSB5	PSB4	PSB3	PSB2	PSB1	PSB0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	PSB14	0	R/W	PTZ3 multiplex Function Select 0: Selects IRQ3 1: Selects SCIF3_TXD
13	PSB13	0	R/W	PTZ4 multiplex Function Select 0: Selects IRQ4 1: Selects SCIF3_RXD
12	PSB12	0	R/W	PTZ5 multiplex Function Select 0: Selects IRQ5 1: Selects SCIF3_SCK
11	PSB11	0	R/W	PTZ6 multiplex Function Select 0: Selects IRQ6 1: Selects $\overline{\text{SCIF3\_RTS}}$
10	PSB10	0	R/W	PTZ7 multiplex Function Select 0: Selects IRQ7 1: Selects $\overline{\text{SCIF3\_CTS}}$
9	PSB9	0	R/W	PTL7, PTL6 multiplex Function Select
8	PSB8	0	R/W	00: DV_D5, DV_D4 01: SCIF3_SCK, SCIF3_RXD 10: RMII_RXD0, RMII_RXD1 11: Prohibit setting

Bit	Bit Name	Initial Value	R/W	Description
7	PSB7	0	R/W	PTK7 to PTK2, PTS1, PTS0 multiplex Function Select
6	PSB6	0	R/W	00: Selects VIO1 01: Selects VIO0 10: Select IDE* (ATAPI) 11: Prohibited setting
5	PSB5	0	R/W	PTN2 multiplex Function Select
4	PSB4	0	R/W	00: Selects DV_HSYNC 01: Selects SCIF2_TXD 10: Prohibited setting 11: Prohibit setting
3	PSB3	0	R/W	PTN1 multiplex Function Select
2	PSB2	0	R/W	00: Selects DV_D7 01: Selects $\overline{\text{SCIF3\_CTS}}$ 10: Selects RMII_RX_ER 11: Prohibit setting
1	PSB1	0	R/W	PTN0 multiplex Function Select
0	PSB0	0	R/W	00: Selects DV_D6 01: Selects $\overline{\text{SCIF3\_RTS}}$ 10: Selects RMII_CRS_DV 11: Prohibit setting

48.2.49 Pin Select Register C (PSELC)

PSELC is a 16-bit readable/writable register that selects the functions of pins on which two or more other functions are multiplexed.

To use one of the other functions of the pins on which two or more other functions are multiplexed, set the corresponding bit in PSELC, and then set the corresponding bit in the port control register to other function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSC15	PSC14	PSC13	PSC12	PSC11	PSC10	PSC9	PSC8	PSC7	PSC6	PSC5	PSC4	—	PSC2	PSC1	PSC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PSC15	0	R/W	PTV5 to PTV2 multiplex Function Select
14	PSC14	0	R/W	00: Selects FSI 01: Selects MSIOF1 10: Prohibited setting 11: Prohibited setting
13	PSC13	0	R/W	PTM7 to PTM4 multiplex Function Select
12	PSC12	0	R/W	00: Selects VOU 01: Selects MSIOF0 10: Prohibited setting 11: Prohibited setting
11	PSC11	0	R/W	PTM3, PTM2 multiplex Function Select
10	PSC10	0	R/W	00: Select DV_D9 and DV_D8 01: Select $\overline{\text{MSIOF0\_SS1}}$ and $\overline{\text{MSIOF0\_SS2}}$ 10: Select MSIOF0_RSCK and MSIOF0_RSYNC 11: Prohibited setting
9	PSC9	0	R/W	PTL2, PTL3, PTL4 multiplex Function Select
8	PSC8	0	R/W	00: Selects DV_D0, DV_D1, DV_D2 01: Selects SCIF1_TXD, SCIF1_RXD, SCIF1_SCK 10: RMII_TXD1, RMII_TXD0, RMII_TX_EN 11: Prohibited setting

Bit	Bit Name	Initial Value	R/W	Description
7	PSC7	0	R/W	PTF1 multiplex Function Select
6	PSC6	0	R/W	00: Selects LCDD23 01: Selects SCIF2_SCK 10: Prohibited setting 11: Prohibited setting
5	PSC5	0	R/W	PTE5, PTF0 multiplex Function Select
4	PSC4	0	R/W	00: Selects LCDC 01: Selects SCIF2 10: Prohibited setting 11: Prohibited setting
3	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	PSC2	0	R/W	PTX3 multiplex Function Select 0: Selects TS0_SDEN 1: Selects MDC
1	PSC1	0	R/W	PTX4 multiplex Function Select 0: Selects TS0_SCK 1: Selects MDIO
0	PSC0	0	R/W	PTX5 multiplex Function Select 0: Selects TS0_SDAT 1: Selects LNKSTA

48.2.50 Pin Select Register D (PSELD)

PSELD is a 16-bit readable/writable register that selects the functions of pins on which two or more other functions are multiplexed.

To use one of the other functions of the pins on which two or more other functions are multiplexed, set the corresponding bit in PSELD, and then set the corresponding bit in the port control register to other function.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSD15	PSD14	PSD13	PSD12	PSD11	PSD10	PSD9	PSD8	PSD7	PSD6	PSD5	PSD4	PSD3	PSD2	PSD1	PSD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PSD15	0	R/W	PTN3 multiplex Function Select
14	PSD14	0		00: Selects DV_VSYNC
				01: Selects SCIF2_RXD
				10: Prohibited setting
				11: Prohibit setting
13	PSD13	0	R/W	PTN4 multiplex Function Select
12	PSD12	0		00: Selects DV_CLK
				01: Selects SCIF2_SCK
				10: Prohibited setting
				11: Prohibit setting
11	PSD11	0	R/W	PTN5 multiplex Function Select
				0: Selects DV_CLKI
				1: Prohibited setting
10	PSD10	0	R/W	PTF6, PTF5, PTF3 multiplexed function select
9	PSD9	0		00: Select LCDDISP, LCDHSYN, LCDDCK
				01: Select LCDRS, $\overline{\text{LCDCS}}$, $\overline{\text{LCDWR}}$
				10: Prohibited setting
				11: Prohibited setting
8	PSD8	0	R/W	PTF7, PTF4 multiplexed function select
				0: Select LCDVSYN, LCDDON
				1: Prohibited setting

Bit	Bit Name	Initial Value	R/W	Description
7 to 6	PSD7	0	R/W	PTV7, PTV6 multiplex Function Select
	PSD6	0	R/W	00: Selects FSI 01: Selects $\overline{\text{MSIOF1\_SS1}}$ and $\overline{\text{MSIOF1\_SS2}}$ 10: Select MSIOF1_RSYNC and MSIOF1_RSCK 11: Setting prohibited
5	PSD5	0	R/W	PTC[7:0], PTD[7:0], PTE[1:0] multiplex Function Select 0: Selects LCDC function 1: Prohibited setting
4	PSD4	0	R/W	PTL1 multiplexed function select 0: Select DV_D15 1: Prohibited setting
3	PSD3	0	R/W	PTE4 multiplex Function Select
2	PSD2	0	R/W	00: Selects LCDD20 01: Selects SCIF4_SCK 10: Prohibited setting 11: Prohibited setting
1	PSD1	0	R/W	PTR0 multiplexed function select
0	PSD0	0	R/W	00: Select $\overline{\text{WE2}}$/$\overline{\text{ICIORD}}$ 01: Select TPUTO2 10: Select IDE (ATAPI) 11: Prohibited setting

48.2.51 Pin Select Register E (PSELE)

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSE15	PSE14	PSE13	PSE12	PSE11	PSE10	PSE9	PSE8	PSE7	PSE6	PSE5	PSE4	PSE3	PSE2	PSE1	PSE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	PSE15	0	R/W	PTB7, PTB6 multiplexed function select
14	PSE14	0	R/W	00: Select D31, D30 (BSC) 01: Select TPU 10: Select IDEA1, IDEA0 (ATAPI) 11: Prohibited setting
13	PSE13	0	R/W	PTW[7:4] multiplex Function Select
12	PSE12	0	R/W	00: Select MMC_D7 to MMC_D4 01: Selects SDHI1 function 10: Selects IDE function (ATAPI) 11: Prohibited setting
11	PSE11	0	R/W	PTB[5:3] multiplexed function select 0: Select D[29:27] (BSC) 1: Select IDE (ATAPI)
10	PSE10	0	R/W	PTV0 multiplexed function select 0: Select FSIIASD function 1: Prohibited setting
9	PSE9	0	R/W	PTS5 multiplex Function Select
8	PSE8	0	R/W	00: Selects VIO1_FLD 01: Selects TPUTI2 10: Select IDE (ATAPI) 11: Prohibited setting
7	PSE7	0	R/W	PTL5 multiplex Function Select
6	PSE6	0	R/W	00: DV_D3 01: SCIF3_TXD 10: RMII_REF_CLK (EtherC) 11: Prohibited setting

Bit	Bit Name	Initial Value	R/W	Description
5	PSE5	0	R/W	PTL0 multiplex Function Select
4	PSE4	0	R/W	00: Selects VOU 01: Selects MSIOF0 10: Prohibited setting 11: Prohibited setting
3	PSE3	0	R/W	PTV1 multiplex Function Select
2	PSE2	0	R/W	00: Selects CLKAUDIOBO 01: Selects MSIOF1_MCK 10: Prohibited setting 11: Prohibited setting
1	PSE1	0	R/W	PTU[5:1] multiplexed function select 0: Select FSI function 1: Prohibited setting
0	PSE0	0	R/W	PTU0 multiplexed function select 0: Select CLKAUDIOAO 1: Prohibited setting

48.2.52 I/O Buffer Hi-Z Control Register A (HIZCRA)

HIZCRA is a 16-bit readable/writable register that controls Hi-Z state of pins for every function unit.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HIZA15	HIZA14	HIZA13	HIZA12	HIZA11	HIZA10	HIZA9	HIZA8	HIZA7	HIZA6	—	—	—	—	HIZA1	HIZA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	HIZA15	0	R/W	Hi-Z Control for PTA Pin (D23 to D16) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
14	HIZA14	0	R/W	Hi-Z Control for PTB7 to PTB3 Pin (D31 to D27, TPUTO1, TPUTO0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
13	HIZA13	0	R/W	Hi-Z Control for PTB2 to PTB0 Pin (D26 to D24, KEYOUT5/IN5, KEYOUT4/IN6, KEYOUT3) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
12	HIZA12	0	R/W	Hi-Z Control for PTC, PTD, PTE1,PTE0 Pin (LCDD0 toLCDD17) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
11	HIZA11	0	R/W	Hi-Z Control for PTE7,PTE6 Pin (FSIMCKA, FSIMCKB) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
10	HIZA10	0	R/W	Hi-Z Control for PTE5 Pin (LCDD21/ SCIF2_TXD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
9	HIZA9	0	R/W	Hi-Z Control for PTE4, PTE3, PTE2 Pin (LCDD20/ SCIF4_SCK,LCDD19/ SCIF4_RXD, LCDD18/ SCIF4_TXD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z

Bit	Bit Name	Initial Value	R/W	Description
8	HIZA8	0	R/W	Hi-Z Control for PTF7 to PTF3 Pin (LCDVSYN, LCDDISP/LCDRS, LCDHSYN/$\overline{\text{LCDCS}}$, LCDDON, LCDDCK/$\overline{\text{LCDWR}}$) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
7	HIZA7	0	R/W	Hi-Z Control for PTF2,PTM1,PTM0 Pin (LCDVEPWC/ SCIF0_TXD, LCDVCPWC/ SCIF0_RXD, $\overline{\text{LCDRD}}$/ SCIF0_SCK) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
6	HIZA6	0	R/W	Hi-Z Control for PTF1 ,PTF0 Pin (LCDD23/ SCIF2_SCK, LCDD22/ SCIF2_RXD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	HIZA1	1	R/W	Hi-Z Control for PTG5 to PTG0 Pin (AUDCK, AUDSYNC, AUDATA3, AUDATA2, AUDATA1, AUDATA0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
0	HIZA0	1	R/W	Hi-Z Control for PTR3 Pin ($\overline{\text{IOIS16}}$/LCDLCLK) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z

48.2.53 I/O Buffer Hi-Z Control Register B (HIZCRB)

HIZCRB is a 16-bit readable/writable register that controls Hi-Z state of pins for every function unit.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HIZB15	HIZB14	HIZB13	HIZB12	HIZB11	HIZB10	HIZB9	HIZB8	HIZB7	HIZB6	HIZB5	HIZB4	HIZB3	HIZB2	HIZB1	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
15	HIZB15	0	R/W	Hi-Z Control for PTH, PTN7,PTN6 Pin (VIO0_VD, VIO0_CLK, VIO0_D7 to D0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
14	HIZB14	0	R/W	Hi-Z Control for PTJ7 to PTJ5 Pin (PDSTATUS, STATUS2, STATUS0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
13	HIZB13	0	R/W	Hi-Z Control for PTJ3to PTJ0 Pin (A22 to A25) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
12	HIZB12	0	R/W	Hi-Z Control for PTK Pin (VIO1_D5/IDED5 to VIO1_D0/IDED0, VIO0_FLD, VIO0_HD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
11	HIZB11	0	R/W	Hi-Z Control for PTL7 to PTL5Pin (DV_D5/ SCIF3_SCK/ RMII_RXD0, DV_D4/ SCIF3_RXD/ RMII_RXD1, DV_D3/ SCIF3_TXD/ RMII_REF_CLK) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
10	HIZB10	0	R/W	Hi-Z Control for PTL4 to PTL2 Pin (DV_D2/ SCIF1_SCK/ RMII_TX_EN, DV_D1/ SCIF1_RXD/ RMII_TXD0, DV_D0/ SCIF1_TXD/ RMII_TXD1) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z

Bit	Bit Name	Initial Value	R/W	Description
9	HIZB9	0	R/W	Hi-Z Control for PTL1 Pin (DV_D15) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
8	HIZB8	0	R/W	Hi-Z Control for PTL0, PTM7 to PTM2,PTN5 Pin (DV_D14/ MSIOF0_MCK, DV_D13/ MSIOF0_TSCK , DV_D12/ MSIOF0_RXD, DV_D11/ MSIOF0_TXD, DV_D10/ MSIOF0_TSYNC, DV_D9/ MSIOF0_SS1/MSIOF0_RSCK , DV_D8/ MSIOF0_SS2/MSIOF0_RSYNC, DV_CLKI) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
7	HIZB7	0	R/W	Hi-Z Control for PTN4 to PTN2 Pin (DV_CLK/ SCIF2_SCK, DV_VSYNC/ SCIF2_RXD, DV_HSYNC/ SCIF2_TXD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
6	HIZB6	0	R/W	Hi-Z Control for PTN1 , PTN0 Pin (DV_D7/ SCIF3_CTS/RMII_RX_ER, DV_D6/ SCIF3_RTS/ RMII_CRS_DV) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
5	HIZB5	0	R/W	Hi-Z Control for PTQ, PTT Pin (D15 toD0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
4	HIZB4	0	R/W	Hi-Z Control for PTR7 to PTR4, PTR2 to PTR0 Pin (CS5A, CS5B, CS6A, CS6B, WE2, WE3, WAIT) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
3	HIZB3	0	R/W	Hi-Z Control for PTS5 to PTS0 Pin (VIO1_FLD/ TPUTI2/ IDEIORDY, VIO1_HD/ SCIF5_SCK, VIO1_VD/ SCIF5_RXD, VIO1_CLK/ SCIF5_TXD, VIO1_D7/ IDED7, VIO1_D6/ IDED6) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
2	HIZB2	0	R/W	Hi-Z Control for PTS6 Pin (VIO_CKO) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z

Bit	Bit Name	Initial Value	R/W	Description
1	HIZB1	0	R/W	Hi-Z Control for PTU7, PYU6 Pin (DACK0, DREQ0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

48.2.54 I/O Buffer Hi-Z Control Register C (HIZCRC)

HIZCRC is a 16-bit readable/writable register that controls Hi-Z state of pins for every function unit.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	HIZC13	HIZC12	HIZC11	HIZC10	HIZC9	HIZC8	HIZC7	HIZC6	HIZC5	HIZC4	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	HIZC13	0	R/W	Hi-Z Control for PTU5 to PTU0 Pin (FSIOASD, FSIIABCK, FSIIALRCK, FSIOABCK, FSIOALRCK, CLKAUDIOAO) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
12	HIZC12	0	R/W	Hi-Z Control for PTV7 to PTV1 Pin (FSIIBSD/ MSIOF1_SS2/MSIOF1_RSYNC, FSIOBSD/ MSIOF1_SS1/MSIOF1_RSCK, FSIIBBCK/ MSIOF1_RXD, FSIIBLRCK/ MSIOF1_TSYNC, FSIOBBCK/ MSIOF1_TSCK, FSIOBLRCK/ MSIOF1_TXD, CLKAUDIOBO/ MSIOF1_MCK) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
11	HIZC11	0	R/W	Hi-Z Control for PTV0 Pin (FSIIASD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
10	HIZC10	0	R/W	Hi-Z Control for PTW7 to PTW4 Pin (MMC_D7/ SDHI1CD/ IODACK, MMC_D6/ SDHI1WP/ IDERST, MMC_D5/ SDHI1D3/ EXBUF_ENB, MMC_D4/ SDHI1D2/ DIRECTION) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z

Bit	Bit Name	Initial Value	R/W	Description
9	HIZC9	0	R/W	Hi-Z Control for PTW3 to PTW0 Pin (MMC_D3/SDHI1D1, MMC_D2/SDHI1D0, MMC_D1/SDHI1CMD, MMC_D0/SDHI1CLK) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
8	HIZC8	0	R/W	Hi-Z Control for PTX7,PTX6 Pin (DACK1/IRDA_OUT, DREQ1/$\overline{\text{IRDA\_IN}}$) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
7	HIZC7	0	R/W	Hi-Z Control for PTX5 to PTX3 Pin (TS_SDAT/LNKSTA, TS_SCK/MDIO, TS_SDEN/MDC) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
6	HIZC6	0	R/W	Hi-Z Control for PTX2Pin (TS_SPSYNC) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
5	HIZC5	0	R/W	Hi-Z Control for PTX1, PTX0 Pin (MMC_CLK, MMC_CMD) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
4	HIZC4	0	R/W	Hi-Z Control for PTY7 to PTY0 pin 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.2.55 I/O Buffer Hi-Z Control Register D (HIZCRD)

HIZCRD is a 16-bit readable/writable register that controls Hi-Z state of pins for every function unit.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HIZD15	HIZD14	HIZD13	HIZD12	HIZD11	HIZD10	HIZD9	HIZD8	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	HIZD15	0	R/W	Hi-Z Control for PTZ7Pin (IRQ7) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
14	HIZD14	0	R/W	Hi-Z Control for PTZ6 Pin (IRQ6) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
13	HIZD13	0	R/W	Hi-Z Control for PTZ5 Pin (IRQ5) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
12	HIZD12	0	R/W	Hi-Z Control for PTZ4 Pin (IRQ4) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
11	HIZD11	0	R/W	Hi-Z Control for PTZ3 Pin (IRQ3) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
10	HIZD10	0	R/W	Hi-Z Control for PTZ2 Pin (IRQ2) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
9	HIZD9	0	R/W	Hi-Z Control for PTZ1 Pin (IRQ1) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z
8	HIZD8	0	R/W	Hi-Z Control for PTZ0 Pin (IRQ0) 0: I/O buffer operates normally 1: I/O buffer input is fixed and output is Hi-Z

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.2.56 Module Function Select Register A (MSELCRA)

MSELCRA is a 16-bit readable/writable register that selects the functions specific to the module.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.2.57 Module Function Select Register B (MSELCRB)

MSELCRB is a 16-bit readable/writable register that selects the functions specific to the module.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MSELB15	MSELB14	—	—	—	—	—	—	MSELB7	—	MSELB5	—	—	—	MSELB1	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R	R/W	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
15	MSELB15	0	R/W	Oscillation control of XTAL_USB
14	MSELB14	0		00: OSC is oscillated
				01: OSC is stopped
				10: External clock is inputted
				11: OSC is stopped
13 to 8	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
7	MSELB7	0	R/W	SCIF2 Port Select
				0: Selects PTE5, PTF1, PTF0 port
				1: Selects PTN[4:2] port
6	—	0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
5	MSELB5	0	R/W	SCIF3 Port Select
				0: Selects PTL[7:5], PTN1, PTN0 port
				1: Selects PTZ[7:3] port
4 to 2	—	All 0	R	Reserved
				These bits are always read as 0. The write value should always be 0.
1	MSELB1	0	R/W	LCDVSYN Input/Output Select
				0: Selects output
				1: Selects input
0	—	0	R	Reserved
				These bits are always read as 0. The write value should always be 0.

48.2.58 Pull-Up Control Register (PULCR)

PULCR is a 16-bit readable/writable register that controls pull-up for pins.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PUL15	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	PUL15	1	R/W	Pull-Up Control for $\overline{\text{TRST}}$ Pin 0: Pull-up MOS is off 1: Pull-up MOS is on
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.2.59 I/O Buffer Drive Control Register A (DRVCRA)

DRVCRA is a 16-bit readable/writable register that selects the drive ability of the I/O buffer.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRVA15	DRVA14	DRVA13	DRVA12	DRVA11	DRVA10	DRVA9	DRVA8	DRVA7	DRVA6	DRVA5	DRVA4	—	—	DRVA1	DRVA0
Initial value:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DRVA15	0	R/W	These bits select the drive ability of the BSC pins
14	DRVA14	1	R/W	D[31:0], A[25:0], $\overline{WE2}$, $\overline{WE3}$, $\overline{RD}$, RDWR, $\overline{WE0}$, $\overline{WE1}$, $\overline{CS0}$, $\overline{CS4}$, $\overline{CS5A}$, $\overline{CS5B}$, $\overline{CS6A}$, $\overline{CS6B}$, $\overline{IOIS16}$, $\overline{WAIT}$, PTU[7:6], PTZ[1:0] 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
13	DRVA13	0	R/W	These bits select the drive ability of the
12	DRVA12	0	R/W	MMC pins PTW[7:0], PTX[1:0] 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
11	DRVA11	0	R/W	These bits select the drive ability of the VIO pins
10	DRVA10	0	R/W	PTN[7:6], PTH[7:0], PTK[7:0], PTS[6:0], SCL1, SDA1 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
9	DRVA9	0	R/W	These bits select the drive ability of the
8	DRVA8	0	R/W	SIU pins PTU[5:0], PTV[7:0], PTE[7:6] 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum

Bit	Bit Name	Initial Value	R/W	Description
7	DRVA7	0	R/W	These bits select the drive ability of the LCDC pins (PTC[7:0], PTD[7:0], PTE[5:0], PTF[7:0], PTM[1:0])
6	DRVA6	0	R/W	00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
5	DRVA5	0	R/W	These bits select the drive ability of the PTL[7:0], PTM[7:2], PTN[5:0], PTX[5:2] pins
4	DRVA4	0	R/W	00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
3	—	All 0	R	Reserved
2				These bits are always read as 0. The write value should always be 0.
1	DRVA1	0	R/W	These bits select the drive ability of the SDHI0 pins (PTY[7:0]), PTZ2
0	DRVA0	0	R/W	00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum

48.2.60 I/O Buffer Drive Control Register B (DRVCRB)

DRVCRB is a 16-bit readable/writable register that selects the drive ability of the I/O buffer.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRVB15	DRVB14	—	—	—	—	—	—	DRVB7	DRVB6	DRVB5	DRVB4	DRVB3	DRVB2	DRVB1	DRVB0
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DRVB15	1	R/W	These bits select the drive ability of the CKO pins
14	DRVB14	0	R/W	00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
13 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	DRVB7	0	R/W	These bits select the drive ability of the $\overline{\text{ASEBRK}}$/BRKAK pins
6	DRVB6	0	R/W	00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
5	DRVB5	0	R/W	These bits select the drive ability of the IRQ[7:3] pins (PTZ[7:3]) SCIF3 pins (PTZ[7:3]) 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
4	DRVB4	0	R/W	
3	DRVB3	0	R/W	These bits select the drive ability of the Ir pins (PTX[7:6]) 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
2	DRVB2	0	R/W	
1	DRVB1	0	R/W	This bit specifies the I/O voltage (VCCQ_VIO) for the SCL1 and SDA1 pins (IIC1) 0: I/O voltage (VCCQ_VIO) is 1.8 V 1: I/O voltage (VCCQ_VIO) is 2.85 V

Bit	Bit Name	Initial Value	R/W	Description
0	DRVB0	0	R/W	This bit specifies the I/O voltage (VCCQ_SR) for the SCL0 and SDA0 pins (IIC0) 0: I/O voltage (VCCQ_SR) is 1.8 V 1: I/O voltage (VCCQ_SR) is 2.85 V

48.2.61 I/O Buffer Drive Control Register C (DRVCRC)

DRVCRC is a 16-bit readable/writable register that selects the drive ability of the I/O buffer.

Bit :	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRVC 15	DRVC 14	DRVC 13	DRVC 12	DRVC 11	DRVC 10	DRVC9	DRVC8	DRVC7	DRVC6	DRVC5	DRVC4	—	—	—	—
Initial value:	1	1	1	0	1	0	1	0	1	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	DRVC15	1	R/W	These bits select the drive ability of the DBSC CLK pins (MCLK,$\overline{\text{MCLK}}$) 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
14	DRVC14	1	R/W	
13	DRVC13	1	R/W	These bits select the drive ability of the DBSC Address/Control pins (MCKE, MODT, $\overline{\text{MRAS}}$, $\overline{\text{NWE}}$, $\overline{\text{MCAS}}$, $\overline{\text{MCS}}$, MBA[2:0], MA[13:0],. 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
12	DRVC12	0	R/W	
11	DRVC11	1	R/W	These bits select the drive ability of the DBSC _DATA pins (MDQ31 to MDQ24), and MDQM3, MDQS3. 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
10	DRVC10	0	R/W	

Bit	Bit Name	Initial Value	R/W	Description
9	DRVC9	1	R/W	These bits select the drive ability of the DBSC _DATA pins (MDQ23 to MDQ16), and MDQM2, MDQS2. 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
8	DRVC8	0	R/W	
7	DRVC7	1	R/W	These bits select the drive ability of the DBSC _DATA pins (MDQ15 to MDQ8), and MDQM1, MDQS1. 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
6	DRVC6	0	R/W	
5	DRVC5	1	R/W	These bits select the drive ability of the DBSC _DATA pins (MDQ7 to MDQ0), and MDQM0, MDQS0. 00: I/O buffer drive ability is minimum 01: I/O buffer drive ability is low 10: I/O buffer drive ability is high 11: I/O buffer drive ability is maximum
4	DRVC4	0	R/W	
3 to0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

48.3 Settings for Each Pin

Table 48.4 list the functions, drive control bits, Hi-Z control bits, and function select bits for each pin.

Table 48.4 List of Control Bits for Each Pin

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTA7	D23	KEYOUT2	IDED15	DRVA[15:14]	HIZA[15]	PACR[15:14]	PSA[15:14]	—
PTA6	D22	KEYOUT1	IDED14	DRVA[15:14]	HIZA[15]	PACR[13:12]	PSA[15:14]	—
PTA5	D21	KEYOUT0	IDED13	DRVA[15:14]	HIZA[15]	PACR[11:10]	PSA[15:14]	—
PTA4	D20	KEYIN4	IDED12	DRVA[15:14]	HIZA[15]	PACR[9:8]	PSA[15:14]	—
PTA3	D19	KEYIN3	IDED11	DRVA[15:14]	HIZA[15]	PACR[7:6]	PSA[15:14]	—
PTA2	D18	KEYIN2	IDED10	DRVA[15:14]	HIZA[15]	PACR[5:4]	PSA[15:14]	—
PTA1	D17	KEYIN1	IDED9	DRVA[15:14]	HIZA[15]	PACR[3:2]	PSA[15:14]	—
PTA0	D16	KEYIN0	IDED8	DRVA[15:14]	HIZA[15]	PACR[1:0]	PSA[15:14]	—
PTB7	D31	TPUTO1	IDEA1	DRVA[15:14]	HIZA[14]	PBCR[15:14]	PSE[15:14]	—
PTB6	D30	TPUTO0	IDEA0	DRVA[15:14]	HIZA[14]	PBCR[13:12]	PSE[15:14]	—
PTB5	D29	—	IODREQ	DRVA[15:14]	HIZA[14]	PBCR[11:10]	PSE[11]	—
PTB4	D28	—	$\overline{\text{IDECS0}}$	DRVA[15:14]	HIZA[14]	PBCR[9:8]	PSE[11]	—
PTB3	D27	—	$\overline{\text{IDECS1}}$	DRVA[15:14]	HIZA[14]	PBCR[7:6]	PSE[11]	—
PTB2	D26	KEYOUT5/IN5	$\overline{\text{IDEIORD}}$	DRVA[15:14]	HIZA[13]	PBCR[5:4]	PSA[15:14]	—
PTB1	D25	KEYOUT4/IN6	$\overline{\text{IDEIOWR}}$	DRVA[15:14]	HIZA[13]	PBCR[3:2]	PSA[15:14]	—
PTB0	D24	KEYOUT3	IDEINT	DRVA[15:14]	HIZA[13]	PBCR[1:0]	PSA[15:14]	—
PTC7	LCDD7	—	—	DRVA[7:6]	HIZA[12]	PCCR[15:14]	PSD[5]	—
PTC6	LCDD6	—	—	DRVA[7:6]	HIZA[12]	PCCR[13:12]	PSD[5]	—
PTC5	LCDD5	—	—	DRVA[7:6]	HIZA[12]	PCCR[11:10]	PSD[5]	—
PTC4	LCDD4	—	—	DRVA[7:6]	HIZA[12]	PCCR[9:8]	PSD[5]	—
PTC3	LCDD3	—	—	DRVA[7:6]	HIZA[12]	PCCR[7:6]	PSD[5]	—
PTC2	LCDD2	—	—	DRVA[7:6]	HIZA[12]	PCCR[5:4]	PSD[5]	—
PTC1	LCDD1	—	—	DRVA[7:6]	HIZA[12]	PCCR[3:2]	PSD[5]	—
PTC0	LCDD0	—	—	DRVA[7:6]	HIZA[12]	PCCR[1:0]	PSD[5]	—
PTD7	LCDD15	—	—	DRVA[7:6]	HIZA[12]	PDCR[15:14]	PSD[5]	—
PTD6	LCDD14	—	—	DRVA[7:6]	HIZA[12]	PDCR[13:12]	PSD[5]	—

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTD5	LCDD13	—	—	DRVA[7:6]	HIZA[12]	PDCR[11:10]	PSD[5]	—
PTD4	LCDD12	—	—	DRVA[7:6]	HIZA[12]	PDCR[9:8]	PSD[5]	—
PTD3	LCDD11	—	—	DRVA[7:6]	HIZA[12]	PDCR[7:6]	PSD[5]	—
PTD2	LCDD10	—	—	DRVA[7:6]	HIZA12]	PDCR[5:4]	PSD[5]	—
PTD1	LCDD9	—	—	DRVA[7:6]	HIZA[12]	PDCR[3:2]	PSD[5]	—
PTD0	LCDD8	—	—	DRVA[7:6]	HIZA[12]	PDCR[1:0]	PSD[5]	—
PTE7	FSIMCKB	—	—	DRVA[9:8]	HIZA[11]	PECR[15:14]	—	—
PTE6	FSIMCKA	—	—	DRVA[9:8]	HIZA[11]	PECR[13:12]	—	—
PTE5	LCDD21	SCIF2_TXD	—	DRVA[7:6]	HIZA[10]	PECR[11:10]	PSC[5:4]	MSELB[7]
PTE4	LCDD20	SCIF4_SCK	—	DRVA[7:6]	HIZA[9]	PECR[9:8]	PSD[3:2]	—
PTE3	LCDD19	SCIF4_RXD	—	DRVA[7:6]	HIZA[9]	PECR[7:6]	PSA[3:2]	—
PTE2	LCDD18	SCIF4_TXD	—	DRVA[7:6]	HIZA[9]	PECR[5:4]	PSA[3:2]	—
PTE1	LCDD17	—	—	DRVA[7:6]	HIZA[12]	PECR[3:2]	PSD[5]	—
PTE0	LCDD16	—	—	DRVA[7:6]	HIZA[12]	PECR[1:0]	PSD[5]	—
PTF7	LCDVSYN	—	—	DRVA[7:6]	HIZA[8]	PFCR[15:14]	PSD[8]	—
PTF6	LCDDISP/LCDRS	—	—	DRVA[7:6]	HIZA[8]	PFCR[13:12]	PSD[10:9]	—
PTF5	LCDHSYN/$\overline{\text{LCDCS}}$	—	—	DRVA[7:6]	HIZA[8]	PFCR[11:10]	PSD[10:9]	—
PTF4	LCDDON	—	—	DRVA[7:6]	HIZA[8]	PFCR[9:8]	PSD[8]	—
PTF3	LCDDCK/$\overline{\text{LCDWR}}$	—	—	DRVA[7:6]	HIZA[8]	PFCR[7:6]	PSD[10:9]]	—
PTF2	LCDVEPWC	SCIF0_TXD	—	DRVA[7:6]	HIZA[7]	PFCR[5:4]	PSA[6]	—
PTF1	LCDD23	SCIF2_SCK	—	DRVA[7:6]	HIZA[6]	PFCR[3:2]	PSC[7:6]	MSELB[7]
PTF0	LCDD22	SCIF2_RXD	—	DRVA[7:6]	HIZA[6]	PFCR[1:0]	PSC[5:4]	MSELB[7]
PTG5	AUDCK	—	—		HIZA[1]	PGCR[11:10]	—	—
PTG4	AUDSYNC	—	—		HIZA[1]	PGCR[9:8]	—	—
PTG3	AUDATA3	—	—		HIZA[1]	PGCR[7:6]	—	—
PTG2	AUDATA2	—	—		HIZA[1]	PGCR[5:4]	—	—
PTG1	AUDATA1	—	—		HIZA[1]	PGCR[3:2]	—	—
PTG0	AUDATA0	—	—		HIZA[1]	PGCR[1:0]	—	—
PTH7	VIO0_VD	—	—	DRVA[11:10]	HIZB[15]	PHCR[15:14]	—	—
PTH6	VIO0_CLK	—	—	DRVA[11:10]	HIZB[15]	PHCR[13:12]	—	—
PTH5	VIO0_D7	—	—	DRVA[11:10]	HIZB[15]	PHCR[11:10]	—	—

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTH4	VIO0_D6	—	—	DRVA[11:10]	HIZB[15]	PHCR[9:8]	—	—
PTH3	VIO0_D5	—	—	DRVA[11:10]	HIZB[15]	PHCR[7:6]	—	—
PTH2	VIO0_D4	—	—	DRVA[11:10]	HIZB[15]	PHCR[5:4]	—	—
PTH1	VIO0_D3	—	—	DRVA[11:10]	HIZB[15]	PHCR[3:2]	—	—
PTH0	VIO0_D2	—	—	DRVA[11:10]	HIZB[15]	PHCR[1:0]	—	—
PTJ7	PDSTATUS	—	—	DRVA[15:14]	HIZB[14]	PJCR[15:14]	—	—
PTJ6	STATUS2	—	—	DRVA[15:14]	HIZB[14]	PJCR[13:12]	—	—
PTJ5	STATUS 0	—	—	DRVA[15:14]	HIZB[14]	PJCR[11:10]	—	—
PTJ3	A25	$\overline{BS}$	—	DRVA[15:14]	HIZB[13]	PJCR[7:6]	PSA[8]	—
PTJ2	A24	—	—	DRVA[15:14]	HIZB[13]	PJCR[5:4]	—	—
PTJ1	A23	—	—	DRVA[15:14]	HIZB[13]	PJCR[3:2]	—	—
PTJ0	A22	—	—	DRVA[15:14]	HIZB[13]	PJCR[1:0]	—	—
PTK7	VIO1_D5	VIO0_D13	IDED5	DRVA[11:10]	HIZB[12]	PKCR[15:14]	PSB[7:6]	—
PTK6	VIO1_D4	VIO0_D12	IDED4	DRVA[11:10]	HIZB[12]	PKCR[13:12]	PSB[7:6]	—
PTK5	VIO1_D3	VIO0_D11	IDED3	DRVA[11:10]	HIZB[12]	PKCR[11:10]	PSB[7:6]	—
PTK4	VIO1_D2	VIO0_D10	IDED2	DRVA[11:10]	HIZB[12]	PKCR[9:8]	PSB[7:6]	—
PTK3	VIO1_D1	VIO0_D9	IDED1	DRVA[11:10]	HIZB[12]	PKCR[7:6]	PSB[7:6]	—
PTK2	VIO1_D0	VIO0_D8	IDED0	DRVA[11:10]	HIZB[12]	PKCR[5:4]	PSB[7:6]	—
PTK1	VIO0_FLD	—	—	DRVA[11:10]	HIZB[12]	PKCR[3:2]	—	—
PTK0	VIO0_HD	—	—	DRVA[11:10]	HIZB[12]	PKCR[1:0]	—	—
PTL7	DV_D5	SCIF3_SCK	RMII_RXD0	DRVA[5:4]	HIZB[11]	PLCR[15:14]	PSB[9:8]	MSELB[5]
PTL6	DV_D4	SCIF3_RXD	RMII_RXD1	DRVA[5:4]	HIZB[11]	PLCR[13:12]	PSB[9:8]	MSELB[5]
PTL5	DV_D3	SCIF3_TXD	RMII_REF_CLK	DRVA[5:4]	HIZB[11]	PLCR[11:10]	PSE[7:6]	MSELB[5]
PTL4	DV_D2	SCIF1_SCK	RMII_TX_EN	DRVA[5:4]	HIZB[10]	PLCR[9:8]	PSC[9:8]	—
PTL3	DV_D1	SCIF1_RXD	RMII_TXD0	DRVA[5:4]	HIZB[10]	PLCR[7:6]	PSC[9:8]	—
PTL2	DV_D0	SCIF1_TXD	RMII_TXD1	DRVA[5:4]	HIZB[10]	PLCR[5:4]	PSC[9:8]	—
PTL1	DV_D15	—	—	DRVA[5:4]	HIZB[9]	PLCR[3:2]	PSD[4]	—
PTL0	DV_D14	MSIOF0_MCK	—	DRVA[5:4]	HIZB[8]	PLCR[1:0]	PSE[5:4]	—
PTM7	DV_D13	MSIOF0_TSCK	—	DRVA[5:4]	HIZB[8]	PMCR[15:14]	PSC[13:12]	—
PTM6	DV_D12	MSIOF0_RXD	—	DRVA[5:4]	HIZB[8]	PMCR[13:12]	PSC[13:12]	—
PTM5	DV_D11	MSIOF0_TXD	—	DRVA[5:4]	HIZB[8]	PMCR[11:10]	PSC[13:12]	—

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTM4	DV_D10	MSIOF0_TSYNC	—	DRVA[5:4]	HIZB[8]	PMCR[9:8]	PSC[13:12]	—
PTM3	DV_D9	$\overline{\text{MSIOF0\_SS1}}$/ MSIOF0_RSCK	—	DRVA[5:4]	HIZB[8]	PMCR[7:6]	PSC[11:10]	—
PTM2	DV_D8	$\overline{\text{MSIOF0\_SS2}}$/ MSIOF0_RSYNC	—	DRVA[5:4]	HIZB[8]	PMCR[5:4]	PSC[11:10]	—
PTM1	LCDVCPWC	SCIF0_RXD	—	DRVA[7:6]	HIZA[7]	PMCR[3:2]	PSA[6]	—
PTM0	$\overline{\text{LCDRD}}$	SCIF0_SCK	—	DRVA[7:6]	HIZA[7]	PMCR[1:0]	PSA[7]	—
PTN7	VIO0_D1	—	—	DRVA[11:10]	HIZB[15]	PNCR[15:14]	—	—
PTN6	VIO0_D0	—	—	DRVA[11:10]	HIZB[15]	PNCR[13:12]	—	—
PTN5	DV_CLKI	—	—	DRVA[5:4]	HIZB[8]	PNCR[11:10]	PSD[11]	—
PTN4	DV_CLK	SCIF2_SCK	—	DRVA[5:4]	HIZB[7]	PNCR[9:8]	PSD[13:12]	MSELB[7]
PTN3	DV_VSYNC	SCIF2_RXD	—	DRVA[5:4]	HIZB[7]	PNCR[7:6]	PSD[15:14]	MSELB[7]
PTN2	DV_HSYNC	SCIF2_TXD	—	DRVA[5:4]	HIZB[7]	PNCR[5:4]	PSB[5:4]	MSELB[7]
PTN1	DV_D7	$\overline{\text{SCIF3\_CTS}}$	RMII_RX_ER	DRVA[5:4]	HIZB[6]	PNCR[3:2]	PSB[3:2]	MSELB[5]
PTN0	DV_D6	$\overline{\text{SCIF3\_RTS}}$	RMII_CRS_DV	DRVA[5:4]	HIZB[6]	PNCR[1:0]	PSB[1:0]	MSELB[5]
PTQ7	D7	—	—	DRVA[15:14]	HIZB[5]	PQCR[15:14]	—	—
PTQ6	D6	—	—	DRVA[15:14]	HIZB[5]	PQCR[13:12]	—	—
PTQ5	D5	—	—	DRVA[15:14]	HIZB[5]	PQCR[11:10]	—	—
PTQ4	D4	—	—	DRVA[15:14]	HIZB[5]	PQCR[9:8]	—	—
PTQ3	D3	—	—	DRVA[15:14]	HIZB[5]	PQCR[7:6]	—	—
PTQ2	D2	—	—	DRVA[15:14]	HIZB[5]	PQCR[5:4]	—	—
PTQ1	D1	—	—	DRVA[15:14]	HIZB[5]	PQCR[3:2]	—	—
PTQ0	D0	—	—	DRVA[15:14]	HIZB[5]	PQCR[1:0]	—	—
PTR7	$\overline{\text{CS6B}}$/$\overline{\text{CE1B}}$	—	—	DRVA[15:14]	HIZB[4]	PRCR[15:14]	—	—
PTR6	$\overline{\text{CS6A}}$/$\overline{\text{CE2B}}$	—	—	DRVA[15:14]	HIZB[4]	PRCR[13:12]	—	—
PTR5	$\overline{\text{CS5B}}$/$\overline{\text{CE1A}}$	—	—	DRVA[15:14]	HIZB[4]	PRCR[11:10]	—	—
PTR4	$\overline{\text{CS5A}}$/$\overline{\text{CE2A}}$	—	—	DRVA[15:14]	HIZB[4]	PRCR[9:8]	—	—
PTR3	$\overline{\text{IOIS16}}$	LCDLCLK	—	DRVA[15:14]	HIZA[0]	PRCR[7:6]	PSA[5]	—
PTR2	$\overline{\text{WAIT}}$	—	—	DRVA[15:14]	HIZB[4]	PRCR[5:4]	—	—
PTR1	$\overline{\text{WE3}}$/$\overline{\text{ICIOWR}}$	TPUTO3	TPUTI3	DRVA[15:14]	HIZB[4]	PRCR[3:2]	PSA[1:0]	—
PTR0	$\overline{\text{WE2}}$/$\overline{\text{ICIORD}}$	TPUTO2	IDEA2	DRVA[15:14]	HIZB[4]	PRCR[1:0]	PSD[1:0]	—
PTS6	VIO_CKO	—	—	DRVA[11:10]	HIZB[2]	PSCR[13:12]	—	—

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTS5	VIO1_FLD	TPUTI2	IDEIORDY	DRVA[11:10]	HIZB[3]	PSCR[11:10]	PSE[9:8]	—
PTS4	VIO1_HD	SCIF5_SCK	—	DRVA[11:10]	HIZB[3]	PSCR[9:8]	PSA[10]	—
PTS3	VIO1_VD	SCIF5_RXD	—	DRVA[11:10]	HIZB[3]	PSCR[7:6]	PSA[9]	—
PTS2	VIO1_CLK	SCIF5_TXD	—	DRVA[11:10]	HIZB[3]	PSCR[5:4]	PSA[9]	—
PTS1	VIO1_D7	VIO0_D15	IDED7	DRVA[11:10]	HIZB[3]	PSCR[3:2]	PSB[7:6]	—
PTS0	VIO1_D6	VIO0_D14	IDED6	DRVA[11:10]	HIZB[3]	PSCR[1:0]	PSB[7:6]	—
PTT7	D15	—	—	DRVA[15:14]	HIZB[5]	PTCR[15:14]	—	—
PTT6	D14	—	—	DRVA[15:14]	HIZB[5]	PTCR[13:12]	—	—
PTT5	D13	—	—	DRVA[15:14]	HIZB[5]	PTCR[11:10]	—	—
PTT4	D12	—	—	DRVA[15:14]	HIZB[5]	PTCR[9:8]	—	—
PTT3	D11	—	—	DRVA[15:14]	HIZB[5]	PTCR[7:6]	—	—
PTT2	D10	—	—	DRVA[15:14]	HIZB[5]	PTCR[5:4]	—	—
PTT1	D9	—	—	DRVA[15:14]	HIZB[5]	PTCR[3:2]	—	—
PTT0	D8	—	—	DRVA[15:14]	HIZB[5]	PTCR[1:0]	—	—
PTU7	DACK0	—	—	DRVA[15:14]	HIZB[1]	PUCR[15:14]	—	—
PTU6	DREQ0	—	—	DRVA[15:14]	HIZB[1]	PUCR[13:12]	—	—
PTU5	FSIOASD	—	—	DRVA[9:8]	HIZC[13]	PUCR[11:10]	PSE[1]	—
PTU4	FSIIABCK	—	—	DRVA[9:8]	HIZC[13]	PUCR[9:8]	PSE[1]	—
PTU3	FSIIALRCK	—	—	DRVA[9:8]	HIZC[13]	PUCR[7:6]	PSE[1]	—
PTU2	FSIOABCK	—	—	DRVA[9:8]	HIZC[13]	PUCR[5:4]	PSE[1]	—
PTU1	FSIOALRCK	—	—	DRVA[9:8]	HIZC[13]	PUCR[3:2]	PSE[1]	—
PTU0	CLKAUDIOAO	—	—	DRVA[9:8]	HIZC[13]	PUCR[1:0]	PSE[0]	—
PTV7	FSIIBSD	$\overline{\text{MSIOF1\_SS2}}$/MSIOF1_RSYNC	—	DRVA[9:8]	HIZC[12]	PVCR[15:14]	PSD[7:6]	—
PTV6	FSIOBSD	$\overline{\text{MSIOF1\_SS1}}$/MSIOF1_RSCK	—	DRVA[9:8]	HIZC[12]	PVCR[13:12]	PSD[7:6]	—
PTV5	FSIIBBCK	MSIOF1_RXD	—	DRVA[9:8]	HIZC[12]	PVCR[11:10]	PSC[15:14]	—
PTV4	FSIIBLRCK	MSIOF1_TSYNC	—	DRVA[9:8]	HIZC[12]	PVCR[9:8]	PSC[15:14]	—
PTV3	FSIOBBCK	MSIOF1_TSCK	—	DRVA[9:8]	HIZC[12]	PVCR[7:6]	PSC[15:14]	—
PTV2	FSIOBLRCK	MSIOF1_TXD	—	DRVA[9:8]	HIZC[12]	PVCR[5:4]	PSC[15:14]	—
PTV1	CLKAUDIOBO	MSIOF1_MCK	—	DRVA[9:8]	HIZC[12]	PVCR[3:2]	PSE[3:2]	—
PTV0	FSIIASD	—	—	DRVA[9:8]	HIZC[11]	PVCR[1:0]	PSE[10]	—

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTW7	MMC_D7	SDHI1CD	IODACK	DRVA[13:12]	HIZC[10]	PWCR[15:14]	PSE[13:12]	—
PTW6	MMC_D6	SDHI1WP	IDERST	DRVA[13:12]	HIZC[10]	PWCR[13:12]	PSE[13:12]	—
PTW5	MMC_D5	SDHI1D3	EXBUF_ENB	DRVA[13:12]	HIZC[10]	PWCR[11:10]	PSE[13:12]	—
PTW4	MMC_D4	SDHI1D2	DIRECTION	DRVA[13:12]	HIZC[10]	PWCR[9:8]	PSE[13:12]	—
PTW3	MMC_D3	SDHI1D1	—	DRVA[13:12]	HIZC[9]	PWCR[7:6]	PSA[13]	—
PTW2	MMC_D2	SDHI1D0	—	DRVA[13:12]	HIZC[9]	PWCR[5:4]	PSA[13]	—
PTW1	MMC_D1	SDHI1CMD	—	DRVA[13:12]	HIZC[9]	PWCR[3:2]	PSA[13]	—
PTW0	MMC_D0	SDHI1CLK	—	DRVA[13:12]	HIZC[9]	PWCR[1:0]	PSA[13]	—
PTX7	DACK1	IRDA_OUT	—	DRVB[3:2]	HIZC[8]	PXCR[15:14]	PSA[12]	—
PTX6	DREQ1	IRDA_IN	—	DRVB[3:2]	HIZC[8]	PXCR[13:12]	PSA[12]	—
PTX5	TS0_SDAT	—	LNKSTA	DRVA[5:4]	HIZC[7]	PXCR[11:10]	PSC[0]	—
PTX4	TS0_SCK	—	MDIO	DRVA[5:4]	HIZC[7]	PXCR[9:8]	PSC[1]	—
PTX3	TS0_SDEN	—	MDC	DRVA[5:4]	HIZC[7]	PXCR[7:6]	PSC[2]	—
PTX2	TS0_SPSYNC	—	—	DRVA[5:4]	HIZC[6]	PXCR[5:4]	—	—
PTX1	MMC_CLK	—	—	DRVA[13:12]	HIZC[5]	PXCR[3:2]	—	—
PTX0	MMC_CMD	—	—	DRVA[13:12]	HIZC[5]	PXCR[1:0]	—	—
PTY7	SDHI0CD	—	—	DRVA[1:0]	HIZC[4]	PYCR[15:14]	—	—
PTY6	SDHI0WP	—	—	DRVA[1:0]	HIZC[4]	PYCR[13:12]	—	—
PTY5	SDHI0D3	—	—	DRVA[1:0]	HIZC[4]	PYCR[11:10]	—	—
PTY4	SDHI0D2	—	—	DRVA[1:0]	HIZC[4]	PYCR[9:8]	—	—
PTY3	SDHI0D1	—	—	DRVA[1:0]	HIZC[4]	PYCR[7:6]	—	—
PTY2	SDHI0D0	—	—	DRVA[1:0]	HIZC[4]	PYCR[5:4]	—	—
PTY1	SDHI0CMD	—	—	DRVA[1:0]	HIZC[4]	PYCR[3:2]	—	—
PTY0	SDHI0CLK	—	—	DRVA[1:0]	HIZC[4]	PYCR[1:0]	—	—
PTZ7	IRQ7	SCIF3_CTS	—	DRVB[5:4]	HIZD[15]	PZCR[15:14]	PSB[10]	MSELB[5]
PTZ6	IRQ6	SCIF3_RTS	—	DRVB[5:4]	HIZD[14]	PZCR[13:12]	PSB[11]	MSELB[5]
PTZ5	IRQ5	SCIF3_SCK	—	DRVB[5:4]	HIZD[13]	PZCR[11:10]	PSB[12]	MSELB[5]
PTZ4	IRQ4	SCIF3_RXD	—	DRVB[5:4]	HIZD[12]	PZCR[9:8]	PSB[13]	MSELB[5]
PTZ3	IRQ3	SCIF3_TXD	—	DRVB[5:4]	HIZD[11]	PZCR[7:6]	PSB[14]	MSELB[5]
PTZ2	IRQ2	—	—	DRVA[1:0]	HIZD[10]	PZCR[5:4]	—	—
PTZ1	IRQ1	—	—	DRVA[15:14]	HIZD[9]	PZCR[3:2]	—	—

Port Pin Name	Function 1 Pin Name	Function 2 Pin Name	Function 3 Pin Name	Drive Control Bit	Hi-Z Control Bit	Port Control Bit	Function 1/2/3 Select Bit	SCIF2/3 Select Bit
PTZ0	IRQ0	—	—	DRVA[15:14]	HIZD[8]	PZCR[1:0]	—	—

Section 49 User Break Controller (UBC)

The user break controller (UBC) provides versatile functions to facilitate program debugging. These functions help to ease creation of a self-monitor/debugger, which allows easy program debugging using this LSI alone, without using the in-circuit emulator. Various break conditions can be set in the UBC: instruction fetch or read/write access of an operand, operand size, data contents, address value, and program stop timing for instruction fetch.

49.1 Features

1. The following break conditions can be set.

   Break channels: Two (channels 0 and 1)

   User break conditions can be set independently for channels 0 and 1, and can also be set as a single sequential condition for the two channels, that is, a sequential break. (Sequential break involves two cases such that the channel 0 break condition is satisfied in a certain bus cycle and then the channel 1 break condition is satisfied in a different bus cycle, and vice versa.)

- Address

  When 40 bits containing ASID and 32-bit address are compared with the specified value, all the ASID bits can be compared or masked.

  32-bit address can be masked bit by bit, allowing the user to mask the address in desired page sizes such as lower 12 bits (4-Kbyte page) and lower 10 bits (1-Kbyte page).

- Data

  32 bits can be masked only for channel 1.

- Bus cycle

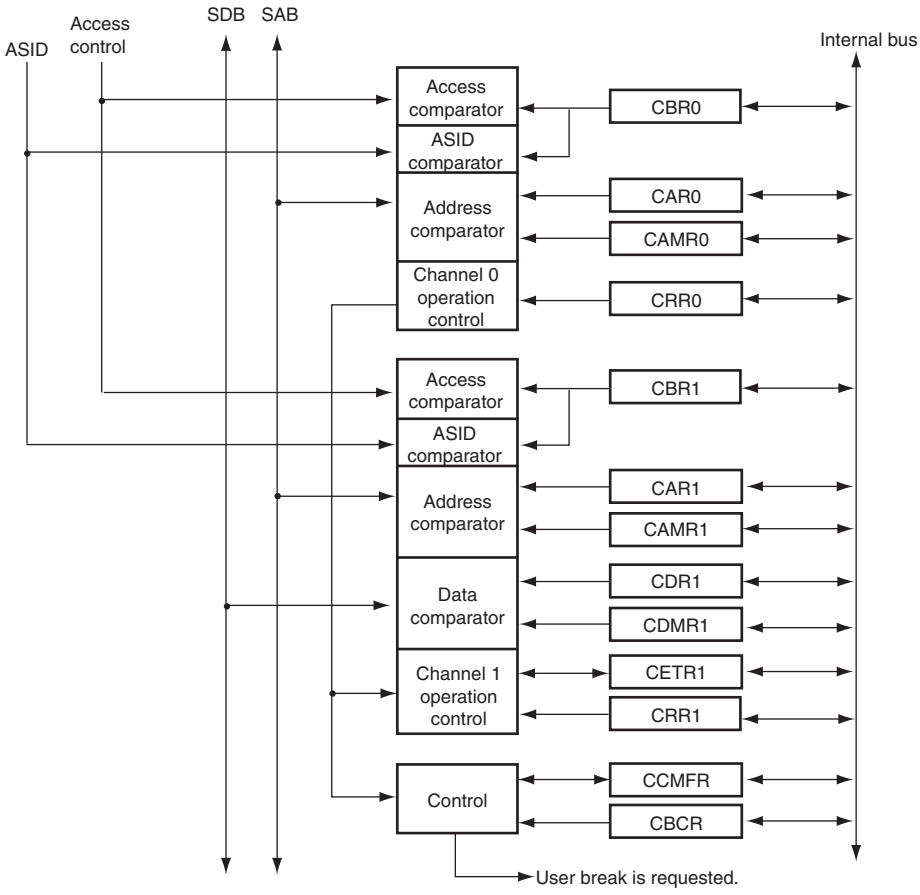
  The program can break either for instruction fetch (PC break) or operand access.

- Read or write access
- Operand sizes

  Byte, word, longword, and quadword are supported.

2. The user-designated exception handling routine for the user break condition can be executed.
3. Pre-instruction-execution or post-instruction-execution can be selected as the PC break timing.
4. A maximum of $2^{12} - 1$ repetition counts can be specified as the break condition (available only for channel 1).

Figure 49.1 shows the UBC block diagram.



[Legend]

CBR0:	Match condition setting register 0	CAMR1:	Match address mask setting register 1
CRR0:	Match operation setting register 0	CDR1:	Match data setting register 1
CAR0:	Match address setting register 0	CDMR1:	Match data mask setting register 1
CAMR0:	Match address mask setting register 0	CETR1:	Execution count break register 1
CBR1:	Match condition setting register 1	CCMFR:	Channel match flag register
CRR1:	Match operation setting register 1	CBCR:	Break control register
CAR1:	Match address setting register 1	SAB:	Operand address bus
		SDB:	Operand data bus

Figure 49.1 Block Diagram of UBC

49.2 Register Descriptions

The UBC has the following registers.

Table 49.1 Register Configuration

Name	Abbreviation	R/W	P4 Address*	Area 7 Address*	Access Size
Match condition setting register 0	CBR0	R/W	H'FF200000	H'1F200000	32
Match operation setting register 0	CRR0	R/W	H'FF200004	H'1F200004	32
Match address setting register 0	CAR0	R/W	H'FF200008	H'1F200008	32
Match address mask setting register 0	CAMR0	R/W	H'FF20000C	H'1F20000C	32
Match condition setting register 1	CBR1	R/W	H'FF200020	H'1F200020	32
Match operation setting register 1	CRR1	R/W	H'FF200024	H'1F200024	32
Match address setting register 1	CAR1	R/W	H'FF200028	H'1F200028	32
Match address mask setting register 1	CAMR1	R/W	H'FF20002C	H'1F20002C	32
Match data setting register 1	CDR1	R/W	H'FF200030	H'1F200030	32
Match data mask setting register 1	CDMR1	R/W	H'FF200034	H'1F200034	32
Execution count break register 1	CETR1	R/W	H'FF200038	H'1F200038	32
Channel match flag register	CCMFR	R/W	H'FF200600	H'1F200600	32
Break control register	CBCR	R/W	H'FF200620	H'1F200620	32

Note: * P4 addresses are used when area P4 in the virtual address space is used, and area 7 addresses are used when accessing the register through area 7 in the physical address space using the TLB.

Table 49.2 Register Status in Each Processing State

Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
CBR0	H'20000000	Retained	Retained	Retained	Retained*	Retained
CRR0	H'00002000	Retained	Retained	Retained	Retained*	Retained
CAR0	Undefined	Retained	Retained	Retained	Retained*	Retained
CAMR0	Undefined	Retained	Retained	Retained	Retained*	Retained
CBR1	H'20000000	Retained	Retained	Retained	Retained*	Retained
CRR1	H'00002000	Retained	Retained	Retained	Retained*	Retained
CAR1	Undefined	Retained	Retained	Retained	Retained*	Retained
CAMR1	Undefined	Retained	Retained	Retained	Retained*	Retained
CDR1	Undefined	Retained	Retained	Retained	Retained*	Retained
CDMR1	Undefined	Retained	Retained	Retained	Retained*	Retained
CETR1	Undefined	Retained	Retained	Retained	Retained*	Retained
CCMFR	H'00000000	Retained	Retained	Retained	Retained*	Retained
CBCR	H'00000000	Retained	Retained	Retained	Retained*	Retained

Note: * Has this state when the MPMD pin is low. When the MPMD pin is high, has the same state as at a power-on reset.

The access size must be the same as the control register size. If the size is different, the register is not written to if attempted, and reading the register returns the undefined value. A desired break may not occur between the time when the instruction for rewriting the control register is executed and the time when the written value is actually reflected on the register. In order to confirm the exact timing when the control register is updated, read the data which has been written most recently. The subsequent instructions are valid for the most recently written register value.

49.2.1 Match Condition Setting Registers 0 and 1 (CBR0 and CBR1)

CBR0 and CBR1 are readable/writable 32-bit registers which specify the break conditions for channels 0 and 1, respectively. The following break conditions can be set in the CBR0 and CBR1: (1) whether or not to include the match flag in the conditions, (2) whether or not to include the ASID, and the ASID value when included, (3) whether or not to include the data value, (4) operand size, (5) whether or not to include the execution count, (6) bus type, (7) instruction fetch cycle or operand access cycle, and (8) read or write access cycle.

- CBR0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MFE	AIE	MFI						AIV							
Initial value:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SZ			—	—	—	—	CD		ID		—	RW		CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MFE	0	R/W	Match Flag Enable Specifies whether or not to include the match flag value specified by the MFI bit of this register in the match conditions. When the specified match flag value is 1, the condition is determined to be satisfied. 0: The match flag is not included in the match conditions; thus, not checked. 1: The match flag is included in the match conditions.
30	AIE	0	R/W	ASID Enable Specifies whether or not to include the ASID specified by the AIV bit of this register in the match conditions. 0: The ASID is not included in the match conditions; thus, not checked. 1: The ASID is included in the match conditions.

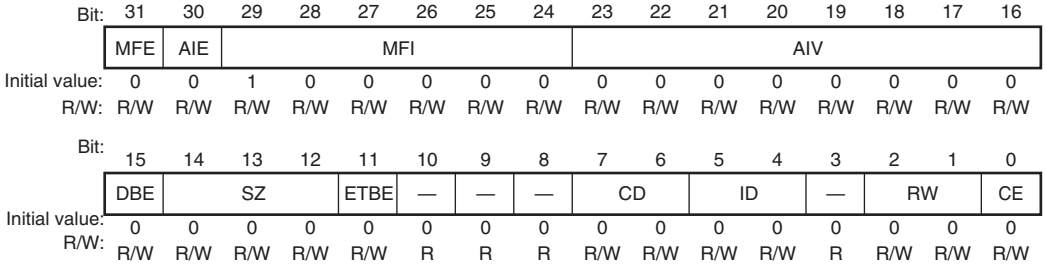
Bit	Bit Name	Initial Value	R/W	Description
29 to 24	MFI	100000	R/W	Match Flag Specify Specifies the match flag to be included in the match conditions. 000000: MF0 bit of the CCMFR register 000001: MF1 bit of the CCMFR register Others: Reserved (setting prohibited) Note: The initial value is the reserved value, but when 1 is written into CBR0[0], MFI must be set to 000000 or 000001. And note that the channel 0 is not hit when MFE bit of this register is 1 and MFI bits are 000000 in the condition of CCRMF.MF0 = 0.
23 to 16	AIV	All 0	R/W	ASID Specify Specifies the ASID value to be included in the match conditions.
15	—	0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
14 to 12	SZ	All 0	R/W	Operand Size Select Specifies the operand size to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 000: The operand size is not included in the match conditions; thus, not checked (any operand size specifies the match condition).*[1] 001: Byte access 010: Word access 011: Longword access 100: Quadword access*[2] Others: Reserved (setting prohibited)
11 to 8	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	CD	All 0	R/W	Bus Select Specifies the bus to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 00: Operand bus for operand access Others: Reserved (setting prohibited)
5, 4	ID	All 0	R/W	Instruction Fetch/Operand Access Select Specifies the instruction fetch cycle or operand access cycle as the match condition. 00: Instruction fetch cycle or operand access cycle 01: Instruction fetch cycle 10: Operand access cycle 11: Instruction fetch cycle or operand access cycle
3	—	0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
2, 1	RW	All 0	R/W	Bus Command Select Specifies the read/write cycle as the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 00: Read cycle or write cycle 01: Read cycle 10: Write cycle 11: Read cycle or write cycle
0	CE	0	R/W	Channel Enable Validates/invalidates the channel. If this bit is 0, all the other bits of this register are invalid. 0: Invalidates the channel. 1: Validates the channel.

Notes: 1. If the data value is included in the match conditions, be sure to specify the operand size.

2. If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and the match data mask setting register.

- CBR1



Bit	Bit Name	Initial Value	R/W	Description
31	MFE	0	R/W	Match Flag Enable Specifies whether or not to include the match flag value specified by the MFI bit of this register in the match conditions. When the specified match flag value is 1, the condition is determined to be satisfied. 0: The match flag is not included in the match conditions; thus, not checked. 1: The match flag is included in the match conditions.
30	AIE	0	R/W	ASID Enable Specifies whether or not to include the ASID specified by the AIV bit of this register in the match conditions. 0: The ASID is not included in the match conditions; thus, not checked. 1: The ASID is included in the match conditions.
29 to 24	MFI	100000	R/W	Match Flag Specify Specifies the match flag to be included in the match conditions. 000000: The MF0 bit of the CCMFR register 000001: The MF1 bit of the CCMFR register Others: Reserved (setting prohibited) Note: The initial value is the reserved value, but when 1 is written into CBR1[0], MFI must be set to 000000 or 000001. And note that the channel 1 is not hit when MFE bit of this register is 1 and MFI bits are 000001 in the condition of CCRMF.MF1 = 0.

Bit	Bit Name	Initial Value	R/W	Description
23 to 16	AIV	All 0	R/W	ASID Specify Specifies the ASID value to be included in the match conditions.
15	DBE	0	R/W	Data Value Enable*3 Specifies whether or not to include the data value in the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 0: The data value is not included in the match conditions; thus, not checked. 1: The data value is included in the match conditions.
14 to 12	SZ	All 0	R/W	Operand Size Select Specifies the operand size to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 000: The operand size is not included in the match condition; thus, not checked (any operand size specifies the match condition). *1 001: Byte access 010: Word access 011: Longword access 100: Quadword access*2 Others: Reserved (setting prohibited)
11	ETBE	0	R/W	Execution Count Value Enable Specifies whether or not to include the execution count value in the match conditions. If this bit is 1 and the match condition satisfaction count matches the value specified by the CETR1 register, the operation specified by the CRR1 register is performed. 0: The execution count value is not included in the match conditions; thus, not checked. 1: The execution count value is included in the match conditions.
10 to 8	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	CD	All 0	R/W	Bus Select Specifies the bus to be included in the match conditions. This bit is valid only when the operand access cycle is specified as a match condition. 00: Operand bus for operand access Others: Reserved (setting prohibited)
5, 4	ID	All 0	R/W	Instruction Fetch/Operand Access Select Specifies the instruction fetch cycle or operand access cycle as the match condition. 00: Instruction fetch cycle or operand access cycle 01: Instruction fetch cycle 10: Operand access cycle 11: Instruction fetch cycle or operand access cycle
3	—	0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
2, 1	RW	All 0	R/W	Bus Command Select Specifies the read/write cycle as the match condition. This bit is valid only when the operand access cycle is specified as a match condition. 00: Read cycle or write cycle 01: Read cycle 10: Write cycle 11: Read cycle or write cycle
0	CE	0	R/W	Channel Enable Validates/invalidates the channel. If this bit is 0, all the other bits in this register are invalid. 0: Invalidates the channel. 1: Validates the channel.

Notes: 1. If the data value is included in the match conditions, be sure to specify the operand size.

2. If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and the match data mask setting register.

3. The OCBI instruction is handled as longword write access without the data value, and the PREF, OCBP, and OCBWB instructions are handled as longword read access without the data value. Therefore, do not include the data value in the match conditions for these instructions.

49.2.2 Match Operation Setting Registers 0 and 1 (CRR0 and CRR1)

CRR0 and CRR1 are readable/writable 32-bit registers which specify the operation to be executed when channels 0 and 1 satisfy the match condition, respectively. The following operations can be set in the CRR0 and CRR1 registers: (1) breaking at a desired timing for the instruction fetch cycle and (2) requesting a break.

- CRR0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCB	BIE
Initial value:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
13	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
12 to 2	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
1	PCB	0	R/W	PC Break Select Specifies either before or after instruction execution as the break timing for the instruction fetch cycle. This bit is invalid for breaks other than the ones for the instruction fetch cycle. 0: Sets the PC break before instruction execution. 1: Sets the PC break after instruction execution.
0	BIE	0	R/W	Break Enable Specifies whether or not to request a break when the match condition is satisfied for the channel. 0: Does not request a break. 1: Requests a break.

