

To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: <http://www.renesas.com>

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (<http://www.renesas.com>)

Send any inquiries to <http://www.renesas.com/inquiry>.

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.

The revision list can be viewed directly by clicking the title page.
The revision list summarizes the locations of revisions and additions. Details should always be checked by referring to the relevant text.

H8S/2258, H8S/2239, H8S/2238, H8S/2237, H8S/2227 Groups

Hardware Manual

Renesas 16-Bit Single-Chip Microcomputer

H8S Family/H8S/2200 Series

| | | | |
|-----------|---|----------------------------------|--|
| H8S/2258 | HD64F2258 HD6432258 HD6432258W | H8S/2236R | HD6432236R HD6432236RW |
| H8S/2256 | HD6432256 HD6432256W | H8S/223 | HD6472237 HD6432237 |
| H8S/2239 | HD64F2239 HD6432239 HD6432239W | H8S/2235 H8S/2233 H8S/2227 | HD6432235 HD6432233 HD64F2227 HD6432227 |
| H8S/2238 | HD64F2238B HD6432238B HD6432238BW | H8S/2225 H8S/2224 H8S/2223 | HD6432225 HD6432224 HD6432223 |
| H8S/2238R | HD64F2238R HD6432238R HD6432238RW | | |
| H8S/2236B | HD6432236B HD6432236BW | | |

Notes regarding these materials

1. This document is provided for reference purposes only so that Renesas customers may select the appropriate Renesas products for their use. Renesas neither makes warranties or representations with respect to the accuracy or completeness of the information contained in this document nor grants any license to any intellectual property rights or any other rights of Renesas or any third party with respect to the information in this document.
2. Renesas shall have no liability for damages or infringement of any intellectual property or other rights arising out of the use of any information in this document, including, but not limited to, product data, diagrams, charts, programs, algorithms, and application circuit examples.
3. You should not use the products or the technology described in this document for the purpose of military applications such as the development of weapons of mass destruction or for the purpose of any other military use. When exporting the products or technology described herein, you should follow the applicable export control laws and regulations, and procedures required by such laws and regulations.
4. All information included in this document such as product data, diagrams, charts, programs, algorithms, and application circuit examples, 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 products listed in this document, please confirm the latest product information with a Renesas sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas such as that disclosed through our website. (<http://www.renesas.com>)
5. Renesas has used reasonable care in compiling the information included in this document, but Renesas assumes no liability whatsoever for any damages incurred as a result of errors or omissions in the information included in this document.
6. When using or otherwise relying on the information in this document, you should evaluate the information in light of the total system before deciding about the applicability of such information to the intended application. Renesas makes no representations, warranties or guaranties regarding the suitability of its products for any particular application and specifically disclaims any liability arising out of the application and use of the information in this document or Renesas products.
7. With the exception of products specified by Renesas as suitable for automobile applications, Renesas products are not designed, manufactured or tested for applications or otherwise in systems the failure or malfunction of which may cause a direct threat to human life or create a risk of human injury or which require especially high quality and reliability such as safety systems, or equipment or systems for transportation and traffic, healthcare, combustion control, aerospace and aeronautics, nuclear power, or undersea communication transmission. If you are considering the use of our products for such purposes, please contact a Renesas sales office beforehand. Renesas shall have no liability for damages arising out of the uses set forth above.
8. Notwithstanding the preceding paragraph, you should not use Renesas products for the purposes listed below:
 - (1) artificial life support devices or systems
 - (2) surgical implantations
 - (3) healthcare intervention (e.g., excision, administration of medication, etc.)
 - (4) any other purposes that pose a direct threat to human lifeRenesas shall have no liability for damages arising out of the uses set forth in the above and purchasers who elect to use Renesas products in any of the foregoing applications shall indemnify and hold harmless Renesas Technology Corp., its affiliated companies and their officers, directors, and employees against any and all damages arising out of such applications.
9. You should use the products described herein within the range specified by Renesas, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas shall have no liability for malfunctions or damages arising out of the use of Renesas products beyond such specified ranges.
10. Although Renesas endeavors to improve the quality and reliability of its products, IC products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Please be sure to implement safety measures to guard against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas 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 applicable measures. Among others, since the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
11. In case Renesas products listed in this document are detached from the products to which the Renesas products are attached or affixed, the risk of accident such as swallowing by infants and small children is very high. You should implement safety measures so that Renesas products may not be easily detached from your products. Renesas shall have no liability for damages arising out of such detachment.
12. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written approval from Renesas.
13. Please contact a Renesas sales office if you have any questions regarding the information contained in this document, Renesas semiconductor products, or if you have any other inquiries.

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins

Handle unused pins in accord with the directions given under Handling of Unused Pins in the manual.

- The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions may occur due to the false recognition of the pin state as an input signal. Unused pins should be handled as described under Handling of Unused Pins in the manual.

2. Processing at Power-on

The state of the product is undefined at the moment when power is supplied.

- The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.

In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.

In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

3. Prohibition of Access to Reserved Addresses

Access to reserved addresses is prohibited.

- The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

4. Clock Signals

After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

- When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

5. Differences between Products

Before changing from one product to another, i.e. to one with a different type number, confirm that the change will not lead to problems.

- The characteristics of MPU/MCU in the same group but having different type numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different type numbers, implement a system-evaluation test for each of the products.

Configuration of This Manual

This manual comprises the following items:

1. General Precautions in the Handling of MPU/MCU Products
2. Configuration of This Manual
3. Preface
4. Main Revisions for This Edition

The list of revisions is a summary of points that have been revised or added to earlier versions. This does not include all of the revised contents. For details, see the actual locations in this manual.

5. Contents
6. Overview
7. 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.

8. List of Registers
9. Electrical Characteristics
10. Appendix
11. Index

Preface

The H8S/2558 Group, H8S/2239 Group, H8S/2238 Group, H8S/2237 Group, and H8S/2227 Group are high-performance microcomputers made up of the internal 32-bit configuration H8S/2000 CPU as their cores, and the peripheral functions required to configure a system.

A single-power flash memory (F-ZTAT^{TM*}) version and masked ROM version are available for these LSIs' ROM. These versions provide flexibility as they can be reprogrammed in no time to cope with all situations from the early stages of mass production to full-scale mass production. This is particularly applicable to application devices of which the specifications frequently changeable.

On-chip peripheral functions of each microcomputer are summarized below.

Note: * F-ZTAT is a trademark of Renesas Technology Corp.

List of On-Chip Peripheral Functions:

| Group Name | H8S/2258 Group | H8S/2239 Group | H8S/2238 Group | H8S/2237 Group | H8S/2227 Group |
|--|-------------------|-------------------|-------------------|-------------------|-------------------|
| Microcomputer | | | H8S/2238B | | H8S/2227 |
| | | | H8S/2238R | H8S/2237 | H8S/2225 |
| | H8S/2258 | | H8S/2236B | H8S/2235 | H8S/2224 |
| | H8S/2256 | H8S/2239 | H8S/2236R | H8S/2233 | H8S/2223 |
| Bus controller (BSC) | O (16 bits) | O (16 bits) | O (16 bits) | O (16bits) | O (16 bits) |
| Data transfer controller (DTC) | O | O | O | O | O |
| DMA controller (DMAC) | — | O | — | — | — |
| PC break controller (PBC) | ×2 | ×2 | ×2 | ×2 | ×2 |
| 16-bit timer pulse unit (TPU) | ×6 | ×6 | ×6 | ×6 | ×3 |
| 8-bit timer (TMR) | ×4 | ×4 | ×4 | ×2 | ×2 |
| Watchdog timer (WDT) | ×2 | ×2 | ×2 | ×2 | ×2 |
| Serial communication interface (SCI) | ×4 | ×4 | ×4 | ×4 | ×3 |
| I ² C bus interface (IIC) | ×2 (option) | ×2 (option) | ×2 (option) | — | — |
| D/A converter | ×2 | ×2 | ×2 | ×2 | — |
| A/D converter | Analog input ×8 | ×8 | ×8 | ×8 | ×8 |
| IEBus TM * controller (IEB) | ×1 | — | — | — | — |

Note: * IEBus (Inter Equipment Bus) is a trademark of NEC Electronics Corp.

Target Users: This manual was written for users who will be using the H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, H8S/2237 Group, and H8S/2227 Group in the design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logical circuits, and microcomputers.

Objective: This manual was written to explain the H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, H8S/2237 Group, and H8S/2227 Group hardware functions and electrical characteristics of this LSI to the target users.
Refer to the H8S/2600 Series, H8S/2000 Series Software Manual for a detailed description of the instruction set.