- CRR1

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PCB	BIE
Initial value:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

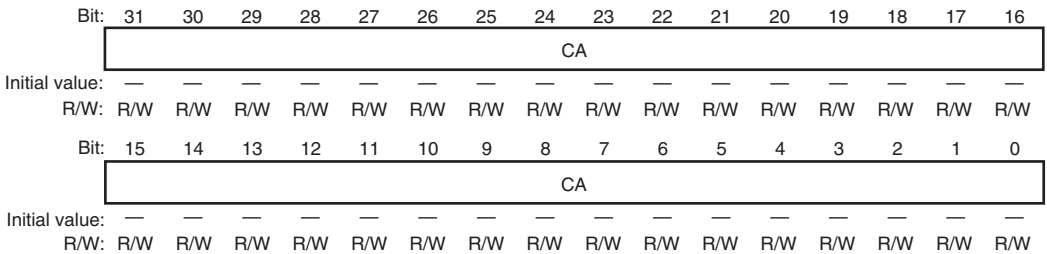
Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
13	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
12 to 2	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.

Bit	Bit Name	Initial Value	R/W	Description
1	PCB	0	R/W	PC Break Select Specifies either before or after instruction execution as the break timing for the instruction fetch cycle. This bit is invalid for breaks other than ones for the instruction fetch cycle. 0: Sets the PC break before instruction execution. 1: Sets the PC break after instruction execution.
0	BIE	0	R/W	Break Enable Specifies whether or not to request a break when the match condition is satisfied for the channel. 0: Does not request a break. 1: Requests a break.

49.2.3 Match Address Setting Registers 0 and 1 (CAR0 and CAR1)

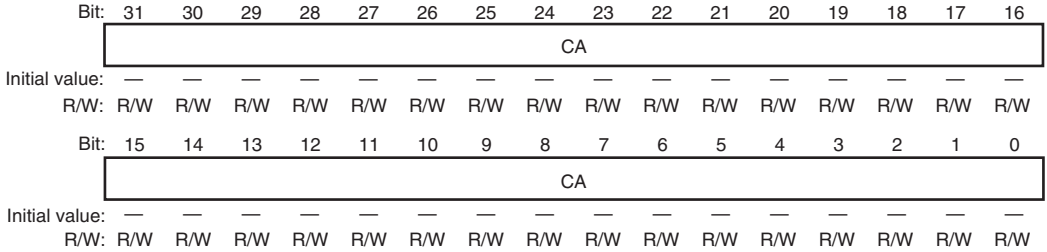
CAR0 and CAR1 are readable/writable 32-bit registers specifying the virtual address to be included in the break conditions for channels 0 and 1, respectively.

- CAR0



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CA	Undefined	R/W	Compare Address Specifies the address to be included in the break conditions. When the operand bus has been specified using the CBR0 register, specify the SAB address in CA[31:0].

- CAR1

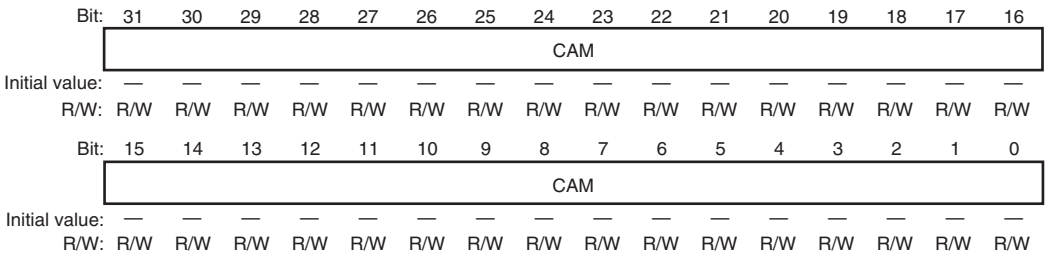


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CA	Undefined	R/W	Compare Address Specifies the address to be included in the break conditions. When the operand bus has been specified using the CBR1 register, specify the SAB address in CA[31:0].

49.2.4 Match Address Mask Setting Registers 0 and 1 (CAMR0 and CAMR1)

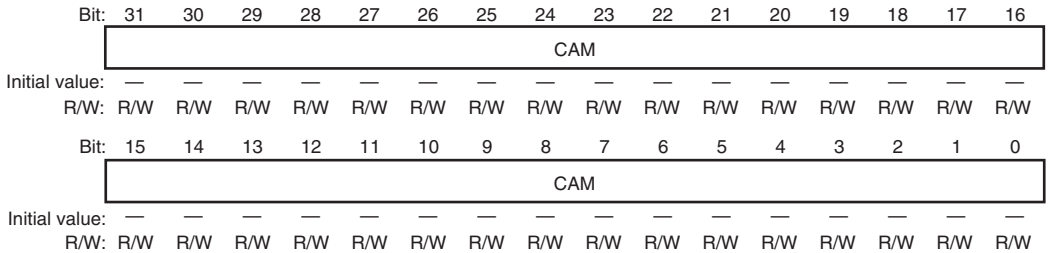
CMAR0 and CMAR1 are readable/writable 32-bit registers which specify the bits to be masked among the address bits specified by using the match address setting register of the corresponding channel. (Set the bits to be masked to 1.)

- CAMR0



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAM	Undefined	R/W	Compare Address Mask Specifies the bits to be masked among the address bits which are specified using the CAR0 register. (Set the bits to be masked to 1.) 0: Address bits CA[n] are included in the break condition. 1: Address bits CA[n] are masked and not included in the break condition. [n] = any values from 31 to 0

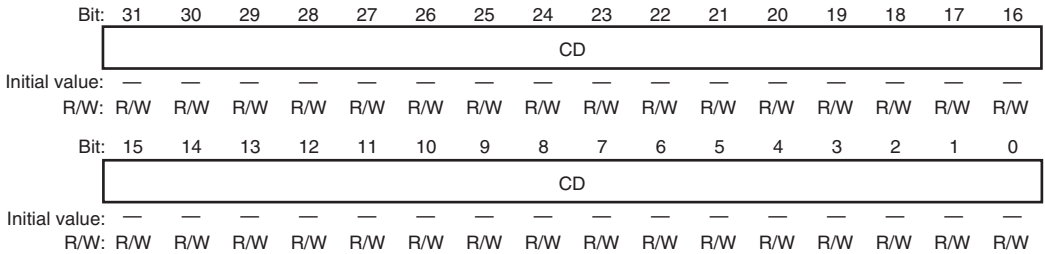
- CAMR1



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CAM	Undefined	R/W	Compare Address Mask Specifies the bits to be masked among the address bits which are specified using the CAR1 register. (Set the bits to be masked to 1.) 0: Address bits CA[n] are included in the break condition. 1: Address bits CA[n] are masked and not included in the break condition. [n] = any values from 31 to 0

49.2.5 Match Data Setting Register 1 (CDR1)

CDR1 is a readable/writable 32-bit register which specifies the data value to be included in the break conditions for channel 1.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CD	Undefined	R/W	Compare Data Value Specifies the data value to be included in the break conditions. When the operand bus has been specified using the CBR1 register, specify the SDB data value in CD[31:0].

Table 49.3 Settings for Match Data Setting Register

Bus and Size Selected Using CBR1	CD[31:24]	CD[23:16]	CD[15:8]	CD[7:0]
Operand bus (byte)	Don't care	Don't care	Don't care	SDB7 to SDB0
Operand bus (word)	Don't care	Don't care	SDB15 to SDB8	SDB7 to SDB0
Operand bus (longword)	SDB31 to SDB24	SDB23 to SDB16	SDB15 to SDB8	SDB7 to SDB0

Notes: 1. If the data value is included in the match conditions, be sure to specify the operand size.
2. The OCBI instruction is handled as longword write access without the data value, and the PREF, OCBP, and OCBWB instructions are handled as longword read access without the data value. Therefore, do not include the data value in the match conditions for these instructions.
3. If the quadword access is specified and the data value is included in the match conditions, the upper and lower 32 bits of 64-bit data are each compared with the contents of both the match data setting register and match data mask setting register.

49.2.6 Match Data Mask Setting Register 1 (CDMR1)

CDMR1 is a readable/writable 32-bit register which specifies the bits to be masked among the data value bits specified using the match data setting register. (Set the bits to be masked to 1.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDM															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDM															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDM	Undefined	R/W	Compare Data Value Mask Specifies the bits to be masked among the data value bits specified using the CDR1 register. (Set the bits to be masked to 1.) 0: Data value bits CD[n] are included in the break condition. 1: Data value bits CD[n] are masked and not included in the break condition. [n] = any values from 31 to 0

49.2.7 Execution Count Break Register 1 (CETR1)

CETR1 is a readable/writable 32-bit register which specifies the number of the channel hits before a break occurs. A maximum value of $2^{12} - 1$ can be specified. When the execution count value is included in the match conditions by using the match condition setting register, the value of this register is decremented by one every time the channel is hit. When the channel is hit after the register value reaches H'001, a break occurs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CET											
Initial value:	0	0	0	0	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
11 to 0	CET	Undefined	R/W	Execution Count Specifies the execution count to be included in the break conditions.

49.2.8 Channel Match Flag Register (CCMFR)

CCMFR is a readable/writable 32-bit register which indicates whether or not the match conditions have been satisfied for each channel. When a channel match condition has been satisfied, the corresponding flag bit is set to 1. To clear the flags, write the data containing value 0 for the bits to be cleared and value 1 for the other bits to this register. (The logical AND between the value which has been written and the current register value is actually written to the register.) Sequential operation using multiple channels is available by using these match flags.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MF1	MF0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
1	MF1	0	R/W	Channel 1 Condition Match Flag This flag is set to 1 when the channel 1 match condition has been satisfied. To clear the flag, write 0 to this bit. 0: Channel 1 match condition has not been satisfied. 1: Channel 1 match condition has been satisfied.
0	MF0	0	R/W	Channel 0 Condition Match Flag This flag is set to 1 when the channel 0 match condition has been satisfied. To clear the flag, write 0 to this bit. 0: Channel 0 match condition has not been satisfied. 1: Channel 0 match condition has been satisfied.

49.2.9 Break Control Register (CBCR)

CBCR is a readable/writable 32-bit register which specifies whether or not to use the user break debugging support function. For details on the user break debugging support function, refer to section 49.4, User Break Debugging Support Function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	UBDE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved For read/write in this bit, refer to General Precautions on Handling of Product.
0	UBDE	0	R/W	User Break Debugging Support Function Enable Specifies whether or not to use the user break debugging support function. 0: Does not use the user break debugging support function. 1: Uses the user break debugging support function.

49.3 Operation Description

49.3.1 Definition of Words Related to Accesses

"Instruction fetch" refers to an access in which an instruction is fetched. For example, fetching the instruction located at the branch destination after executing a branch instruction is an instruction access. "Operand access" refers to any memory access accompanying execution of an instruction. For example, accessing an address (PC + disp × 2 + 4) in the instruction MOV.W@(disp,PC),Rn is an operand access. "Data" is used in contrast to "address".

All types of operand access are classified into read or write access. Special care must be taken in using the following instructions.

- PREF, OCBP, and OCBWB: Instructions for a read access
- MOVCA.L and OCBI: Instructions for a write access
- TAS.B: Instruction for a single read access or a single write access

The operand access accompanying the PREF, OCBP, OCBWB, and OCBI instructions is access without the data value; therefore, do not include the data value in the match conditions for these instructions.

The operand size should be defined for all types of operand access. Available operand sizes are byte, word, longword, and quadword. For operand access accompanying the PREF, OCBP, OCBWB, MOVCA.L, and OCBI instructions, the operand size is defined as longword.

49.3.2 User Break Operation Sequence

The following describes the sequence from when the break condition is set until the user break exception handling is initiated.

1. Specify the operand size, bus, instruction fetch/operand access, and read/write as the match conditions using the match condition setting register (CBR0 or CBR1). Specify the break address using the match address setting register (CAR0 or CAR1), and specify the address mask condition using the match address mask setting register (CAMR0 or CAMR1). To include the ASID in the match conditions, set the AIE bit in the match condition setting register and specify the ASID value by the AIV bit in the same register. To include the data value in the match conditions, set the DBE bit in the match condition setting register; specify the break data using the match data setting register (CDR1); and specify the data mask condition using the match data mask setting register (CDMR1). To include the execution count in the match conditions, set the ETBE bit of the match condition setting register; and specify the execution count using the execution count break register (CETR1). To use the sequential

break, set the MFE bit of the match condition setting register; and specify the number of the first channel using the MFI bit.

2. Specify whether or not to request a break when the match condition is satisfied and the break timing when the match condition is satisfied as a result of fetching the instruction using the match operation setting register (CRR0 or CRR1). After having set all the bits in the match condition setting register except the CE bit and the other necessary registers, set the CE bit and read the match condition setting register again. This ensures that the set values in the control registers are valid for the subsequent instructions immediately after reading the register. Setting the CE bit of the match condition setting register in the initial state after reset via the control registers may cause an undesired break.
3. When the match condition has been satisfied, the corresponding condition match flag (MF1 or MF0) in the channel match flag register (CCMFR) is set. A break is also requested to the CPU according to the set values in the match operation setting register (CRR0 or CRR1). The CPU operates differently according to the BL bit value of the SR register: when the BL bit is 0, the CPU accepts the break request and executes the specified exception handling; and when the BL bit is 1, the CPU does not execute the exception handling.
4. The match flags (MF1 and MF0) can be used to confirm whether or not the corresponding match condition has been satisfied. Although the flag is set when the condition is satisfied, it is not cleared automatically; therefore, write 0 to the flag bit by issuing a memory store instruction to the channel match flag register (CCMFR) in order to use the flag again.
5. Breaks may occur virtually at the same time for channels 0 and 1. In this case, only one break request is sent to the CPU; however, the two condition match flags corresponding to these breaks may be set.
6. While the BL bit in the SR register is 1, no break requests are accepted. However, whether or not the condition has been satisfied is determined. When the condition is determined to be satisfied, the corresponding condition match flag is set.
7. If the sequential break conditions are set, the condition match flag is set every time the match conditions are satisfied for each channel. When the conditions have been satisfied for the first channel in the sequence but not for the second channel in the sequence, clear the condition match flag for the first channel in the sequence in order to release the first channel in the sequence from the match state.

49.3.3 Instruction Fetch Cycle Break

1. If the instruction fetch cycle is set in the match condition setting register (CBR0 or CBR1), the instruction fetch cycle is handled as a match condition. To request a break upon satisfying the match condition, set the BIE bit in the match operation setting register (CRR0 or CRR1) of the corresponding channel. Either before or after executing the instruction can be selected as the break timing according to the PCB bit value. If the instruction fetch cycle is specified as a match condition, be sure to clear the LSB to 0 in the match address setting register (CAR0 or CAR1); otherwise, no break occurs.
2. If pre-instruction-execution break is specified for the instruction fetch cycle, the break is requested when the instruction is fetched and determined to be executed. Therefore, this function cannot be used for the instructions which are fetched through overrun (i.e., the instructions fetched during branching or making transition to the interrupt routine but not executed). For priorities of pre-instruction-execution break and the other exceptions, refer to section 5, Exception Handling. If pre-instruction-execution break is specified for the delayed slot of the delayed branch instruction, the break is requested before the delayed branch instruction is executed. However, do not specify pre-instruction-execution break for the delayed slot of the RTE instruction.
3. If post-instruction-execution break is specified for the instruction fetch cycle, the break is requested after the instruction which satisfied the match condition has been executed and before the next instruction is executed. Similar to pre-instruction-execution break, this function cannot be used for the instructions which are fetched through overrun. For priorities of post-instruction-execution break and the other exceptions, refer to section 5, Exception Handling. If post-instruction-execution break is specified for the delayed branch instruction and its delayed slot, the break does not occur until the first instruction at the branch destination.
4. If the instruction fetch cycle is specified as the channel 1 match condition, the DBE bit of match condition setting register CBR1 becomes invalid, the settings of match data setting register CDR1 and match data mask setting register CDMR1 are ignored. Therefore, the data value cannot be specified for the instruction fetch cycle break.

49.3.4 Operand Access Cycle Break

1. Table 49.4 shows the relation between the operand sizes specified using the match condition setting register (CBR0 or CBR1) and the address bits to be compared for the operand access cycle break.

Table 49.4 Relation between Operand Sizes and Address Bits to be Compared

Selected Operand Size	Address Bits to be Compared
Quadword	Address bits A31 to A3
Longword	Address bits A31 to A2
Word	Address bits A31 to A1
Byte	Address bits A31 to A0
Operand size is not included in the match conditions	Address bits A31 to A3 for quadword access Address bits A31 to A2 for longword access Address bits A31 to A1 for word access Address bits A31 to A0 for byte access

The above table means that if address H'00001003 is set in the match address setting register (CAR0 or CAR1), for example, the match condition is satisfied for the following access cycles (assuming that all the other conditions are satisfied):

— Longword access to address H'00001000
— Word access to address H'00001002
— Byte access to address H'00001003

2. When the data value is included in the channel 1 match conditions:

   If the data value is included in the match conditions, be sure to select the quadword, longword, word, or byte as the operand size using the operand size select bit (SZ) of the match condition setting register (CBR1), and also set the match data setting register (CDR1) and the match data mask setting register (CDMR1). With these settings, the match condition is satisfied when both of the address and data conditions are satisfied. The data value and mask control for byte access, word access, and longword access should be set in bits 7 to 0, 15 to 0, and 31 to 0 in the bits CDR1 and CDMR1, respectively. For quadword access, 64-bit data is divided into the upper and lower 32-bit data units, and each unit is independently compared with the specified condition. When either the upper or lower 32-bit data unit satisfies the match condition, the match condition for the 64-bit data is determined to be satisfied.

3. The operand access accompanying the PREF, OCBP, OCBWB, and OCBI instructions are access without the data value; therefore, if the data value is included in the match conditions for these instructions, the match conditions will never be satisfied.

4. If the operand bus is selected, a break occurs after executing the instruction which has satisfied the conditions and immediately before executing the next instruction. However, if the data value is included in the match conditions, a break may occur after executing several instructions after the instruction which has satisfied the conditions; therefore, it is impossible to identify the instruction causing the break. If such a break has occurred for the delayed branch instruction or its delayed slot, the break does not occur until the first instruction at the branch destination.

   However, do not specify the operand break for the delayed slot of the RTE instruction. And if the data value is included in the match conditions, it is not allowed to set the break for the preceding the RTE instruction by one to six instructions.

49.3.5 Sequential Break

1. Sequential break conditions can be specified by setting the MFE and MFI bits in the match condition setting registers (CBR0 and CBR1). (Sequential break involves two cases such that channel 0 break condition is satisfied then channel 1 break condition is satisfied, and vice versa.) To use the sequential break function, clear the MFE bit of the match condition setting register and the BIE bit of the match operation setting register of the first channel in the sequence, and set the MFE bit and specify the number of the second channel in the sequence using the MFI bit in the match condition setting register of the second channel in the sequence. If the sequential break condition is set, the condition match flag is set every time the match condition is satisfied for each channel. When the condition has been satisfied for the first channel in the sequence but not for the second channel in the sequence, clear the condition match flag for the first channel in the sequence in order to release the first channel in the sequence from the match state.
2. For channel 1, the execution count break condition can also be included in the sequential break conditions.
3. If the match conditions for the first and second channels in the sequence are satisfied within a significantly short time, sequential operation may not be guaranteed in some cases, as shown below.

- When the Match Condition is Satisfied at the Instruction Fetch Cycle for Both the First and Second Channels in the Sequence:

Instruction B is 0 instruction after instruction A	Equivalent to setting the same addresses; do not use this setting.
Instruction B is one instruction after instruction A	Sequential operation is not guaranteed.
Instruction B is two or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the instruction fetch cycle for the first channel in the sequence whereas the match condition is satisfied at the operand access cycle for the second channel in the sequence:

Instruction B is 0 or one instruction after instruction A	Sequential operation is not guaranteed.
Instruction B is two or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the operand access cycle for the first channel in the sequence whereas the match condition is satisfied at the instruction fetch cycle for the second channel in the sequence:

Instruction B is 0 to five instructions after instruction A	Sequential operation is not guaranteed.
Instruction B is six or more instructions after instruction A	Sequential operation is guaranteed.

- When the match condition is satisfied at the operand access cycle for both the first and second channels in the sequence:

Instruction B is 0 to five instructions after instruction A	Sequential operation is not guaranteed.
Instruction B is six or more instructions after instruction A	Sequential operation is guaranteed.

49.3.6 Program Counter Value to be Saved

When a break has occurred, the address of the instruction to be executed when the program restarts is saved in the SPC then the exception handling state is initiated. A unique instruction causing a break can be identified unless the data value is included in the match conditions.

1. When the instruction fetch cycle (before instruction execution) is specified as the match condition:

   The address of the instruction which has satisfied the match conditions is saved in the SPC. The instruction which has satisfied the match conditions is not executed, but a break occurs instead. However, if the match conditions are satisfied for the delayed slot instruction, the address of the delayed branch instruction is saved in the SPC.
2. When the instruction fetch cycle (after instruction execution) is specified as the match condition:

   The address of the instruction immediately after the instruction which has satisfied the match conditions is saved in the SPC. The instruction which has satisfied the match conditions is executed, then a break occurs before the next instruction. If the match conditions are satisfied for the delayed branch instruction or its delayed slot, these instructions are executed and the address of the branch destination is saved in the SPC.
3. When the operand access (address only) is specified as the match condition:

   The address of the instruction immediately after the instruction which has satisfied the break conditions is saved in the SPC. The instruction which has satisfied the match conditions are executed, then a break occurs before the next instruction. However, if the conditions are satisfied for the delayed slot, the address of the branch destination is saved in the SPC.
4. When the operand access (address and data) is specified as the match condition:

   If the data value is added to the match conditions, the instruction which has satisfied the match conditions is executed. A user break occurs before executing an instruction that is one through six instructions after the instruction which has satisfied the match conditions. The address of the instruction is saved in the SPC; thus, it is impossible to identify exactly where a break will occur. If the conditions are satisfied for the delayed slot instruction, the address of the branch destination is saved in the SPC. If a branch instruction follows the instruction which has satisfied the match conditions, a break may occur after the delayed instruction and delayed slot are executed. In this case, the address of the branch destination is also saved in the SPC.

49.4 User Break Debugging Support Function

By using the user break debugging support function, the branch destination address can be modified when the CPU accepts the user break request. Specifically, setting the UBDE bit of break control register CBCR to 1 allows branching to the address indicated by DBR instead of branching to the address indicated by the [VBR + offset]. Figure 49.2 shows the flowchart of the user break debugging support function.

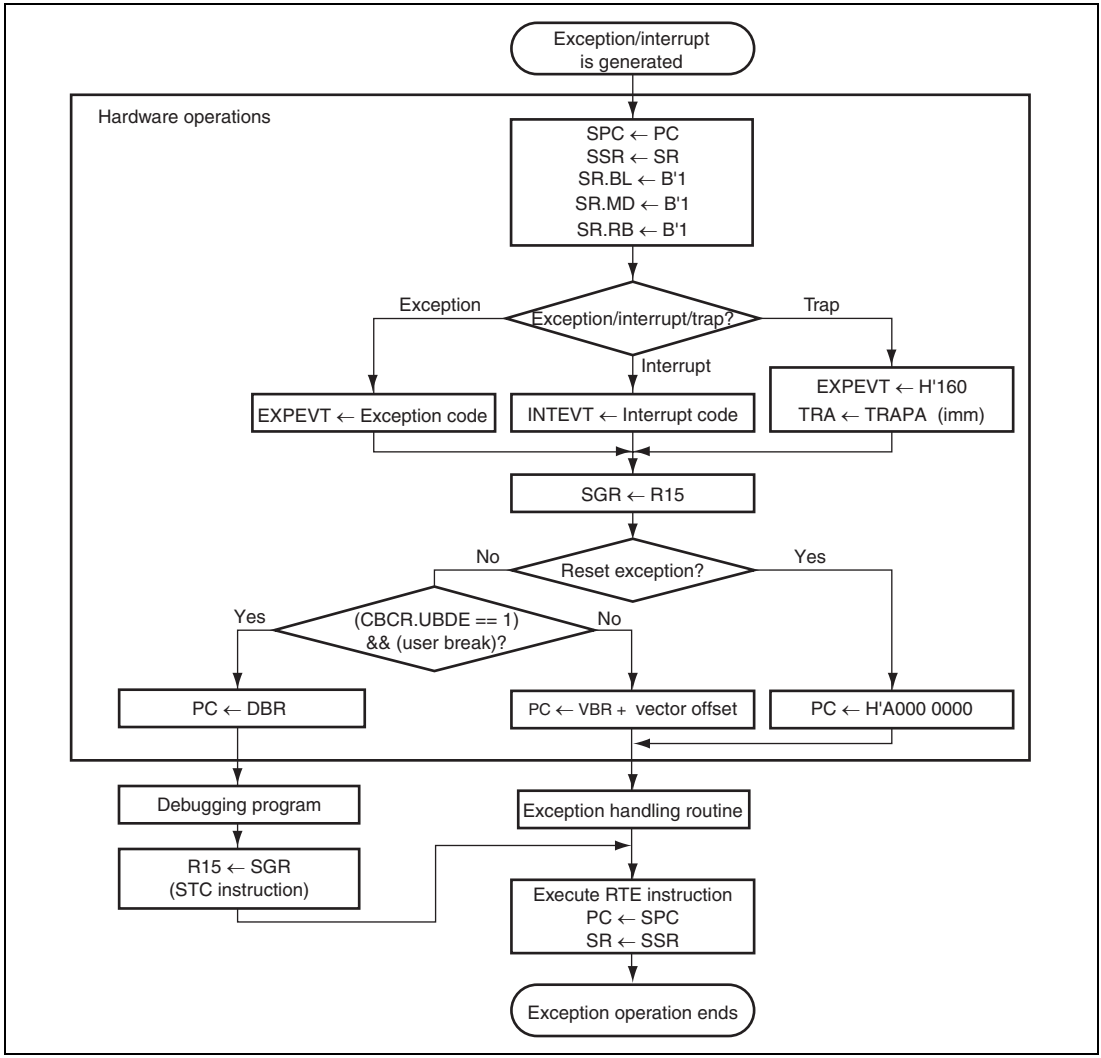


Figure 49.2 Flowchart of User Break Debugging Support Function

49.5 User Break Examples

(1) Match Conditions are Specified for an Instruction Fetch Cycle

- Example 1-1

  Register settings: CBR0 = H'00000013 / CRR0 = H'00002003 / CAR0 = H'00000404 / CAMR0 = H'00000000 / CBR1 = H'00000013 / CRR1 = H'00002001 / CAR1 = H'00008010 / CAMR1 = H'00000006 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  Channel 0

  Address: H'00000404 / Address mask: H'00000000

  Bus cycle: Instruction fetch (after executing the instruction)

  ASID is not included in the conditions.

  Channel 1:

  Address: H'00008010 / Address mask: H'00000006

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing instruction)

  ASID, data values, and execution count are not included in the conditions.

  With the above settings, the user break occurs after executing the instruction at address H'00000404 or before executing the instruction at address H'00008010 to H'00008016.

- Example 1-2

  Register settings: CBR0 = H'40800013 / CRR0 = H'00002000 / CAR0 = H'00037226 / CAMR0 = H'00000000 / CBR1 = H'C0700013 / CRR1 = H'00002001 / CAR1 = H'0003722E / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Channel 0 → Channel1 sequential mode

  Channel 0

  Address: H'00037226 / Address mask: H'00000000 / ASID: H'80

  Bus cycle: Instruction fetch (before executing the instruction)

  Channel 1

  Address: H'0003722E / Address mask: H'00000000 / ASID: H'70

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

With the above settings, the user break occurs after executing the instruction at address H'00037226 where ASID is H'80 before executing the instruction at address H'0003722E where ASID is H'70.

- Example 1-3

  Register settings: CBR0 = H'00000013 / CRR0 = H'00002001 / CAR0 = H'00027128 / CAMR0 = H'00000000 / CBR1 = H'00000013 / CRR1 = H'00002001 / CAR1 = H'00031415 / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  Channel 0

  Address: H'00027128 / Address mask: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  ASID is not included in the conditions.

  Channel 1

  Address: H'00031415 / Address mask: H'00000000

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  ASID, data values, and execution count are not included in the conditions.

  With the above settings, the user break occurs for channel 0 before executing the instruction at address H'00027128. No user break occurs for channel 1 since the instruction fetch is executed only at even addresses.

- Example 1-4

  Register settings: CBR0 = H'40800013 / CRR0 = H'00002000 / CAR0 = H'00037226 / CAMR0 = H'00000000 / CBR1 = H'C0700013 / CRR1 = H'00002001 / CAR1 = H'0003722E / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H′00000000 / CBCR = H'00000000

  Specified conditions: Channel 0 → Channel 1 sequential mode

  Channel 0

  Address: H'00037226 / Address mask: H'00000000 / ASID: H'80

  Bus cycle: Instruction fetch (before executing the instruction)

  Channel 1

  Address: H'0003722E / Address mask: H'00000000 / ASID: H'70

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  Data values and execution count are not included in the conditions.

With the above settings, the user break occurs after executing the instruction at address H'00037226 where ASID is H'80 and before executing the instruction at address H'0003722E where ASID is H'70.

- Example 1-5

  Register settings: CBR0 = H'00000013 / CRR0 = H'00002001 / CAR0 = H'00000500 / CAMR0 = H'00000000 / CBR1 = H'00000813 / CRR1 = H'00002001 / CAR1 = H'00001000 / CAMR1 = H'00000000 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000005 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  Channel 0

  Address: H'00000500 / Address mask: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

  ASID is not included in the conditions.

  Channel 1

  Address: H'00001000 / Address mask: H'00000000

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000005

  Bus cycle: Instruction fetch (before executing the instruction)

  Execution count: 5

  ASID and data values are not included in the conditions.

  With the above settings, the user break occurs for channel 0 before executing the instruction at address H'00000500. The user break occurs for channel 1 after executing the instruction at address H'00001000 four times; before executing the instruction five times.

- Example 1-6

  Register settings: CBR0 = H'40800013 / CRR0 = H'00002003 / CAR0 = H'00008404 / CAMR0 = H'00000FFF / CBR1 = H'40700013 / CRR1 = H'00002001 / CAR1 = H'00008010 / CAMR1 = H'00000006 / CDR1 = H'00000000 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  Channel 0

  Address: H'00008404 / Address mask: H'00000FFF / ASID: H'80

  Bus cycle: Instruction fetch (after executing the instruction)

  Channel 1

  Address: H'00008010 / Address mask: H'00000006 / ASID: H'70

  Data: H'00000000 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Instruction fetch (before executing the instruction)

Data values and execution count are not included in the conditions.

With the above settings, the user break occurs after executing the instruction at address H'00008000 to H'00008FFE where ASID is H'80 or before executing the instruction at address H'00008010 to H'00008016 where ASID is H'70.

(2) Match Conditions are Specified for an Operand Access Cycle

- Example 2-1

  Register settings: CBR0 = H'40800023 / CRR0 = H'00002001 / CAR0 = H'00123456 / CAMR0 = H'00000000 / CBR1 = H'4070A025 / CRR1 = H'00002001 / CAR1 = H'000ABCDE / CAMR1 = H'000000FF / CDR1 = H'0000A512 / CDMR1 = H'00000000 / CETR1 = H'00000000 / CBCR = H'00000000

  Specified conditions: Independent for channels 0 and 1

  Channel 0

  Address: H'00123456 / Address mask: H'00000000 / ASID: H'80

  Bus cycle: Operand bus, operand access, and read (operand size is not included in the conditions.)

  Channel 1

  Address: H'000ABCDE / Address mask: H'000000FF / ASID: H'70

  Data: H'0000A512 / Data mask: H'00000000 / Execution count: H'00000000

  Bus cycle: Operand bus, operand access, write, and word size

  Execution count is not included in the conditions.

  With these settings, the user break occurs for channel 0 for the following accesses: longword read access to address H'000123454, word read access to address H'000123456, byte read access to address H'000123456 where ASID is H'80. The user break occurs for channel 1 when word H'A512 is written to address H'000ABC00 to H'000ABCFE where ASID is H'70.

49.6 Usage Notes

1. A desired break may not occur between the time when the instruction for rewriting the UBC register is executed and the time when the written value is actually reflected on the register.

   After the UBC register is updated, execute one of the following three methods.

   A. Read the updated UBC register, and execute a branch using the RTE instruction.
   (It is not necessary that a branch using the RTE instruction is next to a reading UBC register.)

   B. Execute the ICBI instruction for any address (including non-cacheable area).
   (It is not necessary that the ICBI instruction is next to a reading UBC register.)

   C. Set 0(initial value) to IRMCR.R1 before updating the UBC register and update with following sequence.

      a. Write the UBC register.

      b. Read the UBC register which is updated at 1.

      c. Write the value which is read at 2 to the UBC register.

Note: When two or more UBC registers are updated, executing these methods at each updating the UBC registers is not necessary. At only last updating the UBC register, execute one of these methods.

2. The PCB bit of the CRR0 and CRR1 registers is valid only when the instruction fetch is specified as the match condition.
3. If the sequential break conditions are set, the sequential break conditions are satisfied when the conditions for the first and second channels in the sequence are satisfied in this order. Therefore, if the conditions are set so that the conditions for channels 0 and 1 should be satisfied simultaneously for the same bus cycle, the sequential break conditions will not be satisfied, causing no break.
4. For the SLEEP instruction, do not allow the post-instruction-execution break where the instruction fetch cycle is the match condition. For the instructions preceding the SLEEP instruction by one to five instructions, do not allow the break where the operand access is the match condition.

5. If the user break and other exceptions occur for the same instruction, they are determined according to the specified priority. For the priority, refer to section 5, Exception Handling. If the exception having the higher priority occurs, the user break does not occur.

- The pre-instruction-execution break is accepted prior to any other exception.
- If the post-instruction-execution break and data access break have occurred simultaneously with the re-execution type exception (including the pre-instruction-execution break) having a higher priority, only the re-execution type exception is accepted, and no condition match flags are set. When the exception handling has finished thus clearing the exception source, and when the same instruction has been executed again, the break occurs setting the corresponding flag.
- If the post-instruction-execution break or operand access break has occurred simultaneously with the completion-type exception (TRAPA) having a higher priority, then no user break occurs; however, the condition match flag is set.

6. When conditions have been satisfied simultaneously and independently for channels 0 and 1, resulting in identical SPC values for both of the breaks, the user break occurs only once. However, the condition match flags are set for both channels. For example,

   Instruction at address 110 (post-instruction-execution break for instruction fetch for channel 0) $\rightarrow$ SPC = 112, CCMFR.MF0 = 1

   Instruction at address 112 (pre-instruction-execution break for instruction fetch for channel 1) $\rightarrow$ SPC = 112, CCMFR.MF1 = 1

7. It is not allowed to set the pre-instruction-execution break or the operand break in the delayed slot instruction of the RTE instruction. And if the data value is included in the match conditions of the operand break, do not set the break for the preceding the RTE instruction by one to six instructions.
8. If the re-execution type exception and the post-instruction-execution break are in conflict for the instruction requiring two or more execution states, then the re-execution type exception occurs. Here, the CCMFR.MF0 (or CCMFR.MF1) bit may or may not be set to 1 when the break conditions have been satisfied.

Section 50 User Debugging Interface (H-UDI)

The H-UDI is a serial interface which conforms to the JTAG (IEEE 1149.4: IEEE Standard Test Access Port and Boundary-Scan Architecture) standard. The H-UDI is also used for emulator connection.

50.1 Features

The H-UDI is a serial interface which conforms to the JTAG standard. The H-UDI is also used for emulator connection. When using an emulator, H-UDI functions should not be used. Refer to the appropriate emulator users manual for the method of connecting the emulator.

The H-UDI has six pins: TCK, TMS, TDI, TDO, $\overline{\text{TRST}}$, and $\overline{\text{ASEBRK}}$/BRKACK. The pin functions except $\overline{\text{ASEBRK}}$/BRKACK and serial communications protocol conform to the JTAG standard. This LSI has additional six pins for emulator connection: (AUDSYNC, AUDCK, and AUDATA3 to AUDATA0).

Figure 50.1 shows a block diagram of the H-UDI.

The TAP (Test Access Port) controller and five registers (SDBPR, SDIR, SDDRH, SDDRL, and SDINT). SDBPR supports the JTAG bypass mode, SDIR is used for commands, SDDR is used for data, and SDINT is used for H-UDI interrupts. SDIR is directly accessed from the TDI and TDO pins.

The TAP controller and control registers are initialized by driving the $\overline{\text{TRST}}$ pin low or by applying the TCK signal for five or more clock cycles with the TMS pin set to 1. This initialization sequence is independent of the reset pin for this LSI. Other circuits are initialized by a normal reset.

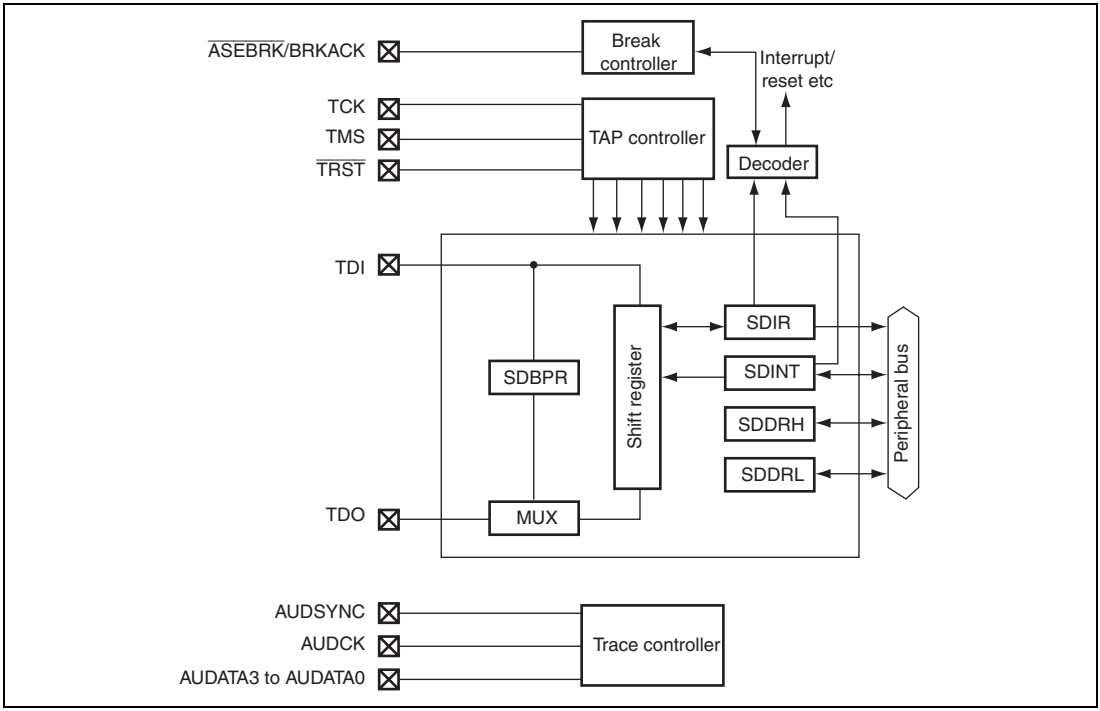


Figure 50.1 H-UDI Block Diagram

50.2 Input/Output Pins

Table 50.1 shows the pin configuration for the H-UDI.

Table 50.1 Pin Configuration

Pin Name	Function	I/O	Description	When Not in Use
TCK	Clock	Input	Functions as the serial clock input pin stipulated in the JTAG standard. Data input to the H-UDI via the TDI pin or data output via the TDO pin is performed in synchronization with this signal.	Open*[1]
TMS	Mode	Input	Mode Select Input Changing this signal in synchronization with the TCK signal determines the significance of data input via the TDI pin. Its protocol conforms to the JTAG standard (IEEE standard 1149.1).	Open*[1]
$\overline{\text{TRST}}$*[2]	Reset	Input	H-UDI Reset Input This signal is received asynchronously with a TCK signal. Asserting this signal resets the JTAG interface circuit. When a power is supplied, the $\overline{\text{TRST}}$ pin should be asserted for a given period regardless of whether or not the JTAG function is used, which differs from the JTAG standard.	Fixed to ground*[3]
TDI	Data input	Input	Data Input Data is sent to the H-UDI by changing this signal in synchronization with the TCK signal.	Open*[1]
TDO	Data output	Output	Data Output Data is read from the H-UDI in synchronization with the TCK signal.	Open
$\overline{\text{ASEBRK}}$/ BRKACK	Emulator	I/O	Pins for an emulator	Open*[1]
AUDSYNC, AUDCK, AUDATA3 to AUDATA0	Emulator	Output	Pins for an emulator	Open
MPMD	ASE (Emulation support mode setting)	Input	A low level on this pin places the chip in ASE mode, enabling use of the emulation support mode functions. When using an emulator such as the E10A, fix this pin at a low level.	Open*[1]

Notes: 1. This pin is pulled up in this LSI. When designing a board emulator is available or using interrupts or resets via the H-UDI or emulator, the use of external pull-up resistors will not cause any problem.

2. When designing a board emulator is available or using interrupts or resets via the H-UDI or emulator, the $\overline{\text{TRST}}$ pin should be designed so that it can be controlled independently and can be controlled to retain low level while the $\overline{\text{RESET}}$ pin is asserted at a power-on reset.

3. This pin should be connected to ground. However, when connected to a ground pin, the following problem occurs. Since the $\overline{\text{TRST}}$ pin is pulled up within this LSI, a weak current flows when the pin is externally connected to a ground pin. The value of the current is determined by a resistance of the pull-up MOS for the port pin. Although this current does not affect the operation of this LSI, it consumes unnecessary power. Pulling up the $\overline{\text{TRST}}$ pin can be disabled by the pull-down control register (PULCR) of the pin function controller (PFC). For details, see section 48, Pin Function Controller (PFC).

The TCK clock or the CPG of this LSI should be set to ensure that the frequency of the TCK clock is less than the peripheral-clock frequency of this LSI.

50.3 Register Descriptions

The H-UDI has the following registers.

Table 50.2 Register Configuration (1)

		CPU Side				
Register Name	**Abbreviation**	**R/W**	**Area P4 Address*[1]**	**Area 7 Address*[1]**	**Size**	**Initial Value*[2]**
Instruction register	SDIR	R	H'FC11 0000	H'1C11 0000	16	H'0EFF
Data register H	SDDR/SDDRH	R/W	H'FC11 0008	H'1C11 0008	32/16	Undefined
Data register L	SDDRL	R/W	H'FC11 000A	H'1C11 000A	16	Undefined
Interrupt source register	SDINT	R/W	H'FC11 0018	H'1C11 0018	16	H'0000
Bypass register	SDBPR	—	—	—	—	Undefined

Notes: 1. The area P4 address is an address when accessing through area P4 in a virtual address space. The area 7 address is an address when accessing through area 7 in a physical space using the TLB.
2. The low level of the $\overline{\text{TRST}}$ pin or the Test-Logic-Reset state of the TAP controller initializes to these values.

Table 50.3 Register Configuration (2)

		H-UDI Side		
Register Name	**Abbreviation**	**R/W**	**Size**	**Initial Value*[1]**
Instruction register	SDIR	R/W	32	H'FFFF FFFD (fixed value*[2])
Data register H	SDDR/SDDRH	—	—	—
Data register L	SDDRL	—	—	—
Interrupt source register	SDINT	W*[3]	32	H'0000 0000
Bypass register	SDBPR	R/W	1	Undefined

Notes: 1. The low level of the $\overline{\text{TRST}}$ pin or the Test-Logic-Reset state of the TAP controller initializes to these values.
2. When reading via the H-UDI, the value is always H'FFFF FFFD.
3. Only 1 can be written to the LSB by the H-UDI interrupt command.

Table 50.4 Register States in Each Operating Mode

Register Abbreviation	Power-On Reset	Manual Reset	Software Standby	Module Standby	R/U-Standby	Sleep
SDIR	H'0EFF	Retained	Retained	Retained	Retained*	Retained
SDDR/SDDRH	Undefined	Retained	Retained	Retained	Retained*	Retained
SDDRL	Undefined	Retained	Retained	Retained	Retained*	Retained
SDINT	H'0000	Retained	Retained	Retained	Retained*	Retained

Note: * Has this state when the MPMD pin is low. When the MPMD pin is high, has the same state as at a power-on reset.

50.3.1 Instruction Register (SDIR)

SDIR is a 16-bit read-only register that can be read from the CPU. Commands are set via the serial input (TDI). SDIR is initialized by $\overline{\text{TRST}}$ or in the Test-Logic-Reset state and can be written by the H-UDI irrespective of the CPU mode. Operation is not guaranteed when a reserved command is set to this register.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TI								—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	TI	B'00001110	R	Test Instruction Bits 7 to 0 11111111: Bypass 01010101: IDCODE 01000000: SAMPLE/PRELOAD 00000000: EXTEST Other than above: Setting prohibited
7 to 0	—	All 1	R	Reserved These bits are always read as 1.

50.3.2 Data Register H and L (SDDRH and SDDRL)

SDDR is a 32-bit register that comprises two registers: SDDRH and SDDRL. SDDRH and SDDRL are also 16-bit registers that can be read from or written to by the CPU. SDDR as a 32-bit register can be read from or written to by the CPU. The register value is not initialized by a reset for the CPU but is initialized by $\overline{\text{TRST}}$.

- SDDRH

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SDDRH data															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- SDDRL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDDRL data															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

50.3.3 Interrupt Source Register (SDINT)

SDINT is a 16-bit register that can be read from or written to by the CPU. Specifying an H-UDI interrupt command in SDIR via H-UDI pin (Update-IR) sets the INTREQ bit to 1. While an H-UDI interrupt command is set in SDIR, SDINT which is connected between the TDI and TDO pins can be read as a 32-bit register. In this case, the upper 16 bits will be 0 and the lower 16 bits represent the SDINT value.

Only 0 can be written to the INTREQ bit by the CPU. While this bit is set to 1, an interrupt request will continue to be generated. This bit, therefore, should be cleared by the interrupt handling routine. It is initialized by $\overline{\text{TRST}}$ or in the Test-Logic-Reset state.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	INTREQ
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved For reading from or writing to these bits, see General Precautions on Handling of Product.
0	INTREQ	0	R/W	Interrupt Request Indicates whether or not an interrupt by an H-UDI interrupt command has occurred. Clearing this bit to 0 by the CPU cancels an interrupt request. When writing 1 to this bit, the previous value is maintained.

50.4 Operation

50.4.1 TAP Control

Figure 50.2 shows the internal states of the TAP controller. The state transitions basically conform to the JTAG standard.

- State transitions occur according to the TMS value at the rising edge of the TCK signal.
- The TDI value is sampled at the rising edge of the TCK signal and shifted at the falling edge of the TCK signal.
- The TDO value is changed at the falling edge of the TCK signal. The TDO signal is in a Hi-Z state other than in the Shift-DR or Shift-IR state.
- A transition to the Test-Logic-Reset by clearing $\overline{\text{TRST}}$ to 0 is performed asynchronously with the TCK signal.

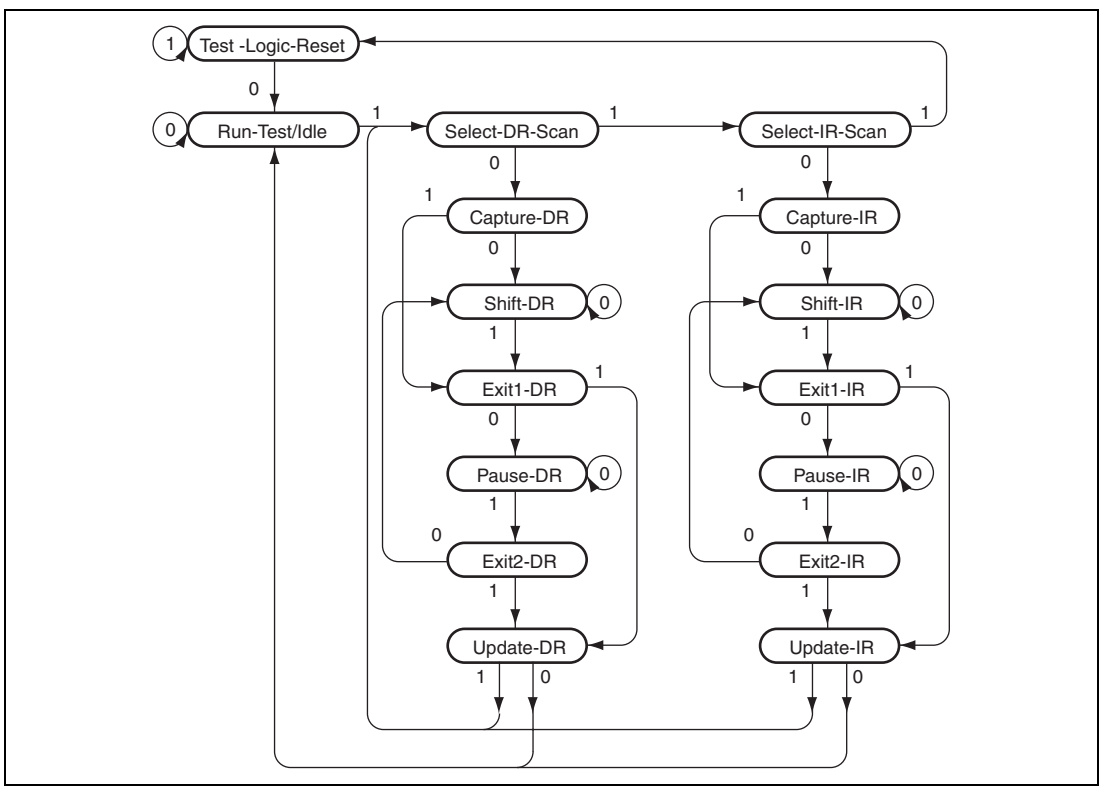


Figure 50.2 TAP Controller State Transitions

50.4.2 H-UDI Reset

A power-on reset is generated by the SDIR command. After the H-UDI reset assert command has been sent from the H-UDI pin, sending the H-UDI reset negate command resets the CPU (see Figure 50.3). The required time between the H-UDI reset assert and H-UDI reset negate commands is the same as the time for holding the reset pin low in order to reset this LSI by a power-on reset.

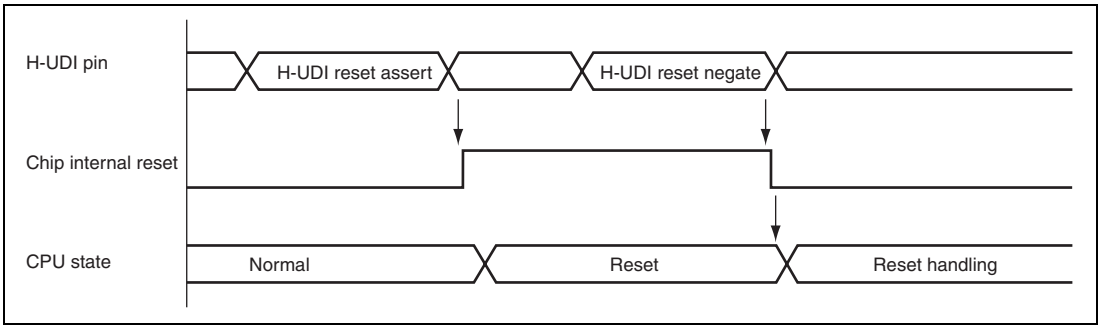


Figure 50.3 H-UDI Reset

50.4.3 H-UDI Interrupt

The H-UDI interrupt function generates an interrupt by setting the appropriate command in SDIR from the H-UDI. An H-UDI interrupt request signal is asserted when the INTREQ bit in SDINT is set to 1 by setting the appropriate command (Update-IR). Since the interrupt request signal is not negated until the INTREQ bit is cleared to 0 by software, it is not possible to lose the interrupt request. While an H-UDI interrupt command is set in SDIR, SDINT is connected between the TDI and TDO pins.

50.5 Usage Notes

- Once an SDIR command is set, it will be changed only by an assertion of the $\overline{\text{TRST}}$ signal, making the TAP controller Test-Logic-Reset state, or writing other commands from the H-UDI.
- The H-UDI is used for emulator connection. Therefore, H-UDI functions cannot be used when using an emulator.

Section 51 List of Registers

The list of registers gives information on the on-chip I/O registers and is configured as described below.

1. Register Addresses (by functional module, in order of the corresponding section numbers):

- Descriptions by functional module, in order of the corresponding section numbers
- Access to reserved addresses which are not described in this list is disabled.
- When registers consist of 16 or 32 bits, the addresses of the MSBs are given, on the presumption of a big-endian system.

2. Register States in Each Operating Mode:

- Register states are described in the same order as the Register Addresses (by functional module, in order of the corresponding section numbers).
- For the initial state of each bit, refer to the description of the register in the corresponding section.
- The register states described are for the basic operating modes. If there is a specific reset for an on-chip module, refer to the section on that on-chip module.

51.1 Register Addresses

Entries under Access Size indicate the numbers of bits.

Note: Access to undefined or reserved addresses is prohibited. Since operation or continued operation is not guaranteed when these registers are accessed, do not attempt such access.

Register Name	Abbreviation	R/W	Address	Module	Access Size
TRAPA exception register	TRA	R/W	H'FF00 0020	Exception Handling	32
Exception event register	EXPEVT	R/W	H'FF00 0024		32
Interrupt event register	INTEVT	R/W	H'FF00 0028		32
Non-support detection exception register	EXPMASK	R/W	H'FF2F 0004		32
Page table entry high register	PTEH	R/W	H'FF00 0000	MMU	32
Page table entry low register	PTEL	R/W	H'FF00 0004		32
Translation table base register	TTB	R/W	H'FF00 0008		32
TLB exception address register	TEA	R/W	H'FF00 000C		32
MMU control register	MMUCR	R/W	H'FF00 0010		32
Page table entry assistance register	PTEA	R/W	H'FF00 0034		32
Physical address space control register	PASCR	R/W	H'FF00 0070		32
Instruction re-fetch inhibit control register	IRMCR	R/W	H'FF00 0078		32
Cache control register	CCR	R/W	H'FF00 001C	Cache	32
On-chip memory control register	RAMCR	R/W	H'FF00 0074		32
Queue address control register 0	QACR0	R/W	H'FF00 0038		32
Queue address control register 1	QACR1	R/W	H'FF00 003C		32
LCK control register	PRLCKCR	R/W	H'FF80 0018	Super Hyway	32
PRI control register 0(SH-X2)	PRPRICR 0	R/W	H'FF80 0028		32
PRI control register 1(DBG)	PRPRICR 1	R/W	H'FF80 0030		32
PRI control register 2(DMAC0)	PRPRICR 2	R/W	H'FF80 0038		32
PRI control register 3(DMAC1)	PRPRICR 3	R/W	H'FF80 0040		32
PRI control register 4	Reserved	R/W	H'FF80 0048		32
PRI control register 5(LMB)	PRPRICR 5	R/W	H'FF80 0050		32
PRI control register 6(2DGcmd)	PRPRICR 6	R/W	H'FF80 0058		32
PRI control register 7(2DG)	PRPRICR 7	R/W	H'FF80 0060		32
PRI control register 8(ATAPI)	PRPRICR 8	R/W	H'FF80 0068		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
PRI control register 9(SPU2)	PRPRICR 9	R/W	H'FF80 0070	Super Hyway	32
PRI control register 10(Ether)	PRPRICR 10	R/W	H'FF80 0078		32
Interrupt control register 0	ICR0	R/W	H'A414 0000	INTC	16
Interrupt control register 1	ICR1	R/W	H'A414 001C		16
Interrupt priority register 00	INTPRI00	R/W	H'A414 0010		32
Interrupt request register 00	INTREQ00	R/W	H'A414 0024		8
Interrupt mask register 00	INTMSK00	R/W	H'A414 0044		8
Interrupt mask clear register 00	INTMSKCLR00	W	H'A414 0064		8
NMI flag control register	NMIFCR	R/W	H'A414 00C0		16
User interrupt mask level register	USERIMSK	R/W	H'A470 0000		32
Interrupt priority register A	IPRA	R/W	H'A408 0000		16
Interrupt priority register B	IPRB	R/W	H'A408 0004		16
Interrupt priority register C	IPRC	R/W	H'A408 0008		16
Interrupt priority register D	IPRD	R/W	H'A408 000C		16
Interrupt priority register E	IPRE	R/W	H'A408 0010		16
Interrupt priority register F	IPRF	R/W	H'A408 0014		16
Interrupt priority register G	IPRG	R/W	H'A408 0018		16
Interrupt priority register H	IPRH	R/W	H'A408 001C		16
Interrupt priority register I	IPRI	R/W	H'A408 0020		16
Interrupt priority register J	IPRJ	R/W	H'A408 0024		16
Interrupt priority register K	IPRK	R/W	H'A408 0028		16
Interrupt priority register L	IPRL	R/W	H'A408 002C		16
Interrupt mask register 0	IMR0	R/W	H'A408 0080		8
Interrupt mask register 1	IMR1	R/W	H'A408 0084		8
Interrupt mask register 2	IMR2	R/W	H'A408 0088		8
Interrupt mask register 3	IMR3	R/W	H'A408 008C		8
Interrupt mask register 4	IMR4	R/W	H'A408 0090		8
Interrupt mask register 5	IMR5	R/W	H'A408 0094		8
Interrupt mask register 6	IMR6	R/W	H'A408 0098		8
Interrupt mask register 7	IMR7	R/W	H'A408 009C		8
Interrupt mask register 8	IMR8	R/W	H'A408 00A0		8
Interrupt mask register 9	IMR9	R/W	H'A408 00A4		8

Register Name	Abbreviation	R/W	Address	Module	Access Size
Interrupt mask register 10	IMR10	R/W	H'A408 00A8	INTC	8
Interrupt mask register 11	IMR11	R/W	H'A408 00AC		8
Interrupt mask register 12	IMR12	R/W	H'A408 00B0		8
Interrupt mask clear register 0	IMCR0	W	H'A408 00C0		8
Interrupt mask clear register 1	IMCR1	W	H'A408 00C4		8
Interrupt mask clear register 2	IMCR2	W	H'A408 00C8		8
Interrupt mask clear register 3	IMCR3	W	H'A408 00CC		8
Interrupt mask clear register 4	IMCR4	W	H'A408 00D0		8
Interrupt mask clear register 5	IMCR5	W	H'A408 00D4		8
Interrupt mask clear register 6	IMCR6	W	H'A408 00D8		8
Interrupt mask clear register 7	IMCR7	W	H'A408 00DC		8
Interrupt mask clear register 8	IMCR8	W	H'A408 00E0		8
Interrupt mask clear register 9	IMCR9	W	H'A408 00E4		8
Interrupt mask clear register 10	IMCR10	W	H'A408 00E8		8
Interrupt mask clear register 11	IMCR11	W	H'A408 00EC		8
Interrupt mask clear register 12	IMCR12	W	H'A408 00F0		8
Memory address-map selection register	MMSELR	R/W	H'FF80 0020	BSC	32
Common control register	CMNCR	R/W	H'FEC1 0000		32
CS0 space bus control register	CS0BCR	R/W	H'FEC1 0004		32
CS4 space bus control register	CS4BCR	R/W	H'FEC1 0010		32
CS5A space bus control register	CS5ABCR	R/W	H'FEC1 0014		32
CS5B space bus control register	CS5BBCR	R/W	H'FEC1 0018		32
CS6A space bus control register	CS6ABCR	R/W	H'FEC1 001C		32
CS6B space bus control register	CS6BBCR	R/W	H'FEC1 0020		32
CS0 space wait control register	CS0WCR	R/W	H'FEC1 0024		32
CS4 space wait control register	CS4WCR	R/W	H'FEC1 0030		32
CS5A space wait control register	CS5AWCR	R/W	H'FEC1 0034		32
CS5B space wait control register	CS5BWCR	R/W	H'FEC1 0038		32
CS6A space wait control register	CS6AWCR	R/W	H'FEC1 003C		32
CS6B space wait control register	CS6BWCR	R/W	H'FEC1 0040		32
Reset bus wait counter	RBWTCNT	—	H'FEC1 0054		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
DBSC2 SDRAM kind select register	DBKIND	R/W	H'FD00 0008	DBSC	32
DBSC2 status register	DBSTATE	R	H'FD00 000C		32
SDRAM operation enable register	DBEN	R/W	H'FD00 0010		32
SDRAM command control register	DBCMDCNT	R/W	H'FD00 0014		32
SDRAM CKE control register	DBCKECNT	R/W	H'FD00 0018		32
SDRAM configuration setting register	DBCONF	R/W	H'FD00 0020		32
SDRAM timing register 0	DBTR0	R/W	H'FD00 0030		32
SDRAM timing register 1	DBTR1	R/W	H'FD00 0034		32
SDRAM timing register 2	DBTR2	R/W	H'FD00 0038		32
SDRAM timing register 3	DBTR3	R/W	H'FD00 003C		32
SDRAM refresh/power-down control register 0	DBRFPDN 0	R/W	H'FD00 0040		32
SDRAM refresh/power-down control register 1	DBRFPDN 1	R/W	H'FD00 0044		32
SDRAM refresh/power-down control register 2	DBRFPDN 2	R/W	H'FD00 0048		32
SDRAM refresh status register	DBRFSTS	R/W	H'FD00 004C		32
SDRAM mode setting register	DBMRCNT	W	H'FD00 0060		32
DDR-PAD control register	DBPDCNT0	R/W	H'FD00 0108		32
DMA0 source address register_0	DMA0_SAR_0	R/W	H'FE00 8020	DMAC	32
DMA0 destination address register_0	DMA0_DAR_0	R/W	H'FE00 8024		32
DMA0 transfer count register_0	DMA0_TCR_0	R/W	H'FE00 8028		32
DMA0 channel control register_0	DMA0_CHCR_0	R/W	H'FE00 802C		32
DMA0 source address register_1	DMA0_SAR_1	R/W	H'FE00 8030		32
DMA0 destination address register_1	DMA0_DAR_1	R/W	H'FE00 8034		32
DMA0 transfer count register_1	DMA0_TCR_1	R/W	H'FE00 8038		32
DMA0 channel control register_1	DMA0_CHCR_1	R/W	H'FE00 803C		32
DMA0 source address register_2	DMA0_SAR_2	R/W	H'FE00 8040		32
DMA0 destination address register_2	DMA0_DAR_2	R/W	H'FE00 8044		32
DMA0 transfer count register_2	DMA0_TCR_2	R/W	H'FE00 8048		32
DMA0 channel control register_2	DMA0_CHCR_2	R/W	H'FE00 804C		32
DMA0 source address register_3	DMA0_SAR_3	R/W	H'FE00 8050		32
DMA0 destination address register_3	DMA0_DAR_3	R/W	H'FE00 8054		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
DMA0 transfer count register_3	DMA0_TCR_3	R/W	H'FE00 8058	DMAC	32
DMA0 channel control register_3	DMA0_CHCR_3	R/W	H'FE00 805C		32
DMA0 operation register	DMA0_DMAOR	R/W	H'FE00 8060		16
DMA0 source address register_4	DMA0_SAR_4	R/W	H'FE00 8070		32
DMA0 destination address register_4	DMA0_DAR_4	R/W	H'FE00 8074		32
DMA0 transfer count register_4	DMA0_TCR_4	R/W	H'FE00 8078		32
DMA0 channel control register_4	DMA0_CHCR_4	R/W	H'FE00 807C		32
DMA0 source address register_5	DMA0_SAR_5	R/W	H'FE00 8080		32
DMA0 destination address register_5	DMA0_DAR_5	R/W	H'FE00 8084		32
DMA0 transfer count register_5	DMA0_TCR_5	R/W	H'FE00 8088		32
DMA0 channel control register_5	DMA0_CHCR_5	R/W	H'FE00 808C		32
DMA0 source address register B_0	DMA0_SARB_0	R/W	H'FE00 8120		32
DMA0 destination address register B_0	DMA0_DARB_0	R/W	H'FE00 8124		32
DMA0 transfer count register B_0	DMA0_TCRB_0	R/W	H'FE00 8128		32
DMA0 source address register B_1	DMA0_SARB_1	R/W	H'FE00 8130		32
DMA0 destination address register B_1	DMA0_DARB_1	R/W	H'FE00 8134		32
DMA0 transfer count register B_1	DMA0_TCRB_1	R/W	H'FE00 8138		32
DMA0 source address register B_2	DMA0_SARB_2	R/W	H'FE00 8140		32
DMA0 destination address register B_2	DMA0_DARB_2	R/W	H'FE00 8144		32
DMA0 transfer count register B_2	DMA0_TCRB_2	R/W	H'FE00 8148		32
DMA0 source address register B_3	DMA0_SARB_3	R/W	H'FE00 8150		32
DMA0 destination address register B_3	DMA0_DARB_3	R/W	H'FE00 8154		32
DMA0 transfer count register B_3	DMA0_TCRB_3	R/W	H'FE00 8158		32
DMA0 extended resource selector 0	DMA0_DMARS_0	R/W	H'FE00 9000		16
DMA0 extended resource selector 1	DMA0_DMARS_1	R/W	H'FE00 9004		16
DMA0 extended resource selector 2	DMA0_DMARS_2	R/W	H'FE00 9008		16
DMA1 source address register_0	DMA1_SAR_0	R/W	H'FDC0 8020		32
DMA1 destination address register_0	DMA1_DAR_0	R/W	H'FDC0 8024		32
DMA1 transfer count register_0	DMA1_TCR_0	R/W	H'FDC0 8028		32
DMA1 channel control register_0	DMA1_CHCR_0	R/W	H'FDC0 802C		32
DMA1 source address register_1	DMA1_SAR_1	R/W	H'FDC0 8030		32
DMA1 destination address register_1	DMA1_DAR_1	R/W	H'FDC0 8034		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
DMA1 transfer count register_1	DMA1_TCR_1	R/W	H'FDC0 8038	DMAC	32
DMA1 channel control register_1	DMA1_CHCR_1	R/W	H'FDC0 803C		32
DMA1 source address register_2	DMA1_SAR_2	R/W	H'FDC0 8040		32
DMA1 destination address register_2	DMA1_DAR_2	R/W	H'FDC0 8044		32
DMA1 transfer count register_2	DMA1_TCR_2	R/W	H'FDC0 8048		32
DMA1 channel control register_2	DMA1_CHCR_2	R/W	H'FDC0 804C		32
DMA1 source address register_3	DMA1_SAR_3	R/W	H'FDC0 8050		32
DMA1 destination address register_3	DMA1_DAR_3	R/W	H'FDC0 8054		32
DMA1 transfer count register_3	DMA1_TCR_3	R/W	H'FDC0 8058		32
DMA1 channel control register_3	DMA1_CHCR_3	R/W	H'FDC0 805C		32
DMA1 operation register	DMA1_DMAOR	R/W	H'FDC0 8060		16
DMA1 source address register_4	DMA1_SAR_4	R/W	H'FDC0 8070		32
DMA1 destination address register_4	DMA1_DAR_4	R/W	H'FDC0 8074		32
DMA1 transfer count register_4	DMA1_TCR_4	R/W	H'FDC0 8078		32
DMA1 channel control register_4	DMA1_CHCR_4	R/W	H'FDC0 807C		32
DMA1 source address register_5	DMA1_SAR_5	R/W	H'FDC0 8080		32
DMA1 destination address register_5	DMA1_DAR_5	R/W	H'FDC0 8084		32
DMA1 transfer count register_5	DMA1_TCR_5	R/W	H'FDC0 8088		32
DMA1 channel control register_5	DMA1_CHCR_5	R/W	H'FDC0 808C		32
DMA1 source address register B_0	DMA1_SARB_0	R/W	H'FDC0 8120		32
DMA1 destination address register B_0	DMA1_DARB_0	R/W	H'FDC0 8124		32
DMA1 transfer count register B_0	DMA1_TCRB_0	R/W	H'FDC0 8128		32
DMA1 source address register B_1	DMA1_SARB_1	R/W	H'FDC0 8130		32
DMA1 destination address register B_1	DMA1_DARB_1	R/W	H'FDC0 8134		32
DMA1 transfer count register B_1	DMA1_TCRB_1	R/W	H'FDC0 8138		32
DMA1 source address register B_2	DMA1_SARB_2	R/W	H'FDC0 8140		32
DMA1 destination address register B_2	DMA1_DARB_2	R/W	H'FDC0 8144		32
DMA1 transfer count register B_2	DMA1_TCRB_2	R/W	H'FDC0 8148		32
DMA1 source address register B_3	DMA1_SARB_3	R/W	H'FDC0 8150		32
DMA1 destination address register B_3	DMA1_DARB_3	R/W	H'FDC0 8154		32
DMA1 transfer count register B_3	DMA1_TCRB_3	R/W	H'FDC0 8158		32
DMA1 extended resource selector 0	DMA1_DMARS_0	R/W	H'FDC0 9000		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
DMA1 extended resource selector 1	DMA1_DMARS_1	R/W	H'FDC0 9004	DMAC	16
DMA1 extended resource selector 2	DMA1_DMARS_2	R/W	H'FDC0 9008		16
Frequency control register A	FRQCRA	R/W	H'A415 0000	CPG	32
Frequency control register B	FRQCRB	R/W	H'A415 0004		32
Video clock frequency control register	VCLKCR	R/W	H'A415 0048		32
FSI clock A frequency control register	FCLKACR	R/W	H'A415 0008		32
FSI clock B frequency control register	FCLKBCR	R/W	H'A415 000C		32
IrDA clock frequency control register	IRDACLKCR	R/W	H'A415 0018		32
SPU clock frequency control register	SPUCLKCR	R/W	H'A415 003C		32
PLL control register	PLLCR	R/W	H'A415 0024		32
FLL multiplication register	FLLFRQ	R/W	H'A415 0050		32
Frequency change status register	LSTATS	R	H'A415 0060		32
Standby control register	STBCR	R/W	H'A415 0020	Reset and Power-Down Modes	32
Module stop register 0	MSTPCR0	R/W	H'A415 0030		32
Module stop register 1	MSTPCR1	R/W	H'A415 0034		32
Module stop register 2	MSTPCR2	R/W	H'A415 0038		32
Boot address register	BAR	R/W	H'A415 0040		32
RCLK watchdog timer counter	RWTCNT	R/W	H'A452 0000	RWDT	8/16*[1]
RCLK watchdog timer control/status register	RWTCSR	R/W	H'A452 0004		8/16*[1]
Timer start register 0	TSTR0	R/W	H'FFD8 0004	TMU	8
Timer constant register 0_0	TCOR0_0	R/W	H'FFD8 0008		32
Timer counter 0_0	TCNT0_0	R/W	H'FFD8 000C		32
Timer control register 0_0	TCR0_0	R/W	H'FFD8 0010		16
Timer constant register 0_1	TCOR0_1	R/W	H'FFD8 0014		32
Timer counter 0_1	TCNT0_1	R/W	H'FFD8 0018		32
Timer control register 0_1	TCR0_1	R/W	H'FFD8 001C		16
Timer constant register 0_2	TCOR0_2	R/W	H'FFD8 0020		32
Timer counter 0_2	TCNT0_2	R/W	H'FFD8 0024		32
Timer control register 0_2	TCR0_2	R/W	H'FFD8 0028		16
Timer start register 1	TSTR1	R/W	H'FFD9 0004		8
Timer constant register 1_0	TCOR1_0	R/W	H'FFD9 0008		32
Timer counter 1_0	TCNT1_0	R/W	H'FFD9 000C		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
Timer control register 1_0	TCR1_0	R/W	H'FFD9 0010	TMU	16
Timer constant register 1_1	TCOR1_1	R/W	H'FFD9 0014		32
Timer counter 1_1	TCNT1_1	R/W	H'FFD9 0018		32
Timer control register 1_1	TCR1_1	R/W	H'FFD9 001C		16
Timer constant register 1_2	TCOR1_2	R/W	H'FFD9 0020		32
Timer counter 1_2	TCNT1_2	R/W	H'FFD9 0024		32
Timer control register 1_2	TCR1_2	R/W	H'FFD9 0028		16
Timer start register	TPU_TSTR	R/W	H'A4C9 0000	TPU	16
Timer control register 0	TPU_TCR0	R/W	H'A4C9 0010		16
Timer mode register 0	TPU_TMDR0	R/W	H'A4C9 0014		16
Timer I/O control register 0	TPU_TIOR0	R/W	H'A4C9 0018		16
Timer interrupt enable register0	TPU_TIER0	R/W	H'A4C9 001C		16
Timer status register 0	TPU_TSR0	R/W	H'A4C9 0020		16
Timer counter 0	TPU_TCNT0	R/W	H'A4C9 0024		16
Timer general register 0A	TPU_TGR0A	R/W	H'A4C9 0028		16
Timer general register 0B	TPU_TGR0B	R/W	H'A4C9 002C		16
Timer general register 0C	TPU_TGR0C	R/W	H'A4C9 0030		16
Timer general register 0D	TPU_TGR0D	R/W	H'A4C9 0034		16
Timer control register 1	TPU_TCR1	R/W	H'A4C9 0050		16
Timer mode register 1	TPU_TMDR1	R/W	H'A4C9 0054		16
Timer I/O control register 1	TPU_TIOR1	R/W	H'A4C9 0058		16
Timer interrupt enable register1	TPU_TIER1	R/W	H'A4C9 005C		16
Timer status register 1	TPU_TSR1	R/W	H'A4C9 0060		16
Timer counter 1	TPU_TCNT1	R/W	H'A4C9 0064		16
Timer general register 1A	TPU_TGR1A	R/W	H'A4C9 0068		16
Timer general register 1B	TPU_TGR1B	R/W	H'A4C9 006C		16
Timer general register 1C	TPU_TGR1C	R/W	H'A4C9 0070		16
Timer general register 1D	TPU_TGR1D	R/W	H'A4C9 0074		16
Timer control register 2	TPU_TCR2	R/W	H'A4C9 0090		16
Timer mode register 2	TPU_TMDR2	R/W	H'A4C9 0094		16
Timer I/O control register 2	TPU_TIOR2	R/W	H'A4C9 0098		16
Timer interrupt enable register2	TPU_TIER2	R/W	H'A4C9 009C		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
Timer status register 2	TPU_TSR2	R/W	H'A4C9 00A0	TPU	16
Timer counter 2	TPU_TCNT2	R/W	H'A4C9 00A4		16
Timer general register 2A	TPU_TGR2A	R/W	H'A4C9 00A8		16
Timer general register 2B	TPU_TGR2B	R/W	H'A4C9 00AC		16
Timer general register 2C	TPU_TGR2C	R/W	H'A4C9 00B0		16
Timer general register 2D	TPU_TGR2D	R/W	H'A4C9 00B4		16
Timer control register 3	TPU_TCR3	R/W	H'A4C9 00D0		16
Timer mode register 3	TPU_TMDR3	R/W	H'A4C9 00D4		16
Timer interrupt enable register3	TPU_TIER3	R/W	H'A4C9 00DC		16
Timer status register 3	TPU_TSR3	R/W	H'A4C9 00E0		16
Timer counter 3	TPU_TCNT3	R/W	H'A4C9 00E4		16
Timer I/O control register 3	TPU_TIOR3	R/W	H'4AC9 00D8		16
Timer general register 3A	TPU_TGR3A	R/W	H'A4C9 00E8		16
Timer general register 3B	TPU_TGR3B	R/W	H'A4C9 00EC		16
Timer general register 3C	TPU_TGR3C	R/W	H'A4C9 00F0		16
Timer general register 3D	TPU_TGR3D	R/W	H'A4C9 00F4		16
Compare much timer start register	CMSTR	R/W	H'A44A 0000	CMT	16
Compare match timer start register	CMCSR	R/W	H'A44A 0060		16
Compare match timer control/status register	CMCNT	R/W	H'A44A 0064		32
Compare match timer counter	CMCOR	R/W	H'A44A 0068		32
Command setting register	CE_CMD_SET	R/W	H'A4CA 0000	MMCIF	16/32
Argument register	CE_ARG	R/W	H'A4CA 0008		16/32
Argument register for automatically-issued CMD12	CE_ARG_CMD12	R/W	H'A4CA 000C		16/32
Command control register	CE_CMD_CTRL	R/W	H'A4CA 0010		16/32
Transfer block setting register	CE_BLOCK_SET	R/W	H'A4CA 0014		16/32
Clock control register	CE_CLK_CTRL	R/W	H'A4CA 0018		16/32
Buffer access configuration register	CE_BUF_ACC	R/W	H'A4CA 001C		16/32
Response register 3	CE_RESP3	R/W	H'A4CA 0020		16/32
Response register 2	CE_RESP2	R/W	H'A4CA 0024		16/32
Response register 1	CE_RESP1	R/W	H'A4CA 0028		16/32
Response register 0	CE_RESP0	R/W	H'A4CA 002C		16/32

Register Name	Abbreviation	R/W	Address	Module	Access Size
Response register for automatically-issued CMD12	CE_RESP_CMD12	R/W	H'A4CA 0030	MMCIF	16/32
Data register	CE_DATA	R/W	H'A4CA 0034		16*2/32
Interrupt flag register	CE_INT	R/W	H'A4CA 0040		16/32
Interrupt mask register	CE_INT_MASK	R/W	H'A4CA 0044		16/32
Status register 1	CE_HOST_STS1	R/W	H'A4CA 0048		16/32
Status register 2	CE_HOST_STS2	R/W	H'A4CA 004C		16/32
Version register	CE_VERSION	R/W	H'A4CA 007C		16/32
MSIOF0 Transmit mode register 1	MSIOF0_SITMDR1	R/W	H'A4C4 0000	MSIOF	32
MSIOF0 Transmit mode register 2	MSIOF0_SITMDR2	R/W	H'A4C4 0004		32
MSIOF0 Transmit mode register 3	MSIOF0_SITMDR3	R/W	H'A4C4 0008		32
MSIOF0 Receive mode register 1	MSIOF0_SIRMDR1	R/W	H'A4C4 0010		32
MSIOF0 Receive mode register 2	MSIOF0_SIRMDR2	R/W	H'A4C4 0014		32
MSIOF0 Receive mode register 3	MSIOF0_SIRMDR3	R/W	H'A4C4 0018		32
MSIOF0 Transmit clock select register	MSIOF0_SITSCR	R/W	H'A4C4 0020		16
MSIOF0 Receive clock select register	MSIOF0_SIRSCR	R/W	H'A4C4 0022		16
MSIOF0 Control register	MSIOF0_SICTR	R/W	H'A4C4 0028		32
MSIOF0 FIFO control register	MSIOF0_SIFCTR	R/W	H'A4C4 0030		32
MSIOF0 Status register	MSIOF0_SISTR	R/W	H'A4C4 0040		32
MSIOF0 Interrupt enable register	MSIOF0_SIIER	R/W	H'A4C4 0044		32
MSIOF0 Transmit control data register 1	MSIOF0_SITDR1	W	H'A4C4 0048		32
MSIOF0 Transmit control data register 2	MSIOF0_SITDR2	W	H'A4C4 004C		32
MSIOF0 Transmit FIFO data register	MSIOF0_SITFDR	W	H'A4C4 0050		32
MSIOF0 Receive control data register 1	MSIOF0_SIRDR1	R	H'A4C4 0058		32
MSIOF0 Receive control data register 2	MSIOF0_SIRDR2	R	H'A4C4 005C		32
MSIOF0 Receive FIFO data register	MSIOF0_SIRFDR	R	H'A4C4 0060		32
MSIOF1 Transmit mode register 1	MSIOF1_SITMDR1	R/W	H'A4C5 0000		32
MSIOF1 Transmit mode register 2	MSIOF1_SITMDR2	R/W	H'A4C5 0004		32
MSIOF1 Transmit mode register 3	MSIOF1_SITMDR3	R/W	H'A4C5 0008		32
MSIOF1 Receive mode register 1	MSIOF1_SIRMDR1	R/W	H'A4C5 0010		32
MSIOF1 Receive mode register 2	MSIOF1_SIRMDR2	R/W	H'A4C5 0014		32
MSIOF1 Receive mode register 3	MSIOF1_SIRMDR3	R/W	H'A4C5 0018		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
MSIOF1 Transmit clock select register	MSIOF1_SITSCR	R/W	H'A4C5 0020	MSIOF	16
MSIOF1 Receive clock select register	MSIOF1_SIRSCR	R/W	H'A4C5 0022		16
MSIOF1 Control register	MSIOF1_SICTR	R/W	H'A4C5 0028		32
MSIOF1 FIFO control register	MSIOF1_SIFCTR	R/W	H'A4C5 0030		32
MSIOF1 Status register	MSIOF1_SISTR	R/W	H'A4C5 0040		32
MSIOF1 Interrupt enable register	MSIOF1_SIIER	R/W	H'A4C5 0044		32
MSIOF1 Transmit control data register 1	MSIOF1_SITDR1	W	H'A4C5 0048		32
MSIOF1 Transmit control data register 2	MSIOF1_SITDR2	W	H'A4C5 004C		32
MSIOF1 Transmit FIFO data register	MSIOF1_SITFDR	W	H'A4C5 0050		32
MSIOF1 Receive control data register 1	MSIOF1_SIRDR1	R	H'A4C5 0058		32
MSIOF1 Receive control data register 2	MSIOF1_SIRDR2	R	H'A4C5 005C		32
MSIOF1 Receive FIFO data register	MSIOF1_SIRFDR	R	H'A4C5 0060		32
Serial mode register 0	SCSMR0	R/W	H'FFE0 0000	SCIF	16
Bit rate register 0	SCBRR0	R/W	H'FFE0 0004		8
Serial control register 0	SCSCR0	R/W	H'FFE0 0008		16
Transmit FIFO data register 0	SCFTDR0	W	H'FFE0 000C		8
Serial status register 0	SCFSR0	R/W	H'FFE0 0010		16
Receive FIFO data register 0	SCFRDR0	R	H'FFE0 0014		8
FIFO control register 0	SCFCR0	R/W	H'FFE0 0018		16
FIFO data count register 0	SCFDR0	R	H'FFE0 001C		16
Line status register 0	SCLSR0	R/W	H'FFE0 0024		16
Serial mode register 1	SCSMR1	R/W	H'FFE1 0000		16
Bit rate register 1	SCBRR1	R/W	H'FFE1 0004		8
Serial control register 1	SCSCR1	R/W	H'FFE1 0008		16
Transmit FIFO data register 1	SCFTDR1	W	H'FFE1 000C		8
Serial status register 1	SCFSR1	R/W	H'FFE1 0010		16
Receive FIFO data register 1	SCFRDR1	R	H'FFE1 0014		8
FIFO control register 1	SCFCR1	R/W	H'FFE1 0018		16
FIFO data count register 1	SCFDR1	R	H'FFE1 001C		16
Line status register 1	SCLSR1	R/W	H'FFE1 0024		16
Serial mode register 2	SCSMR2	R/W	H'FFE2 0000		16
Bit rate register 2	SCBRR2	R/W	H'FFE2 0004		8

Register Name	Abbreviation	R/W	Address	Module	Access Size
Serial control register 2	SCSCR2	R/W	H'FFE2 0008	SCIF	16
Transmit FIFO data register 2	SCFTDR2	W	H'FFE2 000C		8
Serial status register 2	SCFSR2	R/W	H'FFE2 0010		16
Receive FIFO data register 2	SCFRDR2	R	H'FFE2 0014		8
FIFO control register 2	SCFCR2	R/W	H'FFE2 0018		16
FIFO data count register 2	SCFDR2	R	H'FFE2 001C		16
Line status register 2	SCLSR2	R/W	H'FFE2 0024		16
Serial mode register A3	SCASMR3	R/W	H'A4E3 0000	SCIFA	16
Bit rate register A3	SCABRR3	R/W	H'A4E3 0004		8
Serial control register A3	SCASCR3	R/W	H'A4E3 0008		16
Transmit data stop register A3	SCATDSR3	R/W	H'A4E3 000C		8
FIFO error count register A3	SCAFER3	R	H'A4E3 0010		16
Serial status register A3	SCASSR3	R/W	H'A4E3 0014		16
FIFO control register A3	SCAFCR3	R/W	H'A4E3 0018		16
FIFO data count register A3	SCAFDR3	R	H'A4E3 001C		16
Transmit FIFO data register A3	SCAFTDR3	W	H'A4E3 0020		8
Receive FIFO data register A3	SCAFRDR3	R	H'A4E3 0024		8
Serial mode register A4	SCASMR4	R/W	H'A4E4 0000		16
Bit rate register A4	SCABRR4	R/W	H'A4E4 0004		8
Serial control register A4	SCASCR4	R/W	H'A4E4 0008		16
Transmit data stop register A4	SCATDSR4	R/W	H'A4E4 000C		8
FIFO error count register A4	SCAFER4	R	H'A4E4 0010		16
Serial status register A4	SCASSR4	R/W	H'A4E4 0014		16
FIFO control register A4	SCAFCR4	R/W	H'A4E4 0018		16
FIFO data count register A4	SCAFDR4	R	H'A4E4 001C		16
Transmit FIFO data register A4	SCAFTDR4	W	H'A4E4 0020		8
Receive FIFO data register A4	SCAFRDR4	R	H'A4E4 0024		8
Serial mode register A5	SCASMR5	R/W	H'A4E5 0000		16
Bit rate register A5	SCABRR5	R/W	H'A4E5 0004		8
Serial control register A5	SCASCR5	R/W	H'A4E5 0008		16
Transmit data stop register A5	SCATDSR5	R/W	H'A4E5 000C		8
FIFO error count register A5	SCAFER5	R	H'A4E5 0010		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
Serial status register A5	SCASSR5	R/W	H'A4E5 0014	SCIFA	16
FIFO control register A5	SCAFCR5	R/W	H'A4E5 0018		16
FIFO data count register A5	SCAFDR5	R	H'A4E5 001C		16
Transmit FIFO data register A5	SCAFTDR5	W	H'A4E5 0020		8
Receive FIFO data register A5	SCAFRDR5	R	H'A4E5 0024		8
64-Hz counter	R64CNT	R	H'A465 FEC0	RTC	8
Second counter	RSECCNT	R/W	H'A465 FEC2		8
Minute counter	RMINCNT	R/W	H'A465 FEC4		8
Hour counter	RHRCNT	R/W	H'A465 FEC6		8
Day of week counter	RWKCNT	R/W	H'A465 FEC8		8
Date counter	RDAYCNT	R/W	H'A465 FECA		8
Month counter	RMONCNT	R/W	H'A465 FECC		8
Year counter	RYRCNT	R/W	H'A465 FECE		16
Second alarm register	RSECAR	R/W	H'A465 FED0		8
Minute alarm register	RMINAR	R/W	H'A465 FED2		8
Hour alarm register	RHRAR	R/W	H'A465 FED4		8
Day of week alarm register	RWKAR	R/W	H'A465 FED6		8
Date alarm register	RDAYAR	R/W	H'A465 FED8		8
Month alarm register	RMONAR	R/W	H'A465 FEDA		8
RTC control register 1	RCR1	R/W	H'A465 FEDC		8
RTC control register 2	RCR2	R/W	H'A465 FEDE		8
Year alarm register	RYRAR	R/W	H'A465 FEE0		16
RTC control register 3	RCR3	R/W	H'A465 FEE4		8
DMA receive interrupt source clear register	IRIF_RINTCLR	W	H'A45D 0016	IrDA	16/8
DMA transmit interrupt source clear register	IRIF_TINTCLR	W	H'A45D 0018		16/8
IrDA-SIR10 control register	IRIF_SIR0	R/W	H'A45D 0020		16/8
IrDA-SIR10 baud rate error correction register	IRIF_SIR1	R/W	H'A45D 0022		16/8
IrDA-SIR10 baud rate count set register	IRIF_SIR2	R/W	H'A45D 0024		16/8
IrDA-SIR10 status register	IRIF_SIR3	R	H'A45D 0026		16/8
Hardware frame processing set register	IRIF_SIR_FRM	R/W	H'A45D 0028		16/8
EOF value register	IRIF_SIR_EOF	R/W	H'A45D 002A		16/8

Register Name	Abbreviation	R/W	Address	Module	Access Size
Flag clear register	IRIF_SIR_FLG	W	H'A45D 002C	IrDA	16/8
UART status register 2	IRIF_SIR_STS2	R/W	H'A45D 002E		16/8
UART control register	IRIF_UART0	R/W	H'A45D 0030		16/8
UART status register	IRIF_UART1	R	H'A45D 0032		16/8
UART mode register	IRIF_UART2	R/W	H'A45D 0034		16/8
UART transmit data register	IRIF_UART3	W	H'A45D 0036		16/8
UART receive data register	IRIF_UART4	R	H'A45D 0038		16/8
UART interrupt mask register	IRIF_UART5	R/W	H'A45D 003A		16/8
UART baud rate error correction register	IRIF_UART6	R/W	H'A45D 003C		16/8
UART baud rate count set register	IRIF_UART7	R/W	H'A45D 003E		16/8
CRC engine control register	IRIF_CRC0	R/W	H'A45D 0040		16/8
CRC engine input data register	IRIF_CRC1	W	H'A45D 0042		16/8
CRC engine calculation register	IRIF_CRC2	W	H'A45D 0044		16/8
CRC engine output data register 1	IRIF_CRC3	R	H'A45D 0046		16/8
CRC engine output data register 2	IRIF_CRC4	R	H'A45D 0048		16/8
Key scan control register 1	KYCR1	R/W	H'A44B 0000	KEYSC	16
Key scan control register 2	KYCR2	R/W	H'A44B 0004		16
Key scan-in data register	KYINDR	R	H'A44B 0008		16
Key scan-out data register	KYOUTDR	R/W	H'A44B 000C		16
System configuration control register_0	SYSCFG_0	R/W	H'A4D8 0000	USB	16
CPU bus wait setting register_0	BUSWAIT_0	R/W	H'A4D8 0002		16
System configuration status register_0	SYSSTS_0	R	H'A4D8 0004		16
Device state control register_0	DVSTCTR_0	R/W	H'A4D8 0008		16
Test mode register_0	TESTMODE_0	R/W	H'A4D8 000C		16
CFIFO port register_0	CFIFO_0	R/W	H'A4D8 0014		8, 16, 32
D0FIFO port register_0	D0FIFO_0	R/W	H'A4D8 0100		8, 16, 32
D1FIFO port register_0	D1FIFO_0	R/W	H'A4D8 0120		8, 16, 32
CFIFO port select register_0	CFIFOSEL_0	R/W	H'A4D8 0020		16
CFIFO port control register_0	CFIFOCTR_0	R/W	H'A4D8 0022		16
D0FIFO port select register_0	D0FIFOSEL_0	R/W	H'A4D8 0028		16
D0FIFO port control register_0	D0FIFOCTR_0	R/W	H'A4D8 002A		16
D1FIFO port select register_0	D1FIFOSEL_0	R/W	H'A4D8 002C		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
D1FIFO port control register_0	D1FIFOCTR_0	R/W	H'A4D8 002E	USB	16
Interrupt enable register 0_0	INTENB0_0	R/W	H'A4D8 0030		16
Interrupt enable register 1_0	INTENB1_0	R/W	H'A4D8 0032		16
BRDY interrupt enable register_0	BRDYENB_0	R/W	H'A4D8 0036		16
NRDY interrupt enable register_0	NRDYENB_0	R/W	H'A4D8 0038		16
BEMP interrupt enable register_0	BEMPENB_0	R/W	H'A4D8 003A		16
SOF output configuration register_0	SOFCFG_0	R/W	H'A4D8 003C		16
Interrupt status register 0_0	INTSTS0_0	R/W	H'A4D8 0040		16
Interrupt status register 1_0	INTSTS1_0	R/W	H'A4D8 0042		16
BRDY interrupt status register_0	BRDYSTS_0	R/W	H'A4D8 0046		16
NRDY interrupt status register_0	NRDYSTS_0	R/W	H'A4D8 0048		16
BEMP interrupt status register_0	BEMPSTS_0	R/W	H'A4D8 004A		16
Frame number register_0	FRMNUM_0	R/W	H'A4D8 004C		16
µ Frame number register_0	UFRMNUM_0	R/W	H'A4D8 004E		16
USB address register_0	USBADDR_0	R	H'A4D8 0050		16
USB request type register_0	USBREQ_0	R/W	H'A4D8 0054		16
USB request value register_0	USBVAL_0	R/W	H'A4D8 0056		16
USB request index register_0	USBINDX_0	R/W	H'A4D8 0058		16
USB request length register_0	USBLENG_0	R/W	H'A4D8 005A		16
DCP configuration register_0	DCPCFG_0	R/W	H'A4D8 005C		16
DCP maximum packet size register_0	DCPMAXP_0	R/W	H'A4D8 005E		16
DCP control register_0	DCPCTR_0	R/W	H'A4D8 0060		16
Pipe window select register_0	PIPESEL_0	R/W	H'A4D8 0064		16
Pipe configuration register_0	PIPECFG_0	R/W	H'A4D8 0068		16
Pipe buffer setting register_0	PIPEBUF_0	R/W	H'A4D8 006A		16
Pipe maximum packet size register_0	PIPEMAXP_0	R/W	H'A4D8 006C		16
Pipe cycle control register_0	PIPEPERI_0	R/W	H'A4D8 006E		16
Pipe 1 control register_0	PIPE1CTR_0	R/W	H'A4D8 0070		16
Pipe 2 control register_0	PIPE2CTR_0	R/W	H'A4D8 0072		16
Pipe 3 control register_0	PIPE3CTR_0	R/W	H'A4D8 0074		16
Pipe 4 control register_0	PIPE4CTR_0	R/W	H'A4D8 0076		16
Pipe 5 control register_0	PIPE5CTR_0	R/W	H'A4D8 0078		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
Pipe 6 control register_0	PIPE6CTR_0	R/W	H'A4D8 007A	USB	16
Pipe 7 control register_0	PIPE7CTR_0	R/W	H'A4D8 007C		16
Pipe 8 control register_0	PIPE8CTR_0	R/W	H'A4D8 007E		16
Pipe 9 control register_0	PIPE9CTR_0	R/W	H'A4D8 0080		16
Pipe 1 transaction counter enable register_0	PIPE1TRE_0	R/W	H'A4D8 0090		16
Pipe 1 transaction counter register_0	PIPE1TRN_0	R/W	H'A4D8 0092		16
Pipe 2 transaction counter enable register_0	PIPE2TRE_0	R/W	H'A4D8 0094		16
Pipe 2 transaction counter register_0	PIPE2TRN_0	R/W	H'A4D8 0096		16
Pipe 3 transaction counter enable register_0	PIPE3TRE_0	R/W	H'A4D8 0098		16
Pipe 3 transaction counter register_0	PIPE3TRN_0	R/W	H'A4D8 009A		16
Pipe 4 transaction counter enable register_0	PIPE4TRE_0	R/W	H'A4D8 009C		16
Pipe 4 transaction counter register_0	PIPE4TRN_0	R/W	H'A4D8 009E		16
Pipe 5 transaction counter enable register_0	PIPE5TRE_0	R/W	H'A4D8 00A0		16
Pipe 5 transaction counter register_0	PIPE5TRN_0	R/W	H'A4D8 00A2		16
Device address 0 configuration register_0	DEVADD0_0	R/W	H'A4D8 00D0		16
Device address 1 configuration register_0	DEVADD1_0	R/W	H'A4D8 00D2		16
Device address 2 configuration register_0	DEVADD2_0	R/W	H'A4D8 00D4		16
Device address 3 configuration register_0	DEVADD3_0	R/W	H'A4D8 00D6		16
Device address 4 configuration register_0	DEVADD4_0	R/W	H'A4D8 00D8		16
Device address 5 configuration register_0	DEVADD5_0	R/W	H'A4D8 00DA		16
Device address 6 configuration register_0	DEVADD6_0	R/W	H'A4D8 00DC		16
Device address 7 configuration register_0	DEVADD7_0	R/W	H'A4D8 00DE		16
Device address 8 configuration register_0	DEVADD8_0	R/W	H'A4D8 00E0		16
Device address 9 configuration register_0	DEVADD9_0	R/W	H'A4D8 00E2		16
Device address A configuration register_0	DEVADDA_0	R/W	H'A4D8 00E4		16
USB power control register_0	UPONCR0	R/W	H'A40501D4		16
System configuration control register_1	SYSCFG_1	R/W	H'A4D9 0000		16
CPU bus wait setting register_1	BUSWAIT_1	R/W	H'A4D9 0002		16
System configuration status register_1	SYSSTS_1	R	H'A4D9 0004		16
Device state control register_1	DVSTCTR_1	R/W	H'A4D9 0008		16
Test mode register_1	TESTMODE_1	R/W	H'A4D9 000C		16
CFIFO port register_1	CFIFO_1	R/W	H'A4D9 0014		8, 16, 32

Register Name	Abbreviation	R/W	Address	Module	Access Size
D0FIFO port register_1	D0FIFO_1	R/W	H'A4D9 0100	USB	8, 16, 32
D1FIFO port register_1	D1FIFO_1	R/W	H'A4D9 0120		8, 16, 32
CFIFO port select register_1	CFIFOSEL_1	R/W	H'A4D9 0020		16
CFIFO port control register_1	CFIFOCTR_1	R/W	H'A4D9 0022		16
D0FIFO port select register_1	D0FIFOSEL_1	R/W	H'A4D9 0028		16
D0FIFO port control register_1	D0FIFOCTR_1	R/W	H'A4D9 002A		16
D1FIFO port select register_1	D1FIFOSEL_1	R/W	H'A4D9 002C		16
D1FIFO port control register_1	D1FIFOCTR_1	R/W	H'A4D9 002E		16
Interrupt enable register 0_0	INTENB0_1	R/W	H'A4D9 0030		16
Interrupt enable register 1_0	INTENB1_1	R/W	H'A4D9 0032		16
BRDY interrupt enable register_1	BRDYENB_1	R/W	H'A4D9 0036		16
NRDY interrupt enable register_1	NRDYENB_1	R/W	H'A4D9 0038		16
BEMP interrupt enable register_1	BEMPENB_1	R/W	H'A4D9 003A		16
SOF output configuration register_1	SOFCFG_1	R/W	H'A4D9 003C		16
Interrupt status register 0_1	INTSTS0_1	R/W	H'A4D9 0040		16
Interrupt status register 1_1	INTSTS1_1	R/W	H'A4D9 0042		16
BRDY interrupt status register_1	BRDYSTS_1	R/W	H'A4D9 0046		16
NRDY interrupt status register_1	NRDYSTS_1	R/W	H'A4D9 0048		16
BEMP interrupt status register_1	BEMPSTS_1	R/W	H'A4D9 004A		16
Frame number register_1	FRMNUM_1	R/W	H'A4D9 004C		16
µ Frame number register_1	UFRMNUM_1	R/W	H'A4D9 004E		16
USB address register_1	USBADDR_1	R	H'A4D9 0050		16
USB request type register_1	USBREQ_1	R/W	H'A4D9 0054		16
USB request value register_1	USBVAL_1	R/W	H'A4D9 0056		16
USB request index register_1	USBINDX_1	R/W	H'A4D9 0058		16
USB request length register_1	USBLENG_1	R/W	H'A4D9 005A		16
DCP configuration register_1	DCPCFG_1	R/W	H'A4D9 005C		16
DCP maximum packet size register_1	DCPMAXP_1	R/W	H'A4D9 005E		16
DCP control register_1	DCPCTR_1	R/W	H'A4D9 0060		16
Pipe window select register_1	PIPESEL_1	R/W	H'A4D9 0064		16
Pipe configuration register_1	PIPECFG_1	R/W	H'A4D9 0068		16
Pipe buffer setting register_1	PIPEBUF_1	R/W	H'A4D9 006A		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
Pipe maximum packet size register_1	PIPEMAXP_1	R/W	H'A4D9 006C	USB	16
Pipe cycle control register_1	PIPEPERI_1	R/W	H'A4D9 006E		16
Pipe 1 control register_1	PIPE1CTR_1	R/W	H'A4D9 0070		16
Pipe 2 control register_1	PIPE2CTR_1	R/W	H'A4D9 0072		16
Pipe 3 control register_1	PIPE3CTR_1	R/W	H'A4D9 0074		16
Pipe 4 control register_1	PIPE4CTR_1	R/W	H'A4D9 0076		16
Pipe 5 control register_1	PIPE5CTR_1	R/W	H'A4D9 0078		16
Pipe 6 control register_1	PIPE6CTR_1	R/W	H'A4D9 007A		16
Pipe 7 control register_1	PIPE7CTR_1	R/W	H'A4D9 007C		16
Pipe 8 control register_1	PIPE8CTR_1	R/W	H'A4D9 007E		16
Pipe 9 control register_1	PIPE9CTR_1	R/W	H'A4D9 0080		16
Pipe 1 transaction counter enable register_1	PIPE1TRE_1	R/W	H'A4D9 0090		16
Pipe 1 transaction counter register_1	PIPE1TRN_1	R/W	H'A4D9 0092		16
Pipe 2 transaction counter enable register_1	PIPE2TRE_1	R/W	H'A4D9 0094		16
Pipe 2 transaction counter register_1	PIPE2TRN_1	R/W	H'A4D9 0096		16
Pipe cycle control register_1	PIPE3TRE_1	R/W	H'A4D9 0098		16
Pipe 3 transaction counter register_1	PIPE3TRN_1	R/W	H'A4D9 009A		16
Pipe 4 transaction counter enable register_1	PIPE4TRE_1	R/W	H'A4D9 009C		16
Pipe 4 transaction counter register_1	PIPE4TRN_1	R/W	H'A4D9 009E		16
Pipe 5 transaction counter enable register_1	PIPE5TRE_1	R/W	H'A4D9 00A0		16
Pipe 5 transaction counter register_1	PIPE5TRN_1	R/W	H'A4D9 00A2		16
Device address 0 configuration register_1	DEVADD0_1	R/W	H'A4D9 00D0		16
Device address 1 configuration register_1	DEVADD1_1	R/W	H'A4D9 00D2		16
Device address 2 configuration register_1	DEVADD2_1	R/W	H'A4D9 00D4		16
Device address 3 configuration register_1	DEVADD3_1	R/W	H'A4D9 00D6		16
Device address 4 configuration register_1	DEVADD4_1	R/W	H'A4D9 00D8		16
Device address 5 configuration register_1	DEVADD5_1	R/W	H'A4D9 00DA		16
Device address 6 configuration register_1	DEVADD6_1	R/W	H'A4D9 00DC		16
Device address 7 configuration register_1	DEVADD7_1	R/W	H'A4D9 00DE		16
Device address 8 configuration register_1	DEVADD8_1	R/W	H'A4D9 00E0		16
Device address 9 configuration register_1	DEVADD9_1	R/W	H'A4D9 00E2		16
Device address A configuration register_1	DEVADDA_1	R/W	H'A4D9 00E4		16
USB power control register_1	UPONCR1	R/W	H'A4050192		16

Register Name	Abbreviation	R/W	Address	Module	Access Size
I2C bus data register	ICDR	R/W	H'A447 0000	IIC	8
I2C bus control register	ICCR	R/W	H'A447 0004		8
I2C bus status register	ICSR	R/W	H'A447 0008		8
I2C interrupt control register	ICIC	R/W	H'A447 000C		8
I2C clock control register low	ICCL	R/W	H'A447 0010		8
I2C clock control register high	ICCH	R/W	H'A447 0014		8
With the exception of some registers, registers of the CEU have A-plane addresses, B-plane addresses, and mirror addresses. For details, refer to table 51.1.				CEU0/1	
VEU0 start register	VESTR_0	R/W	H'FE92 0000	VEU0	32
VEU0 source memory width register	VESWR_0	R/W	H'FE92 0010		32
VEU0 source size register	VESSR_0	R/W	H'FE92 0014		32
VEU0 source address Y register	VSAYR_0	R/W	H'FE92 0018		32
VEU0 source address C register	VSACR_0	R/W	H'FE92 001C		32
VEU0 bundle source size register	VBSSR_0	R/W	H'FE92 0020		32
VEU0 destination memory width register	VEDWR_0	R/W	H'FE92 0030		32
VEU0 destination address Y register	VDAYR_0	R/W	H'FE92 0034		32
VEU0 destination address C register	VDACR_0	R/W	H'FE92 0038		32
VEU0 transform control register	VTRCR_0	R/W	H'FE92 0050		32
VEU0 resize filter control register	VRFCR_0	R/W	H'FE92 0054		32
VEU0 resize filter size clip register	VRFSR_0	R/W	H'FE92 0058		32
VEU0 enhance register	VENHR_0	R/W	H'FE92 005C		32
VEU0 resize filter sub control register	VRSCR_0	R/W	H'FE92 0064		32
VEU0 resize filter size clip offset register	VRSOR_0	R/W	H'FE92 0068		32
VEU0 filter mode control register	VFMCR_0	R/W	H'FE92 0070		32
VEU0 vertical tap coefficient register	VVTCR_0	R/W	H'FE92 0074		32
VEU0 horizontal tap coefficient register	VHTCR_0	R/W	H'FE92 0078		32
VEU0 designated color register	VAPCR_0	R/W	H'FE92 0080		32
VEU0 conversion color register	VECCR_0	R/W	H'FE92 0084		32
VEU0 fill color specification register	VFLCR_0	R/W	H'FE92 0088		32
VEU0 address fixed register	VAFXR_0	R/W	H'FE92 0090		32
VEU0 swapping register	VSWPR_0	R/W	H'FE92 0094		32
VEU0 event interrupt enable register	VEIER_0	R/W	H'FE92 00A0		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
VEU0 event register	VEVTR_0	R/W	H'FE92 00A4	VEU0	32
VEU0 status register	VSTAR_0	R	H'FE92 00B0		32
VEU0 module reset register	VBSRR_0	R/W	H'FE92 00B4		32
VEU0 resize passband register	VRPBR_0	R/W	H'FE92 00C8		32
VEU1 start register	VESTR_1	R/W	H'FE92 4000	VEU1	32
VEU1 source memory width register	VESWR_1	R/W	H'FE92 4010		32
VEU1 source size register	VESSR_1	R/W	H'FE92 4014		32
VEU1 source address Y register	VSAYR_1	R/W	H'FE92 4018		32
VEU1 source address C register	VSACR_1	R/W	H'FE92 401C		32
VEU1 bundle source size register	VBSSR_1	R/W	H'FE92 4020		32
VEU1 destination memory width register	VEDWR_1	R/W	H'FE92 4030		32
VEU1 destination address Y register	VDAYR_1	R/W	H'FE92 4034		32
VEU1 destination address C register	VDACR_1	R/W	H'FE92 4038		32
VEU1 transform control register	VTRCR_1	R/W	H'FE92 4050		32
VEU1 resize filter size clip register	VRFCR_1	R/W	H'FE92 4054		32
VEU1 enhance register	VRFSR_1	R/W	H'FE92 4058		32
VEU1 resize filter sub control register	VENHR_1	R/W	H'FE92 405C		32
VEU1 resize filter size clip offset register	VRSCR_1	R/W	H'FE92 4064		32
VEU1 filter mode control register	VRSOR_1	R/W	H'FE92 4068		32
VEU1 vertical tap coefficient register	VFMCR_1	R/W	H'FE92 4070		32
VEU1 horizontal tap coefficient register	VVTCR_1	R/W	H'FE92 4074		32
VEU1 designated color register	VHTCR_1	R/W	H'FE92 4078		32
VEU1 resize filter size clip register	VAPCR_1	R/W	H'FE92 4080		32
VEU1 conversion color register	VECCR_1	R/W	H'FE92 4084		32
VEU1 fill color specification register	VFLCR_1	R/W	H'FE92 4088		32
VEU1 address fixed register	VAFXR_1	R/W	H'FE92 4090		32
VEU1 swapping register	VSWPR_1	R/W	H'FE92 4094		32
VEU1 event interrupt enable register	VEIER_1	R/W	H'FE92 40A0		32
VEU1 event register	VEVTR_1	R/W	H'FE92 40A4		32
VEU1 status register	VSTAR_1	R	H'FE92 40B0		32
VEU1 module reset register	VBSRR_1	R/W	H'FE92 40B4		32
VEU1 resize passband register	VRPBR_1	R/W	H'FE92 40C8		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
With the exception of some registers, registers of the BEU have A-plane addresses, B-plane addresses, and mirror addresses. For details, refer to table 51.2.				BEU0/1	
JPEG code mode register	JCMOD	R/W	H'FE98 0000	JPU	32
JPEG code command register	JCCMD	R/W	H'FE98 0004		32
JPEG code status register	JCSTS	R	H'FE98 0008		32
JPEG code quantization table number register	JCQTN	R/W	H'FE98 000C		32
JPEG code Huffman table number register	JCHTN	R/W	H'FE98 0010		32
JPEG code DRI upper register	JCDRIU	R/W	H'FE98 0014		32
JPEG code DRI lower register	JCDRID	R/W	H'FE98 0018		32
JPEG code vertical size upper register	JCVSZU	R/W	H'FE98 001C		32
JPEG code vertical size lower register	JCVSZD	R/W	H'FE98 0020		32
JPEG code horizontal size upper register	JCHSZU	R/W	H'FE98 0024		32
JPEG code horizontal size lower register	JCHSZD	R/W	H'FE98 0028		32
JPEG code data count upper register	JCDTCU	R	H'FE98 002C		32
JPEG code data count middle register	JCDTCM	R	H'FE98 0030		32
JPEG code data count lower register	JCDTCD	R	H'FE98 0034		32
JPEG interrupt enable register	JINTE	R/W	H'FE98 0038		32
JPEG interrupt status register	JINTS	R/W	H'FE98 003C		32
JPEG code decode error register	JCDERR	R/W	H'FE98 0040		32
JPEG code reset register	JCRST	R	H'FE98 0044		32
JPEG interface encoding control register	JIFECNT	R/W	H'FE98 0070		32
JPEG interface encode source Y address register 1	JIFESYA1	R/W	H'FE98 0074		32
JPEG interface encode source C address register 1	JIFESCA1	R/W	H'FE98 0078		32
JPEG interface encode source Y address register 2	JIFESYA2	R/W	H'FE98 007C		32
JPEG interface encode source C address register 2	JIFESCA2	R/W	H'FE98 0080		32
JPEG interface encode source memory width register	JIFESMW	R/W	H'FE98 0084		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
JPEG interface encode source vertical size register	JIFESVSZ	R/W	H'FE98 0088	JPU	32
JPEG interface encode source horizontal size register	JIFESHSZ	R/W	H'FE98 008C		32
JPEG interface encode destination address register 1	JIFEDA1	R/W	H'FE98 0090		32
JPEG interface encode destination address register 2	JIFEDA2	R/W	H'FE98 0094		32
JPEG interface encode data reload size register	JIFEDRSZ	R/W	H'FE98 0098		32
JPEG interface decoding control register	JIFDCNT	R/W	H'FE98 00A0		32
JPEG interface decode source address register 1	JIFDSA1	R/W	H'FE98 00A4		32
JPEG interface decode source address register 2	JIFDSA2	R/W	H'FE98 00A8		32
JPEG interface decode data reload size register	JIFDDRSZ	R/W	H'FE98 00AC		32
JPEG interface decode destination memory width register	JIFDDMW	R/W	H'FE98 00B0		32
JPEG interface decode destination vertical size register	JIFDDVSZ	R	H'FE98 00B4		32
JPEG interface decode destination horizontal size register	JIFDDHSZ	R	H'FE98 00B8		32
JPEG interface decode destination Y address register 1	JIFDDYA1	R/W	H'FE98 00BC		32
JPEG interface decode destination C address register 1	JIFDDCA1	R/W	H'FE98 00C0		32
JPEG interface decode destination Y address register 2	JIFDDYA2	R/W	H'FE98 00C4		32
JPEG interface decode destination C address register 2	JIFDDCA2	R/W	H'FE98 00C8		32
JPEG code quantization table 0 register	JCQTBL0	R/W	H'FE99 0000 to H'FE99 003C		32
JPEG code quantization table 1 register	JCQTBL1	R/W	H'FE99 0040 to H'FE99 007C		32
JPEG code quantization table 2 register	JCQTBL2	R/W	H'FE99 0080 to H'FE99 00BC		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
JPEG code quantization table 3 register	JCQTBL3	R/W	H'FE99 00C0 to H'FE99 00FC	JPU	32
JPEG code Huffman table DC0 register	JCHTBD0	R/W	H'FE99 0100 to H'FE99 010C		32
JPEG code Huffman table DC0 register	JCHTBD0	R/W	H'FE99 0110 to H'FE99 0118		32
JPEG code Huffman table AC0 register	JCHTBA0	R/W	H'FE99 0120 to H'FE99 012C		32
JPEG code Huffman table AC0 register	JCHTBA0	R/W	H'FE99 0130 to H'FE99 01D0		32
JPEG code Huffman table DC1 register	JCHTBD1	R/W	H'FE99 0200 to H'FE99 020C		32
JPEG code Huffman table DC1 register	JCHTBD1	R/W	H'FE99 0210 to H'FE99 0218		32
JPEG code Huffman table AC1 register	JCHTBA1	R/W	H'FE99 0220 to H'FE99 022C		32
JPEG code Huffman table AC1 register	JCHTBA1	R/W	H'FE99 0230 to H'FE99 02D0		32
With the exception of some registers, registers of the LCDC have A-plane addresses, B-plane addresses, and mirror addresses. For details, refer to table 51.3.				LCDC	
With the exception of some registers, registers of the VOU have A-plane addresses, B-plane addresses, and mirror addresses. For details, refer to table 51.4.				VOU	
ICB control register 0	MEVCR0	R/W	H'E800 0000	MERAM	32
ICB control register 1	MEVCR1	R/W	H'E800 0004		32
ICB transfer end interrupt control register	METEIE	R/W	H'E800 0008		32
ICB transaction error interrupt control register	MEILIE	R/W	H'E800 000C		32
ICB active status register	MEACTST	R/W	H'E800 0010		32
ICB transfer end status register	METEST	R/W	H'E800 0014		32
ICB write transaction error status register	MEILWST	R/W	H'E800 0018		32
ICB read transaction error status register	MEILRST	R/W	H'E800 001C		32
ICB00 buffer control register	ME00CTRL	R/W	H'E800 0400		32
ICB00 frame size register	ME00BSIZE	R/W	H'E800 0404		32
ICB00 MERAM set register	ME00MCNF	R/W	H'E800 0408		32
ICB00 reserved	—	R	H'E800 040C		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
ICB00 external memory start address register A	ME00SSARA	R/W	H'E800 0410	MERAM	32
ICB00 external memory start address register B	ME00SSARB	R/W	H'E800 0414		32
ICB00 external memory buffer size register	ME00SBSIZE	R/W	H'E800 0418		32
ICB00 reserved	—	R	H'E800 041C		32
ICB01 ****	ME01****	R/W	H'E800 0420 to H'E800 043C		32
ICB02 ****	ME02****	R/W	H'E800 0440 to H'E800 045C		32
ICB03 ****	ME03****	R/W	H'E800 0460 to H'E800 047C		32
ICB04 ****	ME04****	R/W	H'E800 0480 to H'E800 049C		32
ICB05 ****	ME05****	R/W	H'E800 04A0 to H'E800 04BC		32
ICB06 ****	ME06****	R/W	H'E800 04C0 to H'E800 04DC		32
ICB07 ****	ME07****	R/W	H'E800 04E0 to H'E800 04FC		32
ICB08 ****	ME08****	R/W	H'E800 0500 to H'E800 051C		32
ICB09 ****	ME09****	R/W	H'E800 0520 to H'E800 053C		32
ICB10 ****	ME10****	R/W	H'E800 0540 to H'E800 055C		32
ICB11 ****	ME11****	R/W	H'E800 0560 to H'E800 057C		32
ICB12 ****	ME12****	R/W	H'E800 0580 to H'E800 059C		32
ICB13 ****	ME13****	R/W	H'E800 05A0 to H'E800 05BC		32
ICB14 ****	ME14****	R/W	H'E800 05C0 to H'E800 05DC		32
ICB15 ****	ME15****	R/W	H'E800 05E0 to H'E800 05FC		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
ICB16 ****	ME16****	R/W	H'E800 0600 to H'E800 061C	MERAM	32
ICB17 ****	ME17****	R/W	H'E800 0620 to H'E800 063C		32
ICB18 ****	ME18****	R/W	H'E800 0640 to H'E800 065C		32
ICB19 ****	ME19****	R/W	H'E800 0660 to H'E800 067C		32
ICB20 ****	ME20****	R/W	H'E800 0680 to H'E800 069C		32
ICB21 ****	ME21****	R/W	H'E800 06A0 to H'E800 06BC		32
ICB22 ****	ME22****	R/W	H'E800 06C0 to H'E800 06DC		32
ICB23 ****	ME23****	R/W	H'E800 06E0 to H'E800 06FC		32
ICB24 ****	ME24****	R/W	H'E800 0700 to H'E800 071C		32
ICB25 ****	ME25****	R/W	H'E800 0720 to H'E800 073C		32
ICB26 ****	ME26****	R/W	H'E800 0740 to H'E800 075C		32
ICB27 ****	ME27****	R/W	H'E800 0760 to H'E800 077C		32
ICB28 ****	ME28****	R/W	H'E800 0780 to H'E800 079C		32
ICB29 ****	ME29****	R/W	H'E800 07A0 to H'E800 07BC		32
ICB30 ****	ME30****	R/W	H'E800 07C0 to H'E800 07DC		32
ICB31****	ME31****	R/W	H'E800 07E0 to H'E800 07FC		32
MERAM direct access space	—	R/W	H'E808 0000 to H'E809 FFFF		8/16/32
Interrupt status clear register	CHSTCLR	R/W	H'FEA0 0010	2D-DMAC	32
Channel priority switch register	CHPRI	R/W	H'FEA0 0014		32
CH0 control register	CH0CTRL	R/W	H'FEA0 0020		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
CH1 control register	CH1CTRL	R/W	H'FEA0 0024	2D-DMAC	32
CH2 control register	CH2CTRL	R/W	H'FEA0 0028		32
CH3 control register	CH3CTRL	R/W	H'FEA0 002C		32
CH4 control register	CH4CTRL	R/W	H'FEA0 0120		32
CH5 control register	CH5CTRL	R/W	H'FEA0 0124		32
CH6 control register	CH6CTRL	R/W	H'FEA0 0128		32
CH7 control register	CH7CTRL	R/W	H'FEA0 012C		32
CH0 input/output swap register	CH0SWAP	R/W	H'FEA0 0030		32
CH1 input/output swap register	CH1SWAP	R/W	H'FEA0 0034		32
CH2 input/output swap register	CH2SWAP	R/W	H'FEA0 0038		32
CH3 input/output swap register	CH3SWAP	R/W	H'FEA0 003C		32
CH4 input/output swap register	CH4SWAP	R/W	H'FEA0 0130		32
CH5 input/output swap register	CH5SWAP	R/W	H'FEA0 0134		32
CH6 input/output swap register	CH6SWAP	R/W	H'FEA0 0138		32
CH7 input/output swap register	CH7SWAP	R/W	H'FEA0 013C		32
CH0 source address register	CH0SAR	R/W	H'FEA0 0080		32
CH0 destination address register	CH0DAR	R/W	H'FEA0 0084		32
CH0 destination pixel register	CH0DPXL	R/W	H'FEA0 0088		32
CH0 source format register	CH0SFMT	R/W	H'FEA0 008C		32
CH0 destination format register	CH0DFMT	R/W	H'FEA0 0090		32
CH0 source line address register	CH0SARE	R	H'FEA0 0094		32
CH0 destination line address register	CH0DARE	R	H'FEA0 0098		32
CH0 destination pixel processing register	CH0DPXLE	R	H'FEA0 009C		32
CH1 source address register	CH1SAR	R/W	H'FEA0 00A0		32
CH1 destination address register	CH1DAR	R/W	H'FEA0 00A4		32
CH1 destination pixel register	CH1DPXL	R/W	H'FEA0 00A8		32
CH1 source format register	CH1SFMT	R/W	H'FEA0 00AC		32
CH1 destination format register	CH1DFMT	R/W	H'FEA0 00B0		32
CH1 source line address register	CH1SARE	R	H'FEA0 00B4		32
CH1 destination line address register	CH1DARE	R	H'FEA0 00B8		32
CH1 destination pixel processing register	CH1DPXLE	R	H'FEA0 00BC		32
CH2 source address register	CH2SAR	R/W	H'FEA0 00C0		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
CH2 destination address register	CH2DAR	R/W	H'FEA0 00C4	2D-DMAC	32
CH2 destination pixel register	CH2DPXL	R/W	H'FEA0 00C8		32
CH2 source format register	CH2SFMT	R/W	H'FEA0 00CC		32
CH2 destination format register	CH2DFMT	R/W	H'FEA0 00D0		32
CH2 source line address register	CH2SARE	R	H'FEA0 00D4		32
CH2 destination line address register	CH2DARE	R	H'FEA0 00D8		32
CH2 destination pixel processing register	CH2DPXLE	R	H'FEA0 00DC		32
CH3 source address register	CH3SAR	R/W	H'FEA0 00E0		32
CH3 destination address register	CH3DAR	R/W	H'FEA0 00E4		32
CH3 destination pixel register	CH3DPXL	R/W	H'FEA0 00E8		32
CH3 source format register	CH3SFMT	R/W	H'FEA0 00EC		32
CH3 destination format register	CH3DFMT	R/W	H'FEA0 00F0		32
CH3 source line address register	CH3SARE	R	H'FEA0 00F4		32
CH3 destination line address register	CH3DARE	R	H'FEA0 00F8		32
CH3 destination pixel processing register	CH3DPXLE	R	H'FEA0 00FC		32
CH4 source address register	CH4SAR	R/W	H'FEA0 0180		32
CH4 destination address register	CH4DAR	R/W	H'FEA0 0184		32
CH4 destination pixel register	CH4DPXL	R/W	H'FEA0 0188		32
CH4 source format register	CH4SFMT	R/W	H'FEA0 018C		32
CH4 destination format register	CH4DFMT	R/W	H'FEA0 0190		32
CH4 source line address register	CH4SARE	R	H'FEA0 0194		32
CH4 destination line address register	CH4DARE	R	H'FEA0 0198		32
CH4 destination pixel processing register	CH4DPXLE	R	H'FEA0 019C		32
CH5 source address register	CH5SAR	R/W	H'FEA0 01A0		32
CH5 destination address register	CH5DAR	R/W	H'FEA0 01A4		32
CH5 destination pixel register	CH5DPXL	R/W	H'FEA0 01A8		32
CH5 source format register	CH5SFMT	R/W	H'FEA0 01AC		32
CH5 destination format register	CH5DFMT	R/W	H'FEA0 01B0		32
CH5 source line address register	CH5SARE	R	H'FEA0 01B4		32
CH5 destination line address register	CH5DARE	R	H'FEA0 01B8		32
CH5 destination pixel processing register	CH5DPXLE	R	H'FEA0 01BC		32
CH6 source address register	CH6SAR	R/W	H'FEA0 01C0		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
CH6 destination address register	CH6DAR	R/W	H'FEA0 01C4	2D-DMAC	32
CH6 destination pixel register	CH6DPXL	R/W	H'FEA0 01C8		32
CH6 source format register	CH6SFMT	R/W	H'FEA0 01CC		32
CH6 destination format register	CH6DFMT	R/W	H'FEA0 01D0		32
CH6 source line address register	CH6SARE	R	H'FEA0 01D4		32
CH6 destination line address register	CH6DARE	R	H'FEA0 01D8		32
CH6 destination pixel processing register	CH6DPXLE	R	H'FEA0 01DC		32
CH7 source address register	CH7SAR	R/W	H'FEA0 01E0		32
CH7 destination address register	CH7DAR	R/W	H'FEA0 01E4		32
CH7 destination pixel register	CH7DPXL	R/W	H'FEA0 01E8		32
CH7 source format register	CH7SFMT	R/W	H'FEA0 01EC		32
CH7 destination format register	CH7DFMT	R/W	H'FEA0 01F0		32
CH7 source line address register	CH7SARE	R	H'FEA0 01F4		32
CH7 destination line address register	CH7DARE	R	H'FEA0 01F8		32
CH7 destination pixel processing register	CH7DPXLE	R	H'FEA0 01FC		32
TSIF control register	TSCTLR	R/W	H'A4C8 0000	TSIF	32
TSIF PID data register	TSPIDR	R/W	H'A4C8 0004		32
TSIF command register	TSCMDR	R/W	H'A4C8 0008		32
TSIF interrupt status register	TSSTR	R/W	H'A4C8 000C		32
TSIF TS data register	TSTSDR	R	H'A4C8 0010		32
TSIF buffer clear register	TSBUFCLRR	R/W	H'A4C8 0014		32
TSIF interrupt enable register	TSINTER	R/W	H'A4C8 0018		32
TSIF PSCALE register	TSPSCALER	R/W	H'A4C8 0020		32
TSIF PSCALE_R register	TSPSCALERR	R/W	H'A4C8 0024		32
TSIF PCRADCMODE register	TSPCRADCMDR	R/W	H'A4C8 0028		32
TSIF PCRADC register	TSPCRADCR	R/W	H'A4C8 002C		32
TSIF TR_PCRADC register	TSTRPCRADCR	R/W	H'A4C8 0030		32
TSIF D_PCRADC register	TSDPCRADCR	R/W	H'A4C8 0034		32
TSIF free-running counter	TSFRC	R/W	H'A4C8 0040		32
Port A output serial format register	A_DO_FMT	R/W	H'FE3C0000	FSI	32
Port A output FIFO control register	A_DOFF_CTL	R/W	H'FE3C0004		32
Port A output FIFO status register	A_DOFF_ST	R/W	H'FE3C0008		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
Port A input serial format register	A_DI_FMT	R/W	H'FE3C000C	FSI	32
Port A input FIFO control register	A_DIFF_CTL	R/W	H'FE3C0010		32
Port A input FIFO status register	A_DIFF_ST	R/W	H'FE3C0014		32
Port A clock set 1 register	A_CKG1	R/W	H'FE3C0018		32
Port A clock set 2 register	A_CKG2	R/W	H'FE3C001C		32
Port A read data register	A_DIDT	R	H'FE3C0020		32
Port A write data register	A_DODT	W	H'FE3C0024		32
Port A MUTE state register	A_MUTE_ST	R	H'FE3C0028		32
Port B output serial format register	B_DO_FMT	R/W	H'FE3C0040		32
Port B output FIFO control register	B_DOFF_CTL	R/W	H'FE3C0044		32
Port B output FIFO status register	B_DOFF_ST	R/W	H'FE3C0048		32
Port B input serial format register	B_DI_FMT	R/W	H'FE3C004C		32
Port B input FIFO control register	B_DIFF_CTL	R/W	H'FE3C0050		32
Port B input FIFO status register	B_DIFF_ST	R/W	H'FE3C0054		32
Port B clock set 1 register	B_CKG1	R/W	H'FE3C0058		32
Port B clock set 2 register	B_CKG2	R/W	H'FE3C005C		32
Port B read data register	B_DIDT	R	H'FE3C0060		32
Port B write data register	B_DODT	W	H'FE3C0064		32
Port B MUTE state register	B_MUTE_ST	R	H'FE3C0068		32
Interrupt state register	INT_ST	R/W	H'FE3C0200		32
Interrupt source mask register	IEMSK	R/W	H'FE3C0204		32
Interrupt signal mask register	IMSK	R/W	H'FE3C0208		32
MUTE set register	MUTE	R/W	H'FE3C020C		32
Clock reset register	CLK_RST	R/W	H'FE3C0210		32
Software reset register	SOFT_RST	R/W	H'FE3C0214		32
FIFO size register	FIFO_SZ	R	H'FE3C0218		32
ATAPI control register	ATAPI_CONTROL1	R/W	H'A4DA 2180	ATAPI	32
ATAPI Status Register	ATAPI_STATUS	R/W	H'A4DA 2184		32
Interrupt enable register	ATAPI_INT_ENABLE	R/W	H'A4DA 2188		32
Descriptor Table Base Address Register	ATAPI_DTB_ADR	R/W	H'A4DA 2198		32
DMA Start Address Register	ATAPI_DMA_START_ADR	R/W	H'A4DA 219C		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
DMA Transfer Count Register	ATAPI_DMA_TRANS_CNT	R/W	H'A4DA 21A0	ATAPI	32
ATAPI Control 2 Register	ATAPI_CONTROL2	R/W	H'A4DA 21A4		32
ATAPI Signal Status Register	ATAPI_SIG_ST	R	H'A4DA 21B0		32
Byte swap Register	ATAPI_BYTE_SWAP	R/W	H'A4DA 21BC		32
PIO Timing Register 1	ATAPI_PIO_TIMING1	R/W	H'A4DA 21C0		32
PIO Timing Register 2	ATAPI_PIO_TIMING2	R/W	H'A4DA 21C4		32
Multiword DMA Timing Register	ATAPI_MULTI_TIMING	R/W	H'A4DA 21C8		32
Ultra DMA Timing Register	ATAPI_ULTRA_TIMING	R/W	H'A4DA 21CC		32
E-DMAC mode register	EDMR	R/W	H'A460 0000	EtherMAC	32
E-DMAC transmit request register	EDTRR	R/W	H'A460 0008		32
E-DMAC receive request register	EDRRR	R/W	H'A460 0010		32
Transmit descriptor list start address register	TDLAR	R/W	H'A460 0018		32
Receive descriptor list start address register	RDLAR	R/W	H'A460 0020		32
EtherC/E-DMAC status register	EESR	R/W	H'A460 0028		32
EtherC/E-DMAC status interrupt permission register	EESIPR	R/W	H'A460 0030		32
Transmit/receive status copy enable register	TRSCER	R/W	H'A460 0038		32
Receive missed-frame counter register	RMFCR	R/W	H'A460 0040		32
Transmit FIFO threshold register	TFTR	R/W	H'A460 0048		32
FIFO depth register	FDR	R/W	H'A460 0050		32
Receiving method control register	RMCR	R/W	H'A460 0058		32
Transmit FIFO Undercount Counter	TFUCR	R/W	H'A460 0064		32
Receive FIFO Overflow Counter	RFOCR	R/W	H'A460 0068		32
Flow control start FIFO threshold setting register	FCFTR	R/W	H'A460 0070		32
Transmit Interrupt Setting Register	TRIMD	R/W	H'A460 007C		32
EtherC mode register	ECMR	R/W	H'A460 0100		32
Receive frame length upper limit register	RFLR	R/W	H'A460 0108		32
EtherC status register	ECSR	R/W	H'A460 0110		32
EtherC interrupt permission register	ECSIPR	R/W	H'A460 0118		32
PHY interface register	PIR	R/W	H'A460 0120		32
PHY status register	PSR	R	H'A460 0128		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
Random number generation counter upper limit value	RDMLR	R/W	H'A460 0140	EtherMAC	32
IPG register	IPGR	R/W	H'A460 0150		32
Automatic PAUSE frame register	APR	R/W	H'A460 0154		32
Manual PAUSE frame register	MPR	W	H'A460 0158		32
Receive PAUSE frame counter	RFCF	R	H'A460 0160		32
Automatic PAUSE frame retransmit counter register	TPAUSER	R/W	H'A460 0164		32
PAUSE frame retransmit counter register	TPAUSECR	R	H'A460 0168		32
EtherC mode register	ECMR	R/W	H'A460 0100		32
Receive frame length upper limit register	RFLR	R/W	H'A460 0108		32
EtherC status register	ECSR	R/W	H'A460 0110		32
EtherC interrupt••••••	ECSIPR	R/W	H'A460 0118		32
PHY interface register	PIR	R/W	H'A460 0120		32
PHY status register	PSR	R/W	H'A460 0128		32
Random number generation counter upper limit value	RDMLR	R/W	H'A460 0140		32
IPG register	IPGR	R/W	H'A460 0150		32
Automatic PAUSE frame register	APR	R/W	H'A460 0154		32
Manual PAUSE frame register	MPR	R/W	H'A460 0158		32
Receive PAUSE frame counter	RFCF	R/W	H'A460 0160		32
Automatic PAUSE frame retransmit counter register	TPAUSER	R/W	H'A460 0164		32
PAUSE frame retransmit counter register	TPAUSECR	R/W	H'A460 0168		32
MAC address high register	MAHR	R/W	H'A460 01C0		32
MAC address low register	MALR	R/W	H'A460 01C8		32
Transmit retry over counter register	TROCR	R/W	H'A460 01D0		32
Delayed collision detect counter register	CDCR	R/W	H'A460 01D4		32
Lost carrier counter register	LCCR	R/W	H'A460 01D8		32
Carrier not detect counter register	CNDCR	R/W	H'A460 01DC		32
CRC error frame receive counter register	CEFCR	R/W	H'A460 01E4		32
Frame receive error counter register	FRECR	R/W	H'A460 01E8		32
Too-short frame receive counter register	TSFRCR	R/W	H'A460 01EC		32

Register Name	Abbreviation	R/W	Address	Module	Access Size
Too-long frame receive counter register	TLFRCR	R/W	H'A460 01F0	EtherMAC	32
Residual-bit frame receive counter register	RFCR	R/W	H'A460 01F4		32
Multicast address frame receive counter register	MAFCR	R/W	H'A460 01F8		32
Port A control register	PACR	R/W	H'A405 0100	PFC	16
Port B control register	PBCR	R/W	H'A405 0102		16
Port C control register	PCCR	R/W	H'A405 0104		16
Port D control register	PDCR	R/W	H'A405 0106		16
Port E control register	PECR	R/W	H'A405 0108		16
Port F control register	PFCR	R/W	H'A405 010A		16
Port G control register	PGCR	R/W	H'A405 010C		16
Port H control register	PHCR	R/W	H'A405 010E		16
Port J control register	PJCR	R/W	H'A405 0110		16
Port K control register	PKCR	R/W	H'A405 0112		16
Port L control register	PLCR	R/W	H'A405 0114		16
Port M control register	PMCR	R/W	H'A405 0116		16
Port N control register	PNCR	R/W	H'A405 0118		16
Port Q control register	PQCR	R/W	H'A405 011A		16
Port R control register	PRCR	R/W	H'A405 011C		16
Port S control register	PSCR	R/W	H'A405 011E		16
Port T control register	PTCR	R/W	H'A405 0140		16
Port U control register	PUCR	R/W	H'A405 0142		16
Port V control register	PVCR	R/W	H'A405 0144		16
Port W control register	PWCR	R/W	H'A405 0146		16
Port X control register	PXCR	R/W	H'A405 0148		16
Port Y control register	PYCR	R/W	H'A405 014A		16
Port Z control register	PZCR	R/W	H'A405 014C		16
Pin select register A	PSELA	R/W	H'A405 014E		16
Pin select register B	PSELB	R/W	H'A405 0150		16
Pin select register C	PSELC	R/W	H'A405 0152		16
Pin select register D	PSELD	R/W	H'A405 0154		16
Port A data register	PADR	R/W	H'A405 0120		8

Register Name	Abbreviation	R/W	Address	Module	Access Size
Port B data register	PBDR	R/W	H'A405 0122	PFC	8
Port C data register	PCDR	R/W	H'A405 0124		8
Port D data register	PDDR	R/W	H'A405 0126		8
Port E data register	PEDR	R/W	H'A405 0128		8
Port F data register	PFDR	R/W	H'A405 012A		8
Port G data register	PGDR	R/W	H'A405 012C		8
Port H data register	PHDR	R/W	H'A405 012E		8
Port J data register	PJDR	R/W	H'A405 0130		8
Port K data register	PKDR	R/W	H'A405 0132		8
Port L data register	PLDR	R/W	H'A405 0134		8
Port M data register	PMDR	R/W	H'A405 0136		8
Port N data register	PNDR	R/W	H'A405 0138		8
Port Q data register	PQDR	R/W	H'A405 013A		8
Port R data register	PRDR	R/W	H'A405 013C		8
Port S data register	PSDR	R/W	H'A405 013E		8
Port T data register	PTDR	R/W	H'A405 0160		8
Port U data register	PUDR	R/W	H'A405 0162		8
Port V data register	PVDR	R/W	H'A405 0164		8
Port W data register	PWDR	R/W	H'A405 0166		8
Port X data register	PXDR	R/W	H'A405 0168		8
Port Y data register	PYDR	R/W	H'A405 016A		8
Port Z data register	PZDR	R/W	H'A405 016C		8
I/O buffer Hi-Z control register A	HIZCRA	R/W	H'A405 0158		16
I/O buffer Hi-Z control register B	HIZCRB	R/W	H'A405 015A		16
I/O buffer Hi-Z control register C	HIZCRC	R/W	H'A405 015C		16
I/O buffer Hi-Z control register D	HIZCRD	R/W	H'A405 015E		16
Module function select register A	MSELCRA	R/W	H'A405 0180		16
Module function select register B	MSELCRB	R/W	H'A405 0182		16
Pull-up control register	PULCR	R/W	H'A405 0184		16
I/O buffer drive control register A	DRVCRA	R/W	H'A405 018A		16
I/O buffer drive control register B	DRVCRB	R/W	H'A405 018C		16
I/O buffer drive control register C	DRVCRC	R/W	H'A405 018E		16
Match condition setting register 0	CBR0	R/W	H'FF20 0000	UBC	32

Register Name	Abbreviation	R/W	Address	Module	Access Size
Match operation setting register 0	CRR0	R/W	H'FF20 0004	UBC	32
Match address setting register 0	CAR0	R/W	H'FF20 0008		32
Match address mask setting register 0	CAMR0	R/W	H'FF20 000C		32
Match condition setting register 1	CBR1	R/W	H'FF20 0020		32
Match operation setting register 1	CRR1	R/W	H'FF20 0024		32
Match address setting register 1	CAR1	R/W	H'FF20 0028		32
Match address mask setting register 1	CAMR1	R/W	H'FF20 002C		32
Match data setting register 1	CDR1	R/W	H'FF20 0030		32
Match data mask setting register 1	CDMR1	R/W	H'FF20 0034		32
Execution count break register 1	CETR1	R/W	H'FF20 0038		32
Channel match flag register	CCMFR	R/W	H'FF20 0600		32
Break control register	CBCR	R/W	H'FF20 0620		32
Instruction register	SDIR	R	H'FC11 0000	H-UDI	16
Data register H	SDDR/SDDRH	R/W	H'FC11 0008		32/16
Data register L	SDDRL	R/W	H'FC11 000A		16
Interrupt source register	SDINT	R/W	H'FC11 0018		16
Bypass register	SDBPR	—	—		—

Notes: 1. 8 bits when reading and 16 bits when writing.
2. Only the part of the register at H'A4CA 0034 is accessible through 16-bit access

Table51.1 Register Addresses of CEU (CEU0, CEU1)

Register Name	Abbreviation	R/W	Address (Plane A)	Address B (Plane B)	Mirror Address	Access Size
CEU0 Capture start register	CAPSR_0	R/W	H'FE91 0000	—	—	32
CEU0 Capture control register	CAPCR_0	R/W	H'FE91 0004	—	—	32
CEU0 Capture interface control register*	CAMCR_0	R/W	H'FE91 0008	—	—	32
CEU0 Capture interface cycle register*	CMCYR_0	R/W	H'FE91 000C	—	—	32
CEU0 Capture interface offset register	CAMOR_0	R/W	H'FE91 0010	H'FE91 1010	H'FE91 2010	32
CEU0 Capture interface width register	CAPWR_0	R/W	H'FE91 0014	H'FE91 1014	H'FE91 2014	32
CEU0 Capture interface input format register	CAIFR_0	R/W	H'FE91 0018	—	—	32
CEU0 register control register	CRCNTR_0	R/W	H'FE91 0028	—	—	32
CEU0 register forcible control register	CRCMPR_0	R/W	H'FE91 002C	—	—	32
CEU0 Capture filter control register	CFLCR_0	R/W	H'FE91 0030	H'FE91 1030	H'FE91 2030	32
CEU0 Capture filter size clip register	CFSZR_0	R/W	H'FE91 0034	H'FE91 1034	H'FE91 2034	32
CEU0 Capture destination width register	CDWDR_0	R/W	H'FE91 0038	H'FE91 1038	H'FE91 2038	32
CEU0 Capture data address Y register	CDAYR_0	R/W	H'FE91 003C	H'FE91 103C	H'FE91 203C	32
CEU0 Capture data address C register	CDACR_0	R/W	H'FE91 0040	H'FE91 1040	H'FE91 2040	32
CEU0 Capture data bottom-field address Y register	CDBYR_0	R/W	H'FE91 0044	H'FE91 1044	H'FE91 2044	32
CEU0 Capture data bottom-field address C register	CDBCR_0	R/W	H'FE91 0048	H'FE91 1048	H'FE91 2048	32
CEU0 Capture bundle destination size register	CBDSR_0	R/W	H'FE91 004C	H'FE91 104C	H'FE91 204C	32
CEU0 Firewall operation control register	CFWCR_0	R/W	H'FE91 005C	—	—	32
CEU0 Capture low-pass filter control register	CLFCR_0	R/W	H'FE91 0060	H'FE91 1060	H'FE91 2060	32
CEU0 Capture data output control register	CDOCR_0	R/W	H'FE91 0064	H'FE91 1064	H'FE91 2064	32
CEU0 Capture data complexity level register	CDDCR_0	R/W	H'FE91 0068	H'FE91 1068	H'FE91 2068	32
CEU0 Capture data complexity level address register	CDDAR_0	R/W	H'FE91 006C	H'FE91 106C	H'FE91 206C	32
CEU0 Capture event interrupt enable register	CEIER_0	R/W	H'FE91 0070	—	—	32
CEU0 Capture event flag clear register	CETCR_0	R/W	H'FE91 0074	—	—	32
CEU0 Capture status register	CSTSR_0	R	H'FE91 007C	—	—	32
CEU0 Capture software reset register	CSRTR_0	R/W	H'FE91 0080	—	—	32
CEU0 Capture data size register	CDSSR_0	R/W	H'FE91 0084	—	—	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address B (Plane B)	Mirror Address	Access Size
CEU0 Capture data address Y register 2	CDAYR2_0	R/W	H'FE91 0090	H'FE91 1090	H'FE91 2090	32
CEU0 Capture data address C register 2	CDACR2_0	R/W	H'FE91 0094	H'FE91 1094	H'FE91 2094	32
CEU0 Capture data bottom-field address Y register 2	CDBYR2_0	R/W	H'FE91 0098	H'FE91 1098	H'FE91 2098	32
CEU0 Capture data bottom-field address C register 2	CDBCR2_0	R/W	H'FE91 009C	H'FE91 109C	H'FE91 209C	32
CEU1 Capture start register	CAPSR_1	R/W	H'FE91 4000	—	—	32
CEU1 Capture control register	CAPCR_1	R/W	H'FE91 4004	—	—	32
CEU1 Capture interface control register*	CAMCR_1	R/W	H'FE91 4008	—	—	32
CEU1 Capture interface cycle register*	CMCYR_1	R/W	H'FE91 400C	—	—	32
CEU1 Capture interface offset register	CAMOR_1	R/W	H'FE91 4010	H'FE91 5010	H'FE91 6010	32
CEU1 Capture interface width register	CAPWR_1	R/W	H'FE91 4014	H'FE91 5014	H'FE91 6014	32
CEU1 Capture interface input format register	CAIFR_1	R/W	H'FE91 4018	—	—	32
CEU1 register control register	CRCNTR_1	R/W	H'FE91 4028	—	—	32
CEU1 register forcible control register	CRCMPR_1	R/W	H'FE91 402C	—	—	32
CEU1 Capture filter control register	CFLCR_1	R/W	H'FE91 4030	H'FE91 5030	H'FE91 6030	32
CEU1 Capture filter size clip register	CFSZR_1	R/W	H'FE91 4034	H'FE91 5034	H'FE91 6034	32
CEU1 Capture destination width register	CDWDR_1	R/W	H'FE91 4038	H'FE91 5038	H'FE91 6038	32
CEU1 Capture data address Y register	CDAYR_1	R/W	H'FE91 403C	H'FE91 503C	H'FE91 603C	32
CEU1 Capture data address C register	CDACR_1	R/W	H'FE91 4040	H'FE91 5040	H'FE91 6040	32
CEU1 Capture data bottom-field address Y register	CDBYR_1	R/W	H'FE91 4044	H'FE91 5044	H'FE91 6044	32
CEU1 Capture data bottom-field address C register	CDBCR_1	R/W	H'FE91 4048	H'FE91 5048	H'FE91 6048	32
CEU1 Capture bundle destination size register	CBDSR_1	R/W	H'FE91 404C	H'FE91 504C	H'FE91 604C	32
CEU1 Firewall operation control register	CFWCR_1	R/W	H'FE91 405C	—	—	32
CEU1 Capture low-pass filter control register	CLFCR_1	R/W	H'FE91 4060	H'FE91 5060	H'FE91 6060	32
CEU1 Capture data output control register	CDOCR_1	R/W	H'FE91 4064	H'FE91 5064	H'FE91 6064	32
CEU1 Capture data complexity level register	CDDCR_1	R/W	H'FE91 4068	H'FE91 5068	H'FE91 6068	32
CEU1 Capture data complexity level address register	CDDAR_1	R/W	H'FE91 406C	H'FE91 506C	H'FE91 606C	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address B (Plane B)	Mirror Address	Access Size
CEU1 Capture event interrupt enable register	CEIER_1	R/W	H'FE91 4070	—	—	32
CEU1 Capture event flag clear register	CETCR_1	R/W	H'FE91 4074	—	—	32
CEU1 Capture status register	CSTSR_1	R	H'FE91 407C	—	—	32
CEU1 Capture software reset register	CSRTR_1	R/W	H'FE91 4080	—	—	32
CEU1 Capture data size register	CDSSR_1	R/W	H'FE91 4084	—	—	32
CEU1 Capture data address Y register 2	CDAYR2_1	R/W	H'FE91 4090	H'FE91 5090	H'FE91 6090	32
CEU1 Capture data address C register 2	CDACR2_1	R/W	H'FE91 4094	H'FE91 5094	H'FE91 6094	32
CEU1 Capture data bottom-field address Y register 2	CDBYR2_1	R/W	H'FE91 4098	H'FE91 5098	H'FE91 6098	32
CEU1 Capture data bottom-field address C register 2	CDBCR2_1	R/W	H'FE91 409C	H'FE91 509C	H'FE91 609C	32

Note: * After changing the setting of a register (CAMCR or CMCYR) that is determined by the external module characteristics, do not start capture until at least 10 external input clock cycles have elapsed.