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 descriptions on the CPU, system control functions, peripheral functions, and electrical characteristics.
- In order to understand the details of the CPU's functions
Read the H8S/2600 Series, H8S/2000 Series Software Manual.
- In order to understand the details of a register whose name is already known
Read the index that is the final part of the manual to find the page number of the entry on the register. The addresses, bits, and initial values of the registers are summarized in section 26, List of Registers.

| | | |
|--------|------------------|--|
| Rules: | Register name: | The following notation is used for cases when the same or a similar function, e.g., 16-bit timer pulse unit or serial communication, is implemented on more than one channel: XXX_N (XXX is the register name and N is the channel number) |
| | 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, and decimal is xxxx. |
| | Signal notation: | An overbar is added to a low-active signal: $\overline{\text{xxxx}}$ |

Related Manuals: The latest versions of all related manuals are available from our web site.
Please ensure you have the latest versions of all documents.
<http://www.renesas.com/>

H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, H8S/2237 Group, H8S/2227 Group manuals:

| Document Title | Document No. |
|--|--------------|
| H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, H8S/2237 Group, H8S/2227 Group Hardware Manual | This manual |
| H8S/2600 Series, H8S/2000 Series Software Manual | REJ09B0139 |

User's Manuals for Development Tools:

| Document Title | Document No. |
|--|---------------------|
| H8S, H8/300 Series C/C++ Compiler, Assembler, Optimized Linkage Editor User's Manual | REJ10J2039 |
| High-performance Embedded Workshop User's Manual | REJ10J2037 |

Application Notes:

| Document Title | Document No. |
|--|---------------------|
| H8S, H8/300 Series C/C++ Compiler Package Application Note | REJ05B0464 |

Main Revisions for This Edition

| Item | Page | Revision (See Manual for Details) |
|------|------|-----------------------------------|
|------|------|-----------------------------------|

1.3.2 Pin Arrangements in Each Mode 20 to 23 Table amended

Table 1.1 Pin Arrangements in Each Mode of H8S/2258 Group

| Pin Name | | | | |
|-------------------------------------|--------|--------|--------|---|
| Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode [*] |
| Note added | | | | |
| Note: * The NC should be left open. | | | | |

Table 1.2 Pin Arrangements in Each Mode of H8S/2239 Group 24 to 28 Table amended

| Pin No. | | Pin Name | | | | |
|--|-------------------------|----------|--------|--------|--------|--|
| TFP-100B | | | | | | Flash Memory Programmable Mode ^{*2} |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | TBP-112A ^{*1} | | | | | |
| FP-100BV | TBP-112AV ^{*1} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| Note added | | | | | | |
| Notes: 1. Supported only by HD64F2239. | | | | | | |
| 2. The NC should be left open. | | | | | | |

Table 1.3 Pin Arrangements in Each Mode of H8S/2238 Group 29 to 33 Table amended

| Pin Name | | | | |
|---------------------------------------|--------|--------|--------|--|
| Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*4} |
| Note added | | | | |
| Notes: 4. The NC should be left open. | | | | |

Table 1.4 Pin Arrangements in Each Mode of H8S/2237 Group 34 to 38 Table amended

| Pin Name | | | | |
|-------------------------------------|--------|--------|--------|------------------------|
| Mode 4 | Mode 5 | Mode 6 | Mode 7 | PROM Mode [*] |
| Note added | | | | |
| Note: * The NC should be left open. | | | | |

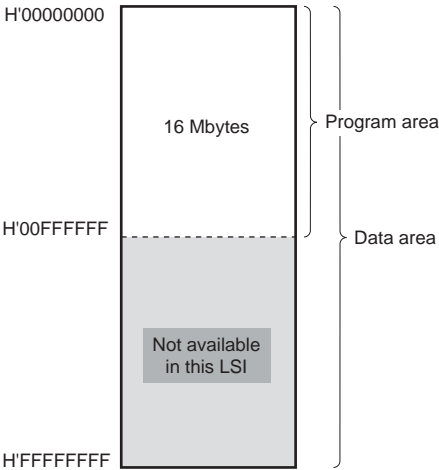
Table 1.5 Pin Arrangements in Each Mode of H8S/2227 Group 39 to 43 Table amended

| Pin Name | | | | |
|---------------------------------------|--------|--------|--------|--|
| Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*3} |
| Note added | | | | |
| Notes: 3. The NC should be left open. | | | | |

2.3 Address Space

Figure 2.5 Memory Map

70 Figure amended



(b) Advanced Mode

2.6 Instruction Set

79

Table amended

Table 2.1 Instruction Classification

| Function | Instructions | Size | Types |
|---------------|--|-------|-------|
| Data transfer | MOV | B/W/L | 5 |
| | POP ^{*1} , PUSH ^{*1} | W/L | |
| | LDM ^{*2} , STM ^{*2} | L | |

Note added

Notes: 5. Only register ER0 to ER6 should be used when using the STM/LDM instruction.

2.6.1 Table of Instructions

81

Classified by Function

Table amended

Table 2.3 Data Transfer Instructions

| Instruction | Size ^{*1} | Function |
|-------------------|--------------------|---|
| LDM ^{*2} | L | @SP+ → Rn (register list) Pops two or more general registers from the stack. |
| STM ^{*2} | L | Rn (register list) → @-SP Pushes two or more general registers onto the stack. |

Note amended

Notes: 1. Refers to the operand size.

B: Byte

W: Word

L: Longword

2. Only register ER0 to ER6 should be used when using the STM/LDM instruction.

| Item | Page | Revision (See Manual for Details) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--|----------|---|-----|---|---------------|-----|-------------|---|-------|---|-----|---|---|-------|---|-----|---|-------|---|-----|---|-------|---|-----|---|-------|---|-----|---|-------|---|-----|---|-------|---|-----|---|-------|---|-----|
| 4.8 Usage Note | 126 | Figure amended | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| <div>Figure 4.3 Operation When SP Value Is Odd</div> <div><p>The diagram illustrates the stack pointer (SP) and register states during two instructions:</p><ul style="list-style-type: none">Initial State: SP points to H'FFFEFF. The stack contains H'FFFEFA, H'FFFEFB, H'FFFEFC, H'FFFEFD, and H'FFFEFF.TRAPA instruction executed: SP is set to H'FFFEFF. Data is saved above SP.MOV.B R1L, @-ER7 executed: The contents of the CCR register are lost.</div> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5.6.5 IRQ Interrupt | 156 | 5.6.5 added | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5.6.6 NMI Interrupts Usage Notes | 156 | 5.6.6 added | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6.3.4 Operation in Transitions to Power-Down Modes | 161 | Description amended <ul style="list-style-type: none">When the SLEEP instruction causes a transition from high speed mode to subactive mode (figure 6.2 (B)). | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7.6.4 Wait Control (2) Pin Wait Insertion | 191 | Description amended <p>Setting the WAITE bit in BCRL to 1 enables wait insertion by means of the WAIT pin.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 9.2.5 DTC Transfer Count Register A (CRA) | 285 | Description amended <p>In repeat mode or block transfer mode, the CRA is divided into two parts; the upper 8 bits (CRAH) and the lower 8 bits (CRAL). In repeat mode, CRAH holds the number of transfers while CRAL functions as an 8-bit transfer counter (1 to 256). In block transfer mode, CRAH holds the block size while CRAL function as an 8-bit block size counter (1 to 256). CRAL is decremented by 1 every time data is transferred, and the contents of CRAH are sent when the count reaches H'00. This operation is repeated.</p> | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 10.1.2 Port 1 Data Register (P1DR) | 310 | Table amended <table><tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr><tr><td>7</td><td>P17DR</td><td>0</td><td>R/W</td><td rowspan="8">Output data for a pin is stored when the pin is specified as a general purpose output port.</td></tr><tr><td>6</td><td>P16DR</td><td>0</td><td>R/W</td></tr><tr><td>5</td><td>P15DR</td><td>0</td><td>R/W</td></tr><tr><td>4</td><td>P14DR</td><td>0</td><td>R/W</td></tr><tr><td>3</td><td>P13DR</td><td>0</td><td>R/W</td></tr><tr><td>2</td><td>P12DR</td><td>0</td><td>R/W</td></tr><tr><td>1</td><td>P11DR</td><td>0</td><td>R/W</td></tr><tr><td>0</td><td>P10DR</td><td>0</td><td>R/W</td></tr></table> | Bit | Bit Name | Initial Value | R/W | Description | 7 | P17DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. | 6 | P16DR | 0 | R/W | 5 | P15DR | 0 | R/W | 4 | P14DR | 0 | R/W | 3 | P13DR | 0 | R/W | 2 | P12DR | 0 | R/W | 1 | P11DR | 0 | R/W | 0 | P10DR | 0 | R/W |
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 7 | P17DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | P16DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 5 | P15DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | P14DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | P13DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | P12DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | P11DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | P10DR | 0 | R/W | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

10.2.2 Port 3 Data Register 316
(P3DR)

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 6 | P36DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 5 | P35DR | 0 | R/W | |
| 4 | P34DR | 0 | R/W | |
| 3 | P33DR | 0 | R/W | |
| 2 | P32DR | 0 | R/W | |
| 1 | P31DR | 0 | R/W | |
| 0 | P30DR | 0 | R/W | |

10.4.2 Port 7 Data Register 323
(P7DR)

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | P77DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | P76DR | 0 | R/W | |
| 5 | P75DR | 0 | R/W | |
| 4 | P74DR | 0 | R/W | |
| 3 | P73DR | 0 | R/W | |
| 2 | P72DR | 0 | R/W | |
| 1 | P71DR | 0 | R/W | |
| 0 | P70DR | 0 | R/W | |

10.6.2 Port A Data Register 328
(PADR)

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 3 | PA3DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 2 | PA2DR | 0 | R/W | |
| 1 | PA1DR | 0 | R/W | |
| 0 | PA0DR | 0 | R/W | |

10.7.2 Port B Data Register 333
(PBDR)

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PB7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PB6DR | 0 | R/W | |
| 5 | PB5DR | 0 | R/W | |
| 4 | PB4DR | 0 | R/W | |
| 3 | PB3DR | 0 | R/W | |
| 2 | PB2DR | 0 | R/W | |
| 1 | PB1DR | 0 | R/W | |
| 0 | PB0DR | 0 | R/W | |

10.8.2 Port C Data Register 340
(PCDR)

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PC7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PC6DR | 0 | R/W | |
| 5 | PC5DR | 0 | R/W | |
| 4 | PC4DR | 0 | R/W | |
| 3 | PC3DR | 0 | R/W | |
| 2 | PC2DR | 0 | R/W | |
| 1 | PC1DR | 0 | R/W | |
| 0 | PC0DR | 0 | R/W | |

10.9.2 Port D Data Register (PDDR) **344**

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PD7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PD6DR | 0 | R/W | |
| 5 | PD5DR | 0 | R/W | |
| 4 | PD4DR | 0 | R/W | |
| 3 | PD3DR | 0 | R/W | |
| 2 | PD2DR | 0 | R/W | |
| 1 | PD1DR | 0 | R/W | |
| 0 | PD0DR | 0 | R/W | |

10.10.2 Port E Data Register (PEDR) **347**

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PE7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PE6DR | 0 | R/W | |
| 5 | PE5DR | 0 | R/W | |
| 4 | PE4DR | 0 | R/W | |
| 3 | PE3DR | 0 | R/W | |
| 2 | PE2DR | 0 | R/W | |
| 1 | PE1DR | 0 | R/W | |
| 0 | PE0DR | 0 | R/W | |

10.11.2 Port F Data Register (PFDR) **351**

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PF7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PF6DR | 0 | R/W | |
| 5 | PF5DR | 0 | R/W | |
| 4 | PF4DR | 0 | R/W | |
| 3 | PF3DR | 0 | R/W | |
| 2 | PF2DR | 0 | R/W | |
| 1 | PF1DR | 0 | R/W | |
| 0 | PF0DR | 0 | R/W | |

10.12.2 Port G Data Register (PGDR) **355**

Table amended

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 5 | | Undefined | | Reserved These bits are always read as undefined value. |
| 4 | PG4DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 3 | PG3DR | 0 | R/W | |
| 2 | PG2DR | 0 | R/W | |
| 1 | PG1DR | 0 | R/W | |
| 0 | PG0DR | 0 | R/W | |

10.13 Handling of Unused Pins **358**

10.13 added

| Item | Page | Revision (See Manual for Details) | | | | | | | | | | | | | | | |
|--|----------|---|-----|---|---------------|-----|-------------|--------|-------|-------|-----|--|---|-------|---|-----|---|
| 11.3.1 Timer Control Register (TCR) | 367 | Table amended | | | | | | | | | | | | | | | |
| <table><tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr><tr><td>4</td><td>CKEG1</td><td>0</td><td>R/W</td><td>Clock Edge 1 and 0</td></tr><tr><td>3</td><td>CKEG0</td><td>0</td><td>R/W</td><td>These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. $\phi/4$ both edges = $\phi/2$ rising edge). If phase counting mode is used on channels 1, 2, 4*, and 5*, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is $\phi/4$ or slower. When the input clock is $\phi/1$ or when overflow/underflow of another channel is selected, this setting is ignored and the input clock is counted at the falling edge of ϕ. 00: Count at rising edge 01: Count at falling edge 1 : Count at both edges Legend: x: Don't care</td></tr></table> | | | Bit | Bit Name | Initial Value | R/W | Description | 4 | CKEG1 | 0 | R/W | Clock Edge 1 and 0 | 3 | CKEG0 | 0 | R/W | These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. $\phi/4$ both edges = $\phi/2$ rising edge). If phase counting mode is used on channels 1, 2, 4*, and 5*, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is $\phi/4$ or slower. When the input clock is $\phi/1$ or when overflow/underflow of another channel is selected, this setting is ignored and the input clock is counted at the falling edge of ϕ . 00: Count at rising edge 01: Count at falling edge 1 : Count at both edges Legend: x: Don't care |
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | |
| 4 | CKEG1 | 0 | R/W | Clock Edge 1 and 0 | | | | | | | | | | | | | |
| 3 | CKEG0 | 0 | R/W | These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. $\phi/4$ both edges = $\phi/2$ rising edge). If phase counting mode is used on channels 1, 2, 4*, and 5*, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is $\phi/4$ or slower. When the input clock is $\phi/1$ or when overflow/underflow of another channel is selected, this setting is ignored and the input clock is counted at the falling edge of ϕ . 00: Count at rising edge 01: Count at falling edge 1 : Count at both edges Legend: x: Don't care | | | | | | | | | | | | | |
| 13.3.1 Timer Counter (TCNT) | 468 | Description added TCNT is an 8-bit readable/writable up-counter. TCNT is initialized to H'00 when the TME bit in TCSR is cleared to 0. To initialize TCNT to H'00 while the timer is operating, write H'00 to TCNT directly. See 13.6.7, Notes on Initializing TCNT by Using the TME Bit. | | | | | | | | | | | | | | | |
| 13.6.3 Changing Value of PSS or CKS2 to CKS0 | 479 | Description amended If the PSS or CKS0 to CKS2 bits in TCSR are written to while the WDT is operating, errors could occur in the incrementation. Software must be used to stop the watchdog timer (by clearing the TME bit to 0) before changing the value of the PSS or CKS0 to CKS2 bits. | | | | | | | | | | | | | | | |
| 13.6.7 Notes on Initializing TCNT by Using the TME Bit | 479 | 13.6.7 added | | | | | | | | | | | | | | | |
| 15.3.8 Smart Card Mode Register (SCMR) | 570 | Table amended | | | | | | | | | | | | | | | |
| <table><tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr><tr><td>7 to 4</td><td>—</td><td>All 1</td><td>—</td><td>Reserved These bits are always read as 1, and cannot be modified.</td></tr><tr><td>3</td><td>SDIR</td><td>0</td><td>R/W</td><td>Smart Card Data Transfer Direction Selects the serial/parallel conversion format. 0: LSB-first in transfer 1: MSB-first in transfer The bit setting is valid only when the transfer data format is 8 bits. Except in the case of 7-bit data in asynchronous mode, either LSB-first or MSB-first may be selected regardless of the serial communication mode. For 7-bit data, set this bit to 0 to select LSB-first in transfer.</td></tr></table> | | | Bit | Bit Name | Initial Value | R/W | Description | 7 to 4 | — | All 1 | — | Reserved These bits are always read as 1, and cannot be modified. | 3 | SDIR | 0 | R/W | Smart Card Data Transfer Direction Selects the serial/parallel conversion format. 0: LSB-first in transfer 1: MSB-first in transfer The bit setting is valid only when the transfer data format is 8 bits. Except in the case of 7-bit data in asynchronous mode, either LSB-first or MSB-first may be selected regardless of the serial communication mode. For 7-bit data, set this bit to 0 to select LSB-first in transfer. |
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | |
| 7 to 4 | — | All 1 | — | Reserved These bits are always read as 1, and cannot be modified. | | | | | | | | | | | | | |
| 3 | SDIR | 0 | R/W | Smart Card Data Transfer Direction Selects the serial/parallel conversion format. 0: LSB-first in transfer 1: MSB-first in transfer The bit setting is valid only when the transfer data format is 8 bits. Except in the case of 7-bit data in asynchronous mode, either LSB-first or MSB-first may be selected regardless of the serial communication mode. For 7-bit data, set this bit to 0 to select LSB-first in transfer. | | | | | | | | | | | | | |

| Item | Page | Revision (See Manual for Details) | | | | | | | | | | | | | | | | | | | | |
|---|-------------------|---|------|--|---------------|------|-------------|-----------------------|------------------|---|-----|--|-----------------------------|-------------------|----------------------|----|--|----------------------------|-------------------|----------------------|----|--|
| 16.3.6 I ² C Bus Control Register (ICCR) | 644 | Table amended | | | | | | | | | | | | | | | | | | | | |
| <table><tr><th>Bit</th><th>Bit Name</th><th>Initial Value</th><th>R/W</th><th>Description</th></tr><tr><td>7</td><td>ICE</td><td>0</td><td>R/W</td><td>I²C Bus Interface Enable When this bit is set to 1, the I²C bus interface module is enabled to send/receive data and drive the bus since it is connected to the SCL and SDA pins. ICMR and ICDR can be accessed. SCL and SDA output is disabled (and input to SCL and SDA is enabled) when this bit is cleared to 0. SAR and SARX can be accessed.</td></tr></table> | | | Bit | Bit Name | Initial Value | R/W | Description | 7 | ICE | 0 | R/W | I ² C Bus Interface Enable When this bit is set to 1, the I ² C bus interface module is enabled to send/receive data and drive the bus since it is connected to the SCL and SDA pins. ICMR and ICDR can be accessed. SCL and SDA output is disabled (and input to SCL and SDA is enabled) when this bit is cleared to 0. SAR and SARX can be accessed. | | | | | | | | | | |
| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | |
| 7 | ICE | 0 | R/W | I ² C Bus Interface Enable When this bit is set to 1, the I ² C bus interface module is enabled to send/receive data and drive the bus since it is connected to the SCL and SDA pins. ICMR and ICDR can be accessed. SCL and SDA output is disabled (and input to SCL and SDA is enabled) when this bit is cleared to 0. SAR and SARX can be accessed. | | | | | | | | | | | | | | | | | | |
| 16.4.6 Slave Transmit Operation | 670 | Description added | | | | | | | | | | | | | | | | | | | | |
| 1. Initialize slave receive mode and wait for slave address reception. When making initial settings for slave receive mode, set the ACKE bit in ICCR to 1. This is necessary in order to enable reception of the acknowledge bit after entering slave transmit mode. | | | | | | | | | | | | | | | | | | | | | | |
| Description amended | | | | | | | | | | | | | | | | | | | | | | |
| 4. The master device drives SDA low at the 9th clock pulse, and returns an acknowledge signal. When the value of the ACKE bit in ICSR is 1, the acknowledge signal state is stored in the ACKB bit, so the ACKB bit can be used to determine whether the transfer operation was performed successfully. | | | | | | | | | | | | | | | | | | | | | | |
| | 671 | Description added | | | | | | | | | | | | | | | | | | | | |
| 10. When the stop condition is detected, that is, when SDA is changed from low to high when SCL is high, the BBSY flag in ICCR is cleared to 0 and the STOP flag in ICSR is set to 1. At the same time, the IRIC flag is set to 1. If the IRIC flag has been set, it is cleared to 0. To restart slave transmit mode operation, make the initial settings once again. | | | | | | | | | | | | | | | | | | | | | | |
| 16.6 Usage Notes | 677 | Table amended | | | | | | | | | | | | | | | | | | | | |
| <table><tr><th>Item</th><th>Symbol</th><th>Output Timing</th><th>Unit</th><th>Notes</th></tr><tr><td>SCL output cycle time</td><td>t_{SCL}</td><td>28 t_{osc} to 256 t_{osc}</td><td>ns</td><td>Figure 27.34</td></tr><tr><td>SCL output high pulse width</td><td>t_{SCLH}</td><td>0.5 t_{SCL}</td><td>ns</td><td></td></tr><tr><td>SCL output low pulse width</td><td>t_{SCLL}</td><td>0.5 t_{SCL}</td><td>ns</td><td></td></tr></table> | | | Item | Symbol | Output Timing | Unit | Notes | SCL output cycle time | t _{SCL} | 28 t _{osc} to 256 t _{osc} | ns | Figure 27.34 | SCL output high pulse width | t _{SCLH} | 0.5 t _{SCL} | ns | | SCL output low pulse width | t _{SCLL} | 0.5 t _{SCL} | ns | |
| Item | Symbol | Output Timing | Unit | Notes | | | | | | | | | | | | | | | | | | |
| SCL output cycle time | t _{SCL} | 28 t _{osc} to 256 t _{osc} | ns | Figure 27.34 | | | | | | | | | | | | | | | | | | |
| SCL output high pulse width | t _{SCLH} | 0.5 t _{SCL} | ns | | | | | | | | | | | | | | | | | | | |
| SCL output low pulse width | t _{SCLL} | 0.5 t _{SCL} | ns | | | | | | | | | | | | | | | | | | | |
| Table 16.7 I ² C Bus Timing (SCL and SDA Output) | | | | | | | | | | | | | | | | | | | | | | |