Table 51.2 Register Addresses of BEU (BEU0, BEU1)

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
BEU0 start register	BESTR_0	R/W	H'FE93 0000	—	—	32
BEU0 source memory width register 1	BSMWR1_0	R/W	H'FE93 0010	H'FE93 1010	H'FE93 2010	32
BEU0 source size register 1	BSSZR1_0	R/W	H'FE93 0014	H'FE93 1014	H'FE93 2014	32
BEU0 source address Y register 1	BSAYR1_0	R/W	H'FE93 0018	H'FE93 1018	H'FE93 2018	32
BEU0 source address C register 1	BSACR1_0	R/W	H'FE93 001C	H'FE93 101C	H'FE93 201C	32
BEU0 source address α register 1	BSAAR1_0	R/W	H'FE93 0020	H'FE93 1020	H'FE93 2020	32
BEU0 source image format register 1	BSIFR1_0	R/W	H'FE93 0024	H'FE93 1024	H'FE93 2024	32
BEU0 source memory width register 2	BSMWR2_0	R/W	H'FE93 0028	H'FE93 1028	H'FE93 2028	32
BEU0 source size register 2	BSSZR2_0	R/W	H'FE93 002C	H'FE93 102C	H'FE93 202C	32
BEU0 source address Y register 2	BSAYR2_0	R/W	H'FE93 0030	H'FE93 1030	H'FE93 2030	32
BEU0 source address C register 2	BSACR2_0	R/W	H'FE93 0034	H'FE93 1034	H'FE93 2034	32
BEU0 source address α register 2	BSAAR2_0	R/W	H'FE93 0038	H'FE93 1038	H'FE93 2038	32
BEU0 source image format register 2	BSIFR2_0	R/W	H'FE93 003C	H'FE93 103C	H'FE93 203C	32
BEU0 source memory width register 3	BSMWR3_0	R/W	H'FE93 0040	H'FE93 1040	H'FE93 2040	32
BEU0 source size register 3	BSSZR3_0	R/W	H'FE93 0044	H'FE93 1044	H'FE93 2044	32
BEU0 source address Y register 3	BSAYR3_0	R/W	H'FE93 0048	H'FE93 1048	H'FE93 2048	32
BEU0 source address C register 3	BSACR3_0	R/W	H'FE93 004C	H'FE93 104C	H'FE93 204C	32
BEU0 source address α register 3	BSAAR3_0	R/W	H'FE93 0050	H'FE93 1050	H'FE93 2050	32
BEU0 source image format register 3	BSIFR3_0	R/W	H'FE93 0054	H'FE93 1054	H'FE93 2054	32
BEU0 tile pattern size register	BTPSR_0	R/W	H'FE93 0058	H'FE93 1058	H'FE93 2058	32
BEU0 multidisplay source memory width register 1	BMSMWR1_0	R/W	H'FE93 0070	—	—	32
BEU0 multidisplay source size register 1	BMSSZR1_0	R/W	H'FE93 0074	—	—	32
BEU0 multidisplay source address Y register 1	BMSAYR1_0	R/W	H'FE93 0078	—	—	32
BEU0 multidisplay source address C register 1	BMSACR1_0	R/W	H'FE93 007C	—	—	32
BEU0 multidisplay source memory width register 2	BMSMWR2_0	R/W	H'FE93 0080	—	—	32
BEU0 multidisplay source size register 2	BMSSZR2_0	R/W	H'FE93 0084	—	—	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
BEU0 multidisplay source address Y register 2	BMSAYR2_0	R/W	H'FE93 0088	—	—	32
BEU0 multidisplay source address C register 2	BMSACR2_0	R/W	H'FE93 008C	—	—	32
BEU0 multidisplay source memory width register 3	BMSMWR3_0	R/W	H'FE93 0090	—	—	32
BEU0 multidisplay source size register 3	BMSSZR3_0	R/W	H'FE93 0094	—	—	32
BEU0 multidisplay source address Y register 3	BMSAYR3_0	R/W	H'FE93 0098	—	—	32
BEU0 multidisplay source address C register 3	BMSACR3_0	R/W	H'FE93 009C	—	—	32
BEU0 multidisplay source memory width register 4	BMSMWR4_0	R/W	H'FE93 00A0	—	—	32
BEU0 multidisplay source size register 4	BMSSZR4_0	R/W	H'FE93 00A4	—	—	32
BEU0 multidisplay source address Y register 4	BMSAYR4_0	R/W	H'FE93 00A8	—	—	32
BEU0 multidisplay source address C register 4	BMSACR4_0	R/W	H'FE93 00AC	—	—	32
BEU0 multidisplay source image format register	BMSIFR_0	R/W	H'FE93 00F0	—	—	32
BEU0 blend control register 0	BBLCR0_0	R/W	H'FE93 0100	H'FE93 1100	H'FE93 2100	32
BEU0 blend control register 1	BBLCR1_0	R/W	H'FE93 0104	—	—	32
BEU0 process control register	BPROCR_0	R/W	H'FE93 0108	H'FE93 1108	H'FE93 2108	32
BEU0 multiwindow control register 0	BMWCR0_0	R/W	H'FE93 010C	—	—	32
BEU0 Blend location register 1	BLOCR1_0	R/W	H'FE93 0114	H'FE93 1114	H'FE93 2114	32
BEU0 Blend location register 2	BLOCR2_0	R/W	H'FE93 0118	H'FE93 1118	H'FE93 2118	32
BEU0 Blend location register 3	BLOCR3_0	R/W	H'FE93 011C	H'FE93 111C	H'FE93 211C	32
BEU0 multidisplay location register 1	BMLOCR1_0	R/W	H'FE93 0120	—	—	32
BEU0 multidisplay location register 2	BMLOCR2_0	R/W	H'FE93 0124	—	—	32
BEU0 multidisplay location register 3	BMLOCR3_0	R/W	H'FE93 0128	—	—	32
BEU0 multidisplay location register 4	BMLOCR4_0	R/W	H'FE93 012C	—	—	32
BEU0 multidisplay transparent color control register 1	BMPCCR1_0	R/W	H'FE93 0130	—	—	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
BEU0 multidisplay transparent color control register 2	BMPCCR2_0	R/W	H'FE93 0134	—	—	32
BEU0 Blend pack form register	BPKFR_0	R/W	H'FE93 0140	H'FE93 1140	H'FE93 2140	32
BEU0 transparent color control register 0	BPCCR0_0	R/W	H'FE93 0144	H'FE93 1144	H'FE93 2144	32
BEU0 transparent color control register 11	BPCCR11_0	R/W	H'FE93 0148	H'FE93 1148	H'FE93 2148	32
BEU0 transparent color control register 12	BPCCR12_0	R/W	H'FE93 014C	H'FE93 114C	H'FE93 214C	32
BEU0 transparent color control register 21	BPCCR21_0	R/W	H'FE93 0150	H'FE93 1150	H'FE93 2150	32
BEU0 transparent color control register 22	BPCCR22_0	R/W	H'FE93 0154	H'FE93 1154	H'FE93 2154	32
BEU0 transparent color control register 31	BPCCR31_0	R/W	H'FE93 0158	H'FE93 1158	H'FE93 2158	32
BEU0 transparent color control register 32	BPCCR32_0	R/W	H'FE93 015C	H'FE93 115C	H'FE93 215C	32
BEU0 destination memory width register	BDMWR_0	R/W	H'FE93 0160	H'FE93 1160	H'FE93 2160	32
BEU0 destination address Y register	BDAYR_0	R/W	H'FE93 0164	H'FE93 1164	H'FE93 2164	32
BEU0 destination address C register	BDACR_0	R/W	H'FE93 0168	H'FE93 1168	H'FE93 2168	32
BEU0 address fixed register	BAFXR_0	R/W	H'FE93 0180	H'FE93 1180	H'FE93 2180	32
BEU0 swapping register	BSWPR_0	R/W	H'FE93 0184	H'FE93 1184	H'FE93 2184	32
BEU0 event interrupt enable register	BEIER_0	R/W	H'FE93 0188	—	—	32
BEU0 event register	BEVTR_0	R/W	H'FE93 018C	—	—	32
BEU0 register control register	BRCNTR_0	R/W	H'FE93 0194	—	—	32
BEU0 status register	BSTAR_0	R	H'FE93 0198	—	—	32
BEU0 module reset register	BBRSTR_0	R/W	H'FE93 019C	—	—	32
BEU0 register-plane forcible setting register	BRCHR_0	R/W	H'FE93 01A0	—	—	32
BEU0 Color Lookup Table	CLUT_0	R/W	H'FE93 3000 to H'FE93 33FF	—	—	32
BEU1 start register	BESTR_1	R/W	H'FE93 4000	—	—	32
BEU1 source memory width register 1	BSMWR1_1	R/W	H'FE93 4010	H'FE93 5010	H'FE93 6010	32
BEU1 source size register 1	BSSZR1_1	R/W	H'FE93 4014	H'FE93 5014	H'FE93 6014	32
BEU1 source address Y register 1	BSAYR1_1	R/W	H'FE93 4018	H'FE93 5018	H'FE93 6018	32
BEU1 source address C register 1	BSACR1_1	R/W	H'FE93 401C	H'FE93 501C	H'FE93 601C	32
BEU1 source address α register 1	BSAAR1_1	R/W	H'FE93 4020	H'FE93 5020	H'FE93 6020	32
BEU1 source image format register 1	BSIFR1_1	R/W	H'FE93 4024	H'FE93 5024	H'FE93 6024	32
BEU1 source memory width register 2	BSMWR2_1	R/W	H'FE93 4028	H'FE93 5028	H'FE93 6028	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
BEU1 source size register 2	BSSZR2_1	R/W	H'FE93 402C	H'FE93 502C	H'FE93 602C	32
BEU1 source address Y register 2	BSAYR2_1	R/W	H'FE93 4030	H'FE93 5030	H'FE93 6030	32
BEU1 source address C register 2	BSACR2_1	R/W	H'FE93 4034	H'FE93 5034	H'FE93 6034	32
BEU1 source address α register 2	BSAAR2_1	R/W	H'FE93 4038	H'FE93 5038	H'FE93 6038	32
BEU1 source image format register 2	BSIFR2_1	R/W	H'FE93 403C	H'FE93 503C	H'FE93 603C	32
BEU1 source memory width register 3	BSMWR3_1	R/W	H'FE93 4040	H'FE93 5040	H'FE93 6040	32
BEU1 source size register 3	BSSZR3_1	R/W	H'FE93 4044	H'FE93 5044	H'FE93 6044	32
BEU1 source address Y register 3	BSAYR3_1	R/W	H'FE93 4048	H'FE93 5048	H'FE93 6048	32
BEU1 source address C register 3	BSACR3_1	R/W	H'FE93 404C	H'FE93 504C	H'FE93 604C	32
BEU1 source address α register 3	BSAAR3_1	R/W	H'FE93 4050	H'FE93 5050	H'FE93 6050	32
BEU1 source image format register 3	BSIFR3_1	R/W	H'FE93 4054	H'FE93 5054	H'FE93 6054	32
BEU1 tile pattern size register	BTPSR_1	R/W	H'FE93 4058	H'FE93 5058	H'FE93 6058	32
BEU1 multidisplay source memory width register 1	BMSMWR1_1	R/W	H'FE93 4070	—	—	32
BEU1 multidisplay source size register 1	BMSSZR1_1	R/W	H'FE93 4074	—	—	32
BEU1 multidisplay source address Y register 1	BMSAYR1_1	R/W	H'FE93 4078	—	—	32
BEU1 multidisplay source address C register 1	BMSACR1_1	R/W	H'FE93 407C	—	—	32
BEU1 multidisplay source memory width register 2	BMSMWR2_1	R/W	H'FE93 4080	—	—	32
BEU1 multidisplay source size register 2	BMSSZR2_1	R/W	H'FE93 4084	—	—	32
BEU1 multidisplay source address Y register 2	BMSAYR2_1	R/W	H'FE93 4088	—	—	32
BEU1 multidisplay source address C register 2	BMSACR2_1	R/W	H'FE93 408C	—	—	32
BEU1 multidisplay source memory width register 3	BMSMWR3_1	R/W	H'FE93 4090	—	—	32
BEU1 multidisplay source size register 3	BMSSZR3_1	R/W	H'FE93 4094	—	—	32
BEU1 multidisplay source address Y register 3	BMSAYR3_1	R/W	H'FE93 4098	—	—	32
BEU1 multidisplay source address C register 3	BMSACR3_1	R/W	H'FE93 409C	—	—	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
BEU1 multidisplay source memory width register 4	BMSMWR4_1	R/W	H'FE93 40A0	—	—	32
BEU1 multidisplay source size register 4	BMSSZR4_1	R/W	H'FE93 40A4	—	—	32
BEU1 multidisplay source address Y register 4	BMSAYR4_1	R/W	H'FE93 40A8	—	—	32
BEU1 multidisplay source address C register 4	BMSACR4_1	R/W	H'FE93 40AC	—	—	32
BEU1 multidisplay source image format register	BMSIFR_1	R/W	H'FE93 40F0	—	—	32
BEU1 blend control register 0	BBLCR0_1	R/W	H'FE93 4100	H'FE93 5100	H'FE93 6100	32
BEU1 blend control register 1	BBLCR1_1	R/W	H'FE93 4104	—	—	32
BEU1 process control register	BPROCR_1	R/W	H'FE93 4108	H'FE93 5108	H'FE93 6108	32
BEU1 multiwindow control register 0	BMWCR0_1	R/W	H'FE93 410C	—	—	32
BEU1 Blend location register 1	BLOCR1_1	R/W	H'FE93 4114	H'FE93 5114	H'FE93 6114	32
BEU1 Blend location register 2	BLOCR2_1	R/W	H'FE93 4118	H'FE93 5118	H'FE93 6118	32
BEU1 Blend location register 3	BLOCR3_1	R/W	H'FE93 411C	H'FE93 511C	H'FE93 611C	32
BEU1 multidisplay location register 1	BMLOCR1_1	R/W	H'FE93 4120	—	—	32
BEU1 multidisplay location register 2	BMLOCR2_1	R/W	H'FE93 4124	—	—	32
BEU1 multidisplay location register 3	BMLOCR3_1	R/W	H'FE93 4128	—	—	32
BEU1 multidisplay location register 4	BMLOCR4_1	R/W	H'FE93 412C	—	—	32
BEU1 multidisplay transparent color control register 1	BMPCCR1_1	R/W	H'FE93 4130	—	—	32
BEU1 multidisplay transparent color control register 2	BMPCCR2_1	R/W	H'FE93 4134	—	—	32
BEU1 Blend pack form register	BPKFR_1	R/W	H'FE93 4140	H'FE93 5140	H'FE93 6140	32
BEU1 transparent color control register 0	BPCCR0_1	R/W	H'FE93 4144	H'FE93 5144	H'FE93 6144	32
BEU1 transparent color control register 11	BPCCR11_1	R/W	H'FE93 4148	H'FE93 5148	H'FE93 6148	32
BEU1 transparent color control register 12	BPCCR12_1	R/W	H'FE93 414C	H'FE93 514C	H'FE93 614C	32
BEU1 transparent color control register 21	BPCCR21_1	R/W	H'FE93 4150	H'FE93 5150	H'FE93 6150	32
BEU1 transparent color control register 22	BPCCR22_1	R/W	H'FE93 4154	H'FE93 5154	H'FE93 6154	32
BEU1 transparent color control register 31	BPCCR31_1	R/W	H'FE93 4158	H'FE93 5158	H'FE93 6158	32
BEU1 transparent color control register 32	BPCCR32_1	R/W	H'FE93 415C	H'FE93 515C	H'FE93 615C	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
BEU1 destination memory width register	BDMWR_1	R/W	H'FE93 4160	H'FE93 5160	H'FE93 6160	32
BEU1 destination address Y register	BDAYR_1	R/W	H'FE93 4164	H'FE93 5164	H'FE93 6164	32
BEU1 destination address C register	BDACR_1	R/W	H'FE93 4168	H'FE93 5168	H'FE93 6168	32
BEU1 address fixed register	BAFXR_1	R/W	H'FE93 4180	H'FE93 5180	H'FE93 6180	32
BEU1 swapping register	BSWPR_1	R/W	H'FE93 4184	H'FE93 5184	H'FE93 6184	32
BEU1 event interrupt enable register	BEIER_1	R/W	H'FE93 4188	—	—	32
BEU1 event register	BEVTR_1	R/W	H'FE93 418C	—	—	32
BEU1 register control register	BRCNTR_1	R/W	H'FE93 4194	—	—	32
BEU1 status register	BSTAR_1	R	H'FE93 4198	—	—	32
BEU1 module reset register	BBRSTR_1	R/W	H'FE93 419C	—	—	32
BEU1 register-plane forcible setting register	BRCHR_1	R/W	H'FE93 41A0	—	—	32
BEU1 Color Lookup Table	CLUT_1	R/W	H'FE93 7000 to H'FE93 73FF	—	—	32

Table 51.3 Register Addresses of LCDC

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
LCDC palette data registers 00 to FF	LDPR00 to LDPRFF	R/W	H'FE94 0000 to H'FE94 03FC	—	—	32
Main LCD dot clock pattern register 1	MLDDCKPAT1R	R/W	H'FE94 0400	—	—	32
Main LCD dot clock pattern register 2	MLDDCKPAT2R	R/W	H'FE94 0404	—	—	32
LCDC dot clock register	LDDCKR	R/W	H'FE94 0410	—	—	32
Dot clock stop register	LDDCKSTPR	R/W	H'FE94 0414	—	—	32
Main LCD module type register 1	MLDMT1R	R/W	H'FE94 0418	H'FE94 1418	H'FE94 2418	32
Main LCD module type register 2	MLDMT2R	R/W	H'FE94 041C	H'FE94 141C	H'FE94 241C	32
Main LCD module type register 3	MLDMT3R	R/W	H'FE94 0420	H'FE94 1420	H'FE94 2420	32
Main LCD data format register	MLDDFR	R/W	H'FE94 0424	H'FE94 1424	H'FE94 2424	32
Main LCD scan mode register 1	MLDSM1R	R/W	H'FE94 0428	H'FE94 1428	H'FE94 2428	32
Main LCD scan mode register 2	MLDSM2R	R/W	H'FE94 042C	—	—	32
Main LCD display data read start address register 1	MLDSA1R	R/W	H'FE94 0430	H'FE94 1430	H'FE94 2430	32
Main LCD display data read start address register 2	MLDSA2R	R/W	H'FE94 0434	H'FE94 1434	H'FE94 2434	32
Main LCD display data storing memory line size register	MLDMLSR	R/W	H'FE94 0438	H'FE94 1438	H'FE94 2438	32
Main LCD horizontal character number register	MLDHCNR	R/W	H'FE94 0448	H'FE94 1448	H'FE94 2448	32
Main LCD horizontal sync signal register	MLDHSYNR	R/W	H'FE94 044C	H'FE94 144C	H'FE94 244C	32
Main LCD vertical line number register	MLDVLNR	R/W	H'FE94 0450	H'FE94 1450	H'FE94 2450	32
Main LCD vertical sync signal register	MLDVSYNR	R/W	H'FE94 0454	H'FE94 1454	H'FE94 2454	32
Main LCD horizontal partial display register	MLDHPDR	R/W	H'FE94 0458	H'FE94 1458	H'FE94 2458	32
Main LCD vertical partial display register	MLDVPDR	R/W	H'FE94 045C	H'FE94 145C	H'FE94 245C	32
Main LCD power management register	MLDPMR	R/W	H'FE94 0460	—	—	32
LCDC palette control register	LDPALCR	R/W	H'FE94 0464	—	—	32
LCDC interrupt register	LDINTR	R/W	H'FE94 0468	—	—	32
LCDC status register	LDSR	R	H'FE94 046C	—	—	32
LCDC control register 1	LDCNT1R	R/W	H'FE94 0470	—	—	32
LCDC control register 2	LDCNT2R	R/W	H'FE94 0474	—	—	32

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
LCDC register side change control register	LDRCNTR	R/W	H'FE94 0478	—	—	32
LCDC input image data swap register	LDDDSR	R/W	H'FE94 047C	—	—	32
LCDC register side forcible select register	LDRCR	R/W	H'FE94 0484	—	—	32
LCDC driver write data register 0	LDDWD0R	R/W	H'FE94 0800	—	—	32
LCDC driver write data register 1	LDDWD1R	R/W	H'FE94 0804	—	—	32
LCDC driver write data register 2	LDDWD2R	R/W	H'FE94 0808	—	—	32
LCDC driver write data register 3	LDDWD3R	R/W	H'FE94 080C	—	—	32
LCDC driver write data register 4	LDDWD4R	R/W	H'FE94 0810	—	—	32
LCDC driver write data register 5	LDDWD5R	R/W	H'FE94 0814	—	—	32
LCDC driver write data register 6	LDDWD6R	R/W	H'FE94 0818	—	—	32
LCDC driver write data register 7	LDDWD7R	R/W	H'FE94 081C	—	—	32
LCDC driver write data register 8	LDDWD8R	R/W	H'FE94 0820	—	—	32
LCDC driver write data register 9	LDDWD9R	R/W	H'FE94 0824	—	—	32
LCDC driver write data register A	LDDWDAR	R/W	H'FE94 0828	—	—	32
LCDC driver write data register B	LDDWDBR	R/W	H'FE94 082C	—	—	32
LCDC driver write data register C	LDDWDCR	R/W	H'FE94 0830	—	—	32
LCDC driver write data register D	LDDWDDR	R/W	H'FE94 0834	—	—	32
LCDC driver write data register E	LDDWDER	R/W	H'FE94 0838	—	—	32
LCDC driver write data register F	LDDWDFR	R/W	H'FE94 083C	—	—	32
LCDC driver read data register	LDDRDR	R/W	H'FE94 0840	—	—	32
LCDC driver write access register	LDDWAR	R/W	H'FE94 0900	—	—	32
LCDC driver read access register	LDDRAR	R/W	H'FE94 0904	—	—	32

Table 51.4 Register Addresses of VOU

Register Name	Abbreviation	R/W	Address (Plane A)	Address (Plane B)	Address	Access Size
VOU execution register	VOUER	R/W	H'FE96 0000	—	—	32
VOU control register	VOUCR	R/W	H'FE96 0004	H'FE96 1004	H'FE96 2004	32
VOU status register	VOUSTR	R/W	H'FE96 0008	—	—	32
VOU video control register	VOUVCR	R/W	H'FE96 000C	H'FE96 100C	H'FE96 200C	32
VOU source image size register	VOUISR	R/W	H'FE96 0010	H'FE96 1010	H'FE96 2010	32
VOU background color register	VOUBCR	R/W	H'FE96 0014	H'FE96 1014	H'FE96 2014	32
VOU display position register	VOUDPR	R/W	H'FE96 0018	H'FE96 1018	H'FE96 2018	32
VOU display size register	VOUDSR	R/W	H'FE96 001C	H'FE96 101C	H'FE96 201C	32
VOU valid pixel start position register	VOUVPR	R/W	H'FE96 0020	H'FE96 1020	H'FE96 2020	32
VOU interrupt register	VOUIR	R/W	H'FE96 0024	—	—	32
VOU reset register	VOUSRR	R/W	H'FE96 0028	—	—	32
VOU mode setting register	VOUMSR	R/W	H'FE96 002C	H'FE96 102C	H'FE96 202C	32
VOU horizontal sync interval register	VOUHIR	R/W	H'FE96 0030	H'FE96 1030	H'FE96 2030	32
VOU source image data format register	VOUDFR	R/W	H'FE96 0034	H'FE96 1034	H'FE96 2034	32
VOU source image data destination start address register 1	VOUAD1R	R/W	H'FE96 0038	H'FE96 1038	H'FE96 2038	32
VOU source image data destination start address register 2	VOUAD2R	R/W	H'FE96 003C	H'FE96 103C	H'FE96 203C	32
VOU source image data address increment register	VOUAIR	R/W	H'FE96 0040	H'FE96 1040	H'FE96 2040	32
VOU source image data swap register	VOUSWR	R/W	H'FE96 0044	—	—	32
VOU register side switch register	VOURCR	R/W	H'FE96 0048	—	—	32
VOU register side forcedly specify register	VOURPR	R/W	H'FE96 0050	—	—	32

51.2 Register States in Each Operating Mode

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
TRA	Undefined	Undefined	Retained	—	Undefined	Undefined	Retained	Exception Handling
EXPEVT	H'0000 0000	H'0000 0020	Retained	—	H'0000 0000	H'0000 0000	Retained	
INTEVT	Undefined	Undefined	Retained	—	Undefined	Undefined	Retained	
EXPMASK	H'0000 001F	H'0000 001F	Retained	—	H'0000 001F	H'0000 001F	Retained	
PTEH	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	MMU
PTEL	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
TTB	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
TEA	Undefined	Retained	Retained	Retained	Undefined	Undefined	Retained	
MMUCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained	
PTEA	H'0000 xxx0	H'0000 xxx0	Retained	Retained	H'0000 xxx0	H'0000 xxx0	Retained	
PASCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained	
IRMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained	
CCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained	Cache
QACR0	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Undefined	
QACR1	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Undefined	
RAMCR	H'0000 0000	H'0000 0000	Retained	Retained	H'0000 0000	H'0000 0000	Retained	
PRPRICR 0 to 10	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	Super-Hyway Packet Router
PRLCKCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICR0	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	INTC
ICR1	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	
INTPRI00	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	
INTREQ00	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	
INTMSK00	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	
INTMSKCLR00	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
NMIFCR	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	
USERIMASK	Initialized	Initialized	Retained	Retained	Retained	Initialized	Retained	
IPRA	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRB	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRC	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
IPRD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	INTC
IPRE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRF	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRH	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRI	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRJ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR6	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR7	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR8	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR9	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR10	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR11	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMR12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR6	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR7	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR8	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR9	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
IMCR10	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	INTC
IMCR11	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMCR12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MMSELR	Initialized	Retained	Retained	—	Initialized	Initialized	Initialized	BSC
CMNCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
CS0BCR	Initialized	Retained	Retained	—	Retained *[1]	Initialized	Retained	
CS4BCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS5ABCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS5BBCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS6ABCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS6BBCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS0WCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
CS4WCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS5AWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS5BWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS6AWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
CS6BWCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
RBWTCNT	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
DBKIND	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	DBSC
DBSTATE	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBEN	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBCMDCNT	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBCKECNT	Initialized	Initialized	Retained	—	Retained	Initialized	Retained	
DBCONF	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBTR0	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBTR1	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBTR2	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBTR3	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBRFPDN 0	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBRFPDN 1	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBRFPDN 2	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DBRFSTS	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
DBMRCNT	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	DBSC
DBPDCNT0	Initialized	Initialized	Retained	—	Initialized	Initialized	Retained	
DMA0_SAR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	DMAC
DMA0_DAR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA0_TCR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA0_CHCR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA0_DMAOR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA0_SARB_0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_DARB_0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA0_TCRB_0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA0_DMARS_0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_SAR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_DAR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_TCR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_CHCR_0 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_DMAOR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_SARB_0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_DARB_0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_TCRB_0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DMA1_DMARS_0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
FRQCRA	Initialized	Retained	Retained	—	Retained	Initialized	Retained	CPG
FRQCRB	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
VCLKCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
FCLKACR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
FCLKBCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
IRDACLKCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
SPUCLKCR	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	
PLLCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
FLLFRQ	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
LSTATS	Initialized	Retained	Retained	—	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
STBCR	Initialized	Retained	Retained	—	Retained	Retained	Retained	Reset and Power-Down Modes
MSTPCR0	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
MSTPCR1	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
MSTPCR2	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
BAR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
RWTCNT	Initialized	Retained	Retained	Retained	Retained	Initialized	Retained	RWDT
RWTCSR	Initialized	Retained	Retained	Retained	Initialized*[1]	Initialized*[1]	Retained	
TSTR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	TMU
TCOR0_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCNT0_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCR0_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCOR0_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCNT0_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCR0_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCOR0_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCNT0_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCR0_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSTR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCOR1_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCNT1_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCR1_0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCOR1_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCNT1_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCR1_1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCOR1_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCNT1_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TCR1_2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TSTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	TPU
TPU_TCR0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TMDR0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TIOR0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TIER0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
TPU_TSR0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	TPU
TPU_TCNT0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TGR0A to 3A	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TGR0B to 3B	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TGR0C to 3C	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPU_TGR0D to 3D	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CMSTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	CMT
CMCSR	Initialized	Initialized*[1]	Retained	Retained	Initialized*[1]	Initialized*[1]	Retained	
CMCNT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CMCOR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_CMD_SET	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	MMCIF
CE_ARG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_ARG_CMD12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_CMD_CTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_BLOCK_SET	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_CLK_CTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_BUF_ACC	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_RESP3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_RESP2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_RESP1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_RESP0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_RESP_CMD12	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_DATA	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_INT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_INT_MASK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_HOST_STS1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_HOST_STS2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CE_VERSION	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIOF0_SITMDR1 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	MSIOF
MSIOF0_SIRMDR1 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIOF0_SITSCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIOF0_SIRSCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
MSIOF0_SICTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	MSIOF
MSIOF0_SIFCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIOF0_SISTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIOF0_SIIER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIOF0_SITDR1, 2	Initialized	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
MSIOF0_SITFDR	Initialized	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
MSIOF0_SIRDR1, 2	Initialized	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
MSIOF0_SIRFDR	Undefined	Undefined	Retained	Initialized	Initialized	Initialized	Retained	
MSIO1_SITMDR1 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SIRMDR1 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SITSCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SIRSCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SICTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SIFCTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SISTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SIIER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MSIO1_SITDR1, 2	Initialized	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
MSIO1_SITFDR	Initialized	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
MSIO1_SIRDR1, 2	Initialized	Initialized	Retained	Initialized	Initialized	Initialized	Retained	
MSIO1_SIRFDR	Undefined	Undefined	Retained	Initialized	Initialized	Initialized	Retained	
SCSMR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	SCIF
SCBRR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCSCR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCFTDR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCFSR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCFRDR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCFCR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCFDR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCLSR0 to 2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCASMR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	SCIFA
SCABRR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCASCR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
SCATDSR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	SCIFA
SCAFER3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCASSR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCAFCR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCAFDR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCAFTDR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SCAFRDR3 to 5	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
R64CNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	RTC
RSECCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RMINCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RHRCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RWKCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RDAYCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RMONCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RYRCNT	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RSECAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RMINAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RHRAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RWKAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RDAYAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RMONAR	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RCR1	Initialized/ Undefined	Initialized	Retained	Retained	Retained	Retained	Retained	
RCR2	Initialized	Initialized/ Undefined	Retained	Retained	Retained	Retained	Retained	
RYRAR	Undefined	Retained	Retained	Retained	Retained	Retained	Retained	
RCR3	Initialized	Initialized	Retained	Retained	Retained	Retained	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
IRIF_RINTCLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	IrDA
IRIF_TINTCLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_SIR0 to 3	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_SIR_FRM	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_SIR_EOF	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_SIR_FLG	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_UART_STS2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_UART0 to 7	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IRIF_CRC0 to 4	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
KYCR1	Initialized	Initialized	Retained	Retained	Initialized*[1]	Initialized	Retained	KEYSC
KYCR2	Initialized*[1]	Initialized*[1]	Retained	Retained	Initialized*[1]	Initialized*[1]	Retained	
KYINDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
KYOUTDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SYSCFG_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	USB
BUSWAIT_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SYSSTS_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DVSTCTR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TESTMODE_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CFIFO_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
D0FIFO_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
D1FIFO_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CFIFOSEL_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CFIFOCTR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
D0FIFOSEL_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
D0FIFOCTR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
D1FIFOSEL_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
D1FIFOCTR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
INTENB0_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
INTENB1_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BRDYENB_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
NRDYENB_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEMPENB_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
SOFCFG_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	USB
INTSTS0_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
INTSTS1_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BRDYSTS_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
NRDYSTS_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEMPSTS_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
FRMNUM_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
UFRMNUM_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
USBADDR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
USBREQ_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
USBVAL_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
USBINDX_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
USBLENG_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DCPCFG_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DCPMAXP_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DCPCTR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPESEL_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPECFG_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPEBUF_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPEMAXP_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPEPERI_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPE1CTR_0/1 to PIPE9CTR_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPE1TRE_0/1 to PIPE5TRE_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIPE1TRN_0/1 to PIPE5TRN_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DEVADD0_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
DEVADD1_0/1 to A_0/1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
UPONCR0/1	Initialized	Retained	Retained	Retained	Retained	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
ICDR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	I^2C
ICCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICSR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICIC0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICCL0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICCH0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICDR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICSR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICIC1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICCL1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ICCH1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CAPSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	CEU
CAPCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CAMCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CMCYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CAMOR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CAPWR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CAIFR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CRCNTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CRCMPR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CFLCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CFSZR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDWDR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDACR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDBYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDBCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CBDSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CFWCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CLFCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDOCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
CDDCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	CEU
CDDAR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CEIER_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CETCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CSTSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CSRTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDSSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDAYR2_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDACR2_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDBYR2_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDBCR2_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VESTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	VEU
VESWR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VESSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VSAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VSACR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VBSSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VEDWR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VDAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VDACR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VTRCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VRFCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VRFSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VENHR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VFMCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VVTCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VHTCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VAPCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VECCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VFLCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VAFXR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VSWPR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
VEIER_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	VEU
VEVTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VSTAR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VBSRR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VRPBR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BESTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	BEU
BSMWR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BSSZR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BSAYR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BSACR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BSAAR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BSIFR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BTPSR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMSMWR1_n to 4_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMSSZR1_n to 4_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMSAYR1_n to 4_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMSACR1_n to 4_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMSIFR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BBLCR0_n, 1_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BPROCR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMWCR0_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BLOCR1_n to 3_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMLOCR1_n to 4_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BMPCCR1_n, 2_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BPCCR0_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BPCCR11_n, 12_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BPCCR21_n, 22_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BPCCR31_n, 32_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDMWR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDAYR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BDACR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BAFXR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
BSWPR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	BEU
BEIER_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEVTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BRCNTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BSTAR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BBRSTR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BRCHR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEIMCR00_n, 01_n, 02_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEIMCR10_n, 11_n, 12_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEIMCR20_n, 21_n, 22_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEICBR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEICOFFR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEBMCR00_n, 01_n, 02_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEBMCR10_n, 11_n, 12_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEBMCR20_n, 21_n, 22_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEBCBR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
BEBCOFFR_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CLUT_n	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCMOD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCCMD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCSTS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCQTN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCHTN	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCDRIU	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCDRID	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCVSZU	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCVSZD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
JCHSZU	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	BEU
JCHSZD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCDTCU	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCDTCM	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCDTCD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	JPU
JINTE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JINTS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCDERR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCRST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFECNT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESYA1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESCA1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESYA2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESCA2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESMW	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESVSZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFESHSZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFEDA1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFEDA2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFEDRSZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDCNT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDSA1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDSA2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDRSZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDMW	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDVSZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDHSZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDYA1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDCA1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDYA2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JIFDDCA2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
JCQTBL0	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
JCQTBL1	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	JPU
JCQTBL2	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCQTBL3	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBD0	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBD0	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBA0	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBA0	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBD1	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBD1	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBA1	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
JCHTBA1	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
LDPR00 to LDPRFF	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	LCDC
MLDDCKPAT1R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDDCKPAT2R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDCKR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDCKSTPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDMT1R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDMT2R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDMT3R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDDFR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDSM1R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDSM2R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDSA1R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDSA2R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDMLSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDWBFR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDWBCNTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDWBAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDHCNR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDHSYNR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDVLNR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDVSYNR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
MLDHPDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	LCDC
MLDVPDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MLDPMR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDPALCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDINTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDCNT1R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDCNT2R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDRCNTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDDSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDRCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDWD0R to LDDWDFR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDRDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDWAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LDDRAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	VOU
VOUCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUSTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUVCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUISR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUBCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUDPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUDSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUVPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUIR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUSRR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUMSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUHIR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUDFR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUAD1R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOUAD2R	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
VOUAIR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	VOU
VOUSWR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOURCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
VOURPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MEVCR0	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	MERAM
MEVCR1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
METEIE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MEILIE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MEACTST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
METEST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MEILWST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MEILRST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ME00CNTRL to ME31CNTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ME00BSIZE to ME31BSIZE	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
ME00MCNF to ME31MCNF	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
ME00SSARA to ME31SSARA	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
ME00SSARB to ME31SSARB	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
ME00SBSIZE to ME31SBSIZE	Undefined	Undefined	Retained	Retained	Undefined	Undefined	Retained	
CHSTCLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	2DDMAC
CHPRI	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0CTRL to CH7CTRL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0SWAP to CH7SWAP	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0SAR to CH7SAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0DAR to CH7DAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0DPXL to CH7DPXL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
CH0SFMT to CH7SFMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	2DDMAC
CH0DFMT to CH7DFMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0SARE to CH7SARE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0DARE to CH7DARE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CH0DPXLE to CH7DPXLE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSCTLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	TSIF
TSPIDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSCMDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSSTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSTSDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSBUFCLRR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSINTER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSPSCALER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSPSCALERR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSPCRADCMDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSPCRADCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSTRPCRADCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSDPCRADCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSFRC	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_DO_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	FSI
A_DOFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_DOFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_DI_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_DIFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_DIFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_CKG1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_CKG2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
A_DIDT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
A_DODT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	FSI
A_MUTE_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DO_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DOFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DOFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DI_FMT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DIFF_CTL	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DIFF_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_CKG1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_CKG2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DIDT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_DODT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
B_MUTE_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
INT_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IEMSK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IMSK	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CLK_RST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
SOFT_RST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
FIFO_SZ	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_CONTROL1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	ATAPI
ATAPI_STATUS	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_INT_ENABLE	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_DTB_ADR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_DMA_ START_ADR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_DMA_ TRANS_CNT	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_CONTROL2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_SIG_ST	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_BYTE_SWAP	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_PIO_TIMING1	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ATAPI_PIO_TIMING2	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
ATAPI_MULTI_TIMING	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	ATAPI
ATAPI_ULTRA_TIMING	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
EDMR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	EtherMAC
EDTRR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
EDRRR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TDLAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RDLAR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
EESR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
EESIPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TRSCER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RMFCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TFTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
FDR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RMCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TFUCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RFOCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
FCFTR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TRIMD	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ECMR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RFLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ECSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ECSIPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RDMLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPGR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
APR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RFCF	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPAUSER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPAUSECR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ECMR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
RFLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	EtherMAC
ECSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
ECSIPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PIR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PSR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RDMLR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
IPGR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
APR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MPR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RFCF	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPAUSER	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TPAUSECR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MAHR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MALR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TROCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CDCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
LCCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CNDCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
CEFCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
FRECR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TSFRCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
TLFRCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
RFCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
MAFCR	Initialized	Initialized	Retained	Retained	Initialized	Initialized	Retained	
PACR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	PFC
PBCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PCCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PDCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PECR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PFCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PGCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PHCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
PJCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	PFC
PKCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PLCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PMCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PNCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PQCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PRCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PSCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PTCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PUCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PVCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PWCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PXCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PYCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PZCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PSELA	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PSELB	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PSELC	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PSELD	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
HIZCRA	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
HIZCRB	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
HIZCRC	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
HIZCRD	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
MSELCRA	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
MSELCRB	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PULCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
DRVCRA	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
DRVCRB	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
SBSCR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PADR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PBDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PCDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
PDDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	PFC
PEDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PFDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PGDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PHDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PJDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PKDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PLDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PMDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PNDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PQDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PRDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PSDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PTDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PUDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PVDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PWDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PXDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PYDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	
PZDR	Initialized	Retained	Retained	—	Retained	Initialized	Retained	

Register Abbreviation	Power-on Reset	Manual Reset	Software Standby	Module Standby	R-Standby	U-Standby	Sleep	Module Name
CBR0	H'2000 0000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	UBC
CRR0	H'0000 2000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CAR0	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CAMR0	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CBR1	H'2000 0000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CRR1	H'0000 2000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CAR1	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CAMR1	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CDR1	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CDMR1	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CETR1	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CCMFR	H'0000 0000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
CBCR	H'0000 0000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
SDIR	H'0EFF	Retained	Retained	Retained	Retained*2	Retained*2	Retained	H-UDI
SDDR/SDDRH	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
SDDRL	Undefined	Retained	Retained	Retained	Retained*2	Retained*2	Retained	
SDINT	H'0000	Retained	Retained	Retained	Retained*2	Retained*2	Retained	

Section 52 Electrical Characteristics

52.1 Absolute Maximum Ratings

Table 52.1 shows the absolute maximum ratings.

Table 52.1 Absolute Maximum Ratings

Item	Symbol	Rating	Unit
Power supply voltage (I/O)	$V_{CC}Q$, $V_{CC}Q$ 1, $V_{CC}Q\_LCD$, $V_{CC}Q\_VIO$, $V_{CC}Q\_MMC$, $V_{CC}Q\_SR$, $V_{CC}Q\_SDC$	–0.3 to 4.6	V
Power supply voltage (DDR)	$V_{CC}Q\_DDR$	–0.3 to 2.6	V
Power supply voltage (Internal)	V_{DD}, V_{DD}_PLL, V_{DD}_FLL	–0.3 to 1.8	V
Input voltage (other than VBUS pin)	V_{in}	–0.3 to $V_{CC}Q$ + 0.3	V
Input voltage (VBUS pin)	V_{in}_VBUS	–0.3 to 5.5	V
Analog power supply voltage (USB I/O)	AV33, DV33	–0.3 to 4.6	V
Analog power supply voltage (USB internal)	AV12, DV12, UV12	–0.3 to 1.8	V
Storage temperature	T_{stg}	–55 to 125	°C

Caution: Operating the chip in excess of the absolute maximum ratings may result in permanent damage.

52.2 Recommended Operating Conditions

Table 52.2 lists the recommended operating conditions. The specification in this section assumes the use under the conditions of table 52.2 unless otherwise noted.

Table 52.2 Recommended Operating Conditions

Item		Symbol	Min.	Typ.	Max.	Unit	Product number
Operating temperature (operating ambient temperature Ta)		T_{opr}	–40	—	85	°C	R8A77240D500BG
			–20	—	70		R8A77240B500BB
Power supply voltage	Core power supply	V_{DD}	1.15	—	1.3	V	M1ϕ ≦ 125 MHz
			1.25	—	1.35	V	M1ϕ ≦ 166 MHz
	I/O power supply	$V_{CC}Q$	2.7	3.3	3.6	V	
		$V_{CC}Q1$	1.65/2.7	1.8/3.3	1.95/3.6	V	
		$V_{CC}Q\_LCD$	1.65/2.7	1.8/3.3	1.95/3.6	V	
		$V_{CC}Q\_VIO$	1.65/2.7	1.8/3.3	1.95/3.6	V	
		$V_{CC}Q\_MMC$	1.65/2.7	1.8/3.3	1.95/3.6	V	
		$V_{CC}Q\_SR$	1.65/2.7	1.8/3.3	1.95/3.6	V	
		$V_{CC}Q\_SDC$	2.7	3.3	3.6	V	
	Power supply for PLL	V_{DD}_PLL	1.15	—	1.3	V	M1ϕ ≦ 125 MHz
			1.25	—	1.35	V	M1ϕ ≦ 166 MHz
	Power supply for FLL	V_{DD}_FLL	1.15	—	1.3	V	M1ϕ ≦ 125 MHz
			1.25	—	1.35	V	M1ϕ ≦ 166 MHz
	Power supply for DDR	$V_{CC}Q\_DDR$	1.7	1.8	1.9	V	
	Analog power supply	AV_{CC}	3.0	3.3	3.6	V	

Item		Symbol	Min.	Typ.	Max.	Unit	Product number
Power supply voltage	USB analog 1.2-V power supply	AV12	1.15	—	1.35	V	
	USB digital 1.2-V power supply	UV12	1.15	—	1.35	V	
	USB digital 1.2-V power supply	DV12	1.15	—	1.35	V	
	USB analog 3.3-V power supply	AV33	3.0	3.3	3.6	V	
	USB digital 3.3-V power supply	DV33	3.0	3.3	3.6	V	
	DDR2 reference voltage (MVREF pin)*	Vref	0.49 × V_{cc}Q_ DDR	0.50 × V_{cc}Q_ DDR	0.51 × V_{cc}Q_ DDR	V	

Note: Be sure to supply power to all the power supply pins, and to connect the Vss pin to the system ground (0 V).

* Generate the Vref voltage through a voltage divider from the V_{cc}Q_DDR power supply.

52.3 Power-On and Power-Off Order

This section describes the specification of the order of powering on and off the master I/O power supply ($V_{CC}Q$), other I/O power supplies ($V_{CC}Q1$, $V_{CC}Q\_LCD$, $V_{CC}Q\_MMC$, $V_{CC}Q\_SDC$, $V_{CC}Q\_SR$, $V_{CC}Q\_VIO$, DV33, AV33, and $V_{CC}Q\_DDR$), and 1.2-V power supplies (V_{DD}, V_{DD}_PLL, V_{DD}_FLL, DV12, AV12, and UV12).

52.3.1 Power-on Order

Figure 52.1 provides an example of the power-on sequence. Follow the procedure below to turn on the power supplies.

(1) Turn on the $V_{CC}Q$ power supply first. At the same time, set RESETP to low level less than VIL.
(2) Wait until the voltage of $V_{CC}Q$ becomes the minimum value specified in table 52.2 Recommended Operating Conditions.
(3) Turn on the I/O power supplies other than $V_{CC}Q$. There is no fixed order of turning them on.
(4) Turn on the 1.2-V power supplies.
(5) After all the power supplies have been turned on, cancel RESETP.

- Set the voltage to be supplied to $V_{CC}Q$ so that the following condition is met.
  VccQ + 0.3 V $\geq$ Other I/O power supplies
- When any of the clock modes 3 to 7 is in use, RTC_CLK supply is required. Input RTC_CLK before turning on the other I/O power supplies, after turning on the $V_{CC}Q$ power supply.

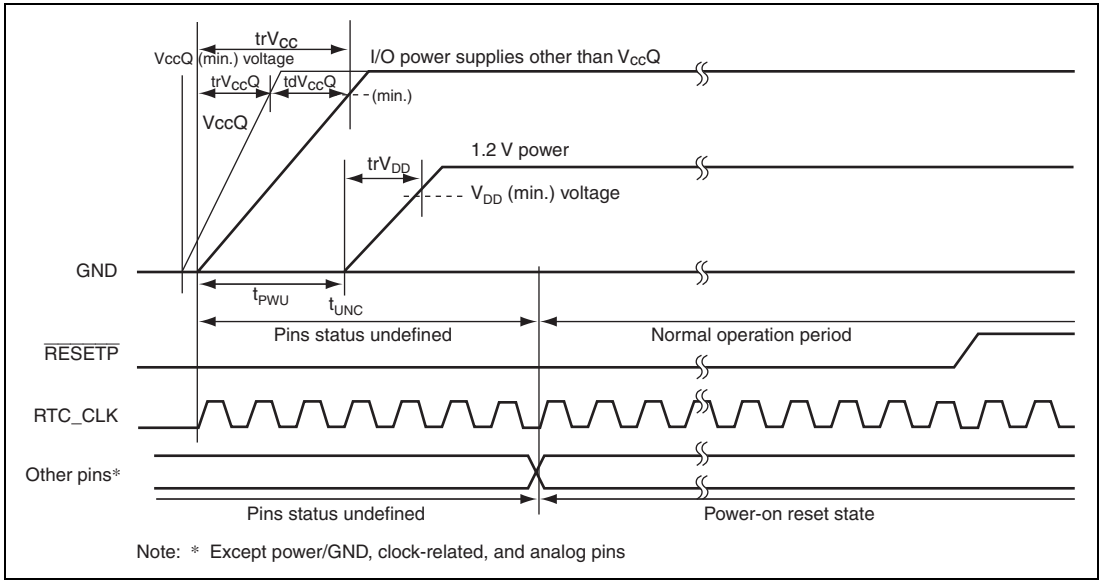


Figure 52.1 Power-on Sequence

Table 52.3 Recommended Timing in Power-On

Item	Symbol	Time	Unit
$V_{CC}Q$ power settling time	$trV_{CC}Q$	≤ 300	µs
Settling time difference between V_{CC} power supplies *1	$tdV_{CC}Q$	≥0	ms
Time difference between 3.3-V V_{CC} and 1.2-V V_{DD} at power-on	t_{PWU}	0 to 10	ms
V_{DD} power settling time	trV_{DD}	≤ 1	ms
Time over which the state is undefined	t_{UNC}	t_{PWU} + trV_{DD} + 3tRCLK *2	ms

Notes: Turn on the I/O power supplies as simultaneously as possible. Shown above are recommended values, which are not strictly required.

1. tdVccQ is a time difference between the time when the $V_{CC}Q$ power supply is stabilized and the time when the other I/O power supplies are stabilized.
2. t_{Rcyc} denotes one cycle period of RCLK clock generated by the CPG.

The pin states are settled in a power-on reset state after a t_{UNC} period during which they are uncertain. Design your system so that this period does not cause it to malfunction.

52.3.2 Power-off order

Figure 52.2 provides an example of the power-off sequence. Follow the procedure below to turn off the power supplies.

(1) Turn off the 1.2-V power supplies.
(2) Turn off the I/O power supplies other than $V_{CC}Q$. There is no fixed order of turning them off.
(3) Lastly, turn off $V_{CC}Q$.

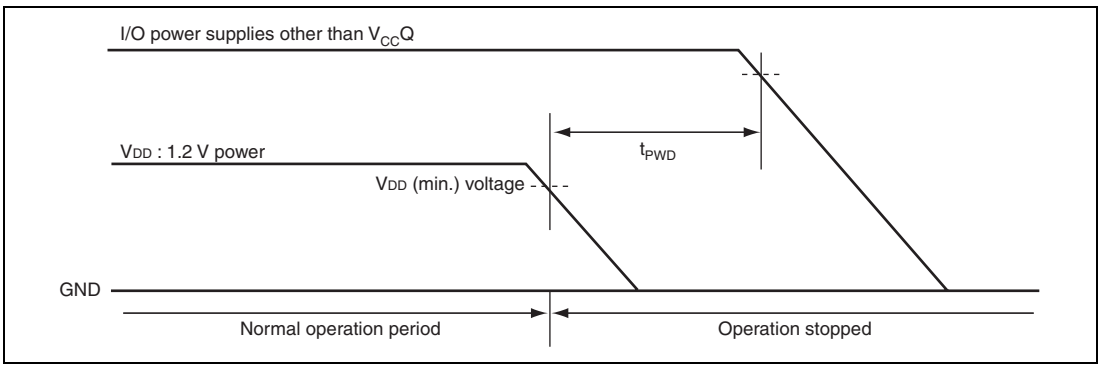


Figure 52.2 Power-off Sequence

Table 52.4 Recommended Timing in Power-Off

Item	Symbol	Maximum Value	Unit
Time difference between the power-off of 1.2-V V_{DD} and 3.3-V $V_{CC}Q$ levels	t_{PWD}	0 to 10	ms

Note: The values in the table above are recommended values, so they represent guidelines rather than strict requirements.

52.4 DC Characteristics

Tables 52.5, 52.6, and 52.7 list the DC characteristics.