16.6 Usage Notes

681

Figure amended

Figure 16.22 Flowchart and Timing of Start Condition Instruction Issuance for Retransmission

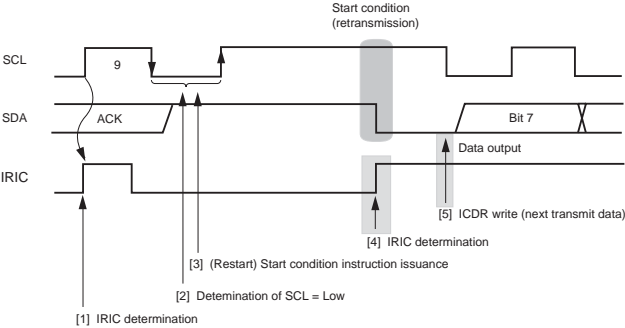


Figure 16.23 Timing of Stop 682 Condition Issuance

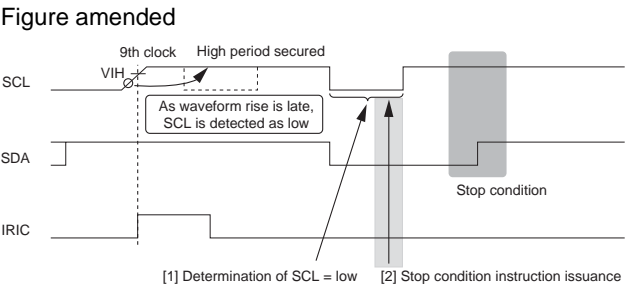
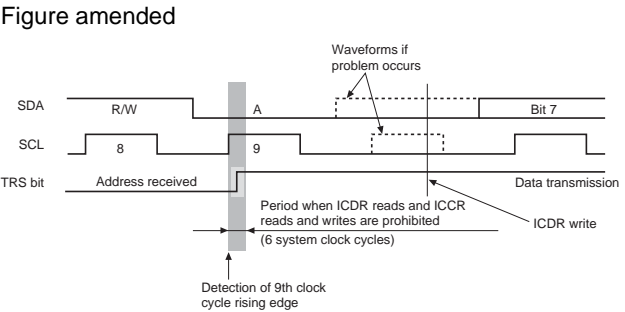


Figure 16.25 ICDR Read and ICCR Access Timing in Slave Transmit Mode



17.2 Input/Output Pins

691

Table amended

Table 17.1 Pin Configuration

| Pin Name | Symbol | I/O | Function |
|-------------------------|------------------|-------|---|
| Analog power supply pin | AV _{cc} | Input | Analog block power supply and reference voltage |
| Analog ground pin | AV _{ss} | Input | Analog block ground and reference voltage |
| Reference voltage pin | V _{ref} | Input | Reference voltage for A/D conversion |
| Analog input pin 0 | AN0 [†] | Input | Group 0 analog input pins |
| Analog input pin 1 | AN1 [†] | Input | |

Note added

Note: * In the case of the H8S/2239 Group, H8S/2227 Group, H8S/2238R, and H8S/2236R, AN0 and AN1 may be used only when Vcc = AVcc.

| Item | Page | Revision (See Manual for Details) |
|--|------|--|
| 17.8.4 Range of Analog Power Supply and Other Pin Settings | 705 | Description added <ul style="list-style-type: none"> Relationship between AV_{CC}, AV_{SS} and V_{CC}, V_{SS} Set AV_{SS} = V_{SS} as the relationship between AV_{CC}, AV_{SS} and V_{CC}, V_{SS}. If the A/D converter is not used, the AV_{CC} and AV_{SS} pins must not be left open. In addition, AN0 and AN1 may be used only when V_{CC} = AV_{CC} in the case of the H8S/2239 Group, H8S/2227 Group, H8S/2238R, and H8S/2236R. |

27.3.2 DC Characteristics 865

Table amended

Table 27.14 DC Characteristics (1)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------|----------------------------------|----------|---------------------|-----|----------------------|------|-----------------|
| Input high voltage | EXTAL, Ports 1, 3, 7, and A to G | V_{IH} | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4 ^{HS} and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3^{HS}$ | V | |

866

Note added

Notes: 5. When $V_{CC} < AV_{CC}$, the maximum value for P40 and P41 is $V_{CC} + 0.3$ V.

| | | |
|--|-----|--|
| 27.3.4 A/D Conversion Characteristics | 883 | Table condition amended |
| Table 27.23 A/D Conversion Characteristics | | Condition A (F-ZTAT version and masked ROM version): $V_{CC} = 2.7$ V to 3.6 V*, $AV_{CC} = 2.7$ V to 3.6 V*, $V_{ref} = 2.7$ V to AV_{CC} , $V_{SS} = AV_{SS} = 0$ V, $\phi = 2$ to 16.0 MHz, $T_a = -20^\circ\text{C}$ to $+75^\circ\text{C}$ (regular specifications) |
| | | Condition B (Masked ROM version): $V_{CC} = 2.2$ V to 3.6 V*, $AV_{CC} = 2.2$ V to 3.6 V*, $V_{ref} = 2.2$ V to AV_{CC} , $V_{SS} = AV_{SS} = 0$ V, $\phi = 2$ to 6.25 MHz, $T_a = -20^\circ\text{C}$ to $+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (wide-range specifications) |
| | | Condition C (F-ZTAT version and masked ROM version): $V_{CC} = 3.0$ V to 3.6 V*, $AV_{CC} = 3.0$ V to 3.6 V*, $V_{ref} = 3.0$ V to AV_{CC} , $V_{SS} = AV_{SS} = 0$ V, $\phi = 10.0$ to 20.0 MHz, $T_a = -20^\circ\text{C}$ to $+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (wide-range specifications) |
| | | Note added Note: * AN0 and AN1 can be used only when $V_{CC} = AV_{CC}$. |

| 27.5.2 DC Characteristics | 908 | Table amended | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------------------|----------------------------------|--|---------------------|-----|----------------------|------|-----------------|-----|------|-----------------|--------------------|----------------------------------|----------|---------------------|---|----------------|---|--|-----------------------------|--|---------------------|---|----------------------|---|--|
| Table 27.39 DC Characteristics (1) | | <table><tr><th>Item</th><th></th><th>Symbol</th><th>Min</th><th>Typ</th><th>Max</th><th>Unit</th><th>Test Conditions</th></tr><tr><td rowspan="2">Input high voltage</td><td>EXTAL, Ports 1, 3, 7, and A to G</td><td>V_{IH}</td><td>$V_{CC} \times 0.8$</td><td>—</td><td>$V_{CC} + 0.3$</td><td>V</td><td></td></tr><tr><td>Ports 4^{HS} and 9</td><td></td><td>$V_{CC} \times 0.8$</td><td>—</td><td>$AV_{CC} + 0.3^{HS}$</td><td>V</td><td></td></tr></table> | Item | | Symbol | Min | Typ | Max | Unit | Test Conditions | Input high voltage | EXTAL, Ports 1, 3, 7, and A to G | V_{IH} | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | | Ports 4 ^{HS} and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3^{HS}$ | V | |
| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions | | | | | | | | | | | | | | | | | | |
| Input high voltage | EXTAL, Ports 1, 3, 7, and A to G | V_{IH} | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | | | | | | | | | | | | | | | | | | | |
| | Ports 4 ^{HS} and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3^{HS}$ | V | | | | | | | | | | | | | | | | | | | |

Item

Page

Revision (See Manual for Details)

27.5.2 DC Characteristics

909

Note added

Table 27.39 DC Characteristics (1)

Notes: 5. When $V_{CC} < AV_{CC}$, the maximum value for P40 and P41 is $V_{CC} + 0.3 V$.