Table 52.5 DC Voltage Characteristics

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input high level voltage	System control ($V_{cc}Q$)	V_{IH}	$V_{cc}Q \times 0.8$	—	$V_{cc}Q + 0.3$	V	
	External bus interface ($V_{cc}Q1$)		$V_{cc}Q1 \times 0.8$	—	$V_{cc}Q1 + 0.3$	V	
	SDRAM interface ($V_{cc}Q\_DDR$, other than MSLD pin)		$V_{cc}Q\_DDR \times 0.8$	—	$V_{cc}Q\_DDR + 0.3$	V	(For LPDDR)
			Vref + 0.2	—	—	V	(For DDR2, AC)
			Vref + 0.125	—	$V_{cc}Q\_DDR + 0.3$	V	(For DDR2, DC)
	SDRAM interface ($V_{cc}Q\_DDR$, MSLD pin)		$V_{cc}Q\_DDR \times 0.8$	—	$V_{cc}Q\_DDR + 0.3$	V	
	LCD interface ($V_{cc}Q\_LCD$)		$V_{cc}Q\_LCD \times 0.8$	—	$V_{cc}Q\_LCD + 0.3$	V	
	VIO ($V_{cc}Q\_VIO$)		$V_{cc}Q\_VIO \times 0.8$	—	$V_{cc}Q\_VIO + 0.3$	V	
	MMC interface ($V_{cc}Q\_MMC$)		$V_{cc}Q\_MMC \times 0.8$	—	$V_{cc}Q\_MMC + 0.3$	V	
	SDHI ($V_{cc}Q\_SDC$)		$V_{cc}Q\_SDC \times 0.8$	—	$V_{cc}Q\_SDC + 0.3$	V	
	Serial interface ($V_{cc}Q\_SR$)		$V_{cc}Q\_SR \times 0.8$	—	$V_{cc}Q\_SR + 0.3$	V	

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input low level voltage	System control ($V_{cc}Q$)	V_{IL}	-0.3	—	$V_{cc}Q \times 0.2$	V	
	External bus interface ($V_{cc}Q1$)		-0.3	—	$V_{cc}Q1 \times 0.2$	V	
	SDRAM interface ($V_{cc}Q\_DDR$, other than MSLD pin)		–0.3	—	$V_{cc}Q\_DDR \times 0.2$	V	(For LPDDR)
			—	—	Vref – 0.2	V	(For DDR2, AC)
			–0.3	—	Vref – 0.125	V	(For DDR2, DC)
	SDRAM interface ($V_{cc}Q\_DDR$, MSLD pin)		–0.3	—	$V_{cc}Q\_DDR \times 0.2$	V	
	LCD interface ($V_{cc}Q\_LCD$)		–0.3	—	$V_{cc}Q\_LCD \times 0.2$	V	
	VIO ($V_{cc}Q\_VIO$)		–0.3	—	$V_{cc}Q\_VIO \times 0.2$	V	
	MMC interface ($V_{cc}Q\_MMC$)		–0.3	—	$V_{cc}Q\_MMC \times 0.2$	V	
	SDHI ($V_{cc}Q\_SDC$)		–0.3	—	$V_{cc}Q\_SDC \times 0.2$	V	
	Serial interface ($V_{cc}Q\_SR$)		–0.3	—	$V_{cc}Q\_SR \times 0.2$	V	

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
AC differential input voltage	SDRAM interface ($V_{CC}Q$_DDR, differential input pin)	V_{ID}	0.5	—	$V_{CC}Q$_DDR + 0.6	V	(For DDR2, AC)
AC differential input cross point voltage	SDRAM interface ($V_{CC}Q$_DDR, differential input pin)	V_{IX}	$V_{CC}Q$_DDR × 0.5 – 0.175	—	$V_{CC}Q$_DDR × 0.5 + 0.175	V	(For DDR2, AC)
Output high level voltage	System control (other than $V_{CC}Q$, XTAL, and XTALUSB pins)	V_{OH}	$V_{CC}Q$ × 0.8	—	—	V	$V_{CC}Q$ = 2.7 to 3.6 V, I_{OH} = -2 mA
			$V_{CC}Q$ × 0.9	—	—	V	$V_{CC}Q$ = 2.7 to 3.6 V, I_{OH} = -200 µA
	External bus interface ($V_{CC}Q1$)		$V_{CC}Q1$ × 0.8	—	—	V	$V_{CC}Q1$ = 2.7 to 3.6 V, I_{OH} = -2 mA
			$V_{CC}Q1$ × 0.9	—	—	V	$V_{CC}Q1$ = 2.7 to 3.6 V, I_{OH} = -200 µA
			$V_{CC}Q1$ × 0.8	—	—	V	$V_{CC}Q1$ = 1.65 to 1.95 V, I_{OH} = -2 mA
			$V_{CC}Q1$ × 0.9	—	—	V	$V_{CC}Q1$ = 1.65 to 1.95 V, I_{OH} = -200 µA
	LCD interface ($V_{CC}Q$_LCD)		$V_{CC}Q$_LCD × 0.8	—	—	V	VCCQ_LCD = 2.7 to 3.6 V, I_{OH} = -2 mA
			$V_{CC}Q$_LCD × 0.9	—	—	V	$V_{CC}Q$_LCD = 2.7 to 3.6 V, I_{OH} = -200 µA
			$V_{CC}Q$_LCD × 0.8	—	—	V	$V_{CC}Q$_LCD = 1.65 to 1.95 V, I_{OH} = -2 mA
			$V_{CC}Q$_LCD × 0.9	—	—	V	$V_{CC}Q$_LCD = 1.65 to 1.95 V, I_{OH} = -200 µA
	VIO ($V_{CC}Q$_VIO)		$V_{CC}Q$_VIO × 0.8	—	—	V	$V_{CC}Q$_VIO = 2.7 to 3.6 V, I_{OH} = -2 mA
			$V_{CC}Q$_VIO × 0.9	—	—	V	VCCQ_VIO = 2.7 to 3.6 V, I_{OH} = -200 µA
			$V_{CC}Q$_VIO × 0.8	—	—	V	$V_{CC}Q$_VIO = 1.65 to 1.95 V, I_{OH} = -2 mA
			$V_{CC}Q$_VIO × 0.9	—	—	V	$V_{CC}Q$_VIO = 1.65 to 1.95 V, I_{OH} = -200 µA

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output high level voltage	MMC interface (V_{cc}Q_MMC)	V_{OH}	V_{cc}Q_MMC × 0.8	—	—	V	V_{cc}Q_MMC = 2.7 to 3.6 V, I_{OH} = -2 mA
			V_{cc}Q_MMC × 0.9	—	—	V	V_{cc}Q_MMC = 2.7 to 3.6 V, I_{OH} = -200 µA
			V_{cc}Q_MMC × 0.8	—	—	V	VCCQ_MMC = 1.65 to 1.95 V, I_{OH} = -2 mA
			V_{cc}Q_MMC × 0.9	—	—	V	V_{cc}Q_MMC = 1.65 to 1.95 V, I_{OH} = -200 µA
	SDHI (V_{cc}Q_SDC)		V_{cc}Q_SDC × 0.8	—	—	V	V_{cc}Q_MMC = 2.7 to 3.6 V, I_{OH} = -2 mA
			V_{cc}Q_SDC × 0.9	—	—	V	V_{cc}Q_MMC = 2.7 to 3.6 V, I_{OH} = -200 µA
	Serial interface (V_{cc}Q_SR)		V_{cc}Q_SR × 0.8	—	—	V	V_{cc}Q_SR = 2.7 to 3.6 V, I_{OH} = -2 mA
			V_{cc}Q_SR × 0.9	—	—	V	V_{cc}Q_SR = 2.7 to 3.6 V, I_{OH} = -200 µA
			V_{cc}Q_SR × 0.8	—	—	V	V_{cc}Q_SR = 1.65 to 1.95 V, I_{OH} = -2 mA
			V_{cc}Q_SR × 0.9	—	—	V	V_{cc}Q_SR = 1.65 to 1.95 V, I_{OH} = -200 µA

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output low level voltage	System control (other than $V_{CC}Q$, XTAL, and XTALUSB pins)	V_{OL}	—	—	$V_{CC}Q \times 0.2$	V	$V_{CC}Q$ = 2.7 to 3.6 V, I_{OL} = 2 mA
	External bus interface ($V_{CC}Q1$)		—	—	$V_{CC}Q1 \times 0.2$	V	$V_{CC}Q1$ = 2.7 to 3.6 V, I_{OL} = 2 mA
			—	—	$V_{CC}Q1 \times 0.2$	V	$V_{CC}Q1$ = 1.65 to 1.95 V, I_{OL} = 2 mA
	LCD interface ($V_{CC}Q\_LCD$)		—	—	$V_{CC}Q\_LCD \times 0.2$	V	$V_{CC}Q\_LCD$ = 2.7 to 3.6 V, I_{OL} = 2 mA
			—	—	$V_{CC}Q\_LCD \times 0.2$	V	$V_{CC}Q\_LCD$ = 1.65 to 1.95 V, I_{OL} = 2 mA
	VIO (other than $V_{CC}Q\_VIO$, SDA1/SCL1)		—	—	$V_{CC}Q\_VIO \times 0.2$	V	$V_{CC}Q\_VIO$ = 2.7 to 3.6 V, I_{OL} = 2 mA
			—	—	$V_{CC}Q\_VIO \times 0.2$	V	$V_{CC}Q\_VIO$ = 1.65 to 1.95 V, I_{OL} = 2 mA
	MMC interface ($V_{CC}Q\_MMC$)		—	—	$V_{CC}Q\_MMC \times 0.2$	V	$V_{CC}Q\_MMC$ = 2.7 to 3.6 V, I_{OL} = 2 mA
			—	—	$V_{CC}Q\_MMC \times 0.2$	V	$V_{CC}Q\_MMC$ = 1.65 to 1.95 V, I_{OL} = 2 mA
	SDHI ($V_{CC}Q\_SDC$)		—	—	$V_{CC}Q\_SDC \times 0.2$	V	$V_{CC}Q\_MMC$ = 2.7 to 3.6 V, I_{OL} = 2 mA
	Serial interface (other than $V_{CC}Q\_SR$, SDA0/SCL0)		—	—	$V_{CC}Q\_SR \times 0.2$	V	$V_{CC}Q\_SR$ = 2.7 to 3.6 V, I_{OL} = 2 mA
			—	—	$V_{CC}Q\_SR \times 0.2$	V	$V_{CC}Q\_SR$ = 1.65 to 1.95 V, I_{OL} = 2 mA
	SDA0, SCL0 Pin		—	—	0.4	V	$V_{CC}Q\_SR$ = 2.7 to 3.6 V
			—	—	$V_{CC}Q\_SR \times 0.15$	V	$V_{CC}Q\_SR$ = 1.65 to 1.95 V
	SDA1, SCL1 Pin		—	—	0.4	V	$V_{CC}Q\_VIO$ = 2.7 to 3.6 V
			—	—	$V_{CC}Q\_VIO \times 0.15$	V	$V_{CC}Q\_VIO$ = 1.65 to 1.95 V
Permissible output low current	SDA0, SCL0 Pin	I_{OL}	—	—	10	mA	$V_{CC}Q\_SR$ = 2.7 to 3.6 V
			—	—	5	mA	$V_{CC}Q\_SR$ = 1.65 to 1.95 V
	SDA1, SCL1 Pin		—	—	10	mA	$V_{CC}Q\_VIO$ = 2.7 to 3.6 V
			—	—	5	mA	$V_{CC}Q\_VIO$ = 1.65 to 1.95 V

Table 52.6 DC Characteristics

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Current consumption	Normal operation	I_{DD}	—	380	520	mA	V_{DD} = 1.2V MCLK = 166.7MHz Iϕ = 500 MHz Bϕ = 83.4 MHz M1ϕ = 83.4 MHz
			—	425	580	mA	V_{DD} = 1.3 V MCLK = 166.7 MHz Iϕ = 500 MHz Bϕ = 83.4 MHz M1ϕ = 166.7 MHz
		I_{CC}	—	—	100	mA	V_{CC}Q = 3.3 V V_{CC}Q_DDR = 1.8 V Bϕ = 83.4 MHz MCLK = 166.7 MHz Data bus width for BSC: 16 bits
		I_{CC}_DDR (DDR2)	—	—	450		
		I_{CC}_DDR (LPDDR)	—	—	80		
	Sleep mode*	I_{DD} (V_{DD} = 1.2V)	—	100	200	mA	*: When external bus cycles other than the refresh cycle are not specified. *: All module stop: On V_{CC}Q = 3.3 V V_{CC}Q_DDR = 1.8 V Bϕ = 83.4 MHz MCLK = 166.7 MHz
		I_{DD} (V_{DD} = 1.3V)	—	110	220		
		I_{CC}	—	—	30		
		I_{CC}_DDR	—	—	50		
	Software standby mode	I_{stby}	—	3	20	mA	Ta = 25°C V_{CC}Q = 3.3 V V_{CC}Q_DDR = 1.8 V V_{DD} = 1.2 V
	R-standby mode	I_{rstby}	—	—	50	µA	Ta = 25°C V_{CC}Q = 3.3 V V_{CC}Q_DDR = 1.8 V V_{DD} = 1.2 V Input clock off
	U-standby mode	I_{ustby}	—	—	40	µA	
Input leak current	Input pins other than $\overline{\text{RESETP}}$ pins and DBSC-related pins	$\|I_{in}\|$	—	—	1	µA	V_{in} = 0.5 to V_{CC}Q – 0.5 V
	DBSC-related pins	$\|I_{inSB}\|$	—	—	3		

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Three-state leak current	I/O pins and output pins other than DBSC-related pins (off state)	$\|I_{STI}\|$	—	—	1	μA	V_{in} = 0.5 to $V_{CC}Q$ – 0.5 V
	DBSC-related pins	$\|I_{inSB}\|$	—	—	3		
Pull-up/ pull-down resistance	Port pins	P_{pull}	40	—	300	kΩ	$V_{CC}Q$ = 2.7 to 3.6 V
			90		800	kΩ	$V_{CC}Q$ = 1.65 to 1.95 V
Pin capacitance	DBSC pins	C_{SB}	—	—	10	pF	
	All pins	C	—	—	10	pF	

Notes: 1. Make sure to supply the electric power to all the power supply pins anytime and the V_{SS} pin to the system ground (0 V).

2. Current consumption values in the table are for V_{IH}min = $V_{CC}Q$ – 0.5 V and V_{IL}max = 0.5 V with all output pins unloaded.

3. I_{DD} is the total current flowing through the V_{DD}, V_{DD}-PLL, V_{DD}-DLL, DV12, AV12, and UV 12 pins. I_{CC} is the total current flowing through the $V_{CC}Q$, $V_{CC}Q1$, $V_{CC}Q\_LCD$, $V_{CC}Q\_VIO$, $V_{CC}Q\_MMC$, $V_{CC}Q\_SR$, $V_{CC}Q$ _SDC, DV33, and AV33 pins. I_{CC}_DDR is the current flowing through the $V_{CC}Q\_DDR$ pin. I_{STBY} is the total of I_{DD}, I_{CC}, and I_{CC}_DDR in standby mode. I_{rstby} is the total of I_{DD}, I_{CC}, and I_{CC}_DDR in R-standby mode. I_{USTBY} is the total of I_{DD}, I_{CC}, and I_{CC}_DDR in U-standby mode.

Table 52.7 Permissible Output Current Values

Item	Symbol	Min.	Typ.	Max.	Unit
Permissible output low current (per pin)	I_{OL}	—	—	2.0	mA
Permissible output low current (total)	ΣI_{OL}	—	—	40	mA
Permissible output high current (per pin)	$-I_{OH}$	—	—	2.0	mA
Permissible output high current (total)	$\Sigma (-I_{OH})$	—	—	40	mA
Permissible I^2C output low current (SCL, SDA)	I_{OL}	—	—	10	mA

Note: * To ensure chip reliability, do not exceed the output current values given in table 52.7.

52.5 AC Characteristics

The inputs of this LSI are synchronous as a rule. The setup and hold time of each input signal must be satisfied unless otherwise noted.

The specification of this LSI assumes that when 1.65 to 1.95-V voltage is supplied using an I/O power supply that allows to select from 1.8 V and 3.3 V, the drive capabilities of I/O buffers are maximized. For setting the drive capabilities of I/O buffers, see section 48, Pin Function Controller (PFC).

52.5.1 Operating Frequency

Table 52.8 Operating Frequency Range

Item		Symbol	Min.	Typ.	Max.	Unit	Remarks
Operating frequency	CPU, FPU, cache (Iϕ)	f	10	—	500	MHz	
	SuperHyway bus (Sϕ), DBSC clock (MCLK)		10	—	166.7		
	VPU clock (M1ϕ)		10	—	125		
			10	—	166.7		Vdd = 1.25 V to 1.35 V
	BSC bus (Bϕ)		10	—	83.4		
	SPU clock (Bϕ)		10	—	83.4		
	Peripheral module (Pϕ)		2.5	—	41.7		
	FSI clock A (FSICKA)		—	—	41.7		
	FSI clock B (FSICKB)		—	—	41.7		
	IrDA clock (IrDACK)		—	—	41.7		
	Video clock (VIO_CKO)		—	—	83.4		

52.5.2 Clock Timing

Table 52.9 Clock Timing

Item	Symbol	Min.	Max.	Unit	Figure
EXTAL clock input frequency	f_{EX}	15	50	MHz	52.3
EXTAL clock input cycle time	t_{EXcyc}	20	66.7	ns	
EXTAL clock input low pulse width	t_{EXL}	4.5	—	ns	
EXTAL clock input high pulse width	t_{EXH}	4.5	—	ns	
EXTAL clock input rise time	t_{EXr}	—	3	ns	
EXTAL clock input fall time	t_{EXf}	—	3	ns	
RTC_CLK clock input frequency	f_{RCLK}	32	33	kHz	
RTC_CLK clock input cycle time	$t_{RCLKcyc}$	30.3	31.3	µs	
RTC_CLK clock input low pulse width	t_{RCLKL}	10	—	µs	
RTC_CLK clock input high pulse width	t_{RCLKH}	10	—	µs	
RTC_CLK clock input rise time	t_{RCLKr}	—	200	ns	
RTC_CLK clock input fall time	t_{RCLKf}	—	200	ns	
CKO clock output frequency	f_{CKO}	5	83.4	MHz	52.4
CKO clock output cycle time	t_{CKOcyc}	12	200	ns	
CKO clock output low pulse width	t_{CKOL}	3	—	ns	
CKO clock output high pulse width	t_{CKOH}	3	—	ns	
CKO clock output rise time	t_{CKOr}	—	3	ns	
CKO clock output fall time	t_{CKOf}	—	3	ns	
RESETP input pulse width (clock mode 0, 1, 3 to 5, 7)	t_{RESPW}	4	—	t_{Rcyc}	52.5
RESETP input pulse width (clock mode 2, 6)	t_{RESPW}	10	—	ms	
RESETOUT assert time (clock modes 0 to 2, 4 to 7)	$t_{RESOUTM0}$	—	300	µs	
RESETOUT assert time (clock mode 3)	$t_{RESOUTM0}$	—	2.3	ms	
Software standby return time (clock modes 0 to 2, 4 to 7)	t_{SOSM0}	—	300	µs	52.6
Software standby return time (clock mode 3)	t_{SOSM0}	—	2.3	ms	
R-standby return time (clock mode 0 to 2, 4 to 7)	t_{ROSM0}	—	560	µs	52.7 to 52.8
R-standby return time (clock mode 3)	t_{ROSM0}	—	2.5	ms	

Note: t_{Rcyc} denotes one cycle period of RCLK clock generated by the CPG.

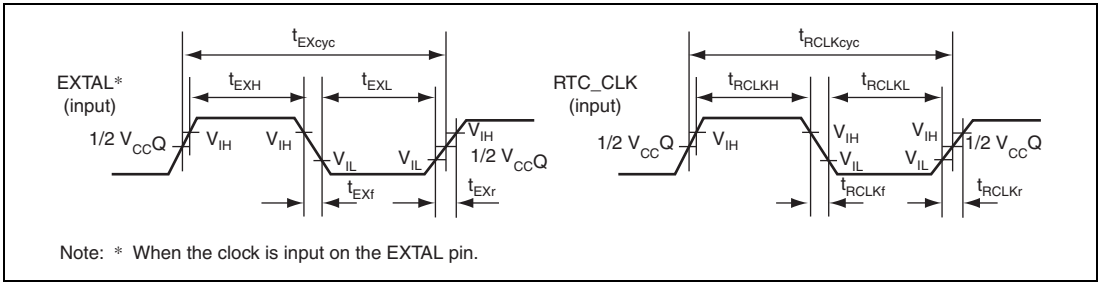


Figure 52.3 Clock Input Timing of EXTAL and RTC_CLK

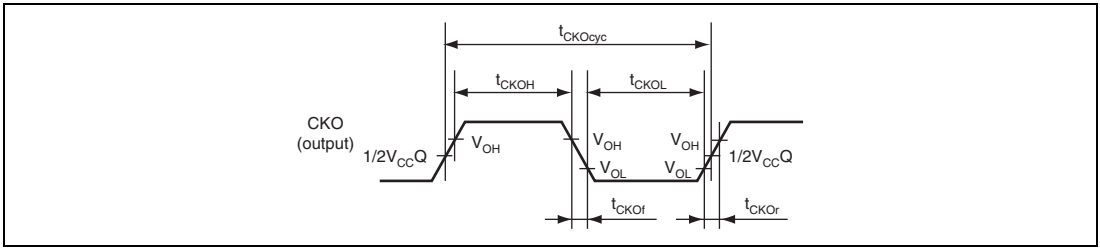


Figure 52.4 Clock Output Timing of CKO

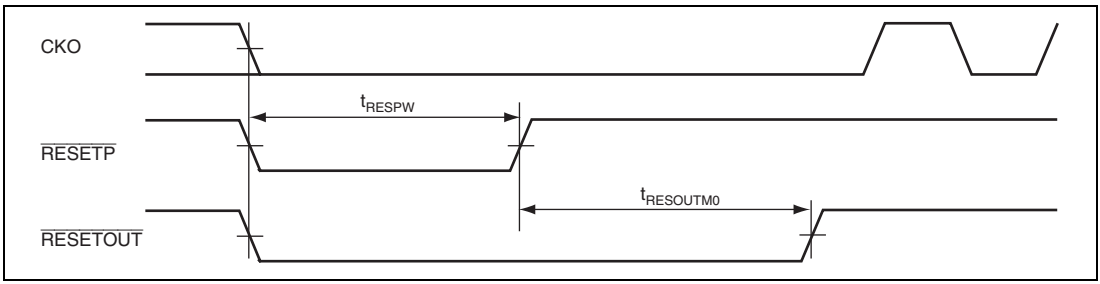


Figure 52.5 Power-On Oscillation Settling Time

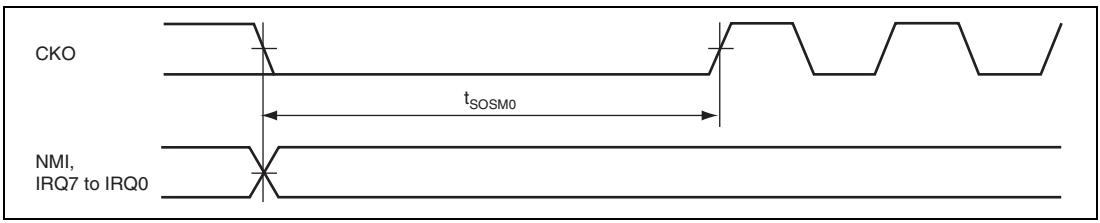


Figure 52.6 Oscillation Settling Time on Return from Software Standby by NMI or IRQ

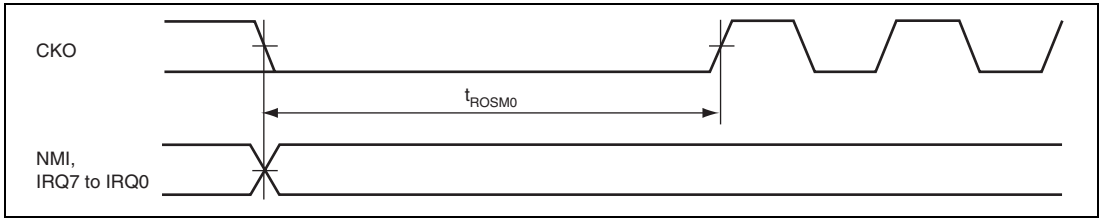


Figure 52.7 Oscillation Settling Time on Return to R-standby due to NMI and IRQ

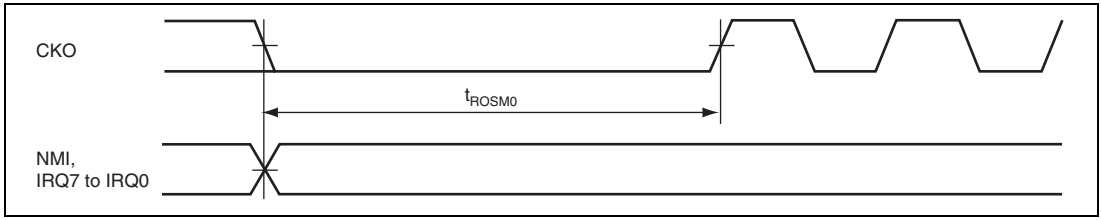


Figure 52.8 Oscillation Settling Time on Return to R-standby due to NMI and IRQ (Clock Mode 3)

52.5.3 Interrupt Signal Timing

Table 52.10 Interrupt Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
NMI setup time*	t_{NMIS}	12	—	ns	52.9
NMI hold time	t_{NMIH}	6	—	ns	
IRQ7 to IRQ0 setup time*	t_{IRQS}	12	—	ns	
IRQ7 to IRQ0 hold time	t_{IRQH}	6	—	ns	

Note: * NMI and IRQ7 to IRQ0 are asynchronous signals. When the setup time in the table is satisfied, a change is detected at the rising edge of the clock. When the setup time is not satisfied, a change may not be detected until the next rising edge of the clock.

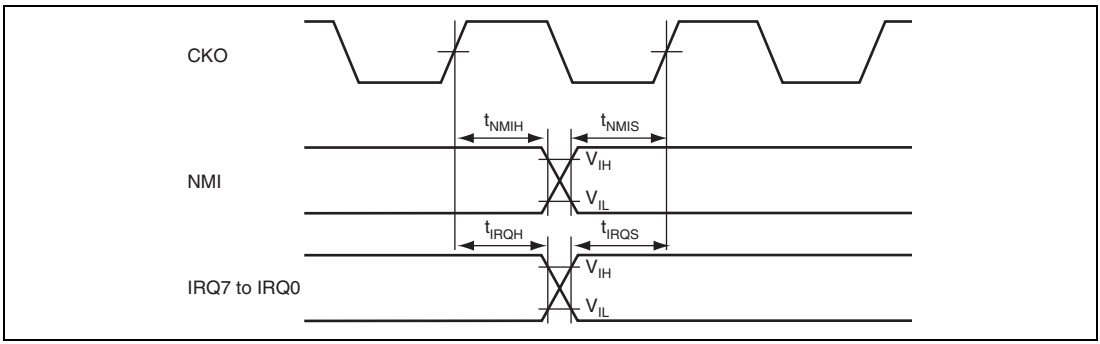


Figure 52.9 Interrupt Signal Input Timing

52.5.4 AC Bus Timing

Table 52.11 Bus Timing

Item	Symbol	Min.	Max.	Unit	Figure
Address delay time 1*	t_{AD1}	1	12	ns	52.10 to 52.23
Address delay time 2	t_{AD2}	$1/2 \times t_{cyc}$	$1/2 \times t_{cyc} + 12$	ns	52.19
Address setup time	t_{AS}	0	—	ns	52.10 to 52.19
Address hold time	t_{AH}	0	—	ns	52.14
CS delay time 1	t_{CSD1}	1	12	ns	52.10 to 52.19, 52.21 to 52.24
Read/write delay time 1	t_{RWD1}	1	12	ns	
Read/write delay time 2	t_{RWD2}	$1/2 \times t_{cyc}$	$1/2 \times t_{cyc} + 12$	ns	52.16
Read strobe delay time	t_{RSD}	$1/2 \times t_{cyc}$	$1/2 \times t_{cyc} + 12$	ns	52.10 to 52.19, 52.21, 52.22
Read data setup time 1	t_{RDS1}	$1/2t_{cyc} + 8$	—	ns	52.10 to 52.16, 52.21 to 52.24
Read data setup time 3	t_{RDS3}	$1/2t_{cyc} + 8$	—	ns	52.17 to 52.19
Read data hold time 1	t_{RDH1}	0	—	ns	52.10 to 52.16, 52.21 to 52.24
Read data hold time 3	t_{RDH3}	0	—	ns	52.17 to 52.19
Write enable delay time 1	t_{WED1}	$1/2 \times t_{cyc}$	$1/2 \times t_{cyc} + 12$	ns	52.10 to 52.18, 52.21, 52.22
Write enable delay time 2	t_{WED2}	0	12	ns	52.16, 52.17
Write data delay time 1	t_{WDD1}	—	12	ns	52.10 to 52.18, 52.21 to 52.24
Write data hold time 1	t_{WDH1}	1	—	ns	52.10 to 52.16, 52.21 to 52.24
Write data hold time 4	t_{WDH4}	0	—	ns	52.10
WAIT setup time 1	t_{WTS1}	$1/2 \times t_{cyc} + 7$	—	ns	52.10 to 52.19, 52.24
WAIT hold time 1	t_{WTH1}	$1/2 \times t_{cyc} + 6$	—	ns	
BS delay time 1	t_{BSD}	—	12	ns	52.10 to 52.16, 52.21 to 52,24

Item	Symbol	Min.	Max.	Unit	Figure
Write data hold time 5	t_{WDH5}	1	—	ns	52.21 to 52.24
ICIORD delay time	t_{ICRSD}	—	$1/2 \times t_{cyc} + 12$	ns	52.23, 52.24
ICIOWR delay time	t_{ICWSD}	—	$1/2 \times t_{cyc} + 12$	ns	52.23, 52.24
IOIS16 setup time	t_{IO16S}	$1/2 \times t_{cyc} + 6$	—	ns	52.24
IOIS16 hold time	t_{IO16H}	$1/2 \times t_{cyc} + 4$	—	ns	52.24
Address delay time 4	t_{AD4}	-4.0	3.5	ns	52.20
CS delay time 4	t_{CSD4}	-4.0	3.0	ns	52.20
BS delay time	t_{BSD}	-4.0	3.5	ns	52.20
Read/write delay time 4	t_{RWD4}	-4.0	3.5	ns	52.20
Read strobe delay time 4	t_{RSD}	-4.0	5.5	ns	52.20
Read data setup time 4	t_{RDS4}	-5.5	—	ns	52.20
Read data hold time 4	t_{RDH4}	1.5	—	ns	52.20
Write enable delay time 4	t_{WED4}	-4.0	5.5	ns	52.20
Write data delay time 4	t_{WDD4}	—	5.5	ns	52.20
Write data hold time 6	t_{WDH6}	-4.0	—	ns	52.20

Notes: t_{cyc} shown in the table denotes the CKO clock output cycle time t_{CKOcyc}.

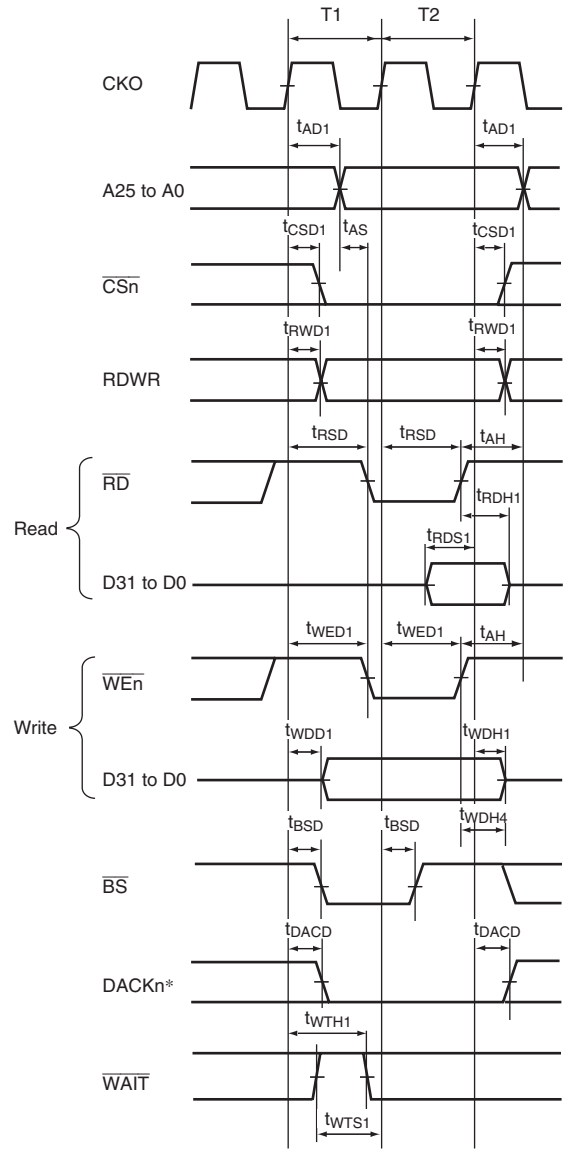
* i. Timing of the start of CS assertion (transition from the high to the low level)

For the normal space, asynchronous burst ROM, and SRAM with byte selection, the minimum value for the time of the first write access via the PCMCIA interface in the specification is invalid.

ii. Timing of CS negation (transition from the low to the high level)

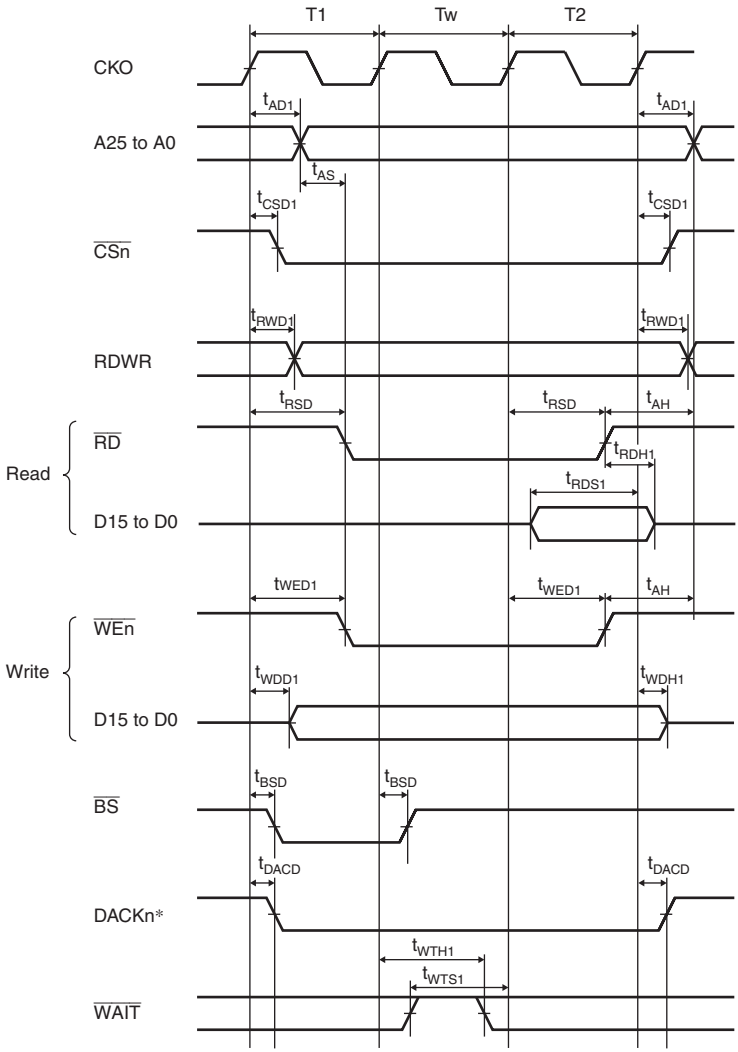
For the normal space and SRAM with byte selection, when successive read access via the PCMCIA interface does not continue, the states of the address signals change one cycle after negation of the CS signal. In this case, the maximum value is invalid.

Furthermore, in write access via the PCMCIA interface (I/O card), since the states of the address signals also change one cycle after negation of the CS signal in the case of continuing access, the maximum value is invalid.



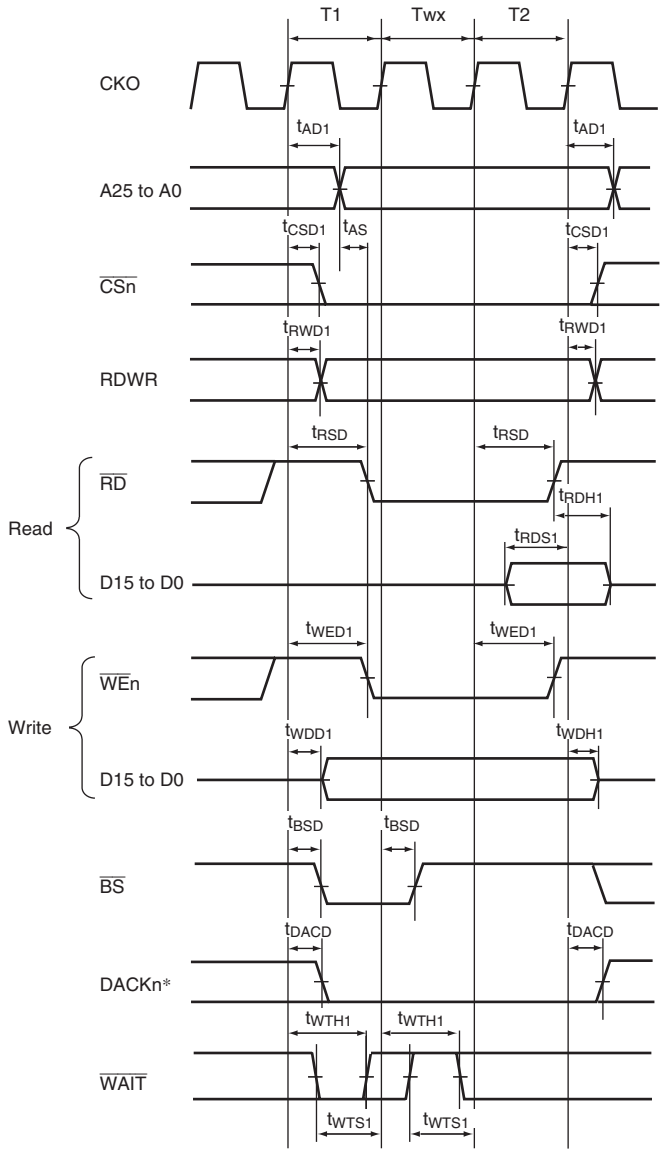
Note: * Waveform when active low is specified for DACKn.

Figure 52.10 Basic Bus Cycle in Normal Space (No Wait)



Note: * Waveform when active low is specified for DACKn.

Figure 52.11 Basic Bus Cycle in Normal Space (Software Wait 1)



Note: * Waveform when active low is specified for DACKn.

Figure 52.12 Basic Bus Cycle in Normal Space (Asynchronous External Wait 1)

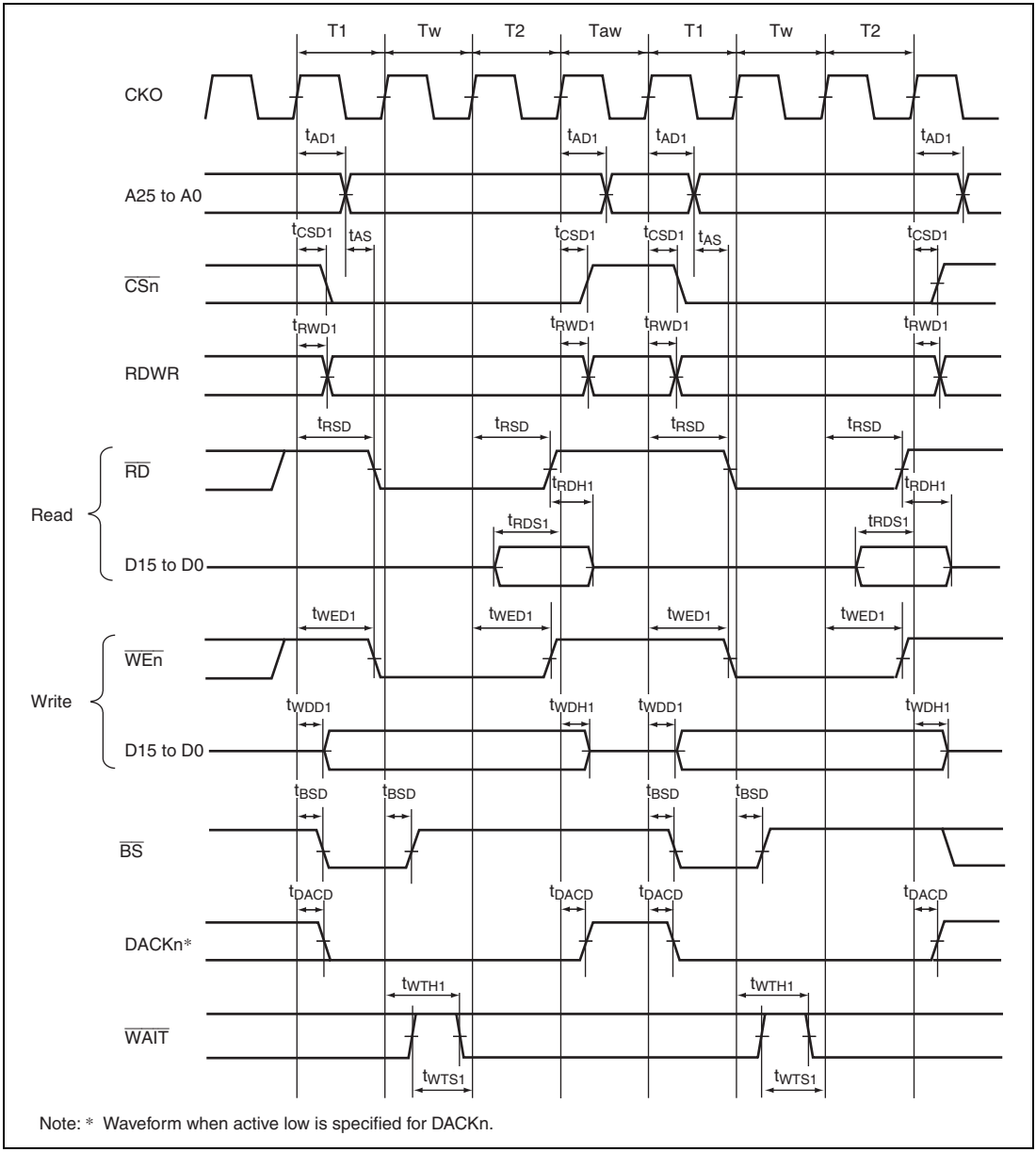
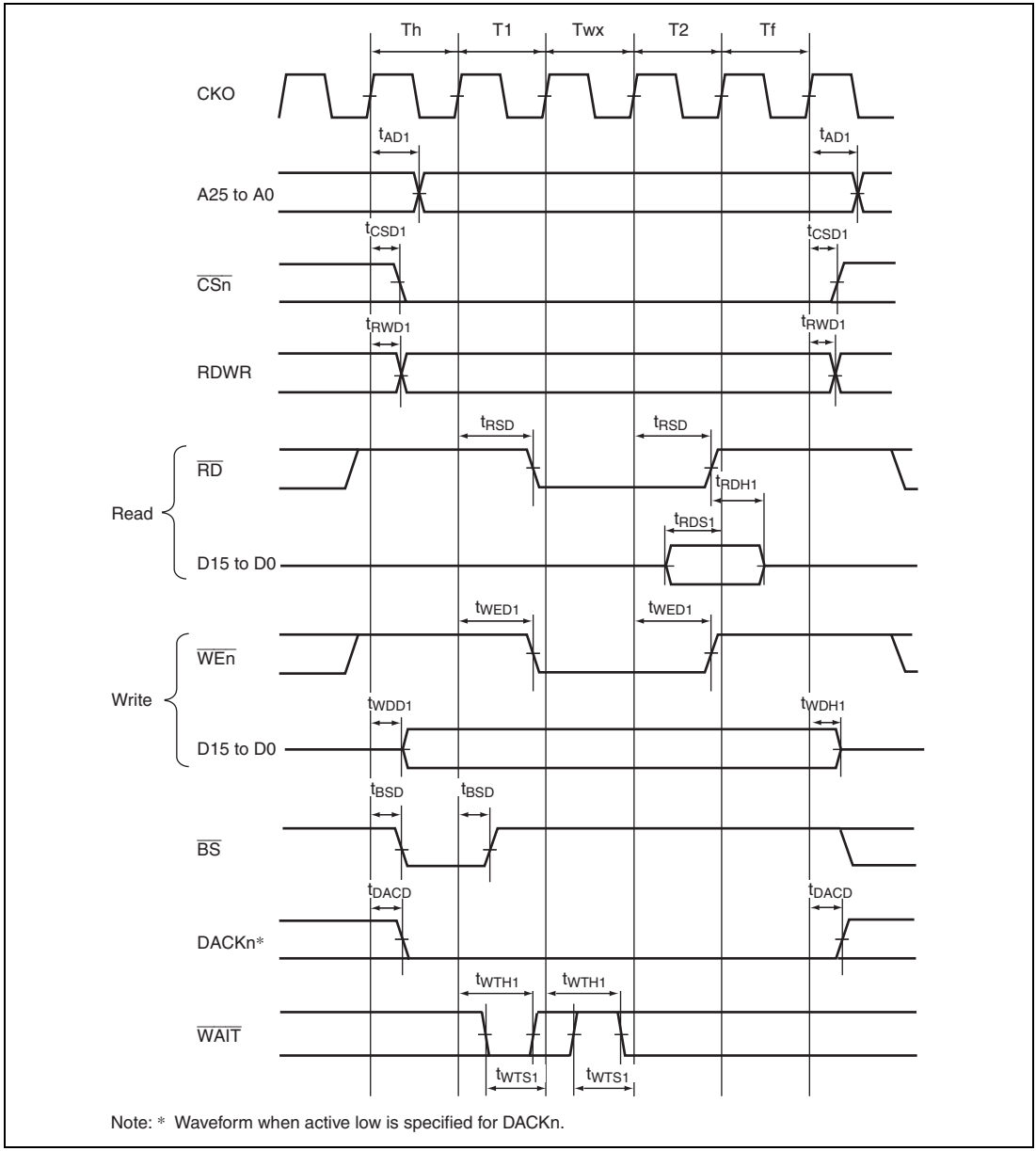


Figure 52.13 Basic Bus Cycle in Normal Space
(Software Wait 1, Asynchronous External Wait Valid (WM Bit = 0), No Idle Cycle)



Note: * Waveform when active low is specified for DACKn.

Figure 52.14 CS Extended Bus Cycle in Normal Space (SW = 1 Cycle, HW = 1 Cycle, Asynchronous External Wait 1)

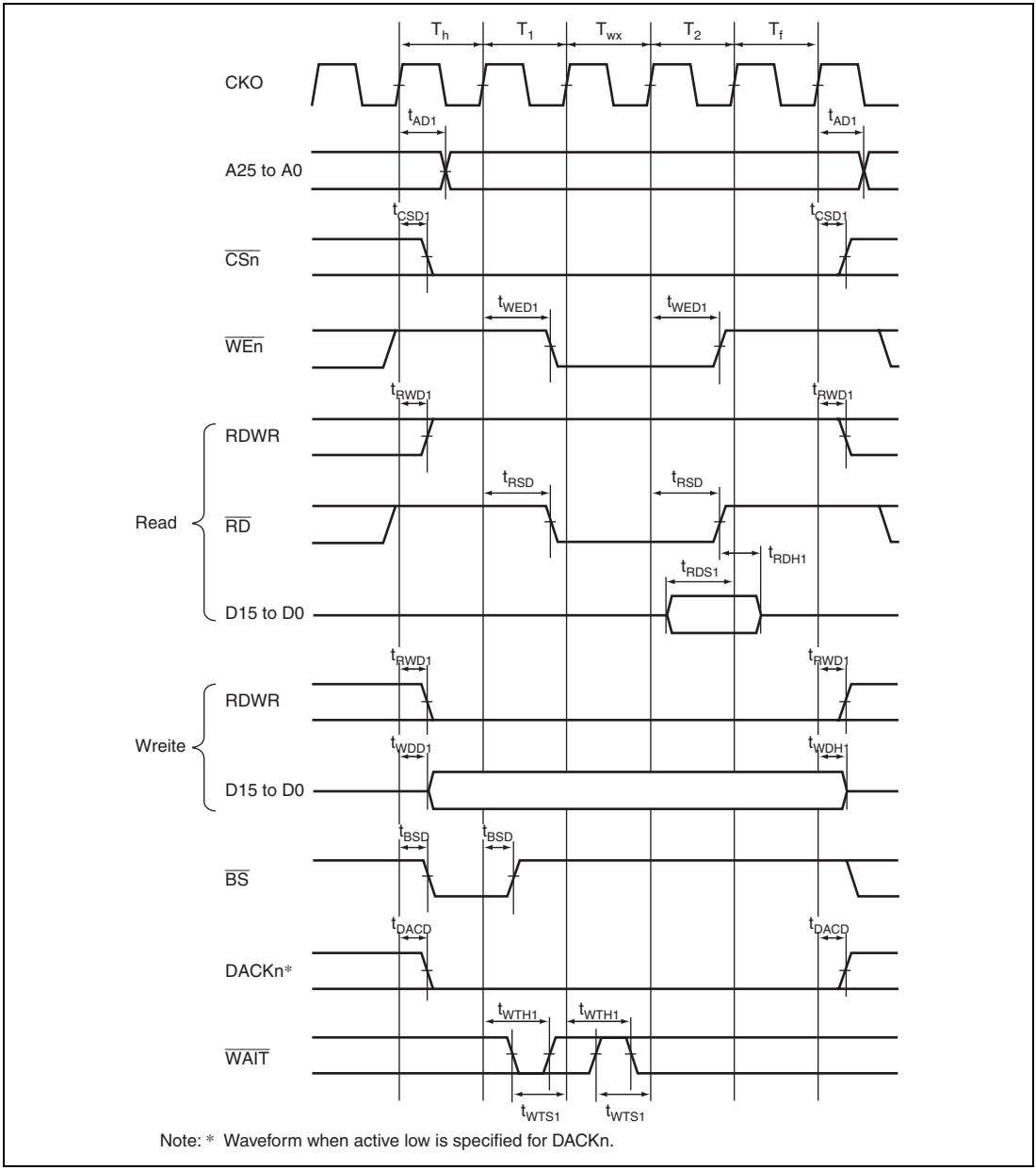
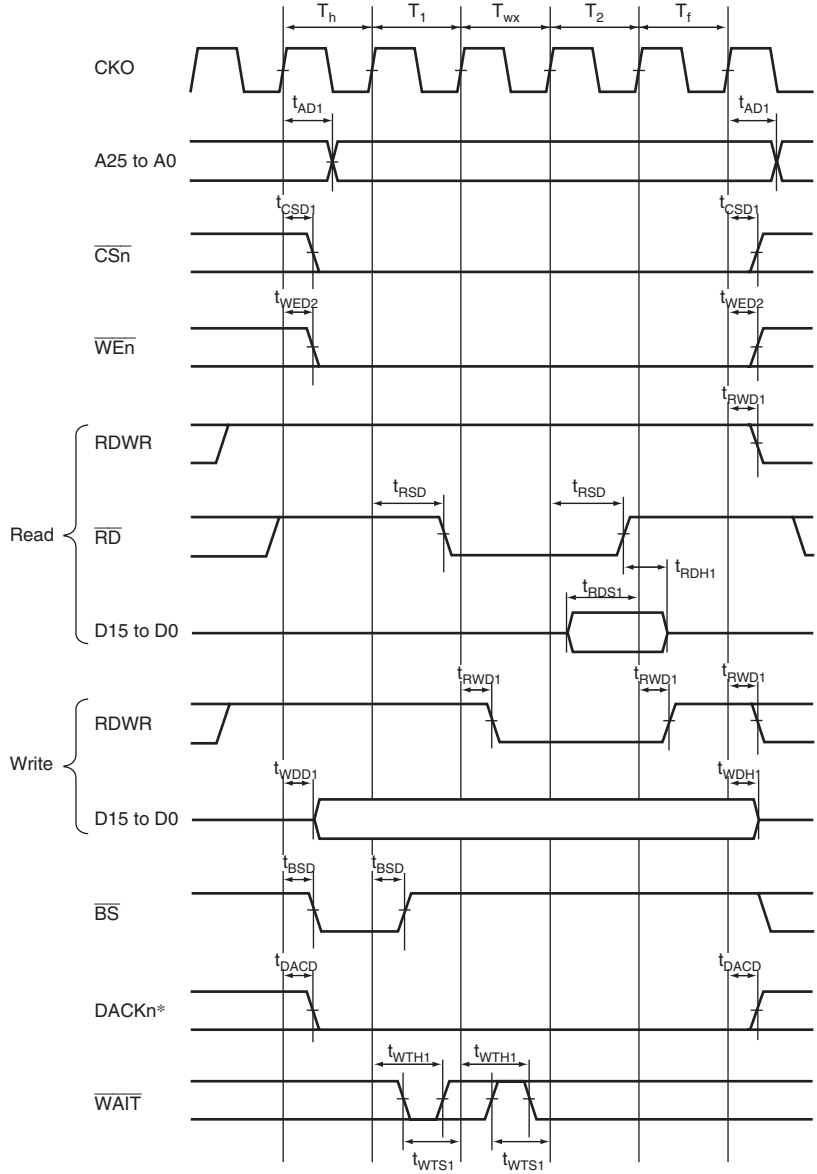


Figure 52.15 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, Asynchronous External Wait 1, BAS = 0 (UB and LB in Write Cycle Controlled))



Note: * Waveform when active low is specified for DACKn.

Figure 52.16 SRAM Bus Cycle with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, Asynchronous External Wait 1, BAS = 1 (Write Cycle WE Control))

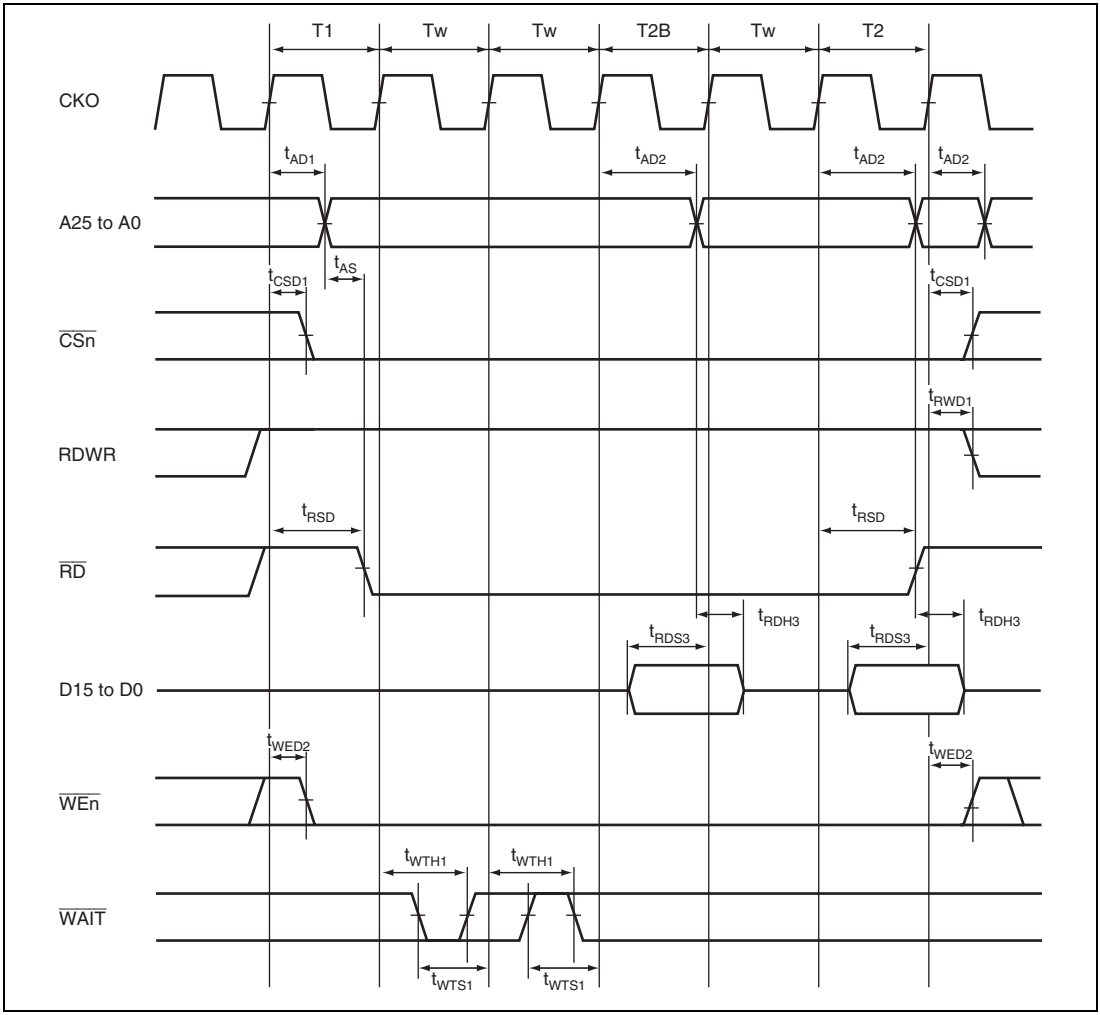


Figure 52.17 SRAM Page Mode Read Bus Cycle with Byte Selection PMD = 1, BAS = 1 (Software Wait 1, Asynchronous External Wait 1, Burst Wait 1, 2 Bursts)

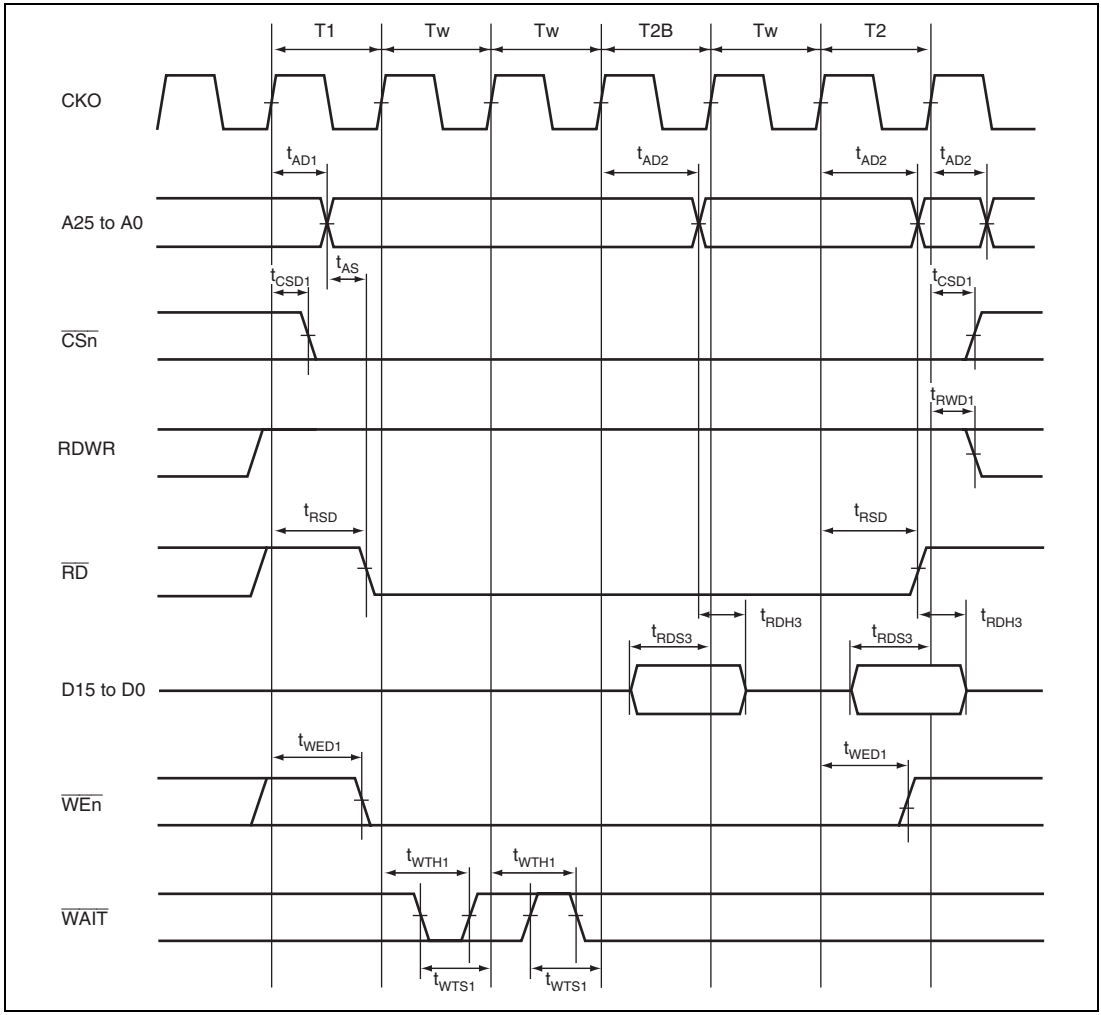


Figure 52.18 SRAM Page Mode Read Bus Cycle with Byte Selection PMD = 1, BAS = 0 (Software Wait 1, Asynchronous External Wait 1, Burst Wait 1, 2 Bursts)

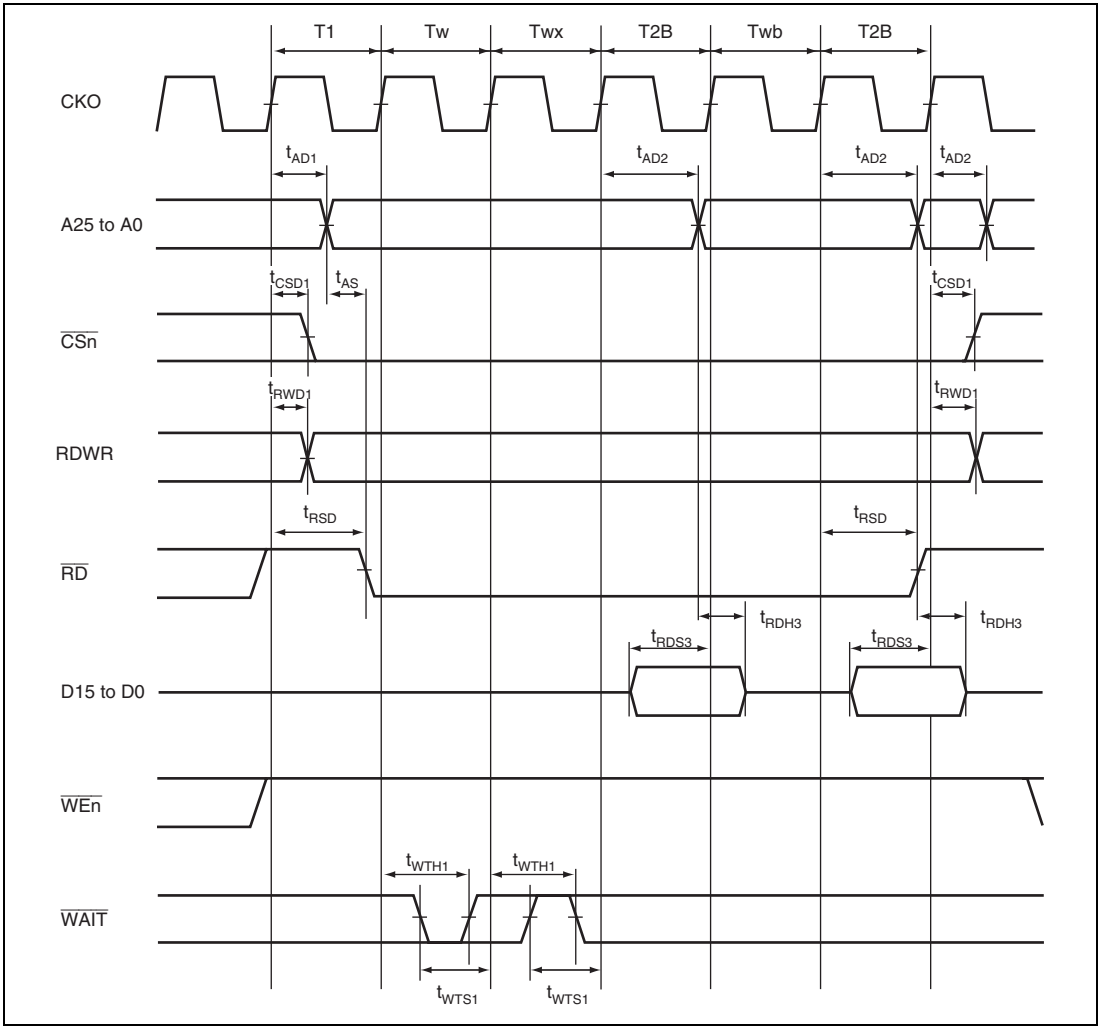


Figure 52.19 Read Bus Cycle of Burst ROM
(Software Wait 1, Asynchronous External Wait 1, Burst Wait 1, 2 Bursts)

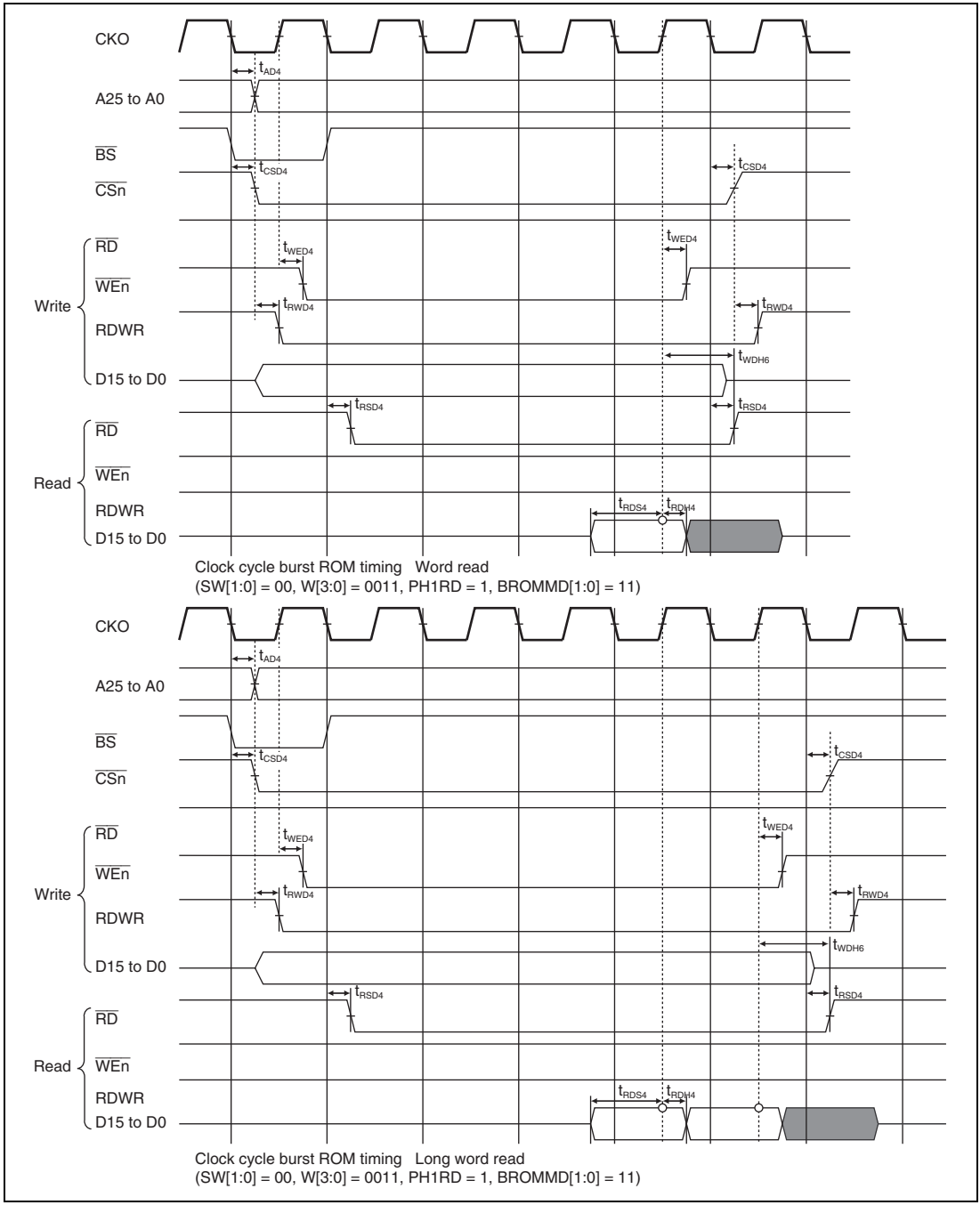


Figure 52.20 Clock-synchronous Burst ROM Timing

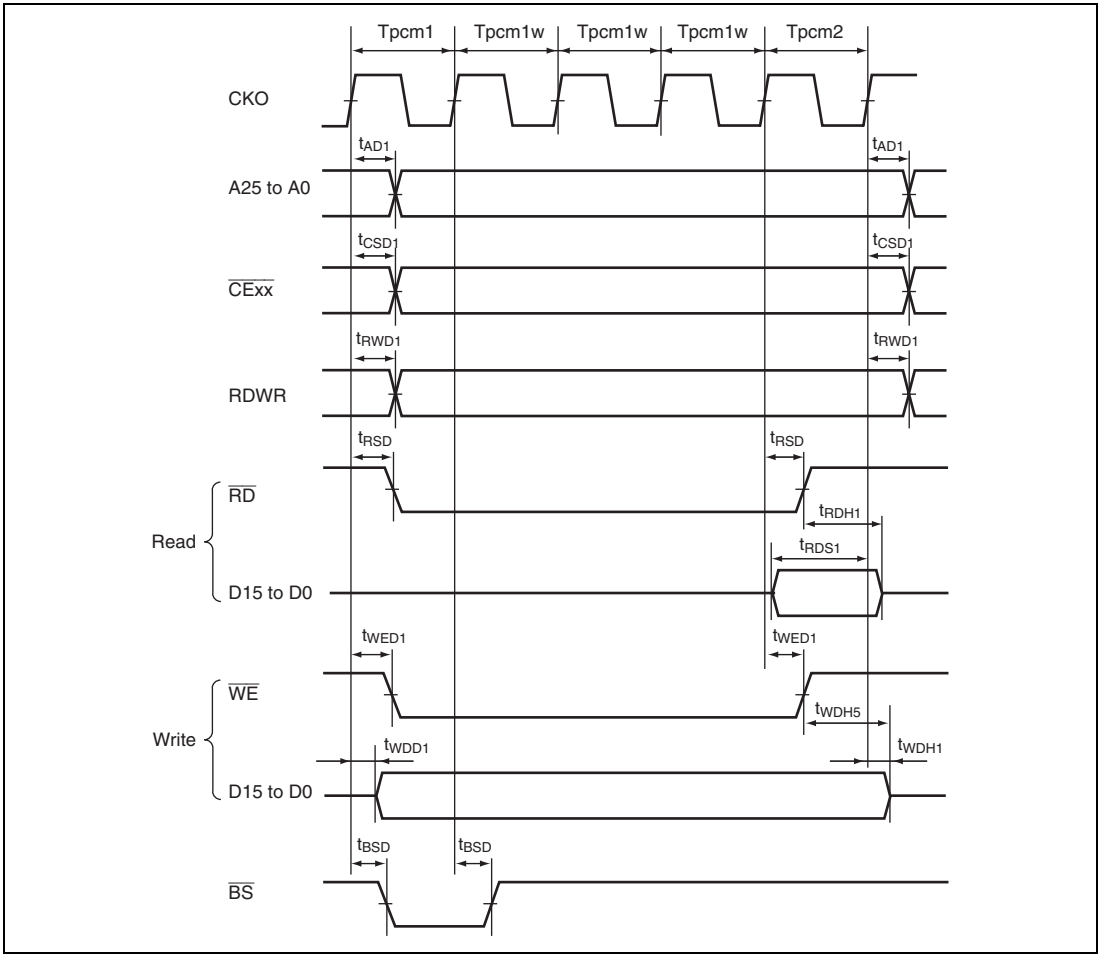


Figure 52.21 PCMCIA Memory Card Interface Bus Timing

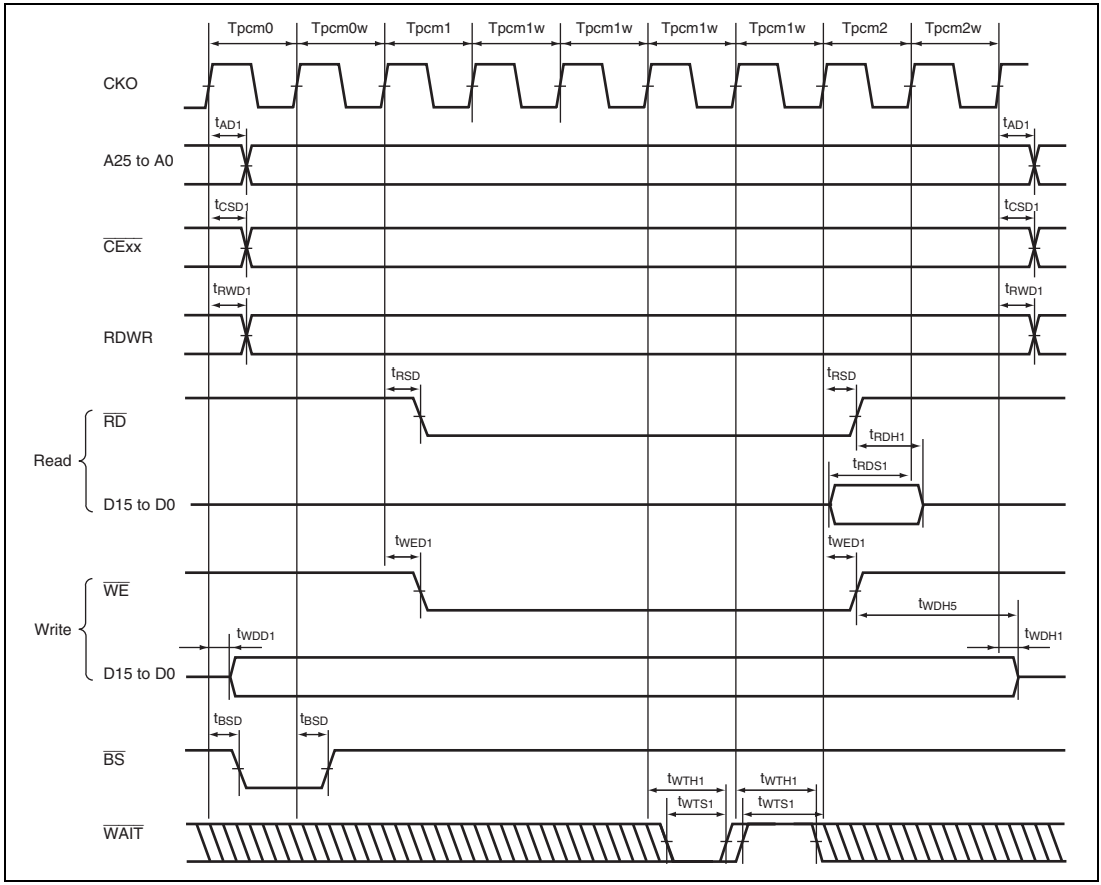


Figure 52.22 PCMCIA Memory Card Interface Bus Timing (TED[3:0] = B'0010, TEH[3:0] = B'0001, Software Wait 1, Hardware Wait 1)