27.5.4 A/D Conversion Characteristics

923

Table condition amended

Table 27.47 A/D Conversion Characteristics

Condition A (F-ZTAT version and masked ROM version):
 $V_{CC} = 2.7 V$ to $3.6 V^*$, $AV_{CC} = 2.7 V$ to $3.6 V^*$,
 $V_{ref} = 2.7 V$ to AV_{CC} , $V_{SS} = AV_{SS} = 0 V$,
 $\phi = 2$ to $13.5 MHz$,
 $T_a = -20^{\circ}C$ to $+75^{\circ}C$ (regular specifications),
 $T_a = -40^{\circ}C$ to $+85^{\circ}C$ (wide-range specifications)
Condition B (F-ZTAT version):
 $V_{CC} = 2.2 V$ to $3.6 V^*$, $AV_{CC} = 2.2 V$ to $3.6 V^*$,
 $V_{ref} = 2.2 V$ to AV_{CC} , $V_{SS} = AV_{SS} = 0 V$,
 $\phi = 2$ to $6.25 MHz$,
 $T_a = -20^{\circ}C$ to $+75^{\circ}C$ (regular specifications)
Condition C (Masked ROM version):
 $V_{CC} = 2.2 V$ to $3.6 V^*$, $AV_{CC} = 2.2 V$ to $3.6 V^*$,
 $V_{ref} = 2.2 V$ to AV_{CC} , $V_{SS} = AV_{SS} = 0 V$,
 $\phi = 2$ to $6.25 MHz$,
 $T_a = -20^{\circ}C$ to $+75^{\circ}C$ (regular specifications),
 $T_a = -40^{\circ}C$ to $+85^{\circ}C$ (wide-range specifications)

Note added

Note: * AN0 and AN1 can be used only when $V_{CC} = AV_{CC}$.

27.6.2 DC Characteristics

928

Table amended

Table 27.51 DC Characteristics (1)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------|----------------------------------|---------------------|-----|------------------------|------|-----------------|
| Input high voltage | EXTAL, Ports 1, 3, 7, and A to G | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4 ^{NS} and 9 | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3^{NS} V$ | | |

929

Note added

Notes: 5. When $V_{CC} < AV_{CC}$, the maximum value for P40 and P41 is $V_{CC} + 0.3 V$.

| Item | Page | Revision (See Manual for Details) |
|--|------|---|
| 27.6.4 A/D Conversion Characteristics | 944 | Table condition amended |
| Table 27.57 A/D Conversion Characteristics | | <p>Condition A (ZTAT version):</p> $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}^*,$ $V_{ref} = 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 10 \text{ MHz},$ $T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications),}$ $T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$ <p>Condition B (F-ZTAT version, Masked ROM version):</p> $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}^*,$ $V_{ref} = 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 13.5 \text{ MHz},$ $T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications),}$ $T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$ <p>Condition C (Masked ROM version):</p> $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*,$ $V_{ref} = 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 6.25 \text{ MHz},$ $T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications),}$ $T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$ |

Note added

Note: * AN0 and AN1 can be used only when $V_{CC} = AV_{CC}$.

Appendix B Product Codes 970

Table amended

Table B.3 Product Codes of H8S/2238 Group

| Product Type | | Product Code | Mark Code | Package (Package Code) |
|--|--|----------------------------|--|--|
| H8S/2238B | Masked ROM version | 5-V version | HD6432238B | HD6432238B(***)TE 100-pin TQFP (TFP-100B) |
| | | | | HD6432238B(***)TF 100-pin TQFP (TFP-100G) |
| | | | | HD6432238B(***)F 100-pin QFP (FP-100A) |
| | | | | HD6432238B(***)FA 100-pin QFP (FP-100B) |
| | On-chip I ² C bus interface product (5-V version) | HD6432238BW | HD6432238BW(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432238BW(***)TF 100-pin TQFP (TFP-100G) |
| | | | | HD6432238BW(***)F 100-pin QFP (FP-100A) |
| | | | | HD6432238BW(***)FA 100-pin QFP (FP-100B) |
| | Masked ROM version | 3-V version, 2.2-V version | HD6432238R | HD6432238R(***)TE 100-pin TQFP (TFP-100B) |
| | | | | HD6432238R(***)TF 100-pin TQFP (TFP-100G) |
| | | | | HD6432238R(***)F 100-pin QFP (FP-100B) |
| | | | | HD6432238R(***)FA 100-pin QFP (FP-100B) |
| On-chip I ² C bus interface product (3-V version) | HD6432238RW | HD6432238RW(***)TE | 100-pin TQFP (TFP-100B) | |
| | | | HD6432238RW(***)TF 100-pin TQFP (TFP-100G) | |
| | | | HD6432238RW(***)F 100-pin QFP (FP-100B) | |
| | | | HD6432238RW(***)FA 100-pin QFP (FP-100B) | |

| Item | Page | Revision (See Manual for Details) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--|---|-------------------------|--------------|-----------|------------------------|------------------------------|------------------------|--------------------|-------------------------|--------------------|-------------------------|-------------------|-----------------------|--------------------|-----------------------|--|---------------------|-------------------------|---------------------|-------------------------|--------------------|-----------------------|---------------------|-----------------------|------------------------------|---------------------------------------|--------------------|-------------------------|--------------------|-------------------------|--------------------|-----------------------|--|--|--|---------------------|-------------------------|---------------------|-------------------------|---------------------|-----------------------|--|--|
| Appendix B Product Codes | 971 | Table amended | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Table B.3 Product Codes of H8S/2238 Group | | <table><tr><th>Product Type</th><th>Product Code</th><th>Mark Code</th><th>Package (Package Code)</th></tr><tr><td rowspan="8">H8S/2236B Masked ROM version</td><td rowspan="4">5-V version HD6432236B</td><td>HD6432236B(***)TE</td><td>100-pin TQFP (TFP-100B)</td></tr><tr><td>HD6432236B(***)TF</td><td>100-pin TQFP (TFP-100G)</td></tr><tr><td>HD6432236B(***)F</td><td>100-pin QFP (FP-100A)</td></tr><tr><td>HD6432236B(***)FA</td><td>100-pin QFP (FP-100B)</td></tr><tr><td rowspan="4">On-chip I²C bus interface product (5-V version) HD6432236BW</td><td>HD6432236BW(***)TE</td><td>100-pin TQFP (TFP-100B)</td></tr><tr><td>HD6432236BW(***)TF</td><td>100-pin TQFP (TFP-100G)</td></tr><tr><td>HD6432236BW(***)F</td><td>100-pin QFP (FP-100A)</td></tr><tr><td>HD6432236BW(***)FA</td><td>100-pin QFP (FP-100B)</td></tr><tr><td rowspan="8">H8S/2236R Masked ROM version</td><td rowspan="4">3-V version, 2.2-V version HD6432236R</td><td>HD6432236R(***)TE</td><td>100-pin TQFP (TFP-100B)</td></tr><tr><td>HD6432236R(***)TF</td><td>100-pin TQFP (TFP-100G)</td></tr><tr><td>HD6432236R(***)FA</td><td>100-pin QFP (FP-100B)</td></tr><tr><td></td><td></td></tr><tr><td rowspan="4">On-chip I²C bus interface product (3-V version) HD6432236RW</td><td>HD6432236RW(***)TE</td><td>100-pin TQFP (TFP-100B)</td></tr><tr><td>HD6432236RW(***)TF</td><td>100-pin TQFP (TFP-100G)</td></tr><tr><td>HD6432236RW(***)FA</td><td>100-pin QFP (FP-100B)</td></tr><tr><td></td><td></td></tr></table> | Product Type | Product Code | Mark Code | Package (Package Code) | H8S/2236B Masked ROM version | 5-V version HD6432236B | HD6432236B(***)TE | 100-pin TQFP (TFP-100B) | HD6432236B(***)TF | 100-pin TQFP (TFP-100G) | HD6432236B(***)F | 100-pin QFP (FP-100A) | HD6432236B(***)FA | 100-pin QFP (FP-100B) | On-chip I ² C bus interface product (5-V version) HD6432236BW | HD6432236BW(***)TE | 100-pin TQFP (TFP-100B) | HD6432236BW(***)TF | 100-pin TQFP (TFP-100G) | HD6432236BW(***)F | 100-pin QFP (FP-100A) | HD6432236BW(***)FA | 100-pin QFP (FP-100B) | H8S/2236R Masked ROM version | 3-V version, 2.2-V version HD6432236R | HD6432236R(***)TE | 100-pin TQFP (TFP-100B) | HD6432236R(***)TF | 100-pin TQFP (TFP-100G) | HD6432236R(***)FA | 100-pin QFP (FP-100B) | | | On-chip I ² C bus interface product (3-V version) HD6432236RW | HD6432236RW(***)TE | 100-pin TQFP (TFP-100B) | HD6432236RW(***)TF | 100-pin TQFP (TFP-100G) | HD6432236RW(***)FA | 100-pin QFP (FP-100B) | | |
| Product Type | Product Code | Mark Code | Package (Package Code) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H8S/2236B Masked ROM version | 5-V version HD6432236B | HD6432236B(***)TE | 100-pin TQFP (TFP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236B(***)TF | 100-pin TQFP (TFP-100G) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236B(***)F | 100-pin QFP (FP-100A) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236B(***)FA | 100-pin QFP (FP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | On-chip I ² C bus interface product (5-V version) HD6432236BW | HD6432236BW(***)TE | 100-pin TQFP (TFP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236BW(***)TF | 100-pin TQFP (TFP-100G) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236BW(***)F | 100-pin QFP (FP-100A) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236BW(***)FA | 100-pin QFP (FP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| H8S/2236R Masked ROM version | 3-V version, 2.2-V version HD6432236R | HD6432236R(***)TE | 100-pin TQFP (TFP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236R(***)TF | 100-pin TQFP (TFP-100G) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236R(***)FA | 100-pin QFP (FP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | On-chip I ² C bus interface product (3-V version) HD6432236RW | HD6432236RW(***)TE | 100-pin TQFP (TFP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236RW(***)TF | 100-pin TQFP (TFP-100G) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | HD6432236RW(***)FA | 100-pin QFP (FP-100B) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Appendix C Product Codes | 973 | Figure replaced | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure C.1 TFP-100B Package Dimensions | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure C.2 TFP-100G Package Dimensions | 974 | Figure replaced | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure C.3 FP-100A Package Dimensions | 975 | Figure replaced | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure C.4 FP-100B Package Dimensions | 976 | Figure replaced | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure C.5 BP-112 Package Dimensions | 977 | Figure replaced | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Figure C.6 TBP-112A, TBP-112AV Package Dimensions | 978 | Figure replaced | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

All trademarks and registered trademarks are the property of their respective owners.

Contents

| | | |
|-----------|--|----|
| Section 1 | Overview | 1 |
| 1.1 | Features | 1 |
| 1.2 | Internal Block Diagram | 4 |
| 1.3 | Pin Description | 9 |
| 1.3.1 | Pin Arrangement | 9 |
| 1.3.2 | Pin Arrangements in Each Mode | 20 |
| 1.3.3 | Pin Functions | 44 |
| Section 2 | CPU | 63 |
| 2.1 | Features | 63 |
| 2.1.1 | Differences between H8S/2600 CPU and H8S/2000 CPU | 64 |
| 2.1.2 | Differences from H8/300 CPU | 65 |
| 2.1.3 | Differences from H8/300H CPU | 65 |
| 2.2 | CPU Operating Modes | 66 |
| 2.2.1 | Normal Mode | 66 |
| 2.2.2 | Advanced Mode | 67 |
| 2.3 | Address Space | 70 |
| 2.4 | Register Configuration | 71 |
| 2.4.1 | General Registers | 72 |
| 2.4.2 | Program Counter (PC) | 73 |
| 2.4.3 | Extended Control Register (EXR) | 73 |
| 2.4.4 | Condition-Code Register (CCR) | 74 |
| 2.4.5 | Initial Values of CPU Registers | 75 |
| 2.5 | Data Formats | 76 |
| 2.5.1 | General Register Data Formats | 76 |
| 2.5.2 | Memory Data Formats | 78 |
| 2.6 | Instruction Set | 79 |
| 2.6.1 | Table of Instructions Classified by Function | 80 |
| 2.6.2 | Basic Instruction Formats | 89 |
| 2.7 | Addressing Modes and Effective Address Calculation | 90 |
| 2.7.1 | Register Direct—Rn | 91 |
| 2.7.2 | Register Indirect—@ERn | 91 |
| 2.7.3 | Register Indirect with Displacement—@(d:16, ERn) or @(d:32, ERn) | 91 |
| 2.7.4 | Register Indirect with Post-Increment—@ERn+ or Register Indirect with Pre-Decrement—@-ERn | 91 |
| 2.7.5 | Absolute Address—@aa:8, @aa:16, @aa:24, or @aa:32 | 91 |
| 2.7.6 | Immediate—#xx:8, #xx:16, or #xx:32 | 92 |

| | | |
|-------------------------------------|--|-----|
| 2.7.7 | Program-Counter Relative—@(d:8, PC) or @(d:16, PC)..... | 92 |
| 2.7.8 | Memory Indirect—@@aa:8 | 93 |
| 2.7.9 | Effective Address Calculation | 94 |
| 2.8 | Processing States..... | 96 |
| 2.9 | Usage Notes | 98 |
| 2.9.1 | TAS Instruction..... | 98 |
| 2.9.2 | STM/LDM Instruction | 98 |
| 2.9.3 | Bit Manipulation Instructions | 98 |
| 2.9.4 | Access Methods for Registers with Write-Only Bits | 100 |
| Section 3 MCU Operating Modes | | 103 |
| 3.1 | Operating Mode Selection | 103 |
| 3.2 | Register Descriptions | 104 |
| 3.2.1 | Mode Control Register (MDCR) | 104 |
| 3.2.2 | System Control Register (SYSCR) | 105 |
| 3.3 | Operating Mode Descriptions | 106 |
| 3.3.1 | Mode 4 | 106 |
| 3.3.2 | Mode 5 | 106 |
| 3.3.3 | Mode 6 | 107 |
| 3.3.4 | Mode 7 | 107 |
| 3.3.5 | Pin Functions | 108 |
| 3.4 | Memory Map in Each Operating Mode | 109 |
| Section 4 Exception Handling | | 119 |
| 4.1 | Exception Handling Types and Priority | 119 |
| 4.2 | Exception Sources and Exception Vector Table | 119 |
| 4.3 | Reset..... | 121 |
| 4.3.1 | Reset Types..... | 121 |
| 4.3.2 | Reset Exception Handling..... | 122 |
| 4.3.3 | Interrupts after Reset..... | 123 |
| 4.3.4 | State of On-Chip Peripheral Modules after Reset Release..... | 123 |
| 4.4 | Traces..... | 123 |
| 4.5 | Interrupts..... | 124 |
| 4.6 | Trap Instruction..... | 124 |
| 4.7 | Stack Status after Exception Handling..... | 125 |
| 4.8 | Usage Note..... | 126 |
| Section 5 Interrupt Controller..... | | 127 |
| 5.1 | Features..... | 127 |
| 5.2 | Input/Output Pins | 129 |

| | | |
|---|---|-----|
| 5.3 | Register Descriptions | 129 |
| 5.3.1 | Interrupt Priority Registers A to L, and O (IPRA to IPRL, IPRO) | 130 |
| 5.3.2 | IRQ Enable Register (IER) | 131 |
| 5.3.3 | IRQ Sense Control Registers H and L (ISCRH and ISCRL) | 131 |
| 5.3.4 | IRQ Status Register (ISR) | 134 |
| 5.4 | Interrupt Sources | 135 |
| 5.4.1 | External Interrupts | 135 |
| 5.4.2 | Internal Interrupts..... | 136 |
| 5.4.3 | Interrupt Exception Handling Vector Table..... | 136 |
| 5.5 | Operation..... | 142 |
| 5.5.1 | Interrupt Control Modes and Interrupt Operation | 142 |
| 5.5.2 | Interrupt Control Mode 0 | 145 |
| 5.5.3 | Interrupt Control Mode 2 | 147 |
| 5.5.4 | Interrupt Exception Handling Sequence | 148 |
| 5.5.5 | Interrupt Response Times | 150 |
| 5.5.6 | DTC and DMAC Activation by Interrupt | 151 |
| 5.6 | Usage Notes | 154 |
| 5.6.1 | Contention between Interrupt Generation and Disabling | 154 |
| 5.6.2 | Instructions that Disable Interrupts | 155 |
| 5.6.3 | When Interrupts are Disabled | 155 |
| 5.6.4 | Interrupts during Execution of EEPMOV Instruction..... | 155 |
| 5.6.5 | IRQ Interrupts Usage Notes | 156 |
| 5.6.6 | NMI Interrupts Usage Notes | 156 |
| Section 6 PC Break Controller (PBC) | | 157 |
| 6.1 | Features | 157 |
| 6.2 | Register Descriptions | 158 |
| 6.2.1 | Break Address Register A (BARA) | 158 |
| 6.2.2 | Break Address Register B (BARB)..... | 159 |
| 6.2.3 | Break Control Register A (BCRA) | 159 |
| 6.2.4 | Break Control Register B (BCRB)..... | 160 |
| 6.3 | Operation..... | 160 |
| 6.3.1 | PC Break Interrupt Due to Instruction Fetch | 160 |
| 6.3.2 | PC Break Interrupt Due to Data Access..... | 161 |
| 6.3.3 | Notes on PC Break Interrupt Handling | 161 |
| 6.3.4 | Operation in Transitions to Power-Down Modes..... | 161 |
| 6.3.5 | When Instruction Execution Is Delayed by One State | 162 |
| 6.4 | Usage Notes | 163 |
| 6.4.1 | Module Stop Mode Setting | 163 |
| 6.4.2 | PC Break Interrupts..... | 163 |