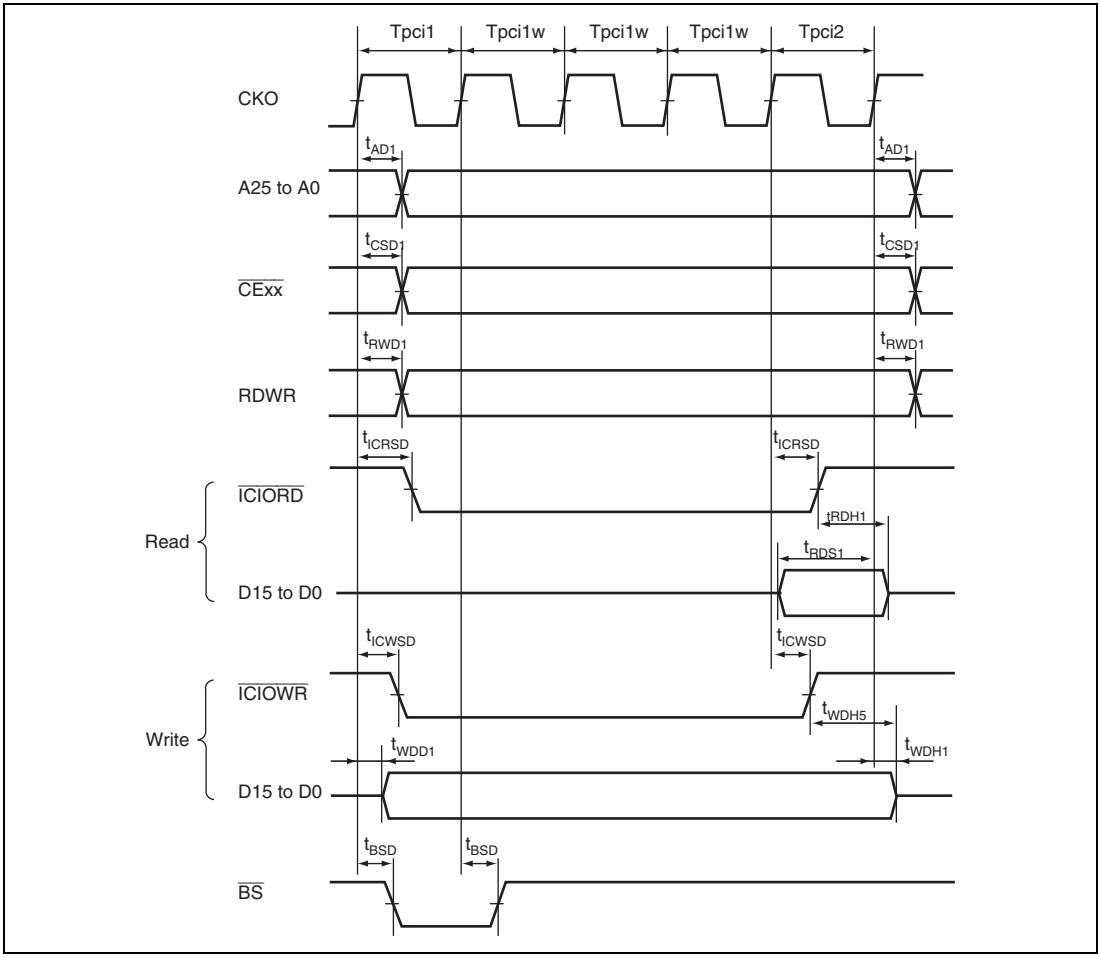


Figure 52.23 PCMCIA I/O Card Interface Bus Timing

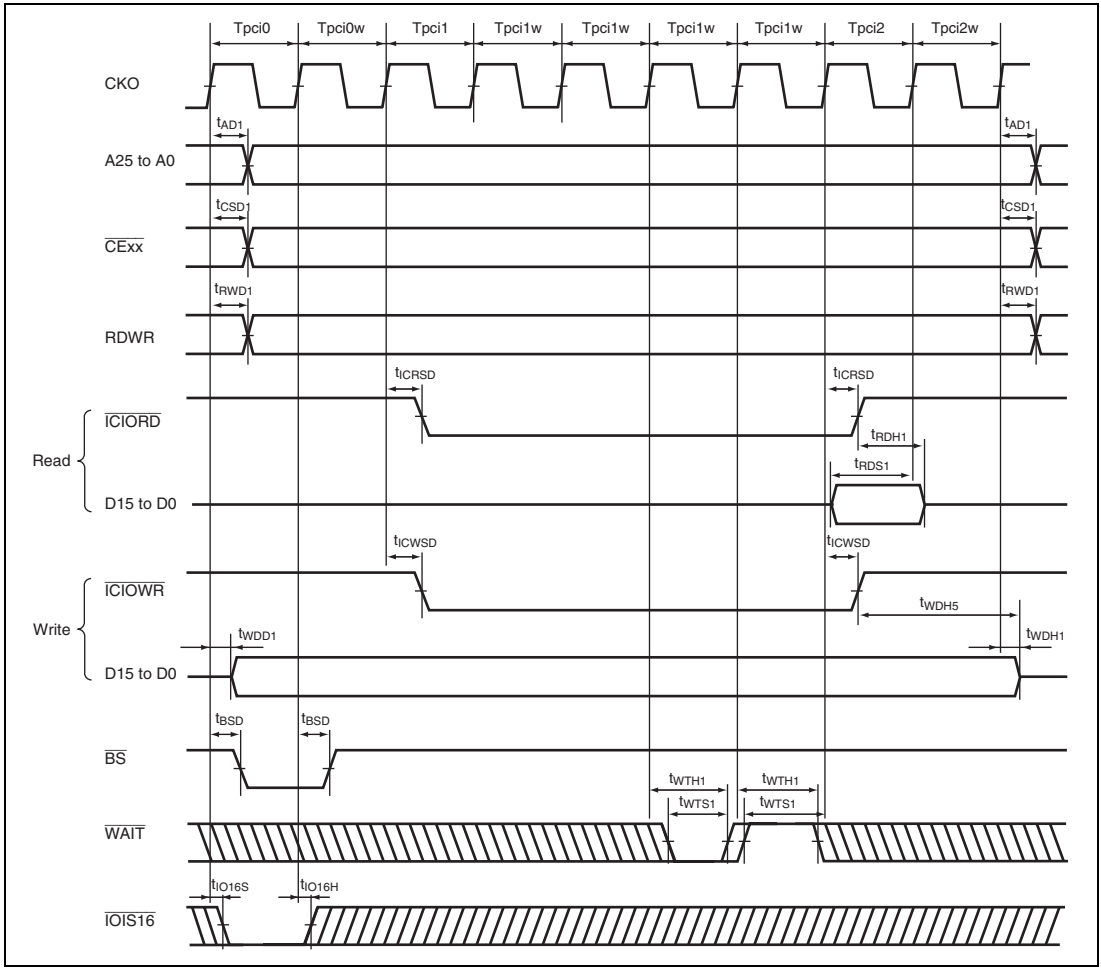


Figure 52.24 PCMCIA I/O Card Interface Bus Timing
(TED[3:0] = B'0010, TEH[3:0] = B'0001, Software Wait 1, Hardware Wait 1)

52.5.5 DBSC Bus Timing

- DDR2-SDRAM

Table 52.12 DDR2-SDRAM Interface Timing

Item	Symbol	Min.	Max.	Unit	Reference
CK cycle	t_{CK}	6.0	6.67	ns	Figure 52.25
CK high period	t_{CH} (abs)	0.45	0.55	t_{CK}	
CK low period	t_{CL} (abs)	0.45	0.55	t_{CK}	
Control signal setup time for CK	t_{IS}	0.88	—	ns	Figure 52.26
Control signal hold time for CK	t_{IH}	0.88	—	ns	
Control/address signal width	t_{IPW}	0.6	—	t_{CK}	
Skew between CK and DQS (read)	t_{RDQSCK}	–0.2	1.4	ns	Figure 52.27
DQS high period (read)	t_{RDQSH}	0.35	0.65	t_{CK}	Figure 52.28
DQS low period (read)	t_{RDQSL}	0.35	0.65	t_{CK}	
DQS pre-amble time (read)	t_{RPRE}	0.9	1.1	t_{CK}	
DQS post-amble time (read)	t_{RPST}	0.4	0.6	t_{CK}	
Skew between DQS and DQ (read)	t_{RDQSQ}	–0.54	0.54	ns	Figure 52.29
DQ hold time for DQS (read)	t_{RQH}	$0.45t_{CK}$–0.47	—	ns	
Rise time of first DQS subsequent to write command issuance (write)	t_{WDQSS}	–0.18	0.18	t_{CK}	Figure 52.30
DQS fall setup time for CK (write)	t_{WDSS}	0.27	—	t_{CK}	
DQS fall hold time for CK (write)	t_{WDSH}	0.27	—	t_{CK}	
DQS high period (write)	t_{WDQSH}	0.35	0.9	t_{CK}	Figure 52.31
DQS low period (write)	t_{WDQSL}	0.35	0.9	t_{CK}	
DQS pre-amble time (write)	t_{WPRE}	0.35	—	t_{CK}	
DQS post-amble time (write)	t_{WPST}	0.4	0.6	t_{CK}	
DQ/DM setup time for DQS (write)	t_{WDS}	0.43	—	ns	Figure 52.32
DQ/DM hold time for DQS (write)	t_{WDH}	0.43	—	ns	
DQ/DM signal width (write)	t_{WDIPW}	0.35	—	t_{CK}	
HiZ transition time of DQ (write)	t_{HZ}	t_{WDH}	t_{CK}	ns	Figure 52.33

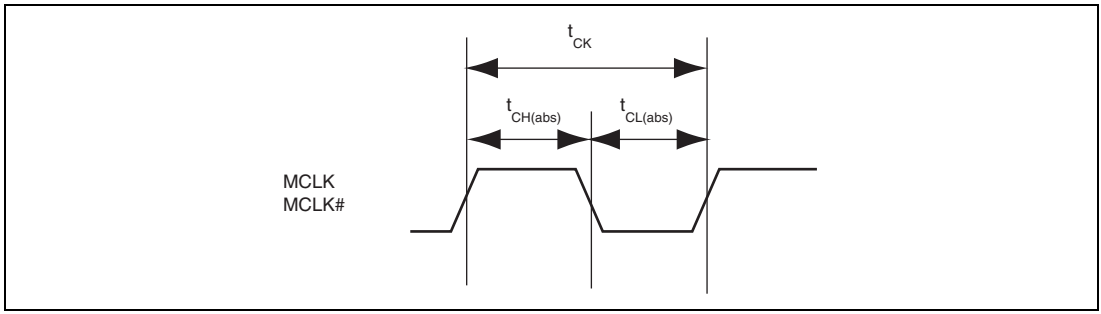


Figure 52.25 Output Clock

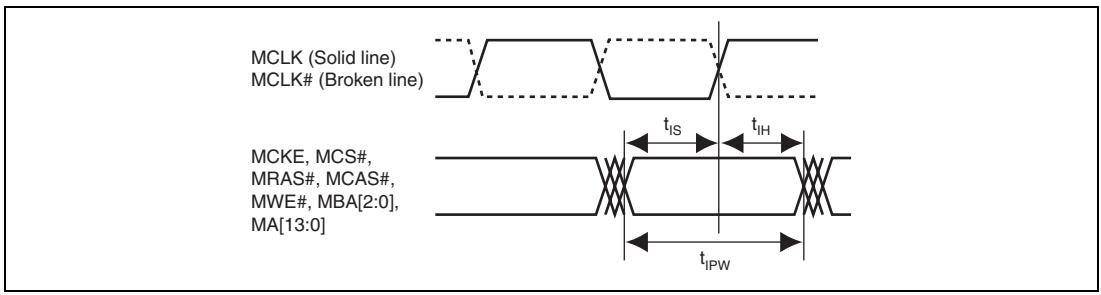


Figure 52.26 Relation between Command Pins and Output Clock

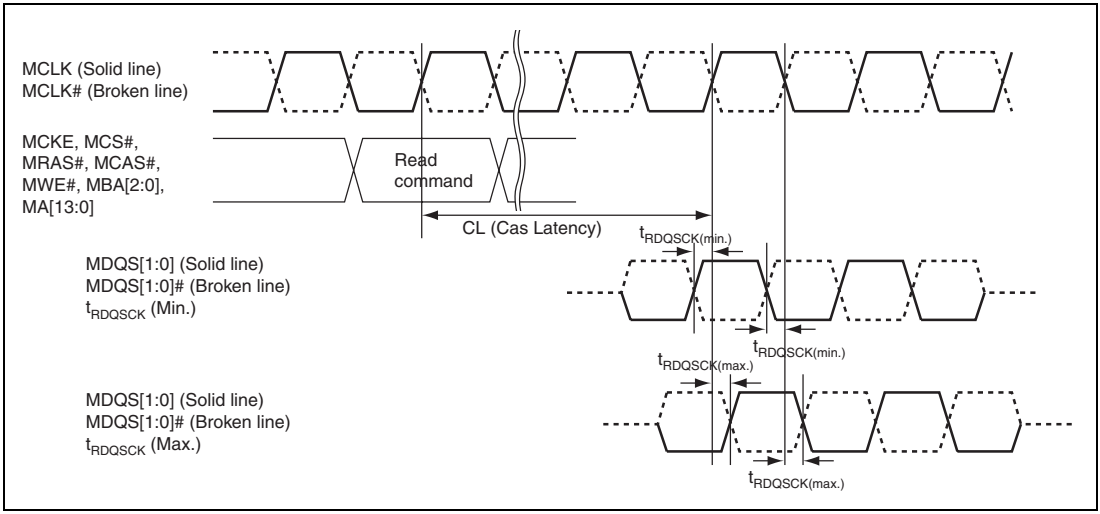


Figure 52.27 DQS Input on Data Read

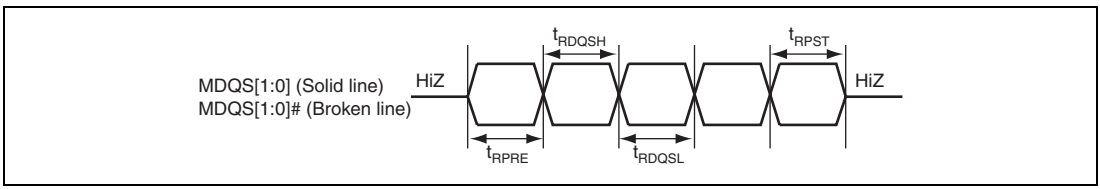


Figure 52.28 DQS Input Waveform Limitation (Read)

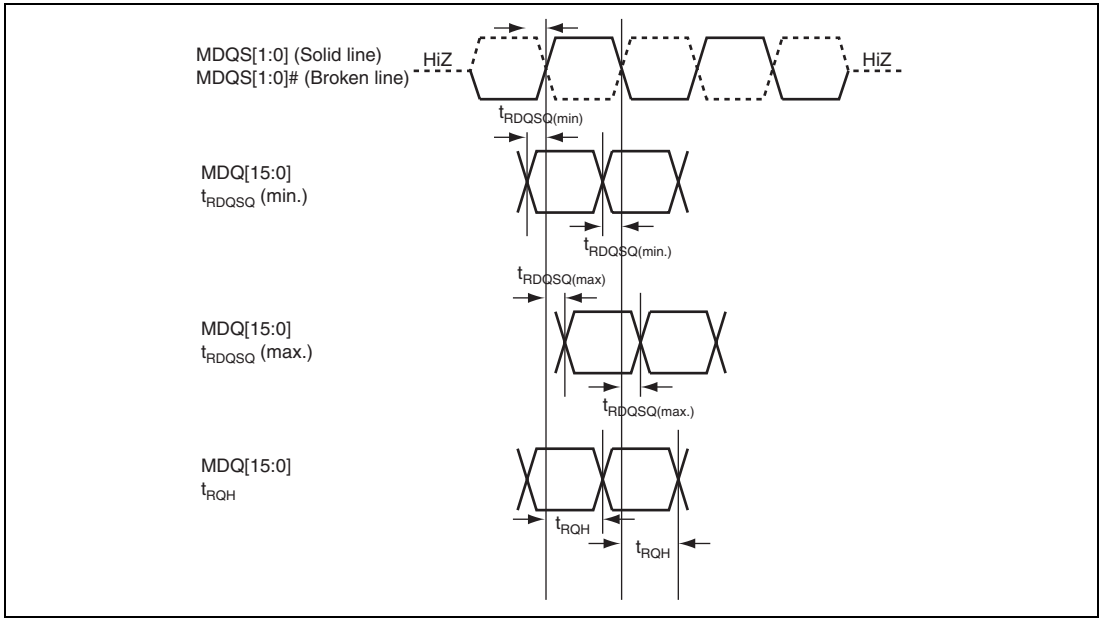


Figure 52.29 DQ Input Waveform Limitation on DQS (Read)

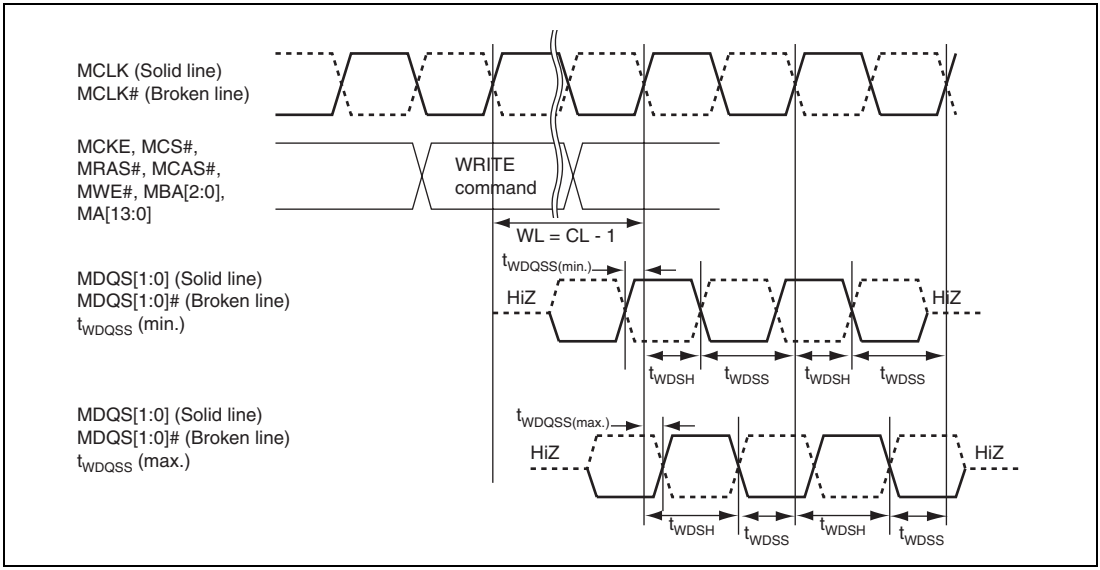


Figure 52.30 Relation of DQS Output Waveform to CK (Write)

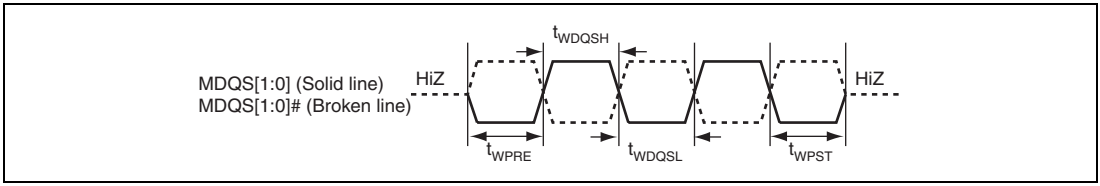


Figure 52.31 Relation between DQS Output Waveforms (Write)

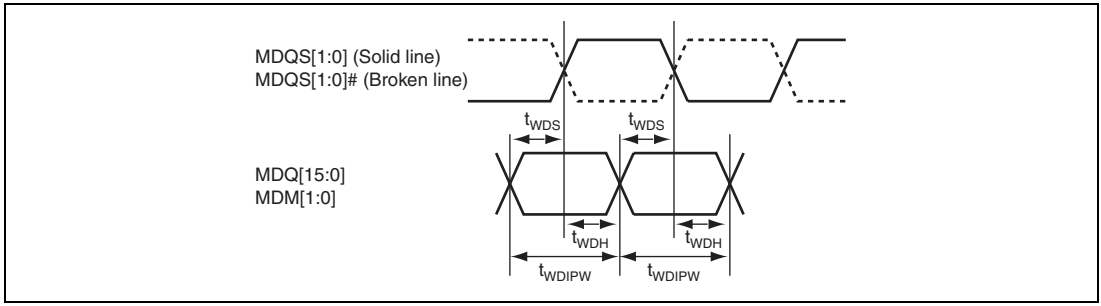


Figure 52.32 Relation between DQS and DQ/DQM Output Waveforms (Write)

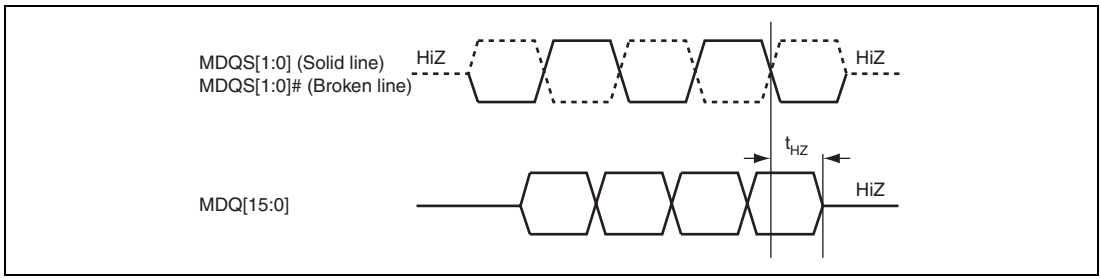


Figure 52.33 Time Required for DQ Output to enter Hi-Z state (Write)

- Mobile-DDR-SDRAM

Table 52.13 Mobile-DDR-SDRAM Interface Timing

		Condition				
Item	**Symbol**	**Min.**	**Typ.**	**Max.**	**Unit**	**Remarks**
CK cycle	t_{CK}	6.0	—	—	ns	Figure 52.34
CK high period	t_{CH}	0.45	—	0.55	t_{CK}	
CK low period	t_{CL}	0.45	—	0.55	t_{CK}	
Control signal setup time for CK	t_{IS}	1.1	—	—	ns	
Control signal hold time for CK	t_{IH}	1.1	—	—	ns	
Control/address signal width	t_{IPW}	2.7	—	—	ns	
First DQS raise time	t_{DQSS}	0.85	—	1.25	t_{CK}	
DQS write pre-amble setup time	t_{WPRES}	0.5	—	—	ns	
DQS output high level pulse width	t_{DQSH}	0.42	—	0.58	t_{CK}	
DQS output low level pulse width	t_{DQSL}	0.42	—	0.58	t_{CK}	
DQS fall setup time	t_{DSS}	0.18	—	—	t_{CK}	
DQS fall hold time	t_{DSH}	0.18	—	—	t_{CK}	
DQS write post-amble time	t_{WPST}	0.4	—	0.6	t_{CK}	
DQM setup time for DQS	t_{DS}	0.7	—	—	ns	
DQM hold time for DQS	t_{DH}	0.7	—	—	ns	
DQS read pre-amble time	t_{RPRE}	0.9	—	1.1	t_{CK}	Figure 52.35
DQS hold time from clock	t_{DQSCK}	2.0	—	7.7	ns	
DQS read post-amble time	t_{RPST}	0.4	—	0.6	t_{CK}	
DQ read access time	t_{AC}	2.0	—	7.7	ns	
DQ/DQS read hold time from DQS	t_{QH}	0.33	—	—	t_{CK}	
Skew between DQS and DQ	t_{DQSQ}	—	—	0.62	ns	

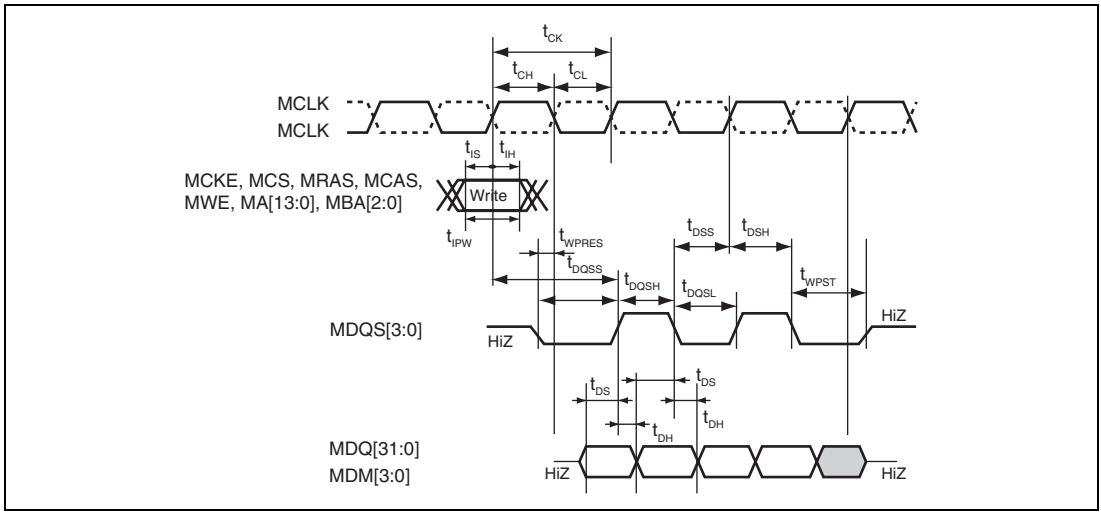


Figure 52.34 Mobile-DDR-SDRAM Output Timing (Write)

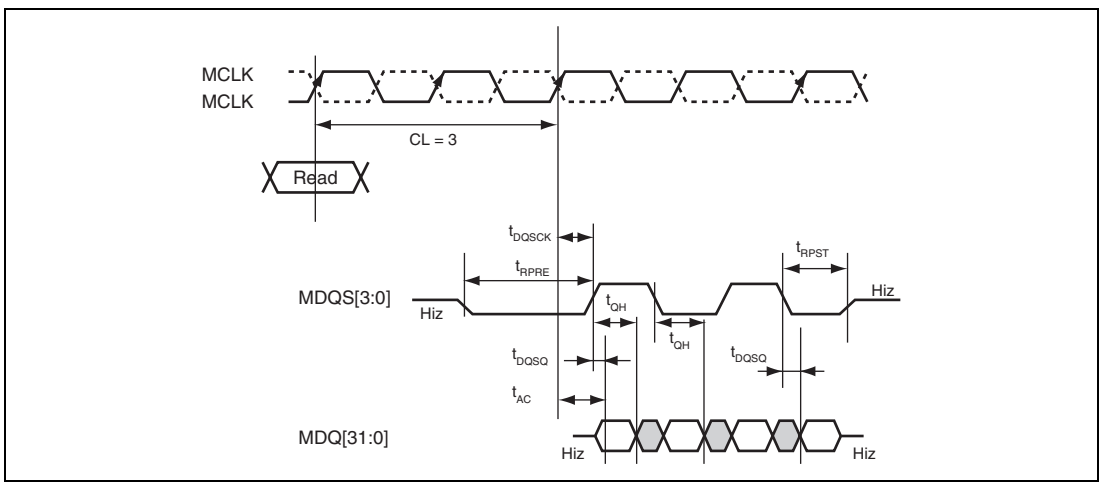


Figure 52.35 Mobile-DDR-SDRAM Input Timing (Read)

52.5.6 I/O Port Signal Timing

Table 52.14 Peripheral Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
Output data delay time	t_{PORTD}	—	17	ns	52.36
Input data setup time	t_{PORTS}	17	—		
Input data hold time	t_{PORTH}	10	—		

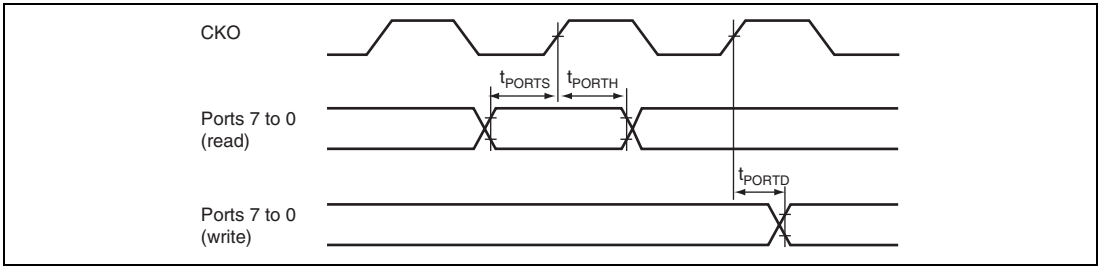


Figure 52.36 I/O Port Timing

52.5.7 DMAC Module Signal Timing

Table 52.15 DMAC Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
DREQ setup time	t_{DREQS}	8	—	ns	52.37
DREQ hold time	t_{DREQH}	8	—		
DACK delay time	t_{DACD}	—	15		52.38

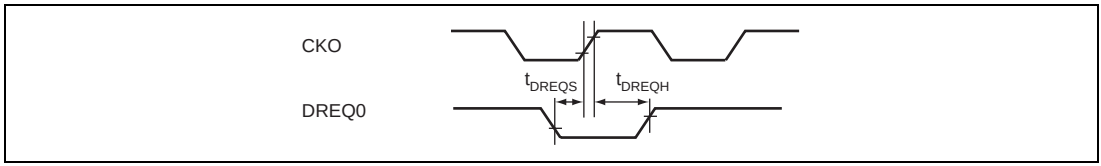


Figure 52.37 DREQ Input Timing (DREQ Low Level Detected)

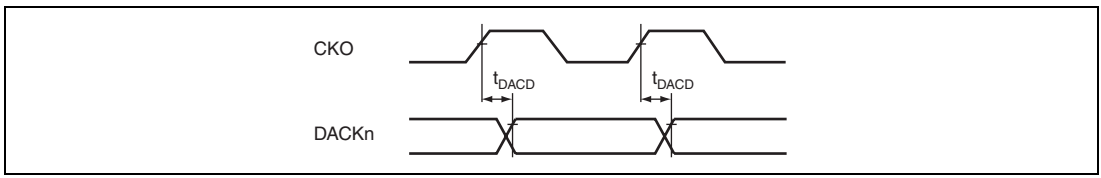


Figure 52.38 DACK Output Timing

52.5.8 TPU Module Signal Timing

Table 52.16 TPU Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
Output data delay time	t_{TOD}	—	15	ns	52.39
Timer clock input setup time	t_{TCKS}	15	—	ns	52.40
Timer clock pulse width	t_{TCKWH} t_{TCKWL}	3	—	tpcyc	52.40

Note: t_{pcyc} denotes one cycle time of the peripheral clock (Pϕ).

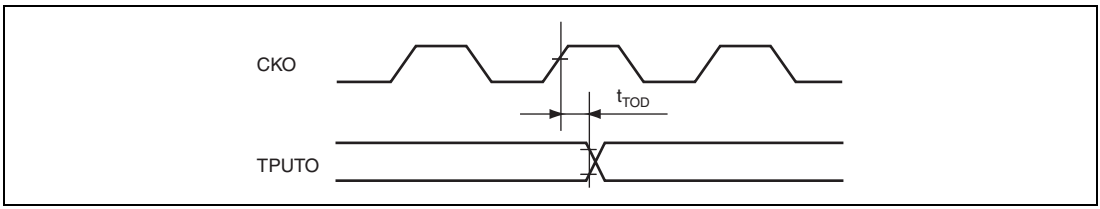


Figure 52.39 TPU Output Timing

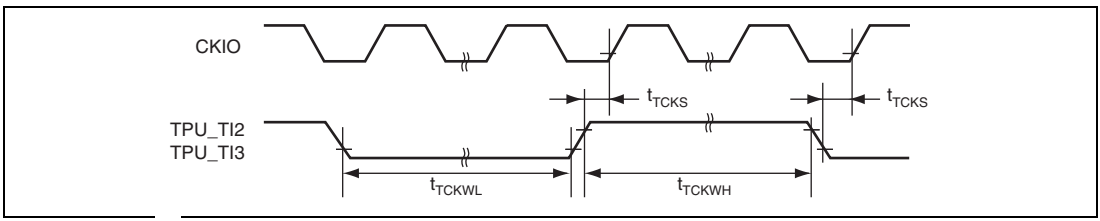


Figure 52.40 TPU Output Timing

52.5.9 MSIOF Module Signal Timing

Table 52.17 MSIOF Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
MSIOF_MCK0/1 clock cycle	t_{MCYC}	$2 \times t_{bcyc}$*	—	ns	52.41
MSIOF_MCK0/1 input high level width	t_{MWH}	$0.4 \times t_{MSCYC}$	—	ns	
MSIOF_MCK0/1 input low level width	t_{MWL}	$0.4 \times t_{MSCYC}$	—	ns	
MSIOF_TSCK (RSCK) clock cycle	t_{MSCYC}	$2 \times t_{bcyc}$*	—	ns	52.42, 52.43
MSIOF_TSCK (RSCK) output high level width	t_{MSWHO}	$0.4 \times t_{MSCYC}$	—	ns	52.42
MSIOF_TSCK (RSCK) output low level width	t_{MSWLO}	$0.4 \times t_{MSCYC}$	—	ns	
MSIOF_TSCK (RSCK) input high level width	t_{MSWHI}	$0.4 \times t_{MSCYC}$	—	ns	
MSIOF_TSCK (RSCK) input low level width	t_{MSWLI}	$0.4 \times t_{MSCYC}$	—	ns	52.43
MSIOF_TSYNC (RSYNC) output delay time	t_{FSD}	—	20	ns	52.42
MSIOF_TSYNC (RSYNC) input setup time	t_{FSS}	20	—	ns	52.43
MSIOF_TSYNC (RSYNC) input hold time	t_{FSH}	20	—	ns	
MSIOF_TXD output delay time	t_{TDD}	—	20	ns	52.42
MSIOF_RXD input setup time	t_{RDS}	20	—	ns	52.43
MSIOF_RXD input hold time	t_{RDH}	20	—	ns	

Note: t_{bcyc} denotes one cycle time of a bus clock (Bϕ).

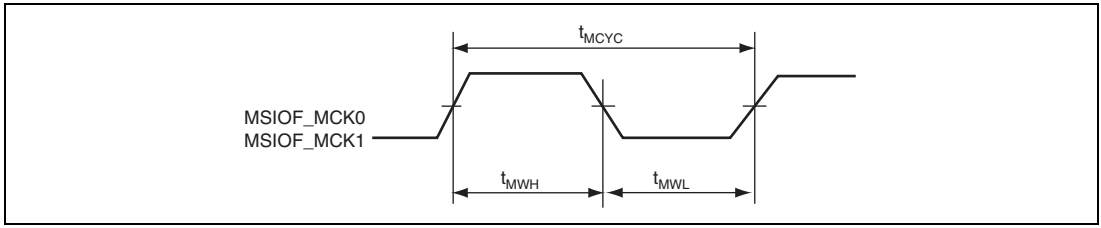


Figure 52.41 MSIOF_MCK Input Timing

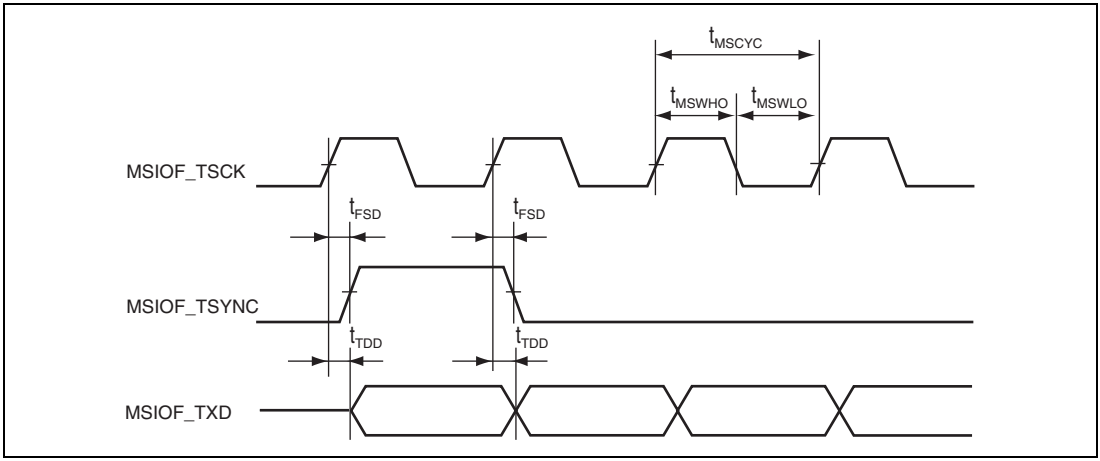


Figure 52.42 MSIOF Transmission/Reception Timing (Master Mode 1)

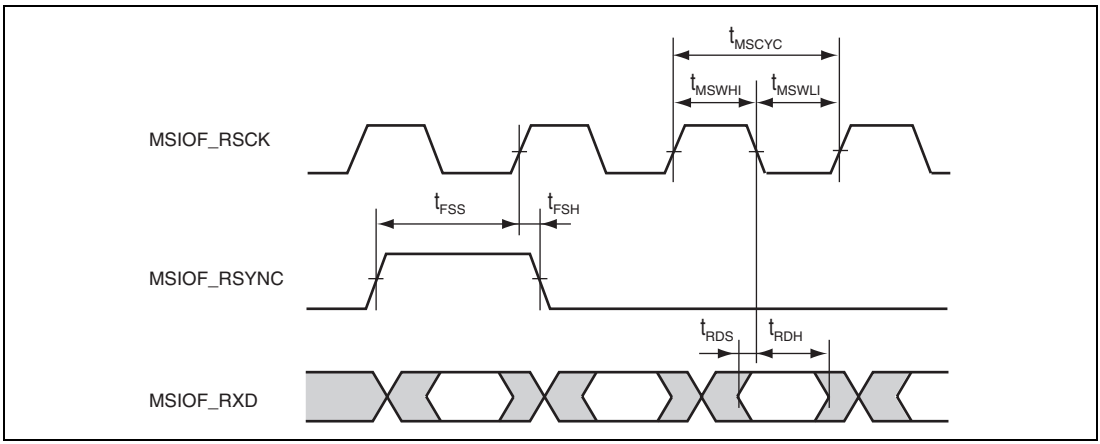


Figure 52.43 MSIOF Transmission/Reception Timing (Slave Mode)

52.5.10 SCIF/SCIFA Module Signal Timing

Table 52.18 SCIF Module Signal Timing (Asynchronous)

Item	Symbol	Min.	Max.	Unit	Figure
SCK input clock cycle	t_{SCYC}	4	—	t_{PCYC}	52.44
SCK input clock high level width	t_{SCWH}	0.4	—	t_{SCYC}	
SCK input clock low level width	t_{SCWL}	0.4	—	t_{SCYC}	
SCK input clock rise time	t_{SCKr}	—	1.5	t_{PCYC}	
SCK input clock fall time	t_{SCKf}	—	1.5	t_{PCYC}	
TXD transfer data delay time	t_{TXD}	—	$3 \times t_{PCYC} + 50$	ns	52.45
RXD input data setup time	t_{RXS}	$2 \times t_{PCYC}$	—	ns	
RTS input data hold time	t_{RXH}	$2 \times t_{PCYC}$	—	ns	
RTS delay time	t_{RTSD}	—	100	ns	
CTS setup time	t_{CTSS}	100	—	ns	
CTS hold time	t_{CTSH}	100	—	ns	

Note: For the SCIF module, t_{pcyc} denotes the period of one cycle of the peripheral clock (Pϕ).
For the SCIFA module, t_{pcyc} denotes the period of one cycle of the bus clock (Bϕ).

Table 52.19 SCIF Module Signal Timing (Clocked Synchronous)

Item	Symbol	Min.	Max.	Unit	Figure
SCK input/output clock cycle	t_{SCYC}	12	—	t_{PCYC}	52.44
SCK input/output clock high level width	t_{SCWH}	0.4	0.6	t_{SCYC}	
SCK input/output clock low level width	t_{SCWL}	0.4	0.6	t_{SCYC}	
SCK input/output rise time (clocked synchronous)	t_{sSCKr}	—	1.5	t_{PCYC}	
SCK input/output fall time (clocked synchronous)	t_{sSCKf}	—	1.5	t_{PCYC}	
TXD output data delay time (SCK input)	t_{TXD}	—	$3 \times t_{pcyc}$ + 50	ns	52.45
TXD output data delay time (SCK output)		—	50	ns	
RXD input data setup time (common to SCK input and output)	t_{RXS}	4	—	t_{PCYC}	
RXD input data hold time (common to SCK input and output)	t_{RXH}	4	—	t_{PCYC}	

Note: In SCIF module t_{pcyc} stands for a cycle time of a peripheral clock (Pϕ).
In SCIFA module t_{pcyc} stands for a cycle time of a bus clock (Bϕ).

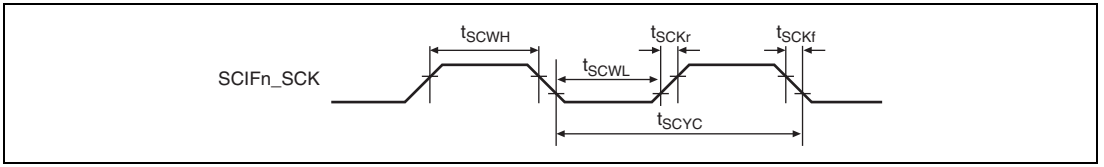


Figure 52.44 SCIF/SCIFA Module Signal Timing

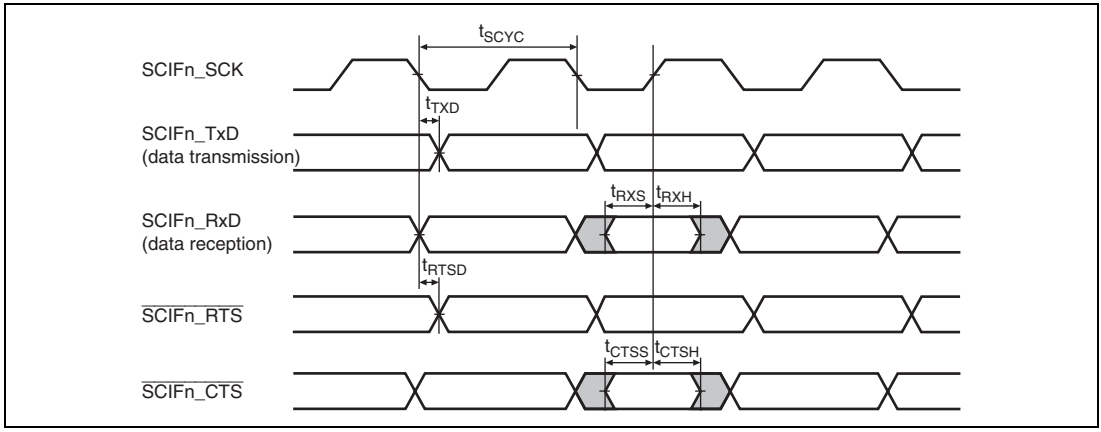


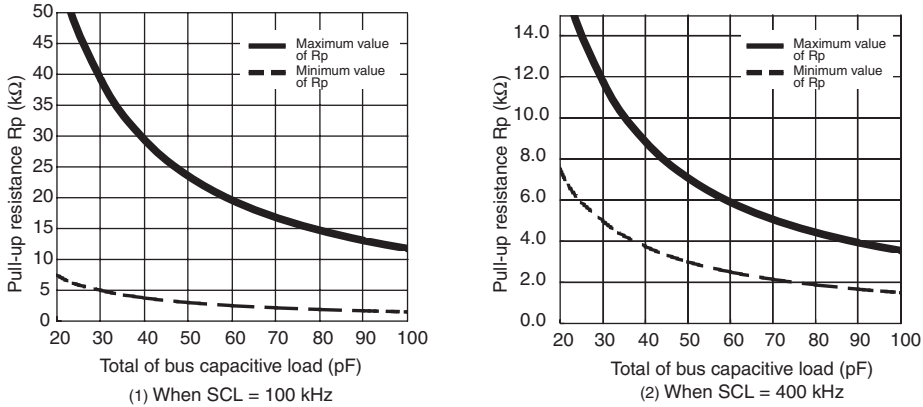
Figure 52.45 SCIF Module Signal Timing

52.5.11 I²C Module Signal Timing

Table 52.20 SDA and SCL Bus Line Characteristics for I²C Bus Device

		Normal Mode		High-Speed Mode			
Item	**Symbol**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**	**Figure**
SCL clock frequency	f_{SCL}	0	100	0	400	kHz	52.46
Hold time (after repeat START condition, first clock pulse is generated)	$t_{HD;STA}$	4.0	—	0.6	—	µs	
Low period in SCL clock	t_{LOW}	4.7	—	1.3	—	µs	
High period in SCL clock	t_{HIGH}	4.0	—	0.6	—	µs	
Setup time for repeat START condition	$t_{SU;STA}$	4.7	—	0.6	—	µs	
Data hold time: for I²C bus device	$t_{HD;DAT}$	—	3.45	—	0.9	µs	
Data setup time	$t_{SU;DAT}$	250	—	100	—	ns	
SDA and SCL signal rise time	t_r	—	1000	—	300	ns	
SDA and SCL signal fall time	t_f	—	300	—	300	ns	
Setup time for STOP condition	$t_{SU;STO}$	4.0	—	0.6	—	µs	
Bus free time between STOP and START conditions	t_{BUF}	4.7	—	1.3	—	µs	
Noise margin at low level of each connected device (including hysteresis)	V_{nL}	$0.1 \times V_{CC}Q$	—	$0.1 \times V_{CC}Q$	—	V	
Noise margin at high level of each connected device (including hysteresis)	V_{nH}	$0.2 \times V_{CC}Q$	—	$0.2 \times V_{CC}Q$	—	V	

Notes: 1. All values are referenced at $V_{CC}Q \times 0.3$ and $V_{CC}Q \times 0.7$ levels.
2. To satisfy the I²C-bus specification, pull-up resistors (Rp) with the appropriate resistance must be included depending on the total of bus capacitive load of each line.
3. Relationship between pull-up resistance and total capacitive load of the I²C bus.



* A hold time of at least 300 ns is internally assured for the SDA signal (relative to V_{IH}min of the SCL signal). The state of the SDA signal is stabilized on falling edges of the SCL signal.

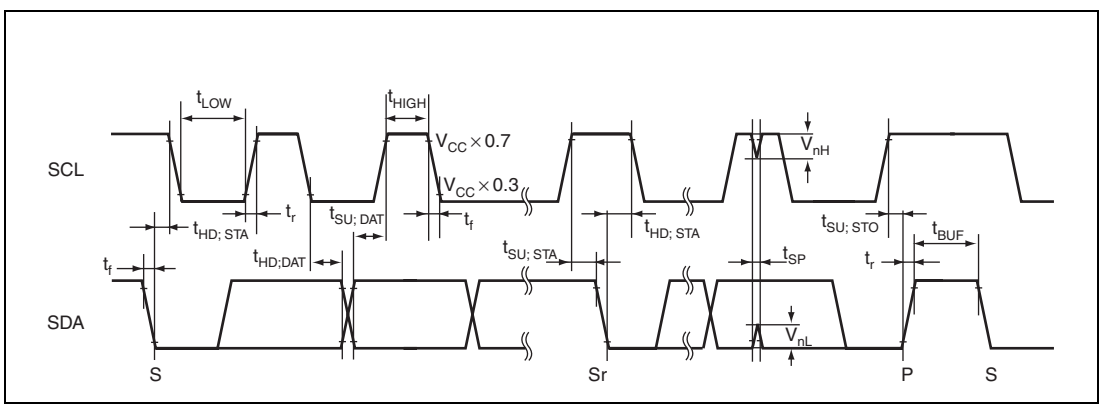


Figure 52.46 Device Timing Definition on I²C Bus

52.5.12 CEU Module Signal Timing

Table 52.21 CEU Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
Vertical sync (VIO_VD) setup time	t_{VVDS}	2	—	ns	52.47
Vertical sync (VIO_VD) hold time	t_{VVDH}	3.5	—	ns	
Horizontal sync (VIO_HD) setup time	t_{VHDS}	2	—	ns	
Horizontal sync (VIO_HD) hold time	t_{VHDH}	3.5	—	ns	
Capture image data (VIO_D) setup time	t_{VDTS}	2	—	ns	
Capture image data (VIO_D) hold time	t_{VDTH}	3.5	—	ns	
Camera clock cycle	t_{VCYC}	t_{bcyc}*	—	ns	
Camera clock high width	t_{VHW}	$0.4 \times t_{Vcyc}$	—	ns	
Camera clock low width	t_{VLW}	$0.4 \times t_{Vcyc}$	—	ns	
Field identification signal (VIO_FLD) setup time	t_{VFDS}	2	—	ns	
Field identification signal (VIO_FLD) hold time	t_{VFDH}	3.5	—	ns	

Note: * t_{bcyc} is a cycle time of an internal bus clock (Bϕ).

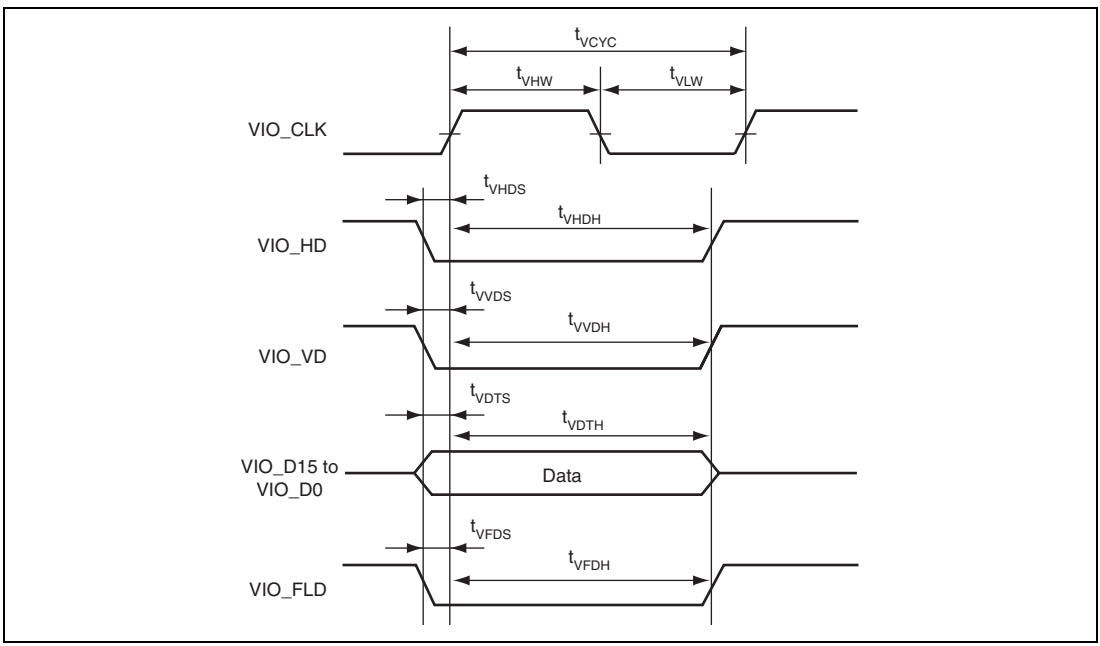


Figure 52.47 VIO Module Signal Timing

52.5.13 LCDC Module Signal Timing

Table 52.22 LCDC Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
Clock (LCDDCK) cycle time	t_{LCC}	12	—	ns	52.48
Clock (LCDLCLK) cycle time	t_{LCLC}	12	—	ns	
Clock (LCDDCK) high pulse time	t_{LCHW}	4	—	ns	
Clock (LCDDCK) low pulse time	t_{LCLW}	4	—	ns	
Data (LCDD) delay time	t_{LDD}	–5.6	4	ns	
Display enable (LCDDISP) delay time	t_{LID}	–5.6	4	ns	
Horizontal sync signal (LCDHSYN) delay time	t_{LHD}	–5.6	4	ns	
Vertical sync signal (LCDVSYN) delay time	t_{LVD}	–5.6	4	ns	
Chip select signal ($\overline{\text{LCDCS}}$) SYS interface command delay time	$t_{LSYSCSD}$	—	22	ns	52.49
Write strobe signal (LCDDCK) SYS interface command delay time	$t_{LSYSWRD}$	—	22	ns	
Register select signal (LCDDISP) SYS interface command delay time	$t_{LSYSRSD}$	—	22	ns	
Data (LCDD) SYS interface command write data delay time	t_{LSYSDD}	—	22	ns	
Read strobe signal ($\overline{\text{LCDRD}}$) SYS interface command delay time	$t_{LSYSRDD}$	—	22	ns	52.50
Data (LCDD) SYS interface read data setup time	$t_{LSYSRDS}$	10	—	ns	
Data (LCDD) SYS interface read data hold time	$t_{LSYSRDH}$	5	—	ns	
Read write signal (LCDVCPWC) SYS interface command delay time	$t_{LSYSRDWRD}$	–12	12	ns	
Write strobe signal ($\overline{\text{LCDWR}}$) SYS interface data cycle time	$t_{LSYSDWRC}$	30	—	ns	52.51
Write strobe signal ($\overline{\text{LCDWR}}$) SYS interface data high pulse time	$t_{LSYSDWRHW}$	9	—	ns	

Item	Symbol	Min.	Max.	Unit	Figure
Write strobe signal ($\overline{\text{LCDWR}}$) SYS interface data low pulse time	$t_{LSYSDWRLW}$	9	—	ns	52.51
Write strobe signal ($\overline{\text{LCDWR}}$) SYS interface data address setup time	$t_{LSYSDAS}$	$t_{LSYSDWRC}$ − 12	$t_{LSYSDWRC}$ + 12	ns	
Write strobe signal ($\overline{\text{LCDWR}}$) SYS interface data address hold time*	$t_{LSYSDAH}$	$t_{LSYSDWRHW}$ − 12	$t_{LSYSDWRHW}$ + 12	ns	
Data (LCDD) SYS interface data delay time	$t_{LSYSDDD}$	–12	12	ns	
Input vertical sync signal (LCDVSYN) setup time	t_{LVIS}	10	—	ns	52.52
Input vertical sync signal (LCDVSYN) hold time	t_{LVIH}	5	—	ns	

Note: * The minimum value of $t_{LSYSDAH}$ is one unit of $t_{LSYSDWRHW}$. $t_{LSYSDWRHW}$ can be set to any value through MLDDCKPAT1R and MLDDCKPAT2R.

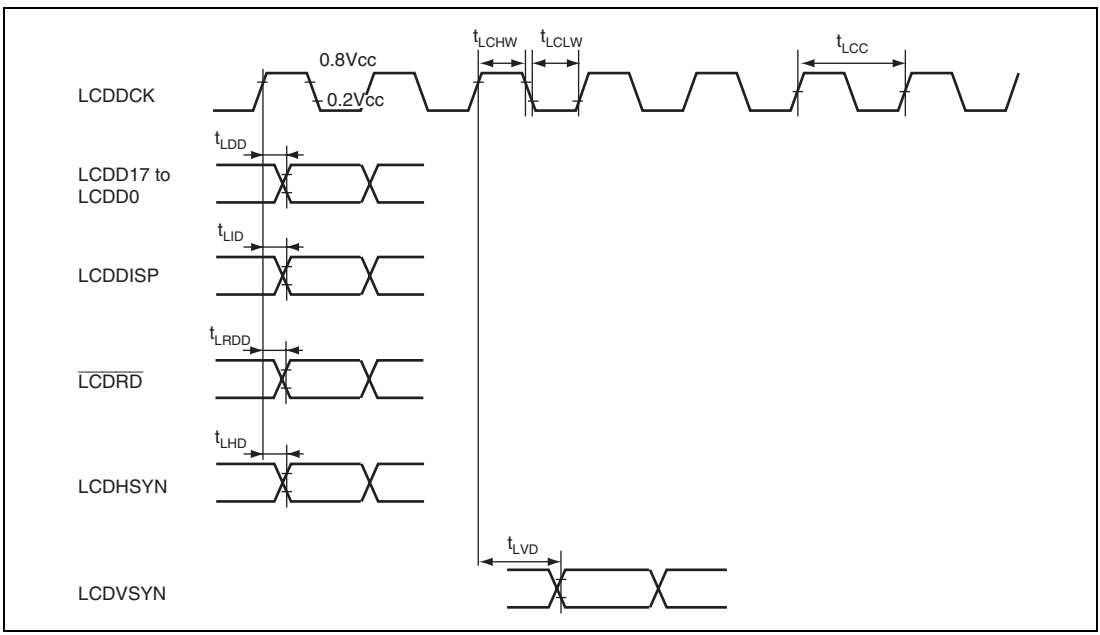


Figure 52.48 LCDC AC Characteristics

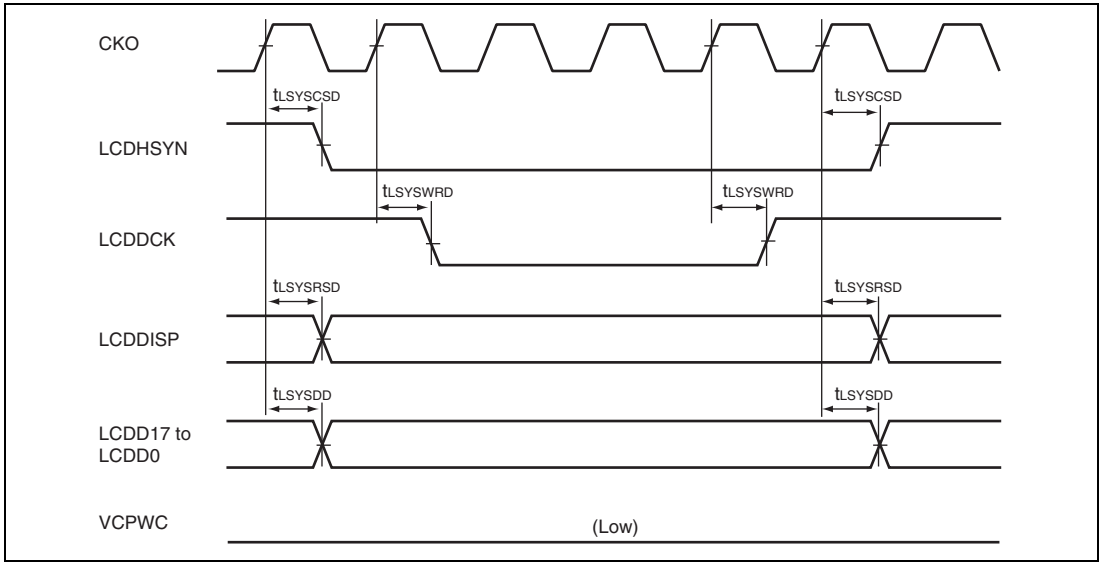


Figure 52.49 LCDC AC Characteristics SYS Interface, Command Write Bus Cycle
(MLDMT2R.WCEC = 5, MLDMT2R.WCLW = 3)

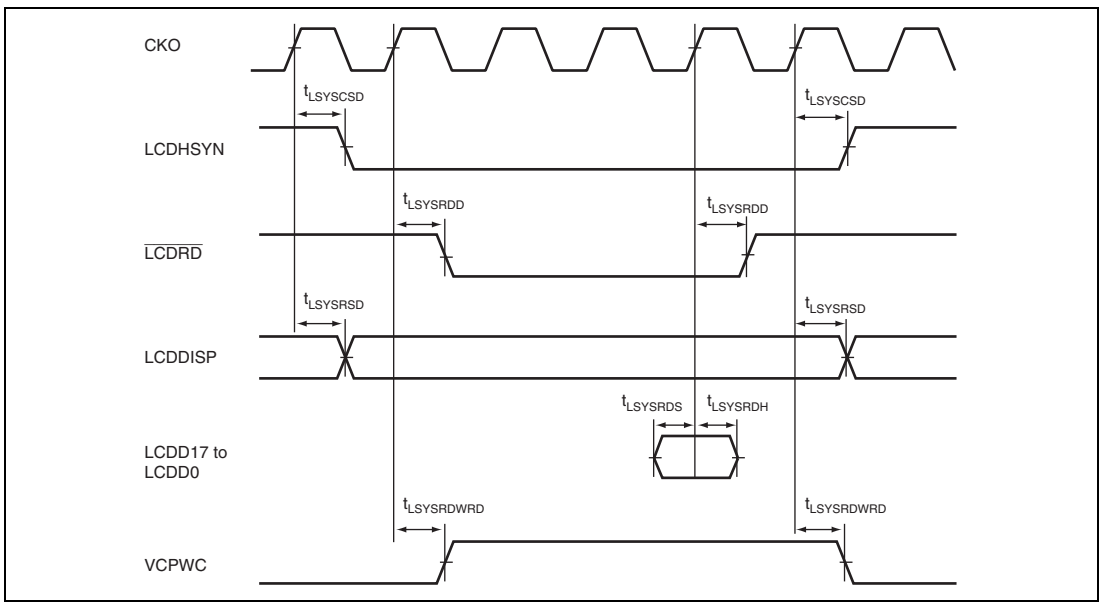


Figure 52.50 LCDC AC Characteristics SYS Interface, Command Read Bus Cycle
(MLDMT3R.RDLC = 4, MLDMT3R.RCEC = 5, MLDMT3R.RCLW = 3)

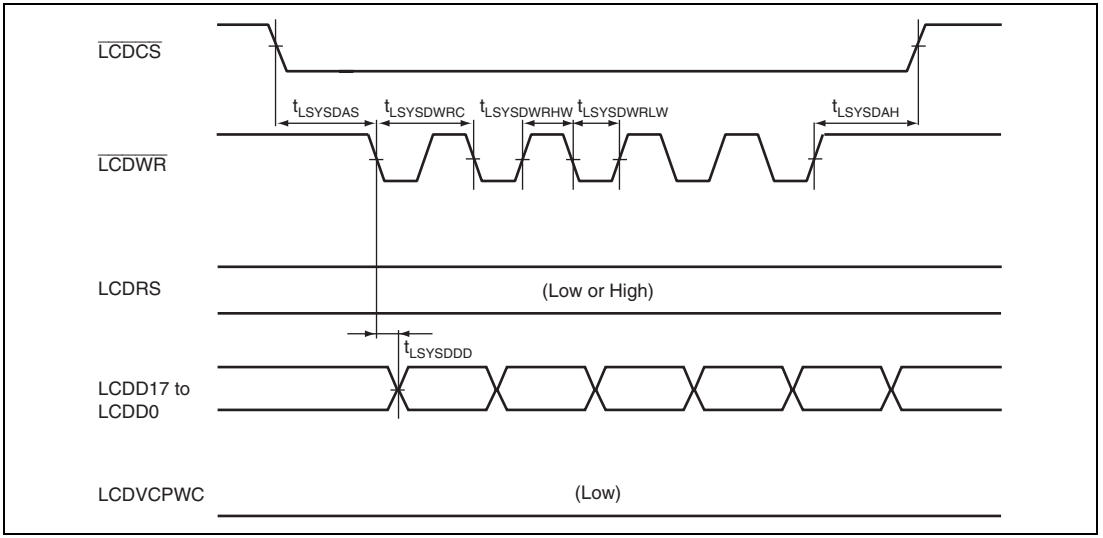


Figure 52.51 LCDC AC Characteristics (SYS Interface, Data Write Bus Cycle)

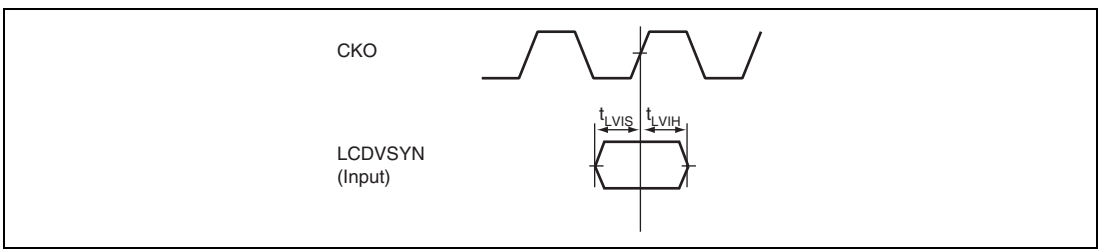


Figure 52.52 LCDC AC Characteristics (VSYNC Input Mode)

52.5.14 VOU Module Signal Timing0

Table 52.23 VOU Module Signal Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Figure
Output clock frequency	f_{px1}	13.5	—	27	MHz	52.53
Output clock cycle	t_{pxcyc1}	37	—	74.1	ns	
Output clock high width	t_{pxwH1}	14	—	—	ns	
Output clock low width	t_{pxwL1}	14	—	—	ns	
Output data delay time	t_{pxd1}	–4	—	4	ns	
Output clock frequency 2	f_{px2}	13.5	—	27	MHz	52.54
Output clock cycle 2	t_{pxcyc2}	37	—	74.1	ns	
Output clock high width 2	t_{pxwH2}	14	—	—	ns	
Output clock low width 2	t_{pxwL2}	14	—	—	ns	
Output data delay time 2	t_{pxd2}	–4	—	4	ns	

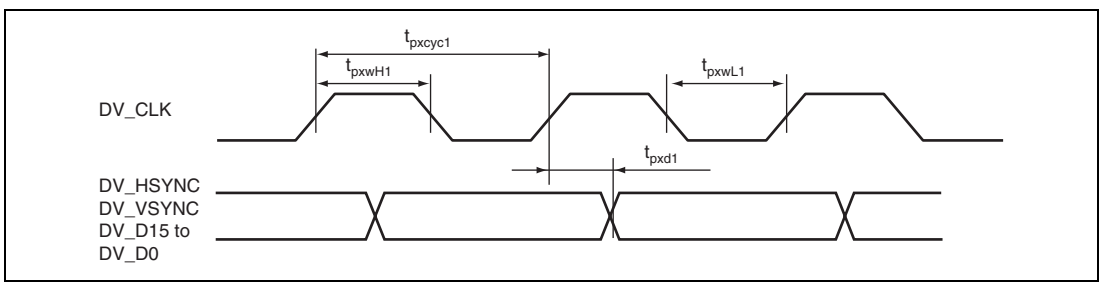


Figure 52.53 VOU AC Characteristics (VOUCR.CKPL = 0)

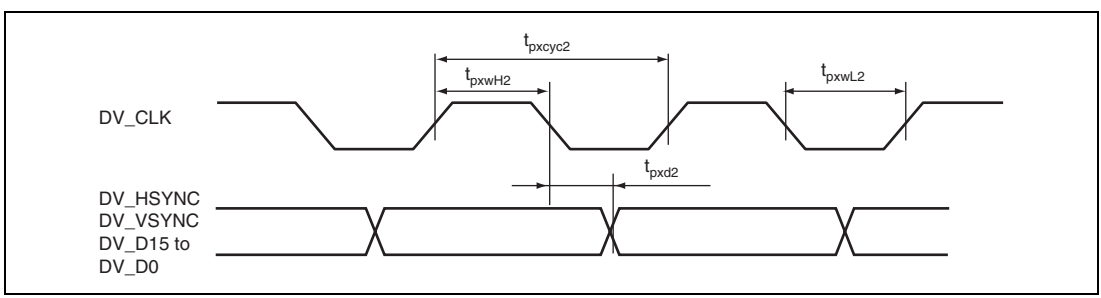


Figure 52.54 VOU AC Characteristics (VOUCR.CKPL = 1)

52.5.15 TSIF Module Signal Timing

Table 52.24 TSIF Module Signal Timing

Item		Symbol	Min.	Max.	Unit	Figure
TSIF input clock cycle	Bϕ ≥ 40 MHz	t_{TSCYC}	25	—	ns	52.55
	Bϕ < 40 MHz	t_{TSCYC}	t_{bcyc}*	—	ns	
TSIF input clock high width		t_{TSHW}	0.4 × t_{TSCYC}	—	ns	
TSIF input clock low width		t_{TSLW}	0.4 × t_{TSCYC}	—	ns	
TSIF input data setup time		t_{TSDTS}	5	—	ns	
TSIF input data hold time		t_{TSDTH}	5	—	ns	
TSIF input data enable signal setup time		t_{TSDES}	5	—	ns	
TSIF input data enable signal hold time		t_{TSDEH}	5	—	ns	
TSIF input data sync signal setup time		t_{TSSYS}	5	—	ns	
TSIF input data sync signal hold time		t_{TSSYH}	5	—	ns	

Note: * t_{bcyc} is a cycle time of an internal bus clock (Bϕ).