| | | |
|--------------------------------------|--|------------|
| 6.4.3 | CMFA and CMFB | 163 |
| 6.4.4 | PC Break Interrupt when DTC and DMAC Is Bus Master | 163 |
| 6.4.5 | PC Break Set for Instruction Fetch at Address Following BSR, JSR, JMP, TRAPA, RTE, and RTS Instruction..... | 163 |
| 6.4.6 | I Bit Set by LDC, ANDC, ORC, and XORC Instruction | 164 |
| 6.4.7 | PC Break Set for Instruction Fetch at Address Following Bcc Instruction..... | 164 |
| 6.4.8 | PC Break Set for Instruction Fetch at Branch Destination Address of Bcc Instruction | 164 |
| Section 7 Bus Controller..... | | 165 |
| 7.1 | Features | 165 |
| 7.2 | Input/Output Pins | 167 |
| 7.3 | Register Descriptions | 167 |
| 7.3.1 | Bus Width Control Register (ABWCR)..... | 168 |
| 7.3.2 | Access State Control Register (ASTCR) | 168 |
| 7.3.3 | Wait Control Registers H and L (WCRH, WCRL)..... | 169 |
| 7.3.4 | Bus Control Register H (BCRH) | 172 |
| 7.3.5 | Bus Control Register L (BCRL) | 173 |
| 7.3.6 | Pin Function Control Register (PFCR) | 174 |
| 7.4 | Bus Control | 175 |
| 7.4.1 | Area Divisions | 175 |
| 7.4.2 | Bus Specifications..... | 176 |
| 7.4.3 | Bus Interface for Each Area..... | 177 |
| 7.4.4 | Chip Select Signals | 178 |
| 7.5 | Basic Timing | 178 |
| 7.5.1 | On-Chip Memory (ROM, RAM) Access Timing | 179 |
| 7.5.2 | On-Chip Peripheral Module Access Timing | 180 |
| 7.5.3 | External Address Space Access Timing | 181 |
| 7.6 | Basic Bus Interface | 181 |
| 7.6.1 | Data Size and Data Alignment..... | 181 |
| 7.6.2 | Valid Stobes..... | 182 |
| 7.6.3 | Basic Timing..... | 183 |
| 7.6.4 | Wait Control | 190 |
| 7.7 | Burst ROM Interface..... | 192 |
| 7.7.1 | Basic Timing..... | 192 |
| 7.7.2 | Wait Control | 194 |
| 7.8 | Idle Cycle..... | 194 |
| 7.9 | Bus Release..... | 197 |
| 7.9.1 | Bus Release Usage Note | 198 |
| 7.10 | Bus Arbitration..... | 199 |

| | | |
|---------------------------------------|--|-----|
| 7.10.1 | Operation | 199 |
| 7.10.2 | Bus Transfer Timing | 200 |
| 7.10.3 | External Bus Release Usage Note | 200 |
| 7.11 | Resets and the Bus Controller | 201 |
| Section 8 DMA Controller (DMAC) | | 203 |
| 8.1 | Features | 203 |
| 8.2 | Input/Output Pins | 205 |
| 8.3 | Register Descriptions | 205 |
| 8.3.1 | Memory Address Registers (MARA and MARB) | 207 |
| 8.3.2 | I/O Address Registers (IOARA and IOARB) | 207 |
| 8.3.3 | Execute Transfer Count Registers (ETCRA and ETCRB) | 208 |
| 8.3.4 | DMA Control Registers (DMACRA and DMACRB) | 209 |
| 8.3.5 | DMA Band Control Registers H and L (DMABCRH and DMABCRL) | 218 |
| 8.3.6 | DMA Write Enable Register (DMAWER) | 229 |
| 8.3.7 | DMA Terminal Control Register (DMATCR) | 231 |
| 8.4 | Activation Sources | 231 |
| 8.4.1 | Activation by Internal Interrupt Request | 232 |
| 8.4.2 | Activation by External Request | 233 |
| 8.4.3 | Activation by Auto-Request | 233 |
| 8.5 | Operation | 234 |
| 8.5.1 | Transfer Modes | 234 |
| 8.5.2 | Sequential Mode | 236 |
| 8.5.3 | Idle Mode | 239 |
| 8.5.4 | Repeat Mode | 241 |
| 8.5.5 | Single Address Mode | 244 |
| 8.5.6 | Normal Mode | 248 |
| 8.5.7 | Block Transfer Mode | 251 |
| 8.5.8 | Basic Bus Cycles | 256 |
| 8.5.9 | DMA Transfer (Dual Address Mode) Bus Cycles | 257 |
| 8.5.10 | DMA Transfer (Single Address Mode) Bus Cycles | 265 |
| 8.5.11 | Multi-Channel Operation | 271 |
| 8.5.12 | Relation between DMAC and External Bus Requests, and DTC | 272 |
| 8.5.13 | DMAC and NMI Interrupts | 272 |
| 8.5.14 | Forced Termination of DMAC Operation | 273 |
| 8.5.15 | Clearing Full Address Mode | 274 |
| 8.6 | Interrupt Sources | 275 |
| 8.7 | Usage Notes | 276 |
| 8.7.1 | DMAC Register Access during Operation | 276 |
| 8.7.2 | Module Stop | 277 |

| | | |
|---|--|------------|
| 8.7.3 | Medium-Speed Mode..... | 277 |
| 8.7.4 | Activation by Falling Edge on $\overline{\text{DREQ}}$ Pin | 278 |
| 8.7.5 | Activation Source Acceptance | 278 |
| 8.7.6 | Internal Interrupt after End of Transfer..... | 278 |
| 8.7.7 | Channel Re-Setting | 279 |
| Section 9 Data Transfer Controller (DTC) | | 281 |
| 9.1 | Features | 281 |
| 9.2 | Register Descriptions | 282 |
| 9.2.1 | DTC Mode Register A (MRA) | 283 |
| 9.2.2 | DTC Mode Register B (MRB)..... | 284 |
| 9.2.3 | DTC Source Address Register (SAR)..... | 285 |
| 9.2.4 | DTC Destination Address Register (DAR)..... | 285 |
| 9.2.5 | DTC Transfer Count Register A (CRA) | 285 |
| 9.2.6 | DTC Transfer Count Register B (CRB)..... | 285 |
| 9.2.7 | DTC Enable Registers A to G, and I (DTCERA to DTCERG, and DTCERI) | 286 |
| 9.2.8 | DTC Vector Register (DTVECR)..... | 287 |
| 9.3 | Activation Sources | 288 |
| 9.4 | Location of Register Information and DTC Vector Table | 289 |
| 9.5 | Operation | 293 |
| 9.5.1 | Normal Mode..... | 294 |
| 9.5.2 | Repeat Mode | 294 |
| 9.5.3 | Block Transfer Mode | 295 |
| 9.5.4 | Chain Transfer | 297 |
| 9.5.5 | Interrupts..... | 298 |
| 9.5.6 | Operation Timing..... | 298 |
| 9.5.7 | Number of DTC Execution States | 300 |
| 9.6 | Procedures for Using DTC..... | 301 |
| 9.6.1 | Activation by Interrupt..... | 301 |
| 9.6.2 | Activation by Software | 301 |
| 9.7 | Examples of Use of the DTC | 302 |
| 9.7.1 | Normal Mode..... | 302 |
| 9.7.2 | Software Activation | 302 |
| 9.8 | Usage Notes | 303 |
| 9.8.1 | Module Stop Mode Setting | 303 |
| 9.8.2 | On-Chip RAM | 303 |
| 9.8.3 | DTCE Bit Setting..... | 303 |
| Section 10 I/O Ports..... | | 305 |
| 10.1 | Port 1 | 309 |

| | | |
|--------|---|-----|
| 10.1.1 | Port 1 Data Direction Register (P1DDR)..... | 309 |
| 10.1.2 | Port 1 Data Register (P1DR)..... | 310 |
| 10.1.3 | Port 1 Register (PORT1)..... | 310 |
| 10.1.4 | Pin Functions | 311 |
| 10.2 | Port 3..... | 315 |
| 10.2.1 | Port 3 Data Direction Register (P3DDR)..... | 315 |
| 10.2.2 | Port 3 Data Register (P3DR)..... | 316 |
| 10.2.3 | Port 3 Register (PORT3)..... | 316 |
| 10.2.4 | Port 3 Open Drain Control Register (P3ODR)..... | 317 |
| 10.2.5 | Pin Functions | 317 |
| 10.3 | Port 4..... | 321 |
| 10.3.1 | Port 4 Register (PORT4)..... | 321 |
| 10.3.2 | Pin Functions | 321 |
| 10.4 | Port 7..... | 322 |
| 10.4.1 | Port 7 Data Direction Register (P7DDR)..... | 322 |
| 10.4.2 | Port 7 Data Register (P7DR)..... | 323 |
| 10.4.3 | Port 7 Register (PORT7)..... | 323 |
| 10.4.4 | Pin Functions | 324 |
| 10.5 | Port 9..... | 327 |
| 10.5.1 | Port 9 Register (PORT9)..... | 327 |
| 10.5.2 | Pin Functions | 327 |
| 10.6 | Port A..... | 328 |
| 10.6.1 | Port A Data Direction Register (PADDDR) | 328 |
| 10.6.2 | Port A Data Register (PADR) | 328 |
| 10.6.3 | Port A Register (PORTA) | 329 |
| 10.6.4 | Port A Pull-Up MOS Control Register (PAPCR) | 329 |
| 10.6.5 | Port A Open Drain Control Register (PAODR)..... | 329 |
| 10.6.6 | Pin Functions | 330 |
| 10.6.7 | Input Pull-Up MOS States in Port A..... | 332 |
| 10.7 | Port B | 332 |
| 10.7.1 | Port B Data Direction Register (PBDDR)..... | 333 |
| 10.7.2 | Port B Data Register (PBDR) | 333 |
| 10.7.3 | Port B Register (PORTB) | 334 |
| 10.7.4 | Port B Pull-Up MOS Control Register (PBPCR)..... | 334 |
| 10.7.5 | Pin Functions | 334 |
| 10.7.6 | Input Pull-Up MOS States in Port B..... | 339 |
| 10.8 | Port C | 339 |
| 10.8.1 | Port C Data Direction Register (PCDDR)..... | 340 |
| 10.8.2 | Port C Data Register (PCDR) | 340 |
| 10.8.3 | Port C Register (PORTC) | 341 |