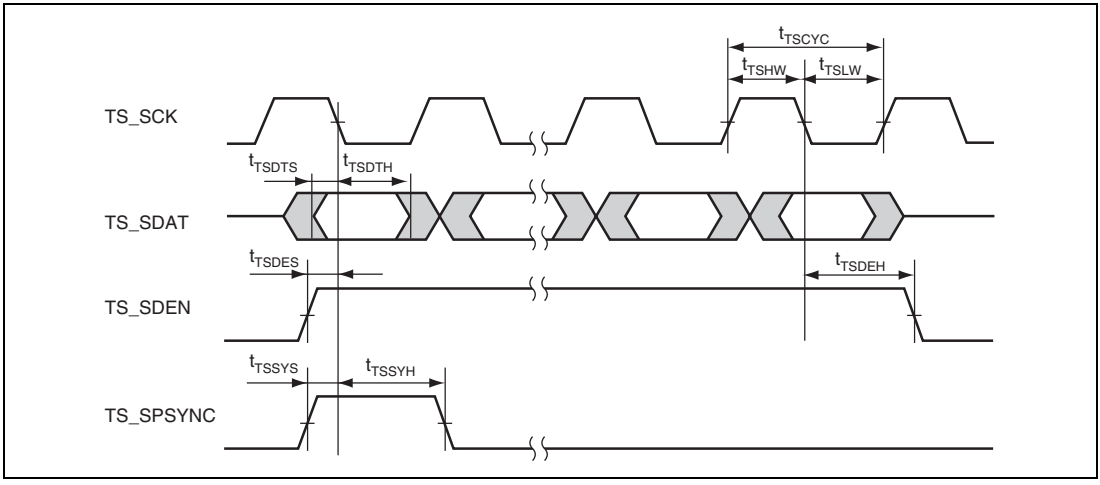
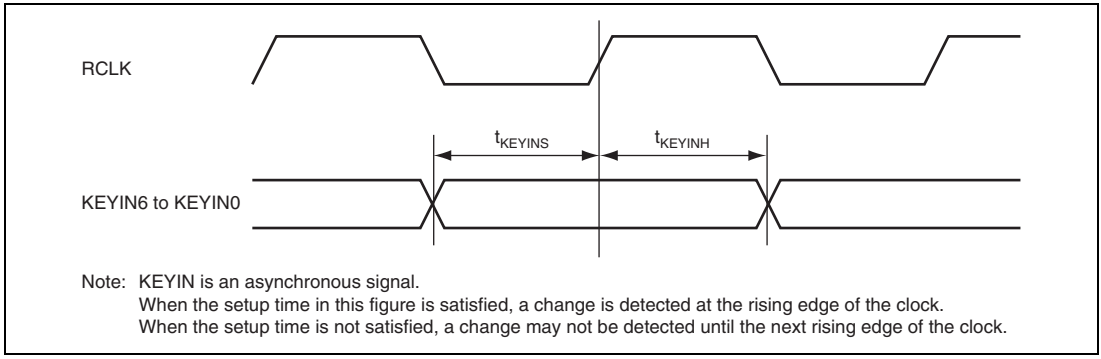


Figure 52.55 TSIF Module Signal Timing (TSCTLR.TSDATP = 0, TSCTLR.TSCLKP = 1, TSCTLR.TSVLDP = 0, TSCTLR.PSYCP = 0)

52.5.16 KEYSC Module Signal Timing

Table 52.25 KEYSC Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
KEYIN input setup time	t_{KEYINS}	15	—	ns	52.56
KEYIN input hold time	t_{KEYINH}	15	—	ns	
KEYOUT delay time	$t_{KEYOUTD}$	—	15	ns	52.57



Note: KEYIN is an asynchronous signal.
When the setup time in this figure is satisfied, a change is detected at the rising edge of the clock.
When the setup time is not satisfied, a change may not be detected until the next rising edge of the clock.

Figure 52.56 KEYIN Input Timing

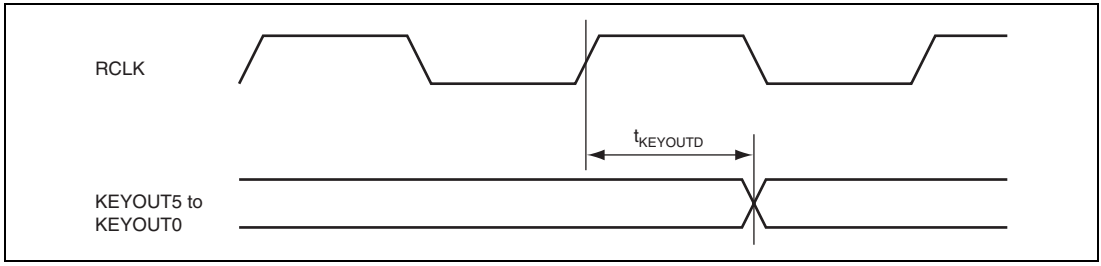


Figure 52.57 KEYOUT Output Timing

52.5.17 ATAPI Interface Module Signal Timing

Table 52.26 Symbols of PIO Transfer Timing of ATAPI Interface

Symbol	Item
t0	Cycle time
t1	Address setup time
t2	IDEIORD/IDEIOWR pulse width 8 bits
t2i	IDEIORD/IDEIOWR recovery time
t3	IDEIOWR data setup time
t4	IDEIOWR data hold time
t5	IDEIORD data setup time
t6	IDEIORD data hold time
t6z	IDEIORD3 state delay time
t9	Address hold time
tRD	IDEIORDY read data valid time
tA	IDEIORDY setup time
tB	IDEIORDY pulse time
tC	Time from IDEIORDY negation to high impedance

Table 52.27 Register Access Timing using PIO Transfer of ATAPI Interface

Test conditions: $V_{CC}Q1 = V_{CC}Q\_VIO = V_{CC}Q\_MMC = 2.7\ V$ to $3.6\ V$

	Mode 0		Mode 1		Mode 2		Mode 3		Mode 4			
Symbol	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**	**Figure**
t0	600	—	383	—	330	—	180	—	120	—	ns	52.58
t1	70	—	50	—	30	—	30	—	25	—	ns	
t2	290	—	290	—	290	—	80	—	70	—	ns	
t2i	—	—	—	—	—	—	70	—	25	—	ns	
t3	60	—	45	—	30	—	30	—	20	—	ns	
t4	30	—	20	—	15	—	10	—	10	—	ns	
t5	50	—	35	—	20	—	20	—	20	—	ns	
t6	5	—	5	—	5	—	5	—	5	—	ns	
t6z	—	30	—	30	—	30	—	30	—	30	ns	
t9	20	—	15	—	10	—	10	—	10	—	ns	
tRD	0	—	0	—	0	—	0	—	0	—	ns	
tA	35	—	35	—	35	—	35	—	35	—	ns	
tB	—	1250	—	1250	—	1250	—	1250	—	1250	ns	
tC	5	—	5	—	5	—	5	—	5	—	ns	

Table 52.28 Data Transfer Timing using PIO Transfer of ATAPI Interface

Test conditions: $V_{CC}Q1 = V_{CC}Q\_VIO = V_{CC}Q\_MMC = 2.7\ V$ to $3.6\ V$

	Mode 0		Mode 1		Mode 2		Mode 3		Mode 4			
Symbol	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**	**Figure**
t0	600	—	383	—	240	—	180	—	120	—	ns	52.58
t1	70	—	50	—	30	—	30	—	25	—	ns	
t2	290	—	290	—	290	—	80	—	70	—	ns	
t2i	—	—	—	—	—	—	70	—	25	—	ns	
t3	60	—	45	—	30	—	30	—	20	—	ns	
t4	30	—	20	—	15	—	10	—	10	—	ns	
t5	50	—	35	—	20	—	20	—	20	—	ns	
t6	5	—	5	—	5	—	5	—	5	—	ns	
t6z	—	30	—	30	—	30	—	30	—	30	ns	
t9	20	—	15	—	10	—	10	—	10	—	ns	
tRD	0	—	0	—	0	—	0	—	0	—	ns	
tA	35	—	35	—	35	—	35	—	35	—	ns	
tB	—	1250	—	1250	—	1250	—	1250	—	1250	ns	
tC	5	—	5	—	5	—	5	—	5	—	ns	

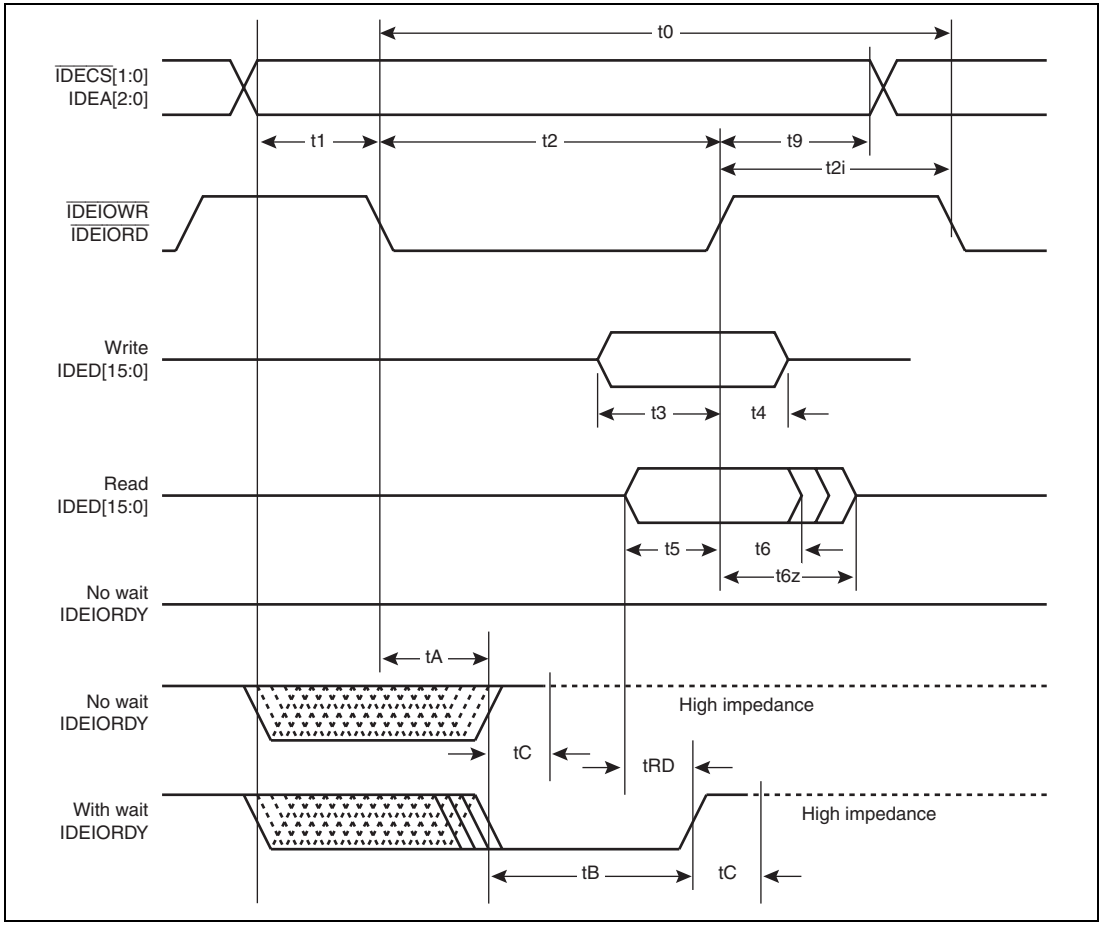


Figure 52.58 PIO Data Transfer and Register Transfer between Devices

Table 52.29 Symbols of Multiword Transfer Timing of ATAPI Interface

Symbol	Item
t0	Cycle time
tD	IDEIORD/IDEIOWR pulse width
tE	IDEIORD data access time
tF	IDEIORD data hold time
tG	IDEIORD/IDEIOWR data setup time
tH	IDEIOWR data hold time
tI	IODACK setup time
tJ	IODACK hold time
tKR	IDEIORD negate pulse width
tKW	IDEIOWR negate pulse width
tLR	IDEIORD · IODREQ delay time
tLW	IDEIOWR · IODREQ delay time
tM	IDECS[1:0] setup time
tN	IDECS[1:0] hold time
tZ	IODACK3 state delay time

Table 52.30 Multiword Transfer Timing of ATAPI Interface

Test conditions: $V_{CC}Q1 = V_{CC}Q\_VIO = V_{CC}Q\_MMC = 2.7\ V$ to $3.6\ V$

	Mode 0		Mode 1		Mode 2			
Symbol	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**	**Figure**
t0	480	—	150	—	120	—	ns	52.60 to 52.62
tD	215	—	80	—	70	—	ns	52.59 to 52.62
tE	—	150	—	60	—	50	ns	
tF	5	—	5	—	5	—	ns	
tG	100	—	30	—	20	—	ns	
tH	20	—	15	—	10	—	ns	
tI	0	—	0	—	0	—	ns	52.59
tJ	20	—	5	—	5	—	ns	52.61, 52.62
tKR	50	—	50	—	25	—	ns	52.60 to 52.62
tKW	215	—	50	—	25	—	ns	
tLR	—	120	—	40	—	35	ns	52.61
tLW	—	40	—	40	—	35	ns	
tM	50	—	30	—	25	—	ns	
tN	15	—	10	—	10	—	ns	52.61, 52.62
tZ	—	20	—	25	—	25	ns	

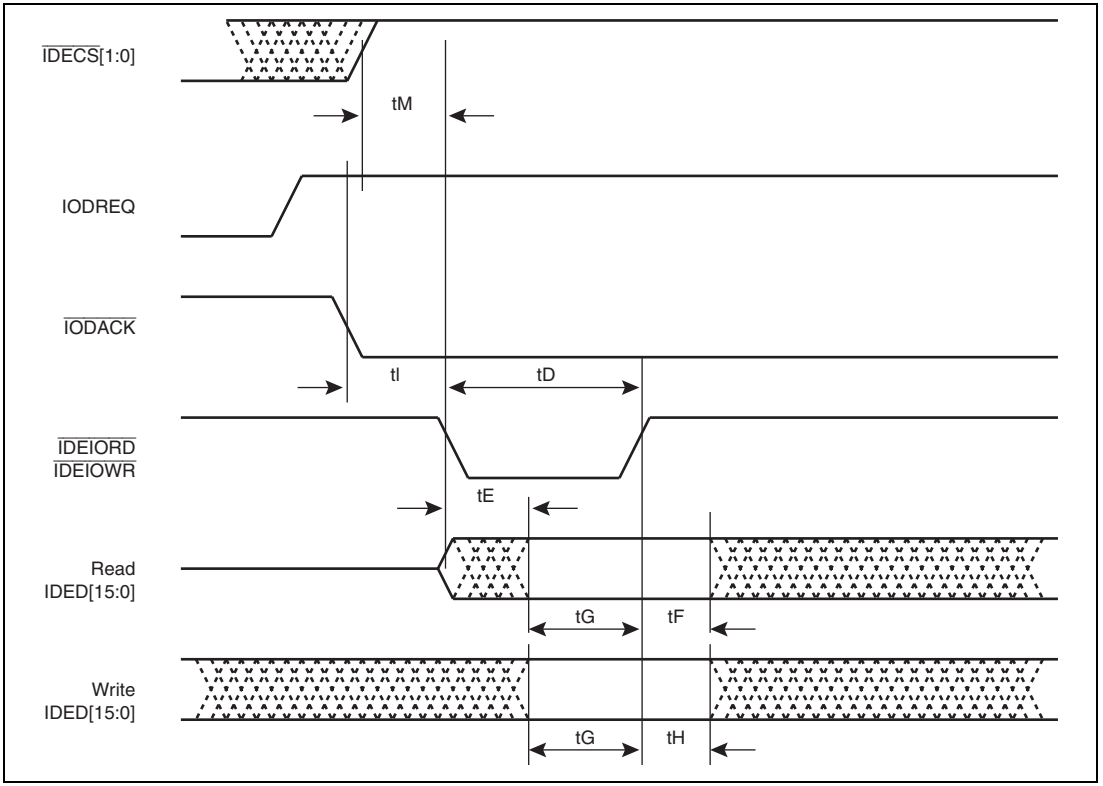


Figure 52.59 Start of Multiword DMA Data Transfer

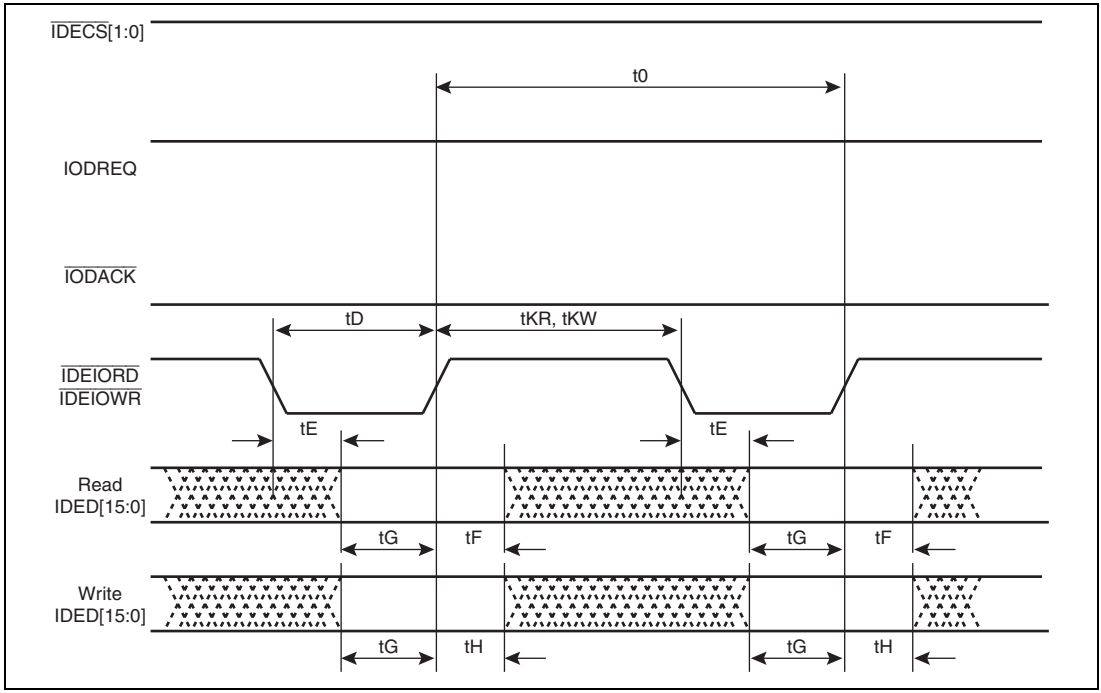


Figure 52.60 Multiword DMA Data Transfer

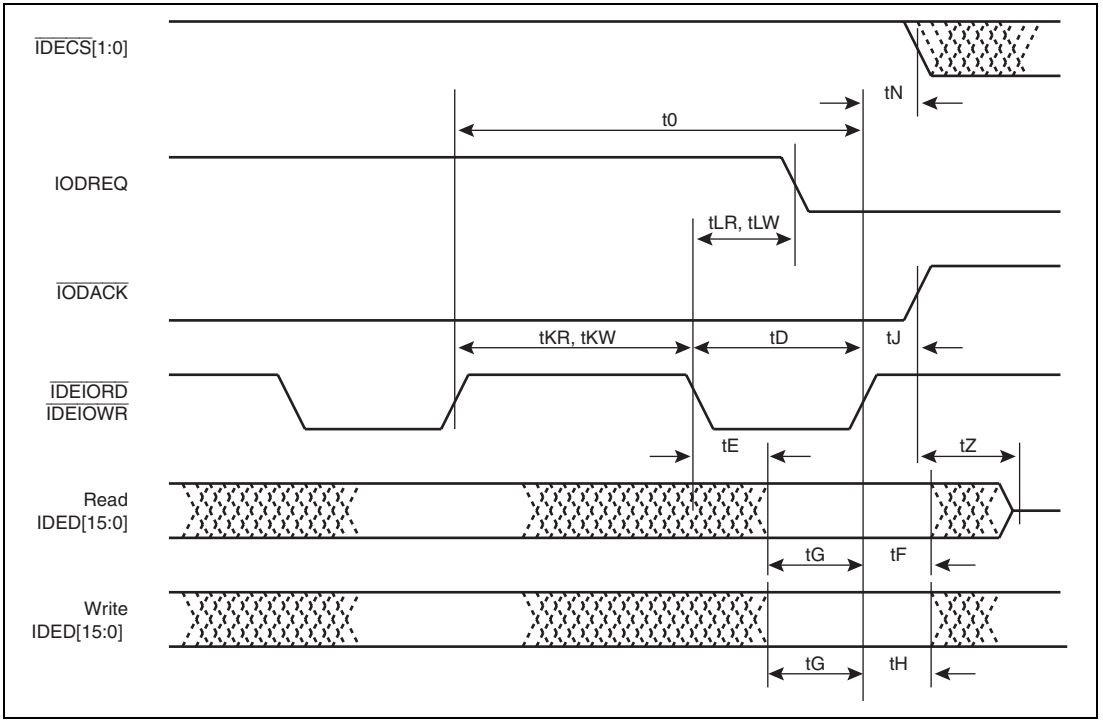


Figure 52.61 End of Multiword DMA Data Transfer from Device

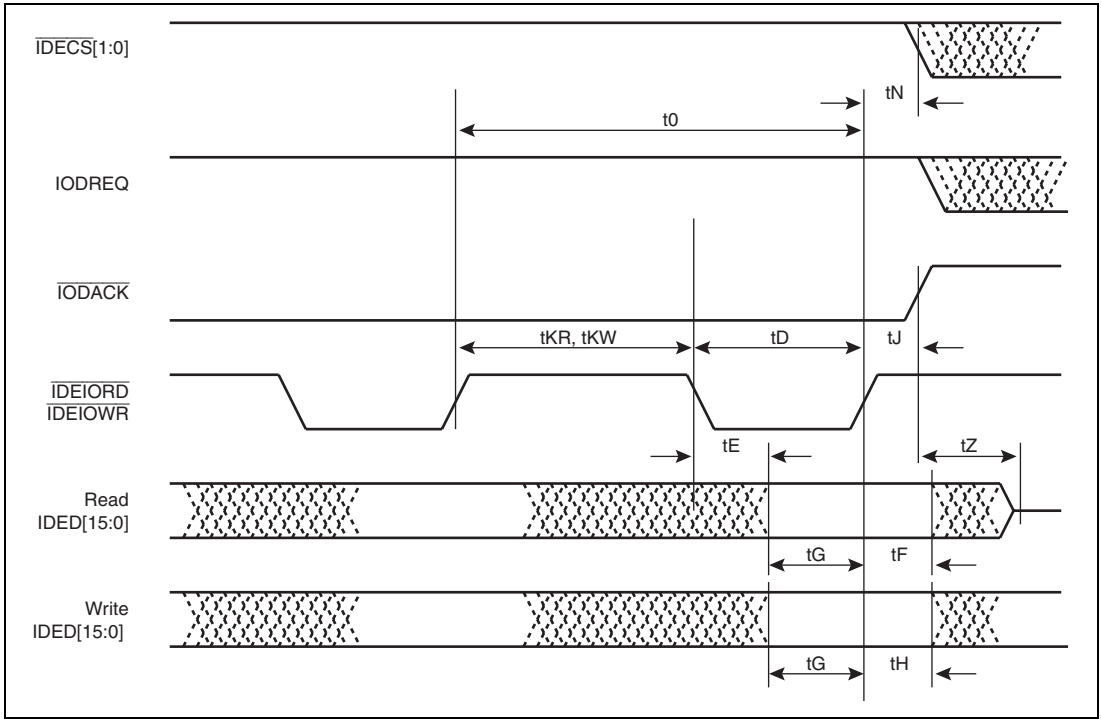


Figure 52.62 End of Multiword DMA Data Transfer from Host

Table 52.31 Symbols of Ultra DMA Transfer Timing of ATAPI Interface

Symbol	Item
t2CYCTYP	Average cycle time (two cycles)
tCYC	Cycle time
t2CYC	Minimum cycle time (two cycles)
tDS	Data setup time (receiver side)
tDH	Data hold time (receiver side)
tDVS	Data setup time (transmitter side)
tDVH	Data hold time (transmitter side)
tCS	CRC data setup time (receiver side)
tCH	CRC data hold time (receiver side)
tCVS	CRC data setup time (transmitter side)
tCVH	CRC data hold time (transmitter side)
tZFS	Setup time from drive of strobe to first STROBE (transmitter side)
tDZFS	Setup time from drive of data to first STROBE (transmitter side)
tFS	First STROBE time
tLI	Limited interlock time
tMLI	Minimum interlock time
tUI	Unlimited interlock time
tAZ	Output release time
tZAH	Output delay time
tZAD	Output determination time (from release)
tENV	Envelope time
tRFS	Last STROBE time
tRP	Time until STOP is asserted or DMARQ is negated
tIORDYZ	Time until IORDY is released
tZIORDY	Time until STROBE is driven
tACK	DMACK setup/hold time
tSS	Strobe stop time

Table 52.32 Ultra DMA Transfer Timing of ATAPI Interface

Test conditions: $V_{CC}Q1 = V_{CC}Q\_VIO = V_{CC}Q\_MMC = 2.7\ V$ to $3.6\ V$

	Mode 0		Mode 1		Mode 2		Mode 3		Mode 4			
Symbol	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**	**Figure**
t2CYCTYP	240	—	160	—	120	—	90	—	60	—	ns	52.64
tCYC	112	—	73	—	54	—	39	—	25	—	ns	52.64, 52.68
t2CYC	230	—	153	—	115	—	86	—	57	—	ns	
tDS	15	—	10	—	7	—	7	—	5	—	ns	
tDH	5	—	5	—	5	—	5	—	5	—	ns	
tDVS	70	—	48	—	31	—	20	—	6.7	—	ns	52.63, 52.64, 52.68, 52.69
tDVH	6.2	—	6.2	—	6.2	—	6.2	—	6.2	—	ns	
tCS	15	—	10	—	7	—	7	—	5	—	ns	
tCH	5	—	5	—	5	—	5	—	5	—	ns	
tCVS	70	—	48	—	31	—	20	—	6.7	—	ns	52.64, 52.67, 52.71, 52.72
tCVH	6.2	—	6.2	—	6.2	—	6.2	—	6.2	—	ns	
tZFS	0	—	0	—	0	—	0	—	0	—	ns	52.61
tDZFS	70	—	48	—	31	—	20	—	6.7	—	ns	52.61, 52.68
tFS	—	230	—	200	—	170	—	130	—	120	ns	52.61
tLI	0	150	0	150	0	150	0	100	0	100	ns	52.66 to 52.70, 52.71, 52.72
tMLI	20	—	20	—	20	—	20	—	20	—	ns	52.68, 52.67, 52.71, 52.72
tUI	0	—	0	—	0	—	0	—	0	—	ns	52.61, 52.68

	Mode 0		Mode 1		Mode 2		Mode 3		Mode 4			
Symbol	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Min.**	**Max.**	**Unit**	**Figure**
tAZ	—	10	—	10	—	10	—	10	—	10	ns	52.63, 52.66, 52.67
tZAH	20	—	20	—	20	—	20	—	20	—	ns	52.66, 52.67
tZAD	0	—	0	—	0	—	0	—	0	—	ns	52.63
tENV	20	70	20	70	20	70	20	55	20	55	ns	52.63, 52.68
tRFS	—	75	—	70	—	60	—	60	—	60	ns	52.65, 52.67, 52.70, 52.72
tRP	160	—	125	—	100	—	100	—	100	—	ns	
tIORDYZ	—	20	—	20	—	20	—	20	—	20	ns	52.66, 52.65, 52.71, 52.72
tZIORDY	0	—	0	—	0	—	0	—	0	—	ns	52.63, 52.68
tACK	20	—	20	—	20	—	20	—	20	—	ns	52.63, 52.66 to 52.68, 52.71, 52.72
tSS	50	—	50	—	50	—	50	—	50	—	ns	52.66, 52.71

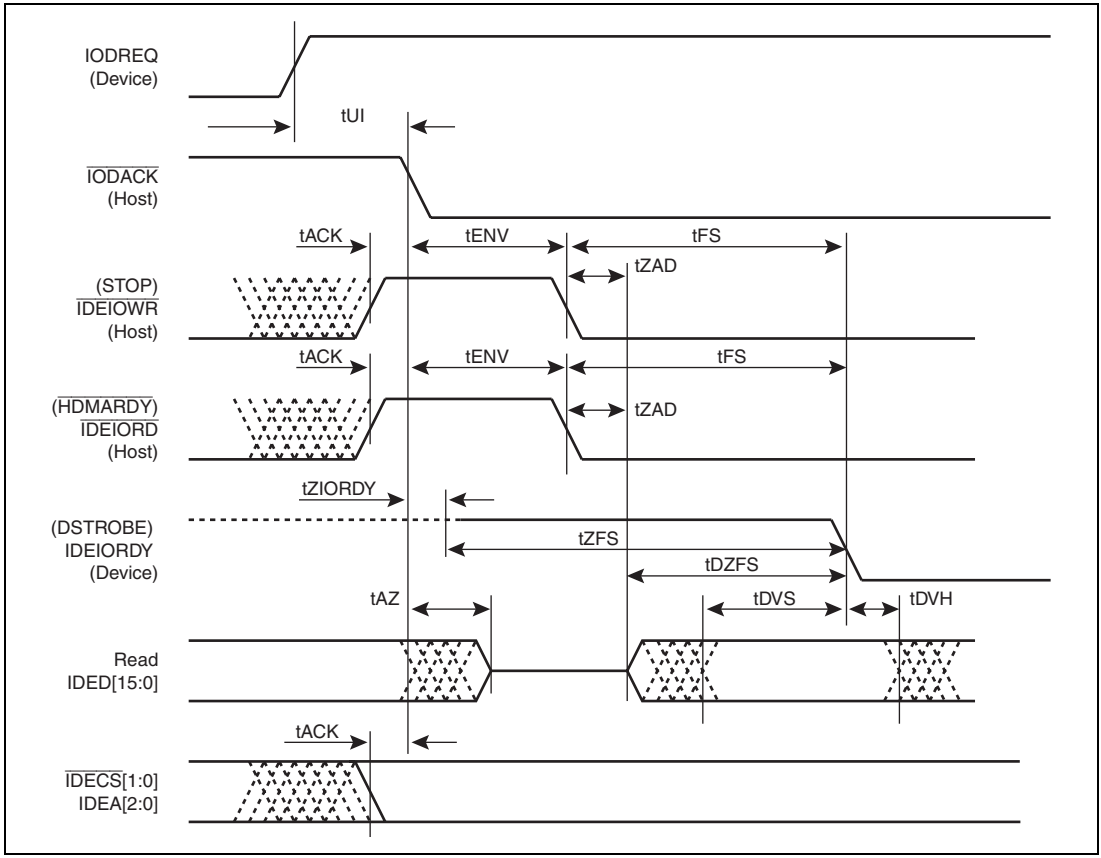


Figure 52.63 Start of Ultra DMA Transfer Data In Burst

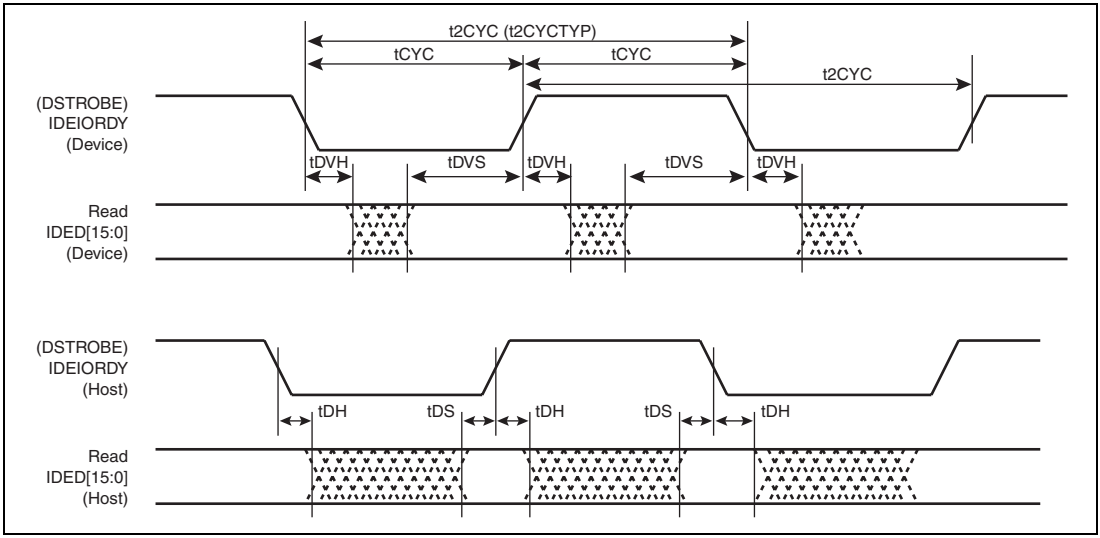


Figure 52.64 Ultra DMA Transfer Data In Burst

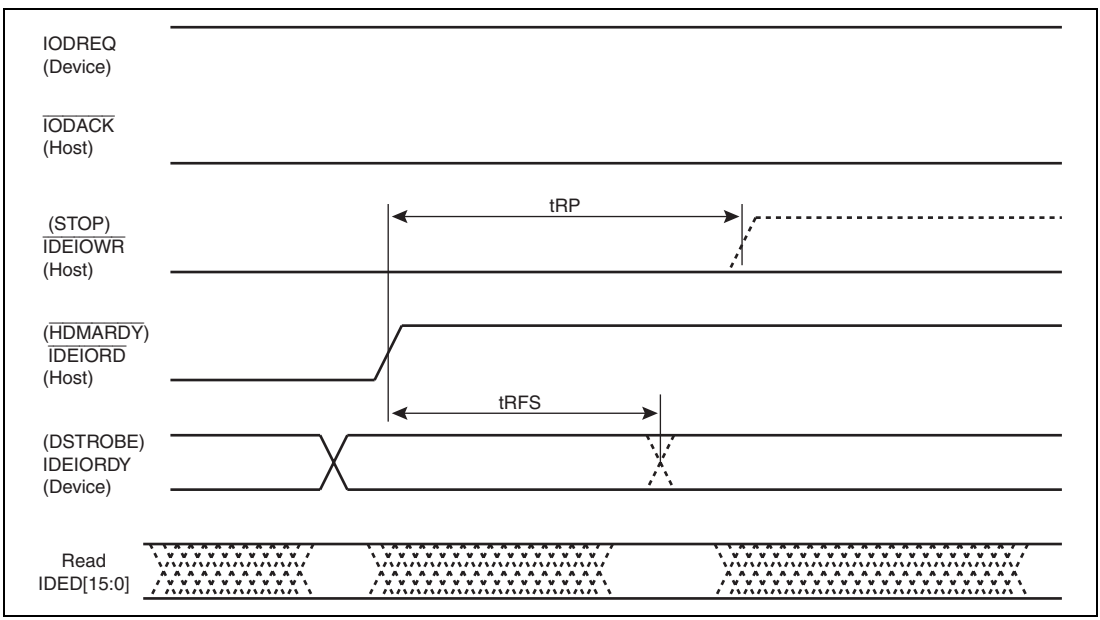


Figure 52.65 Pause of Ultra DMA Transfer Data In Burst from Host

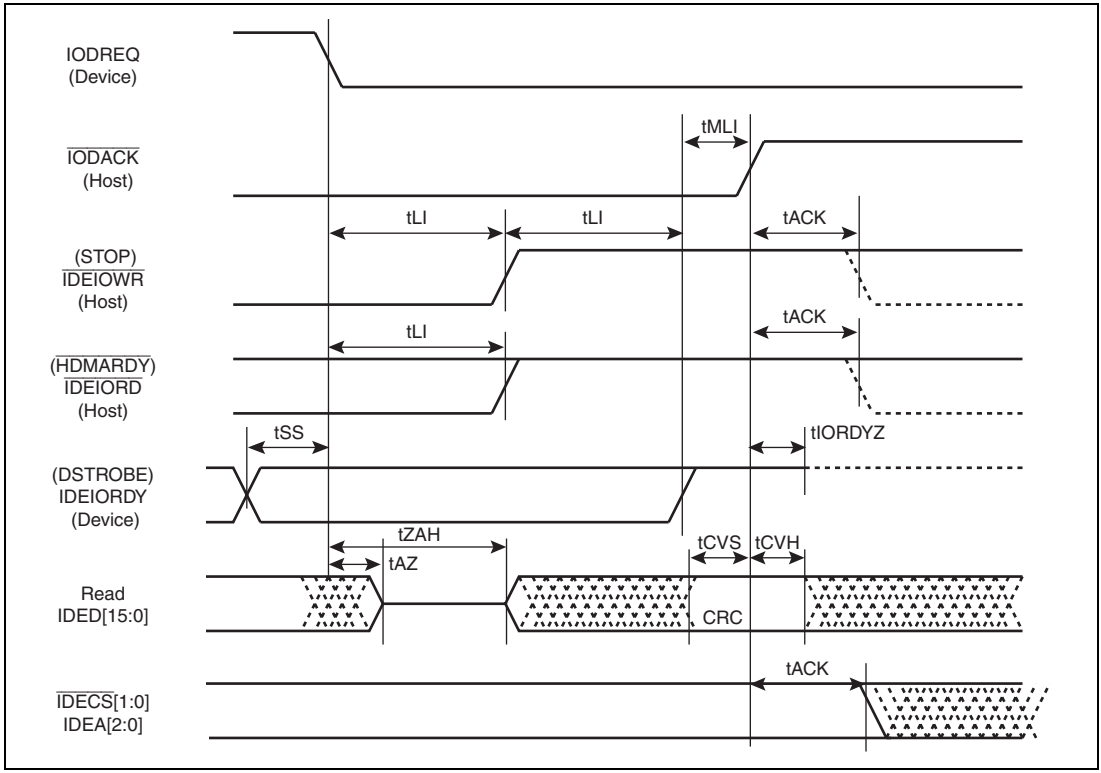


Figure 52.66 End of Ultra DMA Transfer Data In Burst from Device

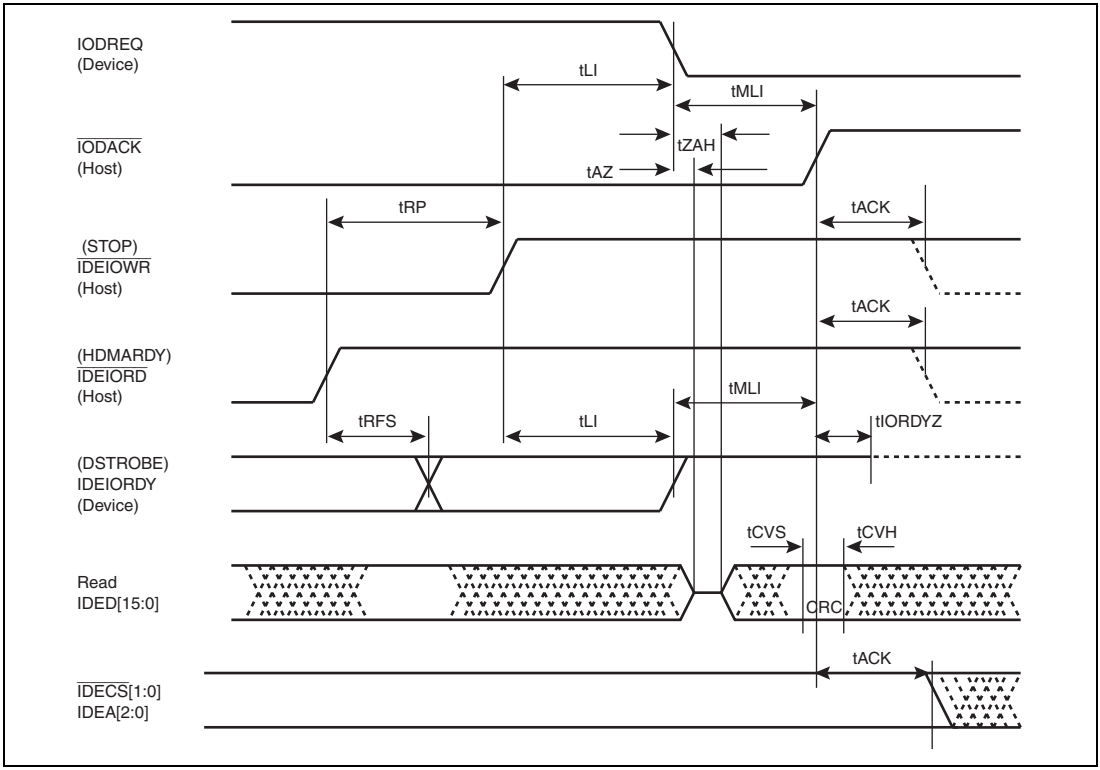


Figure 52.67 End of Ultra DMA Transfer Data In Burst from Host

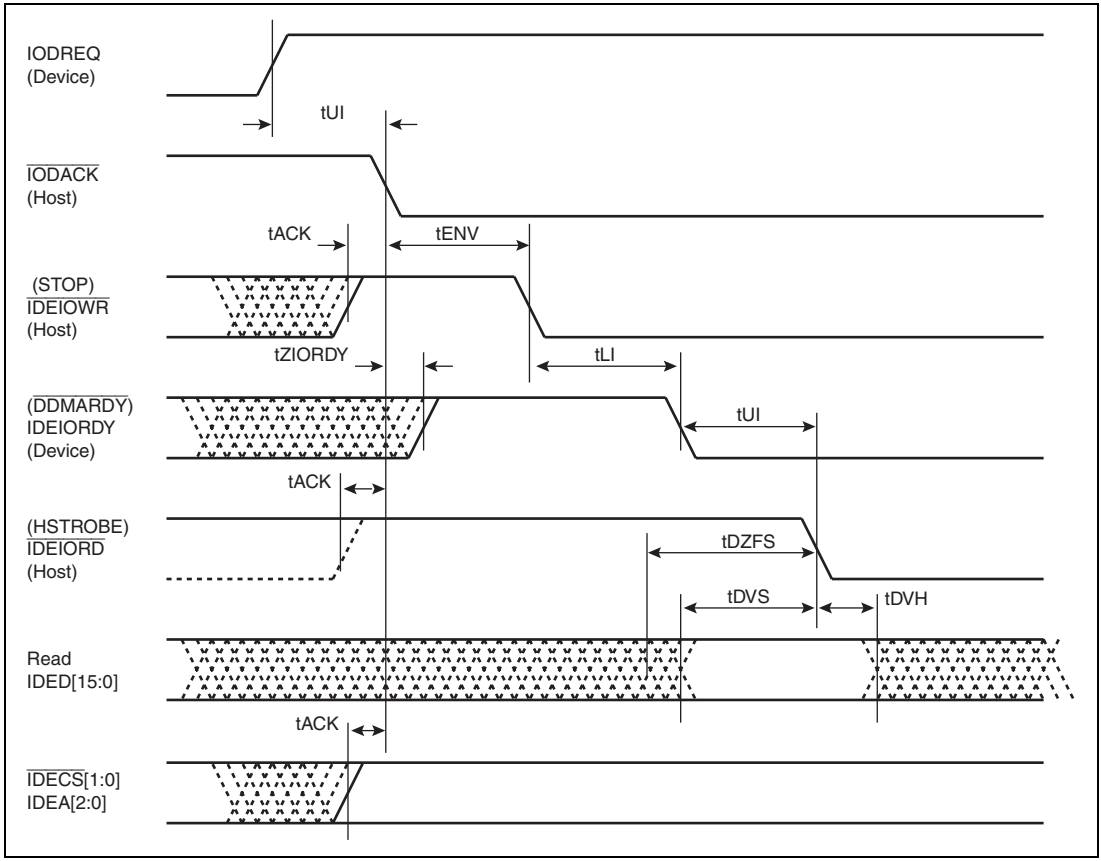


Figure 52.68 Start of Ultra DMA Transfer Data Out Burst

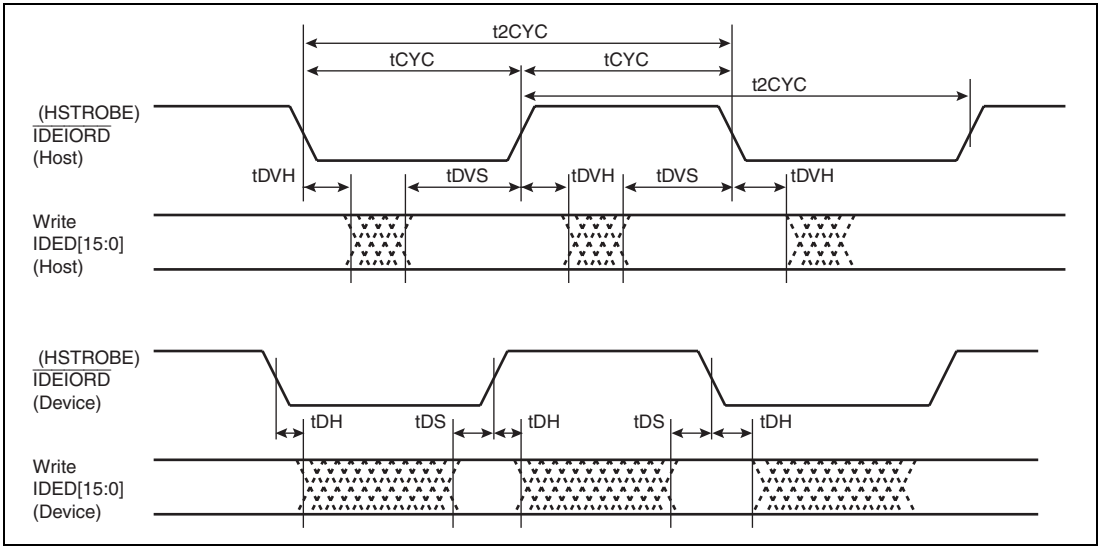


Figure 52.69 Ultra DMA Transfer Data Out Burst

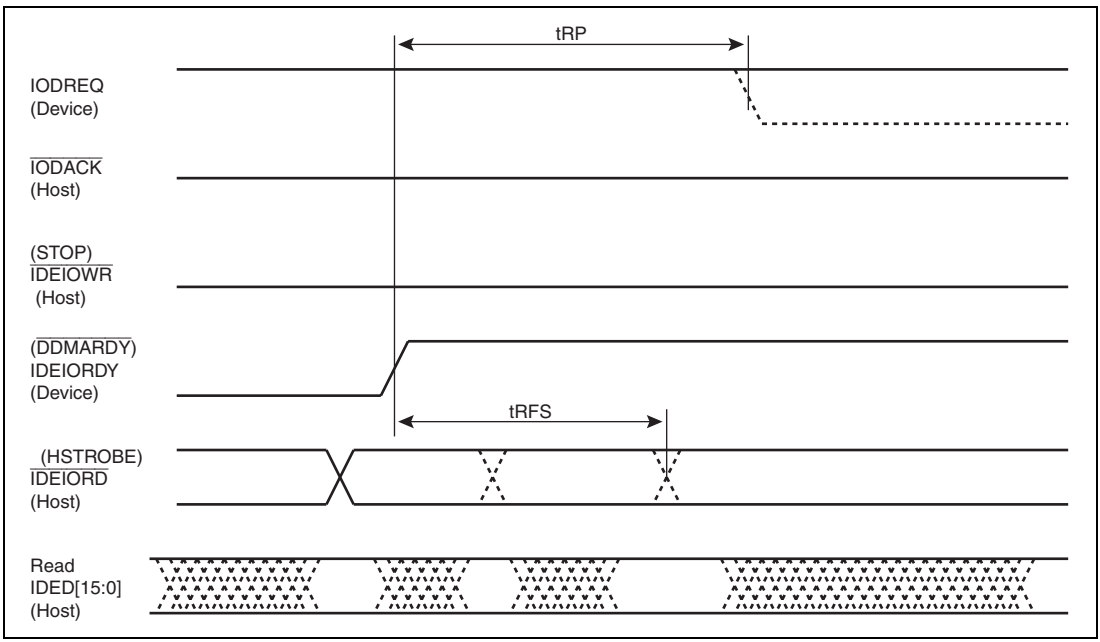


Figure 52.70 Pause of Ultra DMA Transfer Data Out Burst from Device

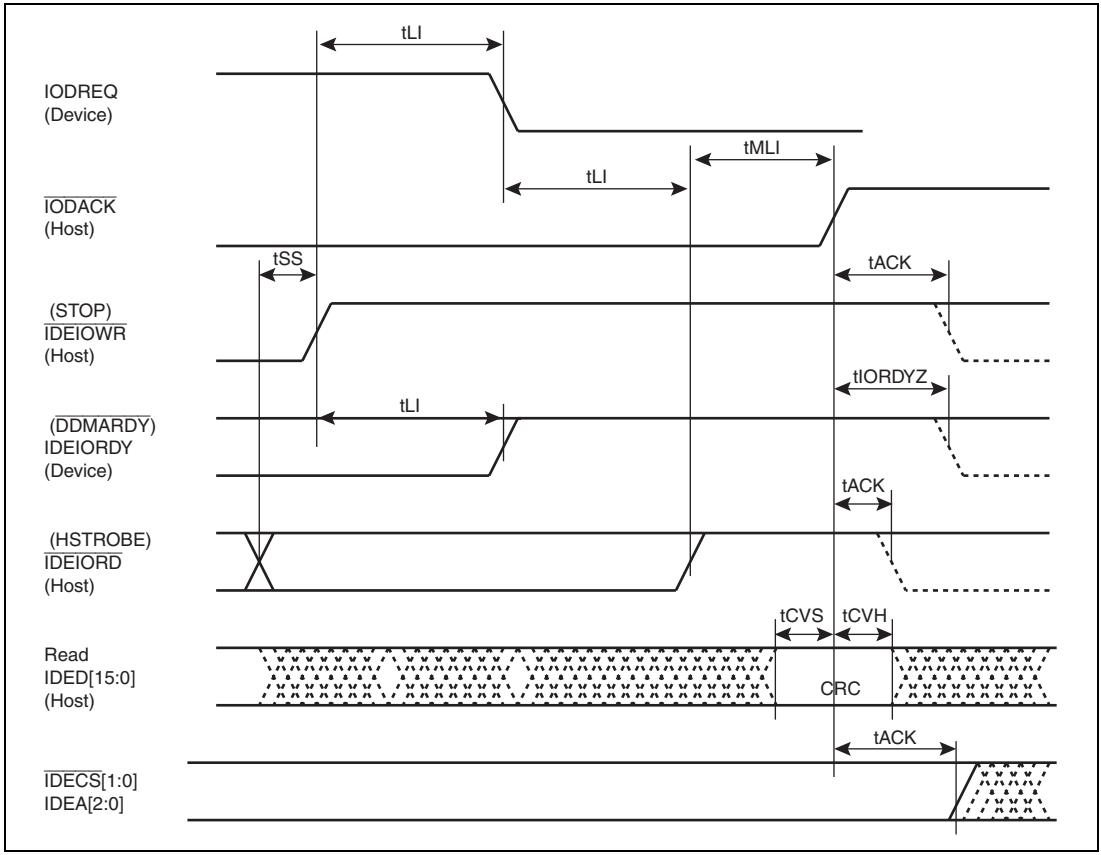


Figure 52.71 End of Ultra DMA Transfer Data Out Burst from Host

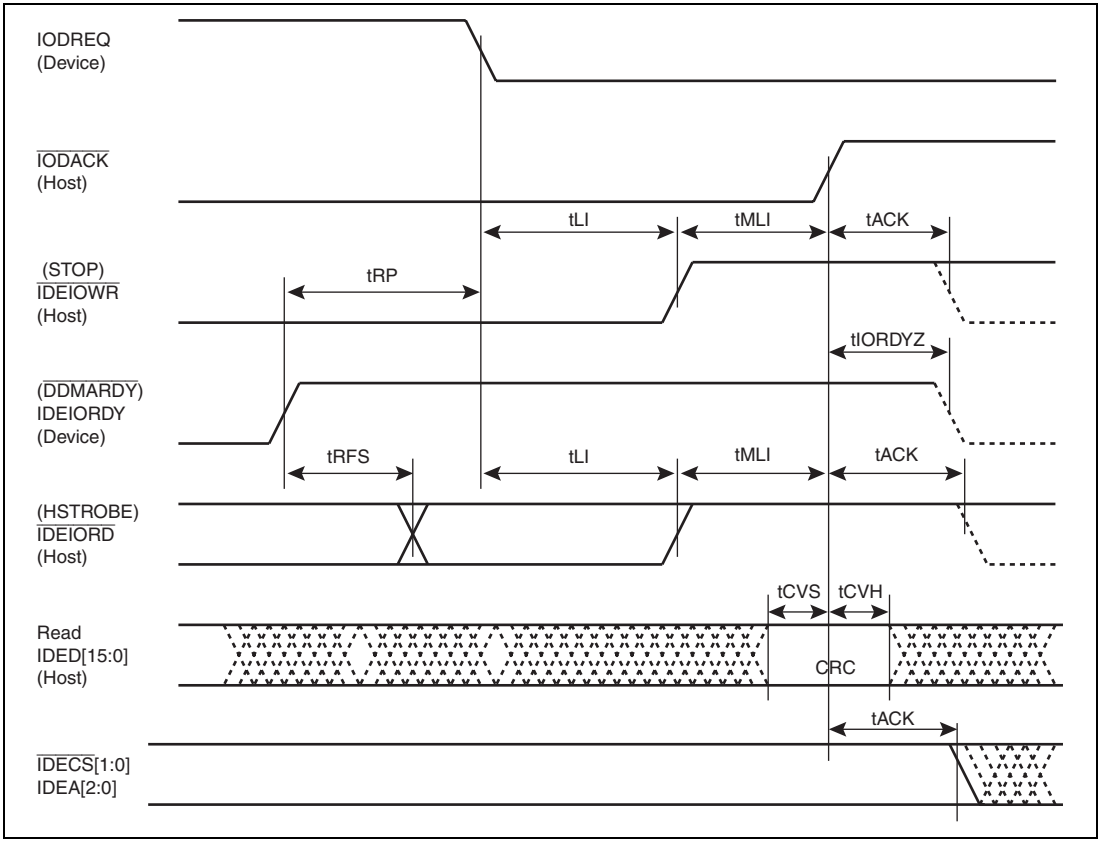


Figure 52.72 End of Ultra DMA Transfer Data Out Burst from Device

Table 52.33 Symbols of DIRECTION Timing of ATAPI Interface

Symbol	Item
tDIRECTION_WF	DIRECTION fall delay time on PIO writing
tDIRECTION_WR	DIRECTION rise delay time on PIO writing
tMDIRECTION_F	Multiword DMA data out DIRECTION fall delay time
tMDIRECTION_R	Multiword DMA data out DIRECTION rise delay time
tUDIRECTION_F (CRC)	DIRECTION fall delay time on ultra DMA data in CRC transmission
tUDIRECTION_R (CRC)	DIRECTION rise delay time on ultra DMA data in CRC transmission
tUDIRECTION_F	DIRECTION fall delay time on ultra DMA data out
tUDIRECTION_R	DIRECTION rise delay time on ultra DMA data out
tDON	Time from fall of DIRECTION to turning on IDED data bus
tDOFF	Time from turning off IDED data bus to rise of DIRECTION

Table 52.34 DIRECTION Timing of ATAPI Interface

Test conditions: $V_{CC}Q1 = V_{CC}Q\_VIO = V_{CC}Q\_MMC = 2.7$ V to 3.6 V

Item and Symbol	Mode 0		Mode 1		Mode 2		Mode 3		Mode 4		Unit	Figure
	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.		
tDIRECTION_WF	79	87	56	65	34	42	34	42	34	42	ns	52.73
tDIRECTION_WR	63	71	63	71	33	41	33	41	33	41	ns	
tMDIRECTION_F	–19	–11	–19	–11	–19	–11	—	—	—	—	ns	52.75
tMDIRECTION_R	3	12	3	12	3	12	—	—	—	—	ns	
tUDIRECTION_F(CRC)	138	147	101	109	86	94	71	79	56	64	ns	52.77, 52.78
tUDIRECTION_R(CRC)	26	34	26	34	26	34	26	34	26	34	ns	
tUDIRECTION_F	48	57	48	57	48	57	48	57	48	57	ns	52.79
tUDIRECTION_R	56	64	56	64	56	64	56	64	56	64	ns	52.80, 52.81
tDON	9	15	9	15	9	15	9	15	18	22	ns	52.73, 52.75, 52.77 to 52.79
tDOFF	6	14	6	14	6	14	6	14	6	14	ns	52.73, 52.75, 52.77, 52.78, 52.80, 52.81

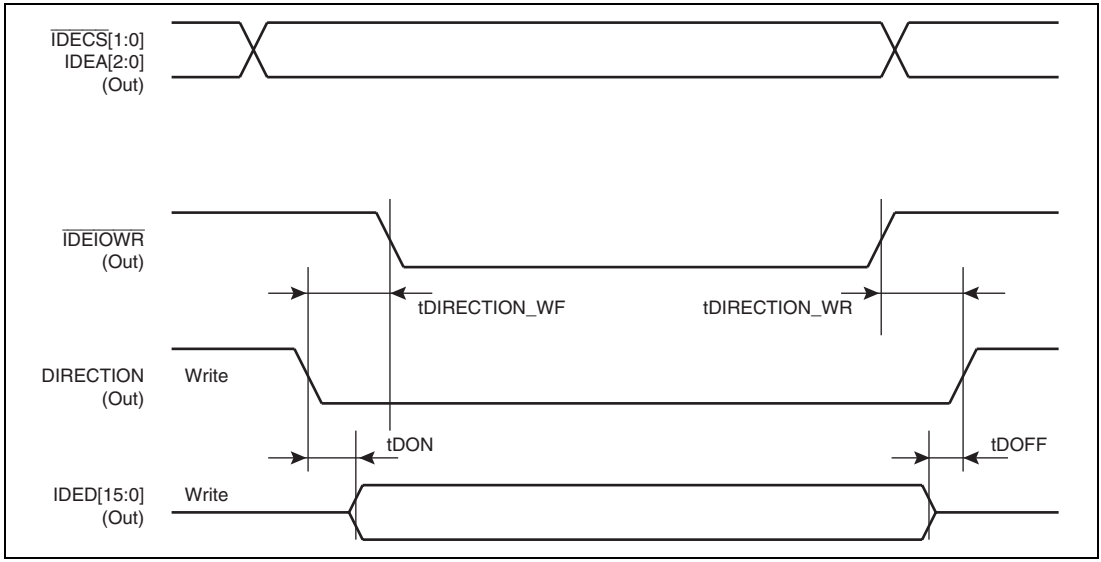


Figure 52.73 PIO Data Transfer to Device (DIRECTION)

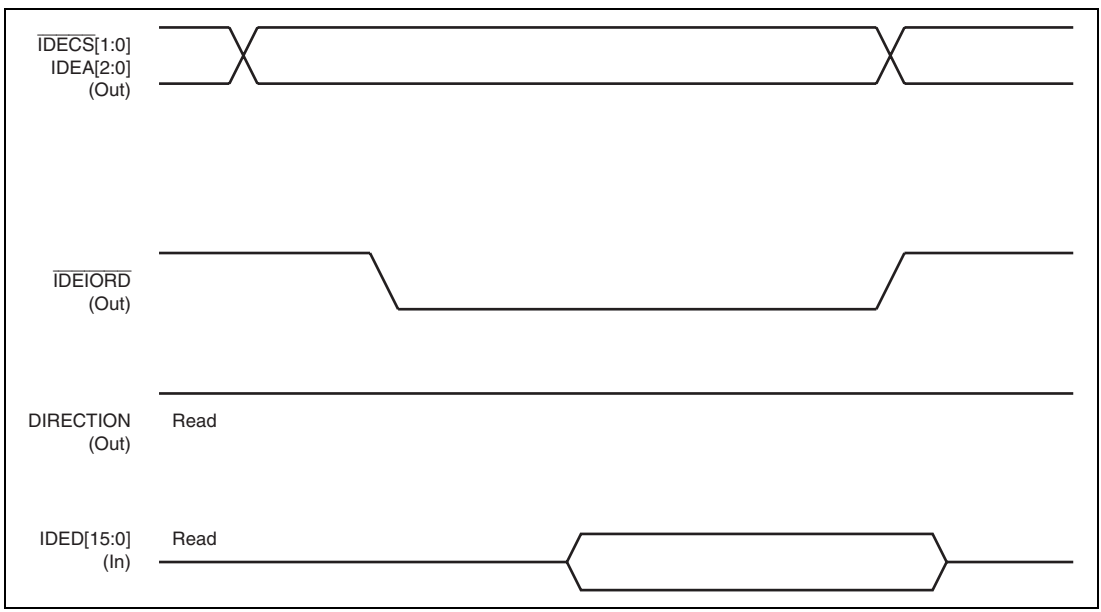


Figure 52.74 PIO Data Transfer from Device (DIRECTION)

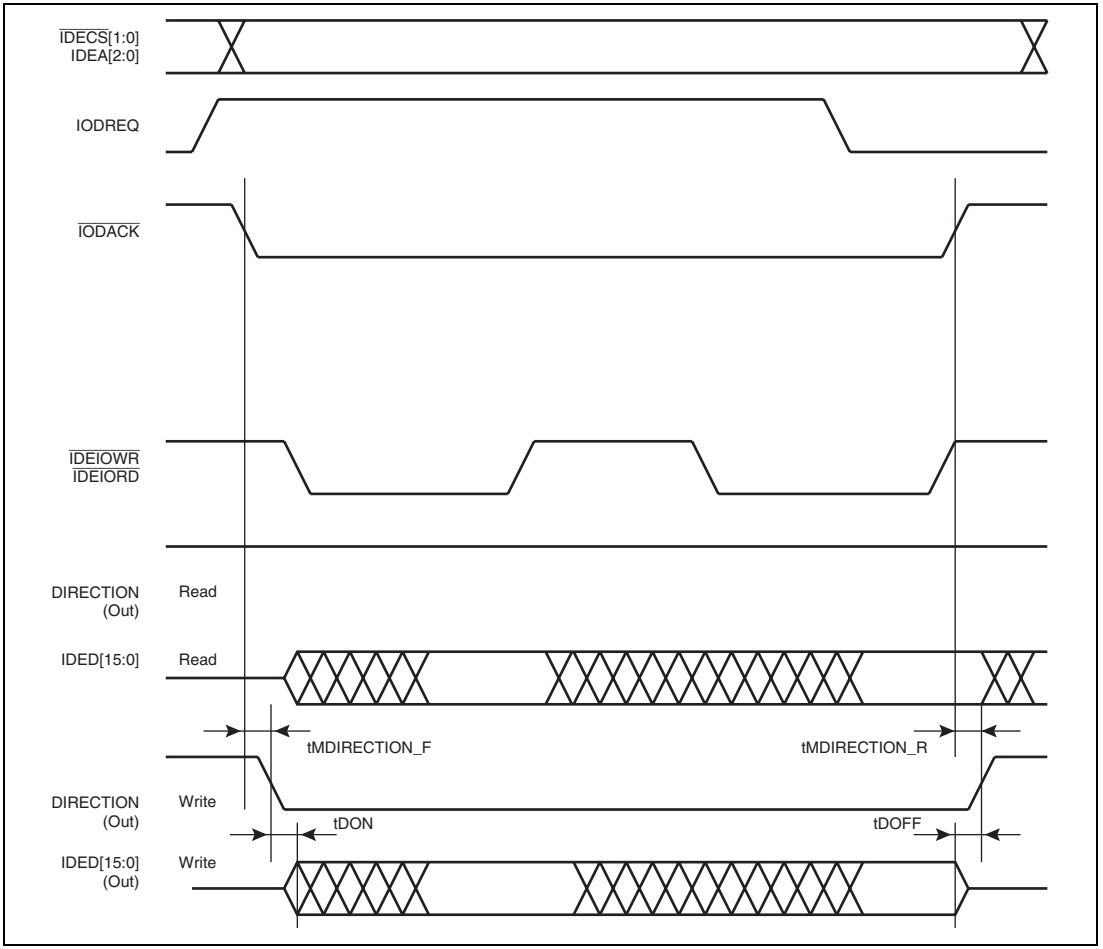


Figure 52.75 Multiword DMA Transfer (DIRECTION)

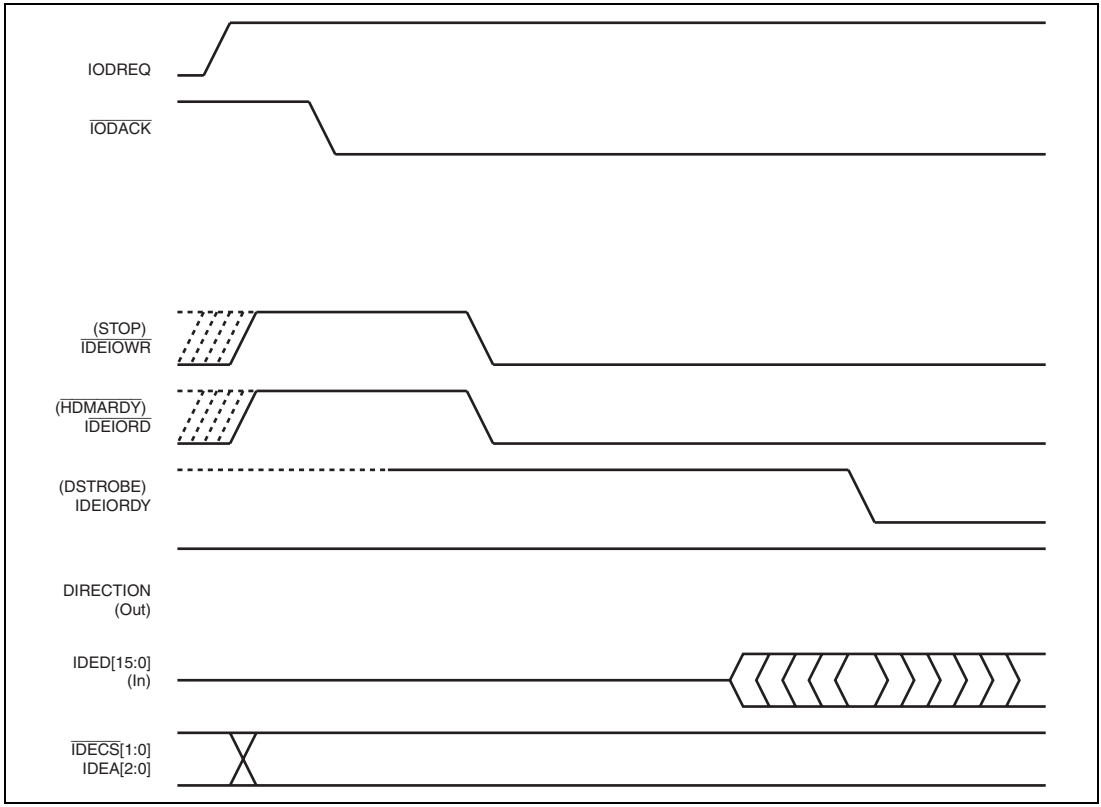


Figure 52.76 Start of Ultra DMA Transfer Data In Burst (DIRECTION)

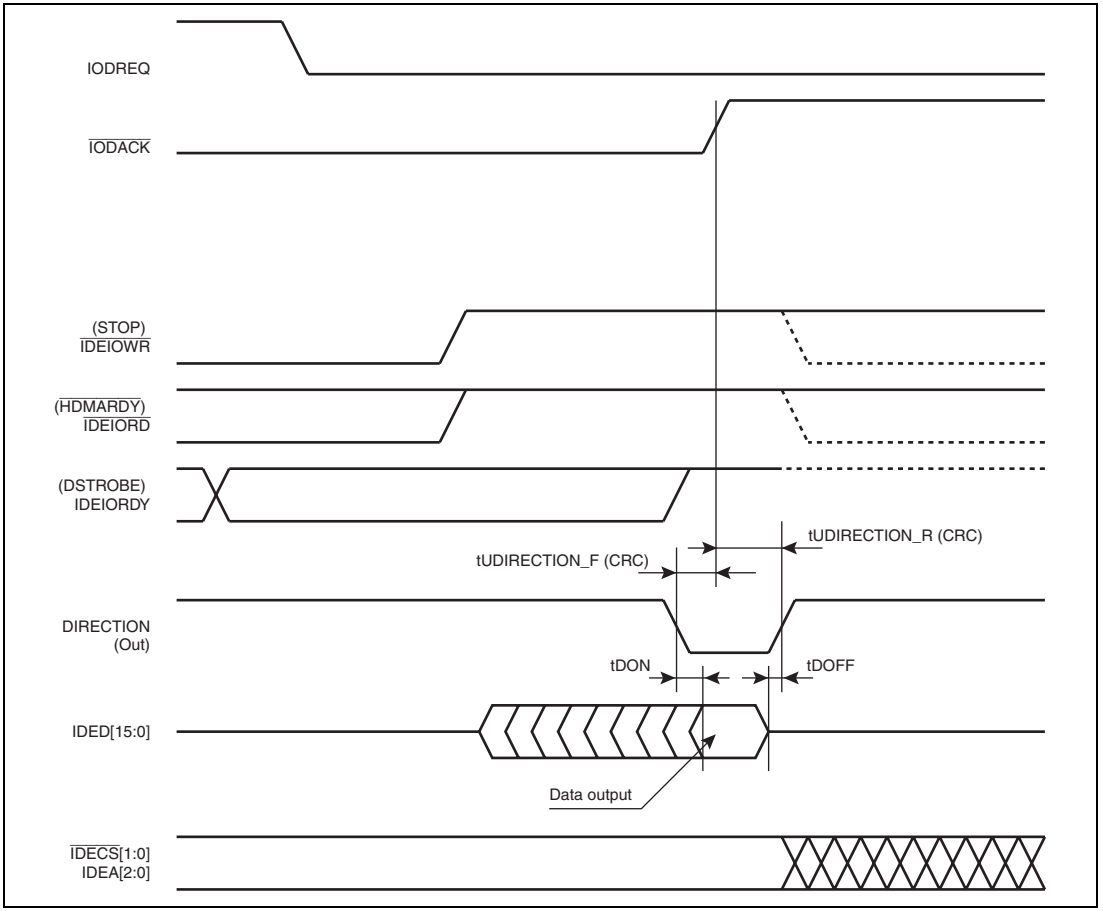


Figure 52.77 End of Ultra DMA Transfer Data In Burst from Device (DIRECTION)

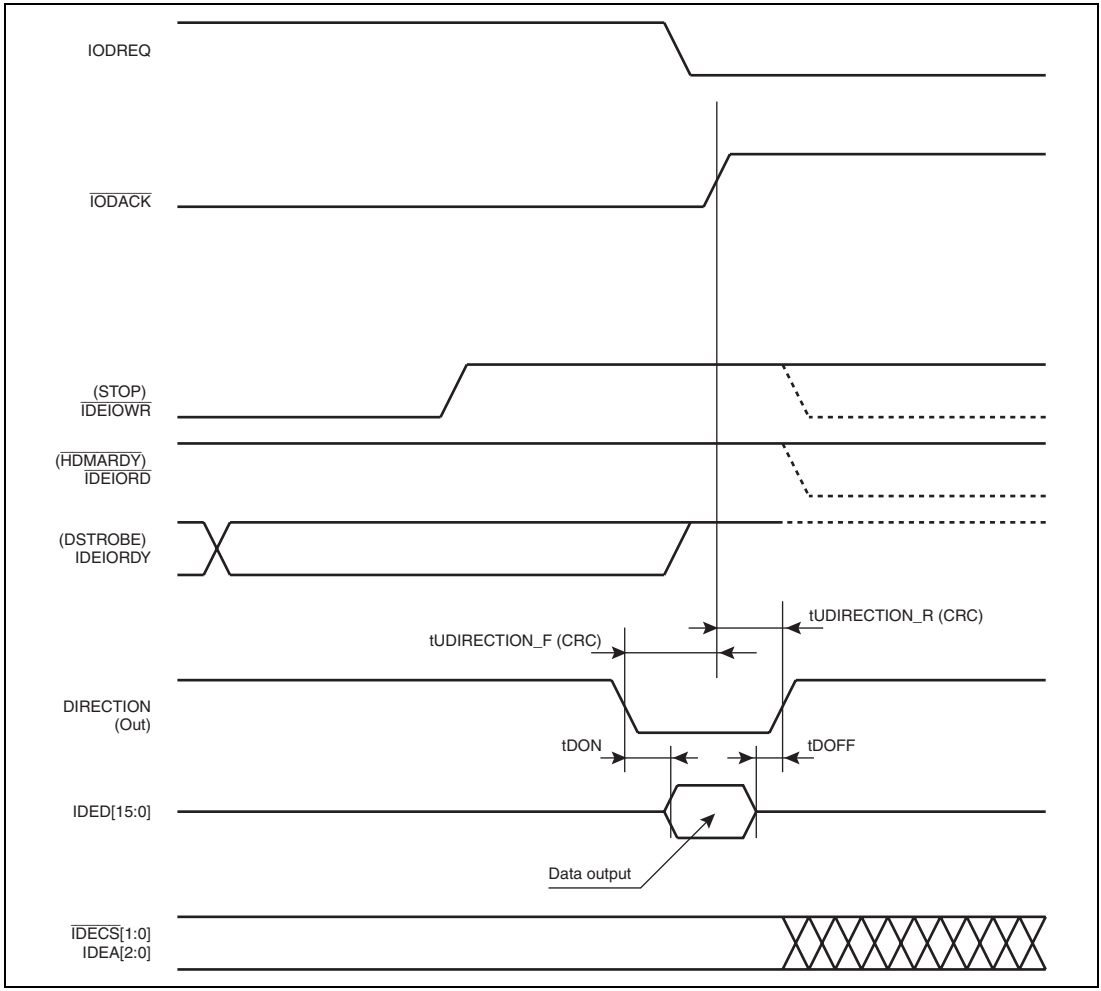


Figure 52.78 End of Ultra DMA Transfer Data In Burst from Host (DIRECTION)

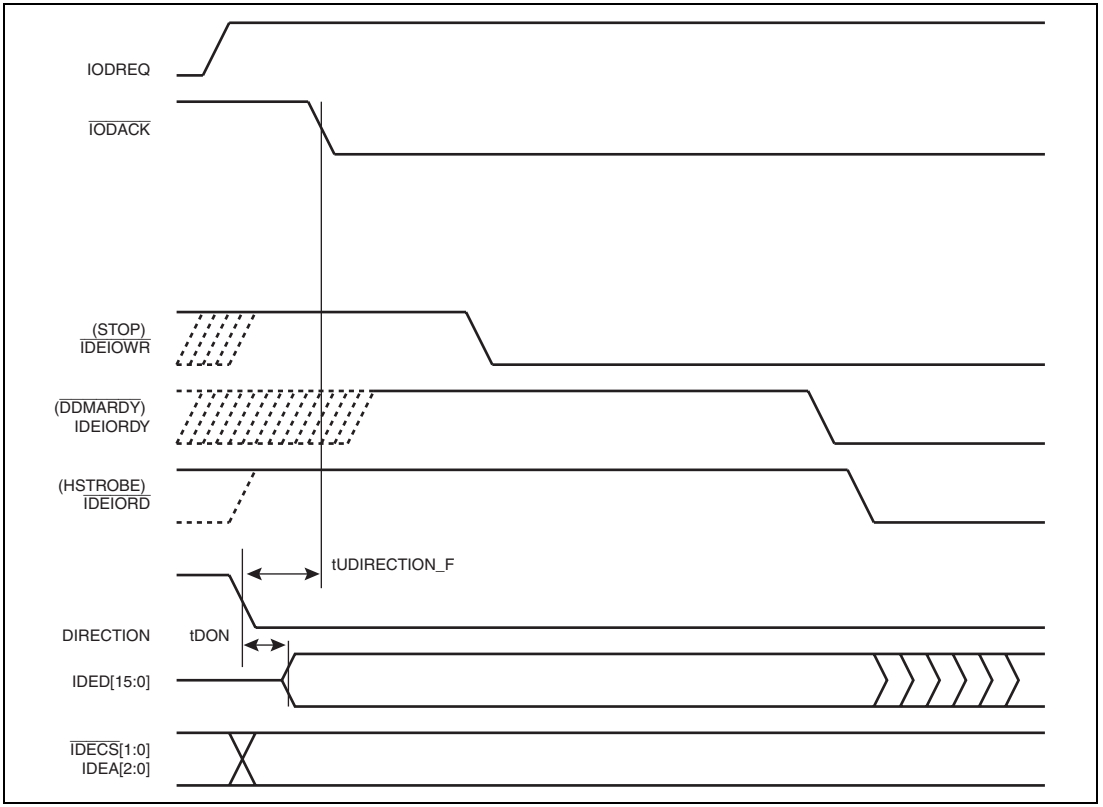


Figure 52.79 Start of Ultra DMA Transfer Data Out Burst (DIRECTION)

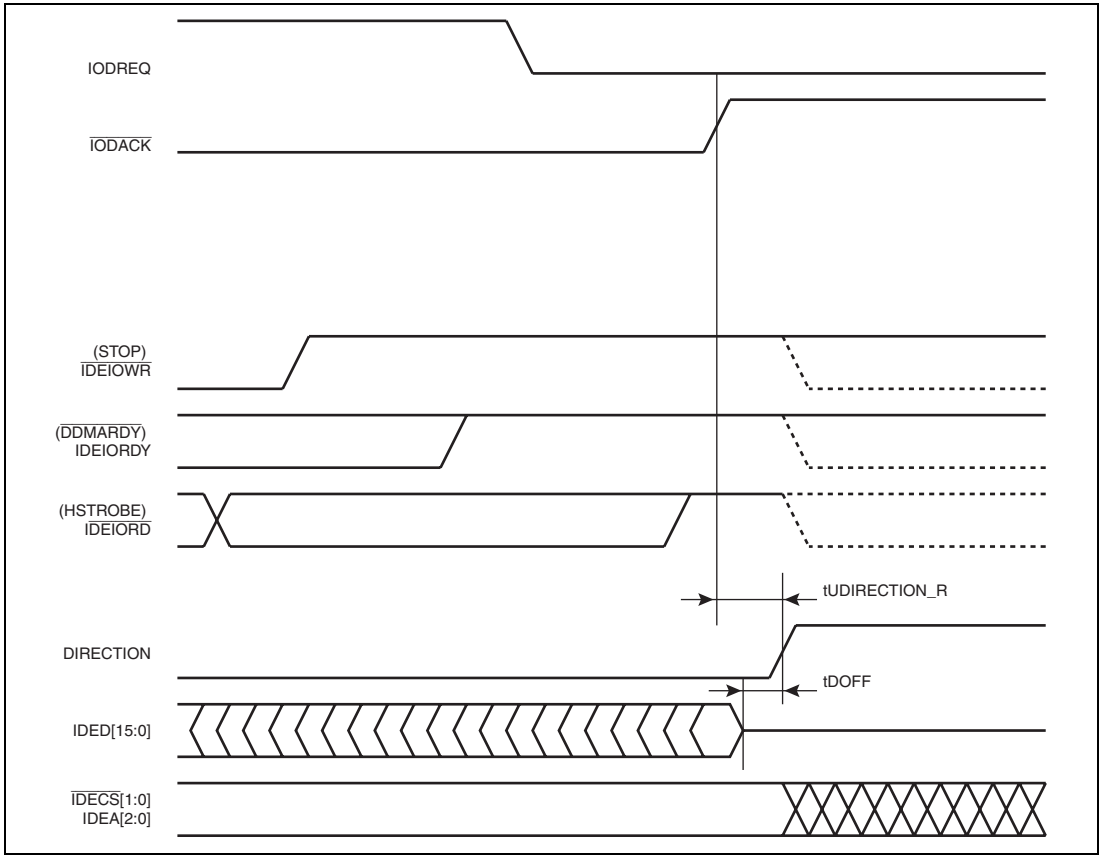


Figure 52.80 End of Ultra DMA Transfer Data Out Burst from Host (DIRECTION)

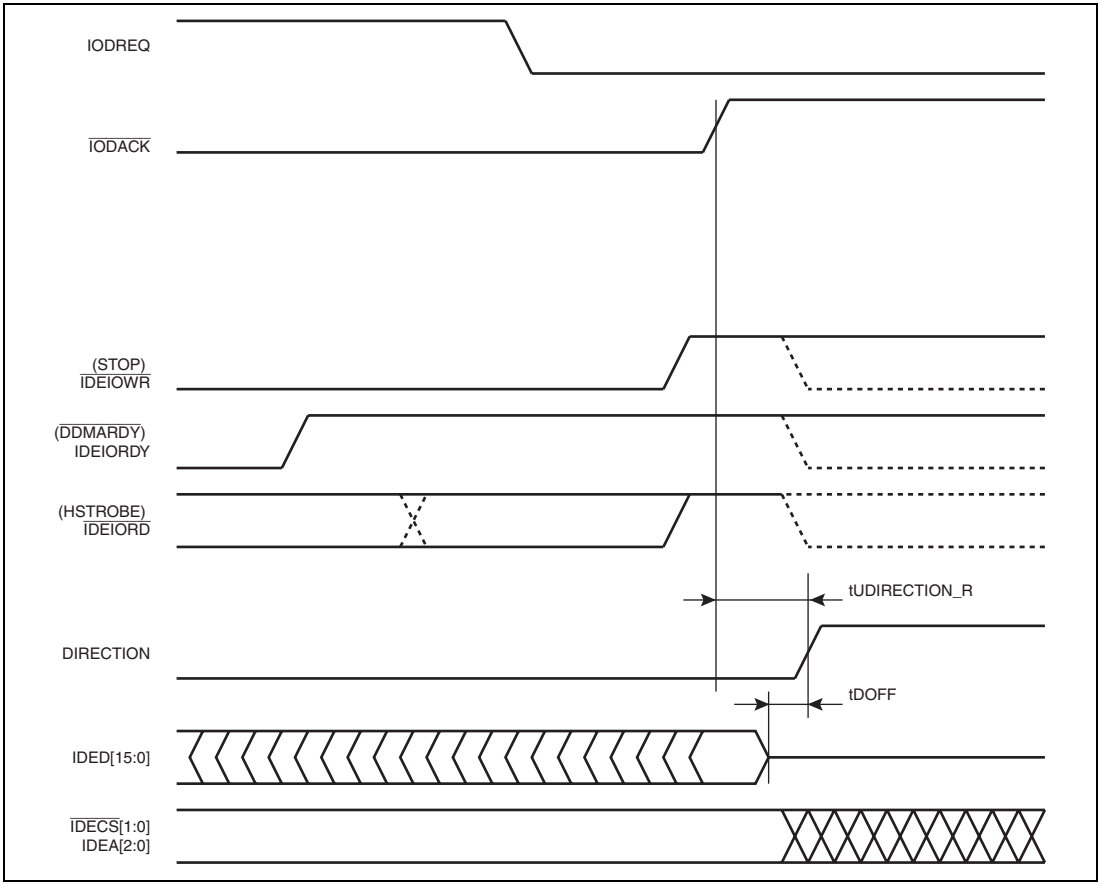


Figure 52.81 End of Ultra DMA Transfer Data Out Burst from Device (DIRECTION)

52.5.18 Ethernet Controller Timing (RMII)

Table 52.35 Ethernet Controller Timing (RMII)

Test conditions: $V_{CC}Q\_SR$ = 2.7 V to 3.6 V

Item	Symbol	Min.	Max.	Unit	Reference
REF50CK cycle time	t_{RTcyc}	20	—	ns	52.82
RMIIn_TXD_EN, RMII1M_TXD_EN output delay time	t_{RTEND}	2.5	15.5	ns	
RMIIn_TXD1, RMIIn_TXD0, RMII1M_TXD1, RMII1M_TXD0 output delay time	t_{RETDD}	2.5	15.5	ns	
RMIIn_CRS_DV, RMII1M_CRS_DV setup time	t_{RRDVS}	4	—	ns	
RMIIn_CRS_DV, RMII1M_CRS_DV hold time	t_{RRDVH}	2.5	—	ns	
RMIIn_RXD1, RMIIn_RXD0, RMII1M_RXD1, RMII1M_RXD0 setup time	t_{RERDS}	4	—	ns	
RMIIn_RXD1, RMIIn_RXD0, RMII1M_RXD1, RMII1M_RXD0 hold time	t_{RERDH}	2.5			
RMIIn_RX_ER setup time	t_{RRERS}	4		ns	52.83, 52.84
RMIIn_RX_ER hold time	t_{RRERH}	2.5		ns	

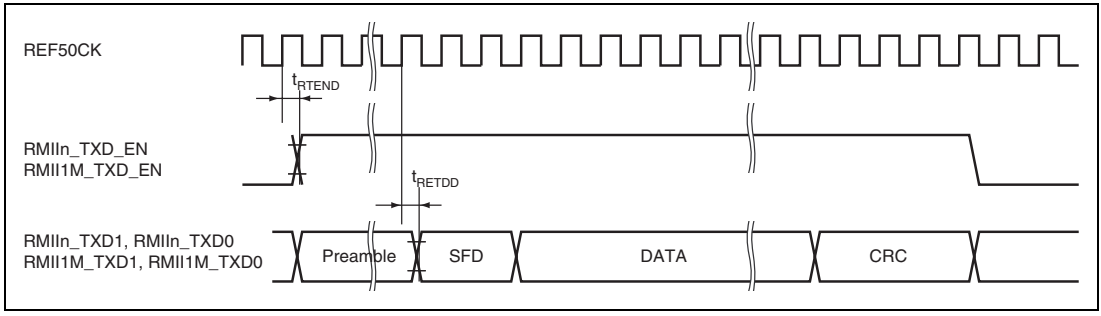


Figure 52.82 RMII Transmission Timing

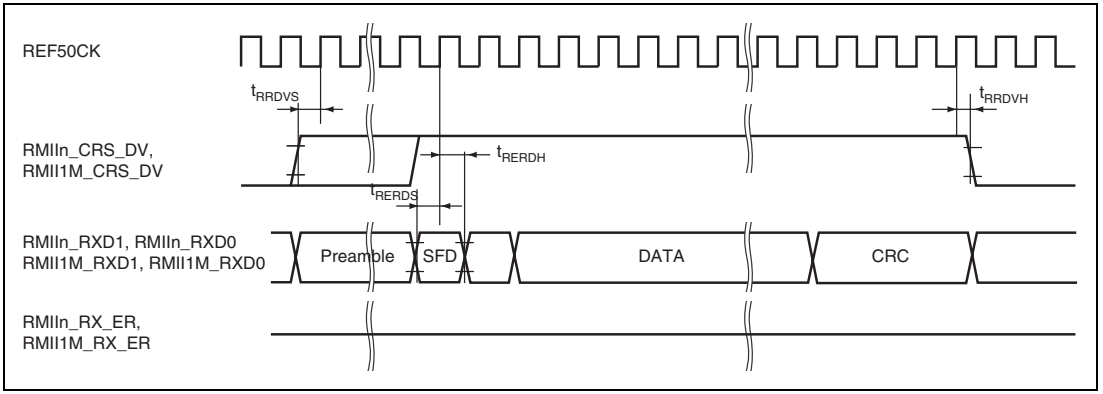


Figure 52.83 RMII Reception Timing (Normal Operation)

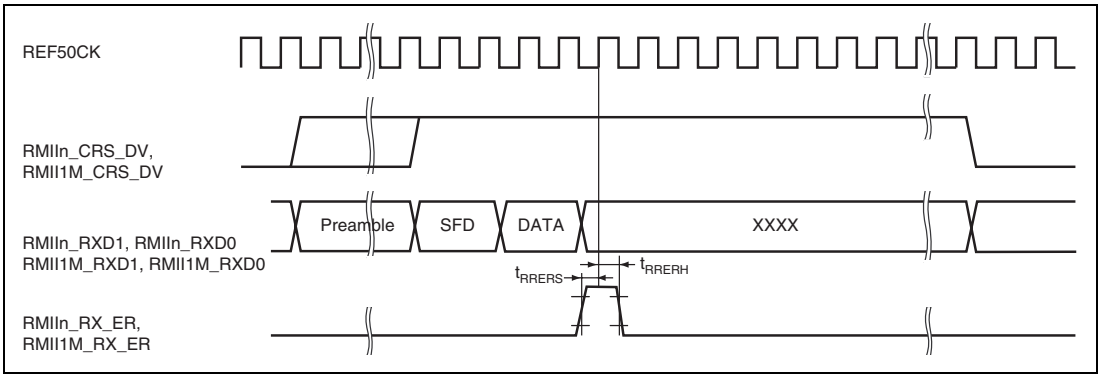


Figure 52.84 RMII Reception Timing (in Case of Error)

52.5.19 SDHI Module Signal Timing

Table 52.36 SDHI Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
SDCLK clock cycle	t_{SDDP}	20	—	ns	52.85
SDCLK clock high level width	t_{SDWH}	0.4 × t_{TSDPP}	—	ns	
SDCLK clock low level width	t_{SDWL}	0.4 × t_{TSDPP}	—	ns	
SDCMD, SDDAT3 to SDDAT0 output data delay (data transfer mode)	t_{SDODLY}	—	5	ns	
SDCMD, SDDAT3 to SDDAT0 input data setup	t_{SDISU}	5	—	ns	
SDCMD, SDDAT3 to SDDAT0 input data hold	t_{SDIH}	2	—	ns	

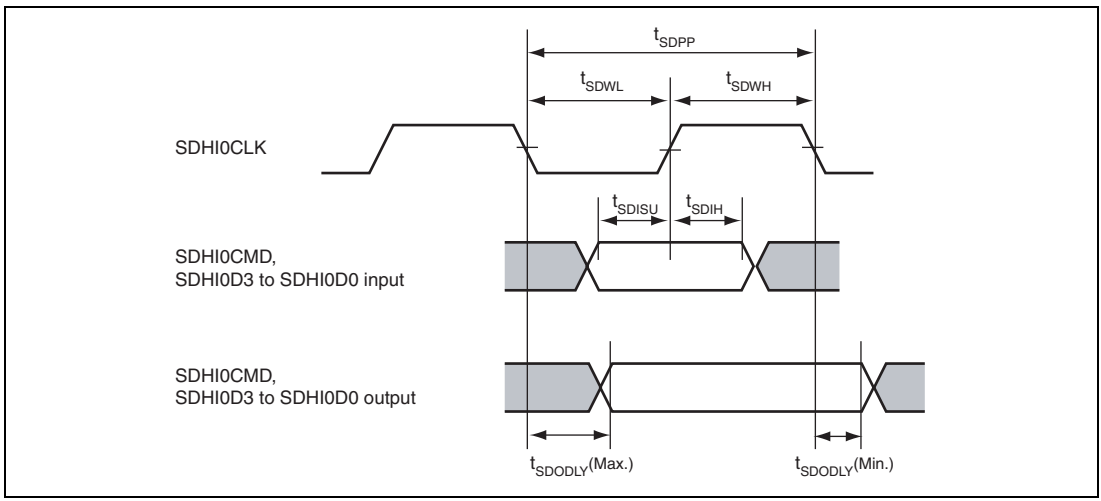


Figure 52.85 SDHI Module Signal Timing

52.5.20 MMCIF Module Signal Timing

Table 52.37 MMCIF Module Signal Timing

Conditions: $V_{CC}Q\_MMC$ = 2.7 V to 3.6 V

Item	Symbol	Min.	Max.	Unit	Reference
MMC_CLK clock cycle	t_{MMCPP}	—	26/52	MHz	52.86
MMC_CLK clock high level width	t_{MMCWH}	6.5	—	ns	
MMC_CLK clock low level width	t_{MMCWL}	6.5	—	ns	
MMC_CMD, MMC_D7 to MMC_D0 output data delay time	$t_{MMCODLY}$	-6.5	6.5	ns	
MMC_CMD, MMC_D7 to MMC_D0 intput data setup time	t_{MMCISU}	4.5	—	ns	
MMC_CMD, MMC_D7 to MMC_D0 intput data hold time	t_{MMCIH}	4.5	—	ns	

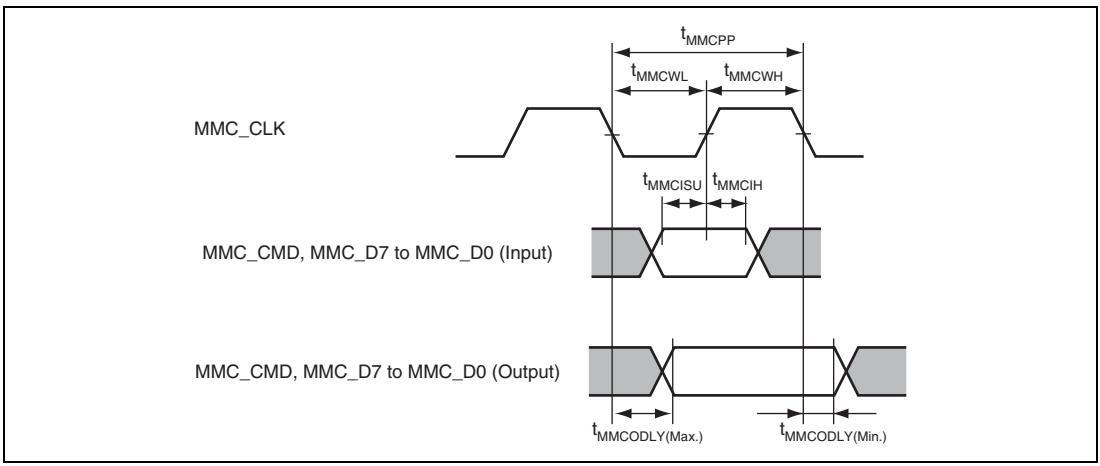


Figure 52.86 MMCIF Module Signal Timing

52.5.21 FSI Module Signal Timing

Table 52.38 FSI Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Reference
FSIMCK input cycle time	$t_{FSIMCYC}$	40	—	ns	52.87
FSIMCK input high level width	t_{FSIMWH}	0.4 × $t_{FSIMCYC}$	—	ns	
FSIMCK input low level width	t_{FSIMWL}	0.4 × $t_{FSIMCYC}$	—	ns	
FSIO_BCK clock cycle time	$t_{FSISICYC}$	80	—	ns	52.88
FSIO_BCK output high level width	$t_{FSISWHO}$	0.4 × $t_{FSISICYC}$	—	ns	
FSIO_BCK output low level width	$t_{FSISWLO}$	0.4 × $t_{FSISICYC}$	—	ns	
FSIO_LRCK output delay time	t_{FSIFSD}	—	20	ns	
FSII_BCK input high level width	$t_{FSISWHI}$	0.4 × $t_{FSISICYC}$	—	ns	
FSII_BCK input low level width	$t_{FSISWLI}$	0.4 × $t_{FSISICYC}$	—	ns	
FSIO_SD output delay time	$t_{FSISTDD}$	—	20	ns	
FSII_SD input setup time	$t_{FSISRDS}$	20	—	ns	
FSII_SD input hold time	$t_{FSISRDH}$	20	—	ns	

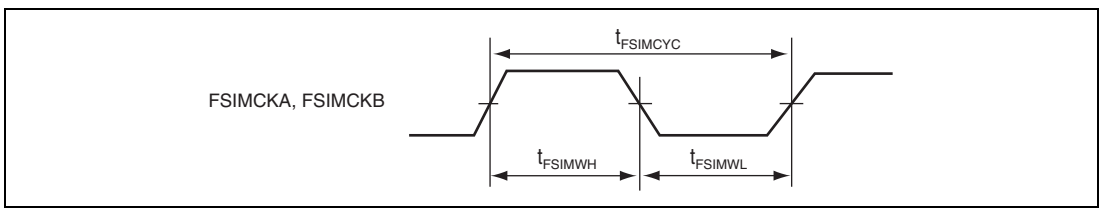


Figure 52.87 FSIMCK Input Timing

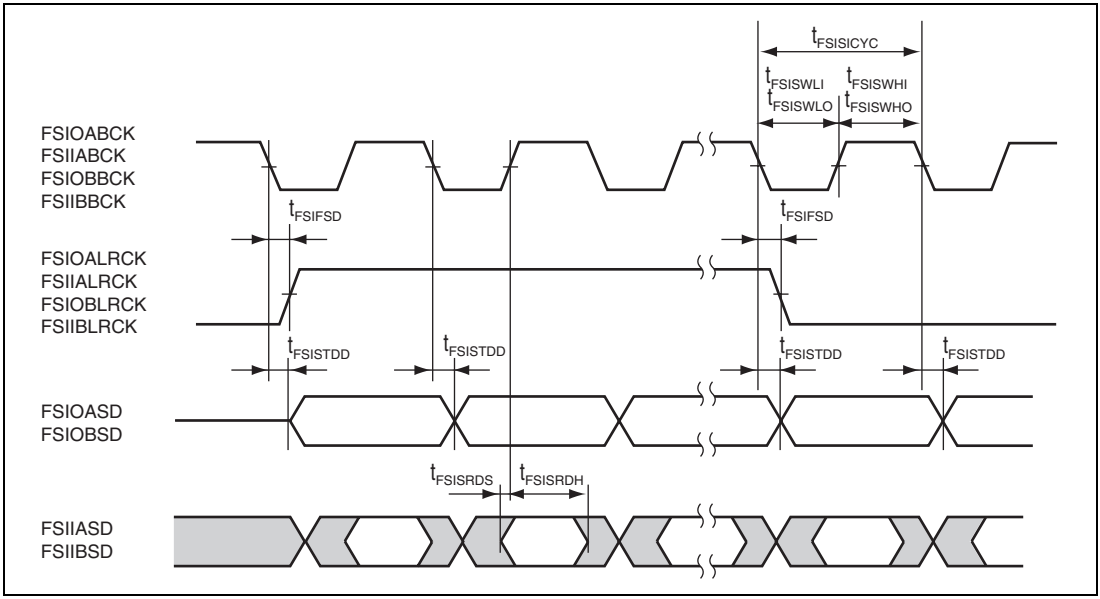


Figure 52.88 FSIMCK Transmission Timing

52.5.22 H-UDI Module Signal Timing

Table 52.39 H-UDI Module Signal Timing

Conditions: $V_{CC}Q$ = 3.3 V ± 0.3V, Ta= -40 to 85 °C, GND = V_{SS} = 0V, CL = 30 pF

Item	Symbol	Min.	Max.	Unit	Reference
Input clock cycle	t_{TCKcyc}	50*		ns	52.89
Input clock pulse width (High)	t_{TCKH}	15		ns	
Input clock pulse width (Low)	t_{TCKL}	15		ns	
Input clock rise time	t_{TCKr}		10	ns	
Input clock fall time	t_{TCKf}		10	ns	
TDI/TMS setup time	t_{TDIS}	15		ns	52.90
TDI/TMS hold time	t_{TDIH}	15		ns	
TDO output delay time	t_{TDO}	0	12	ns	

Note: * 2 MHz during a boundary scan operation.

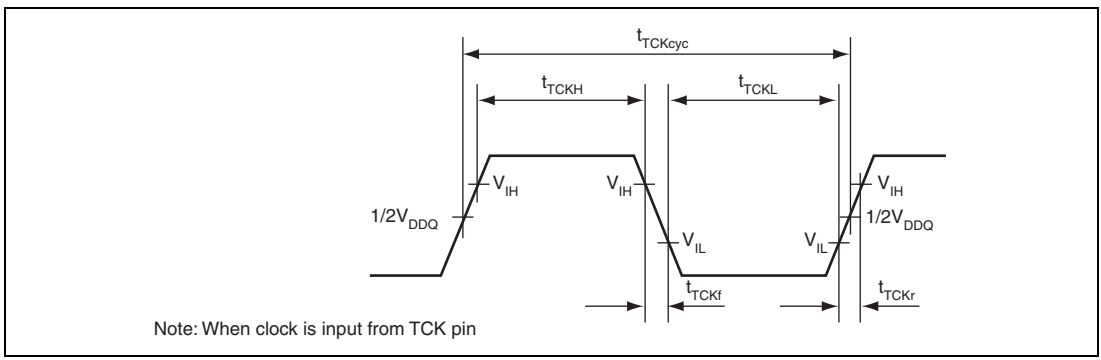


Figure 52.89 TCK Input Timing

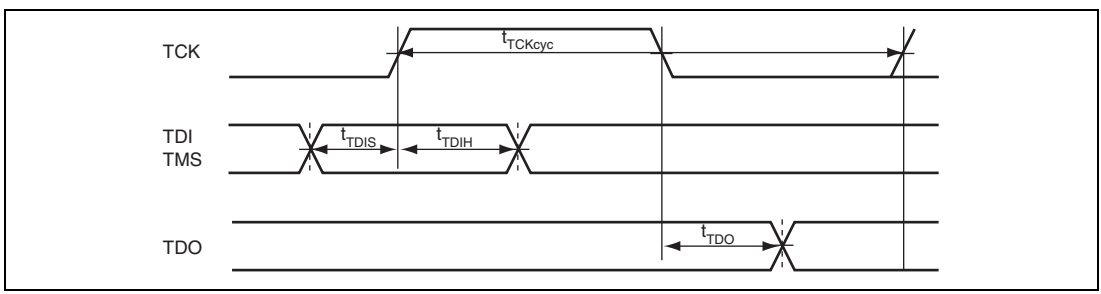


Figure 52.90 H-UDI Data Transfer Timing

52.6 USB Electrical Characteristics

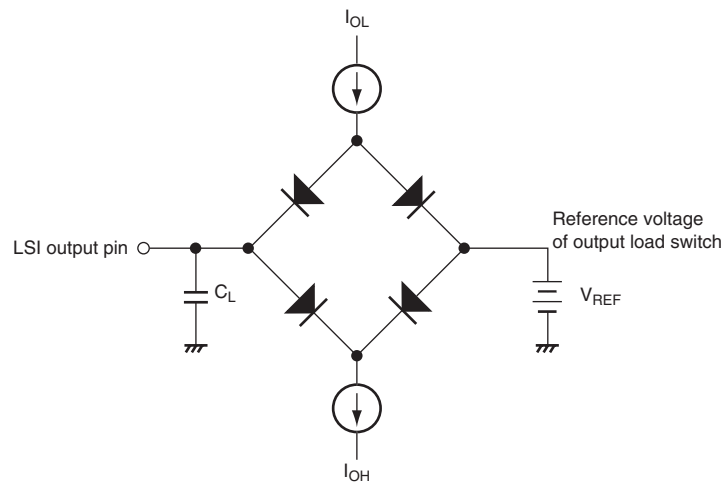
Table 52.40 USB Electrical Characteristics

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Common	VBUS connection detecting voltage	V_{VBUSIH}	4.35	5.0	5.25	V	
	External reference resistance	R_{REF}	5.544	5.6	5.656	kΩ	±1%
	Driver output Hi-Z	R_O	40.5	45	49.5	Ω	
	DP Pull-up resistance (Function mode)	R_{PU}	900	—	1575	Ω	Be in the idle state
			1425	—	3090	Ω	Be in the transmission and reception state
	DP, DM pull-down resistance	R_{PD}	14250	—	24800	Ω	
	EXTAL_USB input clock frequency	f_{EXTAL_USB}	47.9952	48	48.0048	MHz	±100ppm
	Oscillation settling time	t_{UOSC}	10	—	—	ms	
	USB PLL settling time	t_{UPLL}	120	—	—	µs	
FS/LS input	Input high level voltage	V_{IH}	2.0	—	DV33 + 0.3	V	
	Input low level voltage	V_{IL}	–0.3	—	0.8	V	
	Differential input sensitivity	V_{DI}	0.2	—	—	V	|(DP) – (DM)|
	Common mode voltage range	V_{CM}	0.8	—	2.5	V	
	Single ended receiver threshold voltage	V_{SE}	0.8	—	2.0	V	
FS/LS output	Output high level voltage	V_{OH}	2.8	—	—	V	
	Output low level voltage	V_{OL}	—	—	0.3	V	
HS input	Squelch detection threshold voltage (differential)	V_{HSSQ}	100	—	150	mV	
	Differential input sensitivity	V_{HSDI}	150	—	—	mV	|(DP) – (DM)|
	Common mode voltage range	V_{HSCM}	–50	—	500	mV	
	Chirp input voltage	V_{CHIRP_RCV}	700	—	1100	mV	
HS output	Idle state	V_{HSOI}	–10	—	10	mV	
	Output high level voltage	V_{HSOH}	360	—	440	mV	
	Output low level voltage	V_{HSOL}	–10	—	10	mV	
	Chirp output voltage (DP pin)	$V_{HSCHIRP\_P}$	700	—	1100	mV	
	Chirp output voltage (DM pin)	$V_{HSCHIRP\_M}$	–900	—	–500	mV	

Item		Symbol	Min.	Typ.	Max.	Unit	Test Conditions
FS	Rising time (DP, DM)	t_{FDR}	4	—	20	ns	
	Falling time (DP, DM)	tFDF	4	—	20	ns	
	Rising / Rising time ratio	t_{FDR}/t_{FDF}	90	—	111.1	%	
	Output signal crossover voltage (DP, DM)	V_{FCRS}	1.3	—	2.0	V	
LS	Output rising time (DP, DM)	t_{LDR}	75	—	300	ns	
	Output falling time (DP, DM)	t_{LDF}	75	—	300	ns	
	Output rising / rising time ratio	t_{LDR}/t_{LDF}	80	—	125	%	
	Output signal crossover voltage (DP, DM)	V_{LCRS}	1.3	—	2.0	V	
HS	DISCONNECT detection voltage	$V_{DISCONNECT}$	525	—	625	mV	

52.7 AC Characteristic Test Conditions

- I/O signal reference level: VCCQ_DDR × 0.5 (DBSC output)
  V_{REF} (DBSC input)
  $V_{CC}Q \times 0.5$ (other than DBSC)
- Input pulse level: VSS to VCCQ_DDR (DBSC)
  V_{SS} to $V_{CC}Q$ (other than DBSC)
- Input rise and fall times: 1 ns



Notes: 1. C_L is the total value that includes the capacitance of measurement instruments, and is set as follows for each pin:
30 pF: CKO, $\overline{CS0}$, $\overline{CS2}$ to $\overline{CS6B}$
50 pF: All other pins
2. I_{OL} = 0.2 mA, I_{OH} = –0.2 mA

Figure 52.91 Output Load Circuit (other than DDR-SDRAM Interface)

Appendix

A. CPU Operation Mode Register (CPUOPM)

The CPUOPM is used to control the CPU operation mode. This register can be read from or written to the address H'FF2F0000 in P4 area or H'1F2F0000 in area 7 as 32-bit size.

The write value to the reserved bits should be the initial value.

The operation is not guaranteed if the write value is not the initial value.

The CPUOPM register should be updated by the CPU store instruction, not the access from SuperHyway bus master except CPU.

After the CPUOPM is updated, read CPUOPM once, and execute one of the following two methods.

1. Execute a branch using the RTE instruction.
2. Execute the ICBI instruction for any address (including non-cacheable area).

After one of these methods is executed, it is guaranteed that the CPU runs under the updated CPUOPM value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	RABD	—	INTMU	—	—	—
Initial value:	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved The write value must be the initial value.
9 to 6	—	All 1	R	Reserved The write value must be the initial value.
5	RABD	1	R/W	Speculative execution bit for subroutine return 0: Instruction fetch for subroutine return is issued speculatively. When this bit is set to 0, refer to appendix C, Speculative Execution for Subroutine Return. 1: Instruction fetch for subroutine return is not issued speculatively.
4	—	0	R	Reserved The write value must be the initial value.
3	INTMU	0	R/W	Interrupt mode switch bit 0: SR.IMASK is not changed when an interrupt is accepted. 1: SR.IMASK is changed to the accepted interrupt level.
2 to 0	—	All 0	R	Reserved The write value must be the initial value.

B. Instruction Prefetching and Its Side Effects

This LSI is provided with an internal buffer for holding pre-read instructions, and always performs pre-reading. Therefore, program code must not be located in the last 64-byte area of any memory space. If program code is located in these areas, a bus access for instruction prefetch may occur exceeding the memory areas boundary. A case in which this is a problem is shown below.

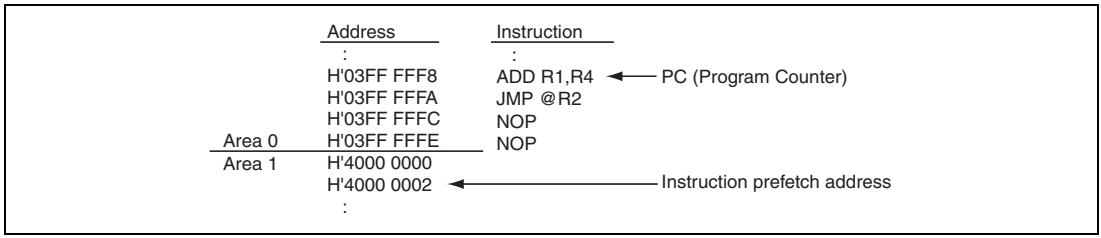


Figure B.1 Instruction Prefetch

Figure B.1 presupposes a case in which the instruction (ADD) indicated by the program counter (PC) and the address H'04000002 instruction prefetch are executed simultaneously. It is also assumed that the program branches to an area other than area 1 after executing the following JMP instruction and delay slot instruction.

In this case, a bus access (instruction prefetch) to area 1 may unintentionally occur from the programming flow.

Instruction Prefetch Side Effects

1. It is possible that an external bus access caused by an instruction prefetch may result in misoperation of an external device, such as a FIFO, connected to the area concerned.
2. If there is no device to reply to an external bus request caused by an instruction prefetch, hang-up will occur.

Remedies

1. These illegal instruction fetches can be avoided by using the MMU.
2. The problem can be avoided by not locating program code in the last 64 bytes of any area.

C. Speculative Execution for Subroutine Return

The SH-4A has the mechanism to issue an instruction fetch speculatively when returning from subroutine. By issuing an instruction fetch speculatively, the execution cycles to return from subroutine may be shortened.

This function is enabled by setting 0 to the bit 5 (RABD) of CPU Operation Mode register (CPUOPM). But this speculative instruction fetch may issue the access to the address that should not be accessed from the program. Therefore, a bus access to an unexpected area or an internal instruction address error may cause a problem. As for the effect of this bus access to unexpected memory area, refer to appendix B, Instruction Prefetching and Its Side Effects.

Usage Condition: When the speculative execution for subroutine return is enabled, the RTS instruction should be used to return to the address set in PR by the JSR, BSR, or BSRF instructions. It can prevent the access to unexpected address and avoid the problem.

D. Pin States after Reset and in Power-Down Modes

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby		U-Standby
							During	After	
Clock	EXTAL	I	I	I	I	I	I	I	I
	XTAL	O*3	O*3	O*3	O*3	O*3	O*3	O*3	O*3
	RTC_CLK	I	I	I	I	I	I	I	I
Operating mode	MD3 to MD0	I	I	I	I	I	I	I	I
	MD8	I	I	I	I	I	I	I	I
	MD5	I	I	I	I	I	I	I	I
	MSLD	I	I	I	I	I	I	I	I
	$\overline{TST}$	I	I	I	I	I	I	I	I
	$\overline{TSTMD}$	I	I	I	I	I	I	I	I
System control	BOOT	I	I	I	I	I	I	I	I
	$\overline{RESETP}$	I	I	I	I	I	I	I	I
	$\overline{RESETA}$	I	I	I	I	I	I	I	I
	$\overline{RESETOUT}$	L	L	O	O	O	O	O	O
	STATUS0	L	L	O	L	H	H	L	H
	STATUS2	L	L	O	L	L	H	L	L
	PDSTATUS	L	L	O	L	L	H	L	H
Interrupt	IRQ7 to IRQ0	—	—	—	I	I	I	I	I
	NMI	Z	I	I	I	I	I	I	I
BSC	A25 to A0	L	L	O	O	O/Z*4	O/Z*4	L	O/Z*4
	$\overline{BS}$	—	—	—	O	H/Z*4	H/Z*4	H	H/Z*4
	CKO	L	O	O	O	O/Z*4	O/Z*4	O	O/Z*4
	$\overline{CS4}$, $\overline{CS0}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{CS5A}$ / $\overline{CE2A}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{CS5B}$ / $\overline{CE1A}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{CS6A}$ / $\overline{CE2B}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{CS6B}$ / $\overline{CE1B}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	D31 to D0	Z	Z	Z	Z/I/O	Z	Z	Z	Z
	$\overline{IOIS16}$	Z	Z	Z	I	Z	Z	I	Z
	RDWR	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{RD}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
BSC	$\overline{\text{WAIT}}$	ZU	IU	IU	IU	IU	IU	IU	IU
	$\overline{\text{WE1}}$, $\overline{\text{WE0}}$ / $\overline{\text{WE}}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{\text{WE2}}$ / $\overline{\text{ICIORD}}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
	$\overline{\text{WE3}}$ / $\overline{\text{ICIOWR}}$	H	H	O	O	H/Z*4	H/Z*4	H	H/Z*4
DBSC	MA13 to MA0	L	H	O	O	O/Z*5	O/Z*5	H	O/Z*5
	MBA2 to MBA0	L	H	O	O	O/Z*5	O/Z*5	H	O/Z*5
	MCLK	L	O	O	IO	IO/I*5	IO/I*5	IO	IO/I*5
	$\overline{\text{MCLK}}$	H	O	O	IO	IO/I*5	IO/I*5	IO	IO/I*5
	$\overline{\text{MCS}}$	L	H	O	O	O/Z*5	O/Z*5	H	O/Z*5
	MDQ31 to MDQ0	Z	Z	Z	Z/I/O	Z	Z	Z	Z
	MDQM3 to MDQM0	L	H	O	O	H/Z*5	H/Z*5	H	H/Z*5
	MDQS3 to MDQS0	Z	Z	Z	Z/I/O*5	H/Z*5	H/Z*5	Z	H/Z*5
	$\overline{\text{MDQS3}}$ to $\overline{\text{MDQS0}}$	Z	Z	Z	Z/I/O*5	L/Z*5	L/Z*5	Z	L/Z*5
	$\overline{\text{MCAS}}$	L	H	O	O	O	O	H	O
	$\overline{\text{MRAS}}$	L	H	O	O	O/Z*5	O/Z*5	H	O/Z*5
	$\overline{\text{MWE}}$	L	H	O	O	O/Z*5	O/Z*5	H	O/Z*5
	MCKE	L/H*9	L/H*9	O	O	O	O	O	O
	MODT	L	L	O	O	O/Z*5	O/Z*5	L	O/Z*5
DMAC	DREQ1, DREQ0	—	—	—	I	Z	Z	I	Z
	DACK1, DACK0	—	—	—	O	O	O	O	O
FSI	CLKAUDIOAO	—	—	—	O	O	O	O	O
	CLKAUDIOBO	—	—	—	O	O	O	O	O
	FSIIABCK	—	—	—	I	Z	Z	I	Z
	FSIIALRCK	—	—	—	I	Z	Z	I	Z
	FSIIASD	—	—	—	I	Z	Z	I	Z
	FSIIBBCK	—	—	—	I	Z	Z	I	Z
	FSIIBLRCK	—	—	—	I	Z	Z	I	Z
	FSIIBSD	—	—	—	I	Z	Z	I	Z
	FSIMCKA	—	—	—	I	Z	Z	I	Z
	FSIMCKB	—	—	—	I	Z	Z	I	Z
	FSIOABCK	—	—	—	O	O	O	O	O
	FSIOALRCK	—	—	—	O	O	O	O	O

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
FSI	FSIOASD	—	—	—	O	O	O	O	O
	FSIOBBCK	—	—	—	O	O	O	O	O
	FSIOBLRCK	—	—	—	O	O	O	O	O
	FSIOBSD	—	—	—	O	O	O	O	O
IIC	SCL1	Z	Z	Z	I/O	Z	Z	Z	Z
	SCL0	Z	Z	Z	I/O	Z	Z	Z	Z
	SDA1	Z	Z	Z	I/O	Z	Z	Z	Z
	SDA0	Z	Z	Z	I/O	Z	Z	Z	Z
LCDC	LCDD23 to LCDD0	—	—	—	O	O	O	L	O
	LCDDCK	—	—	—	O	O	O	O	O
	LCDDISP	—	—	—	O	O	O	L	O
	LCDDON	—	—	—	O	O	O	L	O
	LCDHSYN	—	—	—	O	O	O	L	O
	$\overline{\text{LCDRD}}$	—	—	—	O	O	O	H	O
	$\overline{\text{LCDCS}}$	—	—	—	O	O	O	H	O
	LCDLCLK	—	—	—	I	Z	Z	I	Z
	LCDRS	—	—	—	O	O	O	L	O
	$\overline{\text{LCDWR}}$	—	—	—	O	O	O	H	O
	LCDVCPWC	—	—	—	O	O	O	L	O
	LCDVEPWC	—	—	—	O	O	O	L	O
	LCDVSYN	—	—	—	O	O	O	L	O
TSIF	TS_SCK	—	—	—	I	Z	Z	I	Z
	TS_SDAT	—	—	—	I	Z	Z	I	Z
	TS_SDEN	—	—	—	I	Z	Z	I	Z
	TS_SPSYNC	—	—	—	I	Z	Z	I	Z
USB	DM1, DM0	Z	Z	Z	Z/I/O*5	Z/I/O*5	Z/I/O*5	Z/I/O*5	Z/I/O*5
	DP1, DP0	Z	Z	Z	Z/I/O*5	Z/I/O*5	Z/I/O*5	Z/I/O*5	Z/I/O*5
	EXTALUSB	I	I	I	I	I	I	I	I
	XTALUSB	O	O	O	O	O	O	O	O
	VBUS1, VBUS0	I	I	I	I	I	I	I	I

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
VIO (CEU)	VIO0_CLK	—	—	—	I	Z	Z	I	Z
	VIO0_D15 to VIO0_D0	—	—	—	I	Z	Z	I	Z
	VIO0_FLD	—	—	—	I	Z	Z	I	Z
	VIO0_HD	—	—	—	I	Z	Z	I	Z
	VIO0_VD	—	—	—	I	Z	Z	I	Z
	VIO1_CLK	—	—	—	I	Z	Z	I	Z
	VIO1_D7 to VIO1_D0	—	—	—	I	Z	Z	I	Z
	VIO1_FLD	—	—	—	I	Z	Z	I	Z
	VIO1_HD	—	—	—	I	Z	Z	I	Z
	VIO1_VD	—	—	—	I	Z	Z	I	Z
	VIO_CKO	—	—	—	O	O	O	O	O
VOU	DV_CLK	—	—	—	O	O	O	O	O
	DV_CLKI	—	—	—	I	Z	Z	I	Z
	DV_D15 to DV_D0	—	—	—	O	O	O	O	O
	DV_HSYNC	—	—	—	O	O	O	O	O
	DV_VSYNC	—	—	—	O	O	O	O	O
KEYSC	KEYIN4 to KEYIN0	—	—	—	IU	IU	IU	IU	IU
	KEYOUT3 to KEYOUT0	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	KEYOUT4 / IN6	—	—	—	O/Z/IU*5	O/Z/IU*5	O/Z/IU*5	O/Z/IU*5	O/Z/IU*5
	KEYOUT5 / IN5	—	—	—	O/Z/IU*5	O/Z/IU*5	O/Z/IU*5	O/Z/IU*5	O/Z/IU*5
IrDA	$\overline{\text{IRDA\_IN}}$	—	—	—	I	Z	Z	I	Z
	IRDA_OUT	—	—	—	O	O	O	O	O
MSIOF0	MSIOF0_MCK	—	—	—	I	Z	Z	I	Z
	MSIOF0_RXD	—	—	—	I	Z	Z	I	Z
	$\overline{\text{MSIOF0\_SS1}}$ / MSIOF0_RSCK	—	—	—	O/I*5	O/I*5	O/I*5	O/I*5	O/I*5
	$\overline{\text{MSIOF0\_SS2}}$ / MSIOF0_RSYNC	—	—	—	O/I*5	O/I*5	O/I*5	O/I*5	O/I*5
	MSIOF0_TSCK	—	—	—	O/I*5	O/I*5	O/I*5	O/I*5	O/I*5
	MSIOF0_TSYNC	—	—	—	O/I*5	O/I*5	O/I*5	O/I*5	O/I*5
	MSIOF0_TXD	—	—	—	O	O	O	O	O
MSIOF1	MSIOF1_MCK	—	—	—	I	Z	Z	I	Z
	MSIOF1_RXD	—	—	—	I	Z	Z	I	Z
	$\overline{\text{MSIOF1\_SS1}}$ / MSIOF1_RSCK	—	—	—	O/I*5	O/I*5	O/I*5	O/I*5	O/I*5

Module	Pin Name	During Reset*[1]	During Reset*[2]	After Reset*[2]	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
MSIOF1	$\overline{\text{MSIOF1\_SS2}}$ / MSIOF1_RSYNC	—	—	—	O/I*[5]	O/I*[5]	O/I*[5]	O/I*[5]	O/I*[5]
	MSIOF1_TSCK	—	—	—	O/I*[5]	O/I*[5]	O/I*[5]	O/I*[5]	O/I*[5]
	MSIOF1_TSYNC	—	—	—	O/I*[5]	O/I*[5]	O/I*[5]	O/I*[5]	O/I*[5]
	MSIOF1_TXD	—	—	—	O	O	O	O	O
SCIF	SCIF0_RXD	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	SCIF0_SCK	—	—	—	IO/Z*[5]	O/Z*[5]	O/Z*[5]	IO/Z*[5]	O/Z*[5]
	SCIF0_TXD	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
	SCIF1_RXD	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	SCIF1_SCK	—	—	—	IO/Z*[5]	O/Z*[5]	O/Z*[5]	IO/Z*[5]	O/Z*[5]
	SCIF1_TXD	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
	SCIF2_RXD	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	SCIF2_SCK	—	—	—	IO/Z*[5]	O/Z*[5]	O/Z*[5]	IO/Z*[5]	O/Z*[5]
	SCIF2_TXD	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
SCIFA	$\overline{\text{SCIF3\_CTS}}$	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	$\overline{\text{SCIF3\_RTS}}$	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
	SCIF3_RXD	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	SCIF3_SCK	—	—	—	IO/Z*[5]	O/Z*[5]	O/Z*[5]	IO/Z*[5]	O/Z*[5]
	SCIF3_TXD	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
	SCIF4_RXD	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	SCIF4_SCK	—	—	—	IO/Z*[5]	O/Z*[5]	O/Z*[5]	IO/Z*[5]	O/Z*[5]
	SCIF4_TXD	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
	SCIF5_RXD	—	—	—	I/Z*[5]	Z	Z	I/Z*[5]	Z
	SCIF5_SCK	—	—	—	IO/Z*[5]	O/Z*[5]	O/Z*[5]	IO/Z*[5]	O/Z*[5]
	SCIF5_TXD	—	—	—	O/Z*[5]	Z	Z	O/Z*[5]	Z
MMC	MMC_CLK	—	—	—	L/O	L	L	L	L
	MMC_CMD	—	—	—	I/O	Z	Z	I	Z
	MMC_D7 to MMC_D0	—	—	—	I/O	Z	Z	I	Z
SDHI0	$\overline{\text{SDHI0CD}}$	—	—	—	I	Z	Z	I	Z
	SDHI0CLK	—	—	—	O	L	L	L	L
	SDHI0CMD	—	—	—	I/O	Z	Z	I	Z
	SDHI0D3 to SDHI0D0	—	—	—	I/O	Z	Z	I	Z
	SDHI0WP	—	—	—	I	Z	Z	I	Z

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
SDHI1	$\overline{\text{SDHI1CD}}$	—	—	—	I	Z	Z	I	Z
	SDHI1CLK	—	—	—	O	L	L	L	L
	SDHI1CMD	—	—	—	I/O	Z	Z	I	Z
	SDHI1D3 to SDHI1D0	—	—	—	I/O	Z	Z	I	Z
	SDHI1WP	—	—	—	I	Z	Z	I	Z
TPU	TPUTI3, TPUTI2	—	—	—	I	Z	Z	I	Z
	TPUTO3 to TPUTO0	—	—	—	O	O	O	O	O
ATAPI	DIRECTION	—	—	—	O	H	H	O	H
	$\overline{\text{EXBUF\_ENB}}$	—	—	—	O	H	H	O	H
	IDEA2 to IDEA0	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	$\overline{\text{IDECS1}}$, $\overline{\text{IDECS0}}$	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	IDED15 to IDED0	—	—	—	Z/I/O	Z	Z	Z/I/O	Z
	IDEINT	—	—	—	I	Z	Z	I	Z
	$\overline{\text{IDEIORD}}$	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	IDEIORDY	—	—	—	I	Z	Z	I	Z
	$\overline{\text{IDEIOWR}}$	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	$\overline{\text{IDERST}}$	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	$\overline{\text{IODACK}}$	—	—	—	O/Z*5	O/Z*5	O/Z*5	O/Z*5	O/Z*5
	IODREQ	—	—	—	I	Z	Z	I	Z
EtherC	LNKSTA	—	—	—	I/Z*5	Z	Z	I/Z*5	Z
	MDC	—	—	—	O/Z*5	Z	Z	O/Z*5	Z
	MDIO	—	—	—	IO/Z*5	O/Z*5	O/Z*5	IO/Z*5	O/Z*5
	RMII_CRS_DV	—	—	—	I/Z*5	Z	Z	I/Z*5	Z
	RMII_REF_CLK	—	—	—	I/Z*5	Z	Z	I/Z*5	Z
	RMII_RX_ER	—	—	—	I/Z*5	Z	Z	I/Z*5	Z
	RMII_RXD0	—	—	—	I/Z*5	Z	Z	I/Z*5	Z
	RMII_RXD1	—	—	—	I/Z*5	Z	Z	I/Z*5	Z
	RMII_TX_EN	—	—	—	O/Z*5	Z	Z	O/Z*5	Z
	RMII_TXD0	—	—	—	O/Z*5	Z	Z	O/Z*5	Z
	RMII_TXD1	—	—	—	O/Z*5	Z	Z	O/Z*5	Z

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
Port A	PTA7 to PTA0	—	—	—	P	K	K	P	K
Port B	PTB7 to PTB0	—	—	—	P	K	K	P	K
Port C	PTC7 to PTC0	ZU	ZU	IU	P	K	K	P	K
Port D	PTD7 to PTD0	ZU	ZU	IU	P	K	K	P	K
Port E	PTE1 to PTE0	ZU	ZU	IU	P	K	K	P	K
	PTE5 to PTE2	ZD	ZD	ID	P	K	K	P	K
	PTE7 to PTE6	ZU	ZU	IU	P	K	K	P	K
Port F	PTF1 to PTF0	ZD	ZD	ID	P	K	K	P	K
	PTF7 to PTF2	ZU	ZU	IU	P	K	K	P	K
Port G	PTG5 to PTG0	Z	Z	Z	P	K	K	P	K
Port H	PTH7 to PTH0	ZU	ZU	IU	P	K	K	P	K
Port J	PTJ7 to PTJ0	—	—	—	P	K	K	P	K
Port K	PTK7 to PTK0	ZU	ZU	IU	P	K	K	P	K
Port L	PTL7 to PTL0	ZU	ZU	IU	P	K	K	P	K
Port M	PTM7 to PTM0	ZU	ZU	IU	P	K	K	P	K
Port N	PTN4 to PTN0	ZU	ZU	IU	P	K	K	P	K
	PTN5	ZD	ZD	ID	P	K	K	P	K
	PTN7 to PTN6	ZU	ZU	IU	P	K	K	P	K
Port Q	PTQ7 to PTQ0	—	—	—	P	K	K	P	K
Port R	PTR7 to PTR0	—	—	—	P	K	K	P	K
Port S	PTS6 to PTS0	ZU	ZU	IU	P	K	K	P	K
Port T	PTT7 to PTT0	—	—	—	P	K	K	P	K
Port U	PTU7 to PTU0	ZU	ZU	IU	P	K	K	P	K
Port V	PTV7 to PTV0	ZU	ZU	IU	P	K	K	P	K
Port W	PTW0	L	L	O	P	K	K	P	K
	PTW7 to PTW1	ZU	ZU	IU	P	K	K	P	K
Port X	PTX1, PTX0	ZU	ZU	IU	P	K	K	P	K
	PTX5 to PTX2	ZD	ZD	ID	P	K	K	P	K
	PTX7, PTX6	ZU	ZU	IU	P	K	K	P	K
Port Y	PTY0	L	L	O	P	K	K	P	K
	PTY7 to PTY1	ZU	ZU	IU	P	K	K	P	K
Port Z	PTZ7 to PTZ0	ZU	ZU	IU	P	K	K	P	K

Module	Pin Name	During Reset*1	During Reset*2	After Reset*2	Sleep	Software Standby	R-Standby During	R-Standby After	U-Standby
H-UDI	TCK	IU	IU	IU	IU	IU	IU	IU	IU
	TMS	IU	IU	IU	IU	IU	IU	IU	IU
	TDI	IU	IU	IU	IU	IU	IU	IU	IU
	TDO	Z	Z/O	Z/O	Z/O*6	Z/O*6	Z/O*6	Z/O*6	Z/O*6
	$\overline{\text{TRST}}$	IU	IU	IU	IU*7	IU*7	IU*7	IU*7	IU*7
	$\overline{\text{ASEBRK}}$ / BRKAK	Z	IU/OU*8	IU/OU*8	IU/OU*8	IU/OU*8	IU/OU*8	IU/OU*8	IU/OU*8
	MPMD	IU	IU	IU	IU	IU	IU	IU	IU
AUD	AUDATA3 to AUDATA0	—	—	—	O	O	O	O	O
	AUDCK	—	—	—	O	O	O	O	O
	AUDSYNC	—	—	—	O	O	O	O	O

[Legend]

I: Input (pull-up and pull-down MOS off)

IU: Input (pull-up MOS on)

ID: Input (pull-down MOS on)

H: High-level output

L: Low-level output

O: Output

OU: Output (pull-up MOS on)

P: Functions as a port. (Selection of input or output and state of pull-up and pull-down MOS depend on the register settings)

K: Port state is retained. (input buffer off; output buffer state retained; pull-up and pull-down state retained).

Z: High-impedance state. (input buffer off; output buffer off; pull-up and pull-down MOS off)

ZU: Pulled up. (input buffer off; output buffer off; pull-up MOS on)

ZD: Pulled down. (input buffer off; output buffer off; pull-down MOS on)

The state written on the left of a slash (/) indicates the default state.

—: Cannot be selected.

Notes: *1. Indicates the duration in which $\overline{\text{RESETP}}$ is asserted and a maximum of two RCLK cycles after $\overline{\text{RESETP}}$ is negated.

*2. Indicates the duration in which $\overline{\text{RESETOUT}}$ is asserted after $\overline{\text{RESETP}}$ is negated.

*3. Depends on the clock mode selected (MD0 to MD2).

*4. Z or [H/L] depending on the settings of the HIZMEM and HIZCNT bits in CMNCR register of the BSC.

*5. Depends on the register settings

*6. Depends on the state of the TAP controller when the MPMD pin = H. When the MPMD pin = L, the pin is placed in the state written on the right of a slash (/).

*7. The pull-up MOS can be turned on or off according to the PULCR register setting.

*8. With respect to I/O pins with a pull-up MOS, input or output is selected according to the register setting when the MPMD pin = L. When the $\overline{\text{TRST}}$ pin = L, the pin provides input. When the MPMD pin = H, the pin always provides input.

*9. Depends on the state of the MSLD pin. When the MSLD pin = H, MCKE = H. When the MSLD pin = L, MCKE = L.

E. Package Dimension

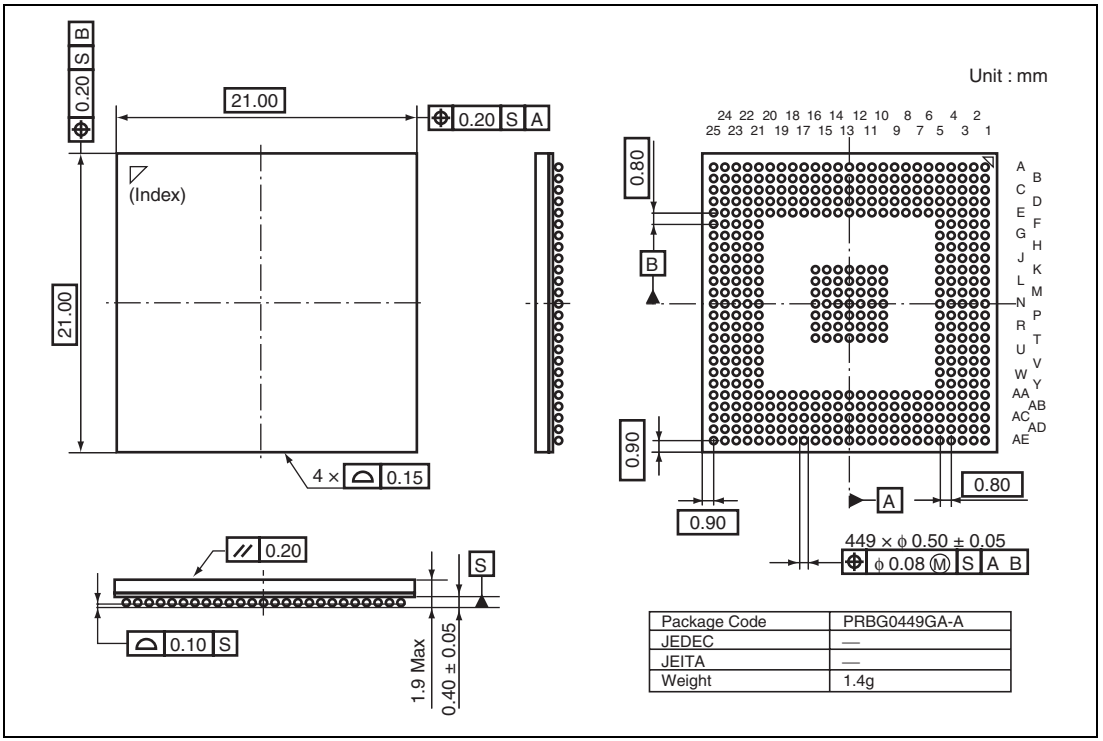


Figure E.1 Package Dimension

Index

M

N

O

P

Q

R

S

T

U

V

W

SH7724 User's Manual: Hardware

Publication Date: Rev.1.00 May 20, 2010

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
7F, No. 363 Fu Shing North Road Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.0

SH7724

Renesas Electronics Corporation

REJ09B0560-0100