| | | |
|--|---|-----|
| 10.8.4 | Port C Pull-Up MOS Control Register (PCPCR) | 341 |
| 10.8.5 | Pin Functions | 342 |
| 10.8.6 | Input Pull-Up MOS States in Port C | 342 |
| 10.9 | Port D | 343 |
| 10.9.1 | Port D Data Direction Register (PDDDR) | 343 |
| 10.9.2 | Port D Data Register (PDDR) | 344 |
| 10.9.3 | Port D Register (PORTD) | 344 |
| 10.9.4 | Port D Pull-Up MOS Control Register (PDPCR) | 345 |
| 10.9.5 | Pin Functions | 345 |
| 10.9.6 | Input Pull-Up MOS States in Port D | 346 |
| 10.10 | Port E | 346 |
| 10.10.1 | Port E Data Direction Register (PEDDR) | 347 |
| 10.10.2 | Port E Data Register (PEDR) | 347 |
| 10.10.3 | Port E Register (PORTE) | 348 |
| 10.10.4 | Port E Pull-Up MOS Control Register (PEPCR) | 348 |
| 10.10.5 | Pin Functions | 349 |
| 10.10.6 | Input Pull-Up MOS States in Port E | 349 |
| 10.11 | Port F | 350 |
| 10.11.1 | Port F Data Direction Register (PFDDR) | 350 |
| 10.11.2 | Port F Data Register (PFDR) | 351 |
| 10.11.3 | Port F Register (PORTF) | 351 |
| 10.11.4 | Pin Functions | 352 |
| 10.12 | Port G | 354 |
| 10.12.1 | Port G Data Direction Register (PGDDR) | 354 |
| 10.12.2 | Port G Data Register (PGDR) | 355 |
| 10.12.3 | Port G Register (PORTG) | 355 |
| 10.12.4 | Pin Functions | 355 |
| 10.13 | Handling of Unused Pins | 358 |
| Section 11 16-Bit Timer Pulse Unit (TPU) | | 359 |
| 11.1 | Features | 359 |
| 11.2 | Input/Output Pins | 364 |
| 11.3 | Register Descriptions | 365 |
| 11.3.1 | Timer Control Register (TCR) | 367 |
| 11.3.2 | Timer Mode Register (TMDR) | 372 |
| 11.3.3 | Timer I/O Control Register (TIOR) | 373 |
| 11.3.4 | Timer Interrupt Enable Register (TIER) | 391 |
| 11.3.5 | Timer Status Register (TSR) | 393 |
| 11.3.6 | Timer Counter (TCNT) | 396 |
| 11.3.7 | Timer General Register (TGR) | 396 |

| | | |
|------------|---|-----|
| 11.3.8 | Timer Start Register (TSTR)..... | 396 |
| 11.3.9 | Timer Synchronous Register (TSYR)..... | 397 |
| 11.4 | Operation..... | 398 |
| 11.4.1 | Basic Functions..... | 398 |
| 11.4.2 | Synchronous Operation..... | 403 |
| 11.4.3 | Buffer Operation..... | 405 |
| 11.4.4 | Cascaded Operation..... | 409 |
| 11.4.5 | PWM Modes..... | 411 |
| 11.4.6 | Phase Counting Mode..... | 416 |
| 11.5 | Interrupt Sources..... | 423 |
| 11.6 | DTC Activation..... | 425 |
| 11.7 | DMAC Activation (H8S/2239 Group Only)..... | 425 |
| 11.8 | A/D Converter Activation..... | 426 |
| 11.9 | Operation Timing..... | 426 |
| 11.9.1 | Input/Output Timing..... | 426 |
| 11.9.2 | Interrupt Signal Timing..... | 430 |
| 11.10 | Usage Notes..... | 433 |
| 11.10.1 | Module Stop Mode Setting..... | 433 |
| 11.10.2 | Input Clock Restrictions..... | 433 |
| 11.10.3 | Caution on Cycle Setting..... | 434 |
| 11.10.4 | Contention between TCNT Write and Clear Operations..... | 434 |
| 11.10.5 | Contention between TCNT Write and Increment Operations..... | 435 |
| 11.10.6 | Contention between TGR Write and Compare Match..... | 436 |
| 11.10.7 | Contention between Buffer Register Write and Compare Match..... | 436 |
| 11.10.8 | Contention between TGR Read and Input Capture..... | 437 |
| 11.10.9 | Contention between TGR Write and Input Capture..... | 438 |
| 11.10.10 | Contention between Buffer Register Write and Input Capture..... | 438 |
| 11.10.11 | Contention between Overflow/Underflow and Counter Clearing..... | 439 |
| 11.10.12 | Contention between TCNT Write and Overflow/Underflow..... | 440 |
| 11.10.13 | Multiplexing of I/O Pins..... | 440 |
| 11.10.14 | Interrupts and Module Stop Mode..... | 440 |
| Section 12 | 8-Bit Timers..... | 441 |
| 12.1 | Features..... | 441 |
| 12.2 | Input/Output Pins..... | 443 |
| 12.3 | Register Descriptions..... | 443 |
| 12.3.1 | Timer Counter (TCNT)..... | 444 |
| 12.3.2 | Time Constant Register A (TCORA)..... | 444 |
| 12.3.3 | Time Constant Register B (TCORB)..... | 445 |
| 12.3.4 | Timer Control Register (TCR)..... | 445 |

| | | |
|------------|---|-----|
| 12.3.5 | Timer Control/Status Register (TCSR)..... | 447 |
| 12.4 | Operation | 452 |
| 12.4.1 | Pulse Output..... | 452 |
| 12.5 | Operation Timing..... | 453 |
| 12.5.1 | TCNT Incrementation Timing | 453 |
| 12.5.2 | Timing of CMFA and CMFB Setting when a Compare-Match Occurs | 454 |
| 12.5.3 | Timing of Timer Output when a Compare-Match Occurs | 455 |
| 12.5.4 | Timing of Compare-Match Clear when a Compare-Match Occurs | 455 |
| 12.5.5 | TCNT External Reset Timing..... | 456 |
| 12.5.6 | Timing of Overflow Flag (OVF) Setting | 456 |
| 12.6 | Operation with Cascaded Connection | 457 |
| 12.6.1 | 16-Bit Count Mode | 457 |
| 12.6.2 | Compare-Match Count Mode | 457 |
| 12.7 | Interrupt Sources..... | 458 |
| 12.7.1 | Interrupt Sources and DTC Activation | 458 |
| 12.7.2 | A/D Converter Activation..... | 458 |
| 12.8 | Usage Notes | 459 |
| 12.8.1 | Contention between TCNT Write and Clear..... | 459 |
| 12.8.2 | Contention between TCNT Write and Increment | 459 |
| 12.8.3 | Contention between TCOR Write and Compare-Match | 460 |
| 12.8.4 | Contention between Compare-Matches A and B..... | 461 |
| 12.8.5 | Switching of Internal Clocks and TCNT Operation..... | 461 |
| 12.8.6 | Contention between Interrupts and Module Stop Mode | 463 |
| 12.8.7 | Mode Setting of Cascaded Connection..... | 463 |
| Section 13 | Watchdog Timer (WDT) | 465 |
| 13.1 | Features..... | 465 |
| 13.2 | Input/Output Pins | 467 |
| 13.3 | Register Descriptions | 467 |
| 13.3.1 | Timer Counter (TCNT)..... | 468 |
| 13.3.2 | Timer Control/Status Register (TCSR)..... | 468 |
| 13.3.3 | Reset Control/Status Register (RSTCSR) (only WDT_0) | 472 |
| 13.4 | Operation | 473 |
| 13.4.1 | Watchdog Timer Mode | 473 |
| 13.4.2 | Interval Timer Mode..... | 474 |
| 13.4.3 | Timing of Setting Overflow Flag (OVF) | 475 |
| 13.4.4 | Timing of Setting Watchdog Timer Overflow Flag (WOVF) | 476 |
| 13.5 | Interrupt Sources..... | 476 |
| 13.6 | Usage Notes | 477 |
| 13.6.1 | Notes on Register Access..... | 477 |

| | | |
|-------------------|--|------------|
| 13.6.2 | Contention between Timer Counter (TCNT) Write and Increment | 478 |
| 13.6.3 | Changing Value of PSS or CKS2 to CKS0..... | 479 |
| 13.6.4 | Switching between Watchdog Timer Mode and Interval Timer Mode..... | 479 |
| 13.6.5 | Internal Reset in Watchdog Timer Mode..... | 479 |
| 13.6.6 | OVF Flag Clearing in Interval Timer Mode | 479 |
| 13.6.7 | Notes on Initializing TCNT by Using the TME Bit..... | 479 |
| Section 14 | IEBus™ Controller (IEB) [H8S/2258 Group] | 481 |
| 14.1 | Features..... | 481 |
| 14.1.1 | IEBus Communications Protocol..... | 483 |
| 14.1.2 | Communications Protocol..... | 485 |
| 14.1.3 | Transfer Data (Data Field Contents)..... | 493 |
| 14.1.4 | Bit Format | 496 |
| 14.2 | Input/Output Pins | 497 |
| 14.3 | Register Descriptions | 497 |
| 14.3.1 | IEBus Control Register (IECTR)..... | 498 |
| 14.3.2 | IEBus Command Register (IECMR) | 500 |
| 14.3.3 | IEBus Master Control Register (IEMCR)..... | 502 |
| 14.3.4 | IEBus Master Unit Address Register 1 (IEAR1) | 504 |
| 14.3.5 | IEBus Master Unit Address Register 2 (IEAR2) | 505 |
| 14.3.6 | IEBus Slave Address Setting Register 1 (IESA1)..... | 505 |
| 14.3.7 | IEBus Slave Address Setting Register 2 (IESA2)..... | 506 |
| 14.3.8 | IEBus Transmit Message Length Register (IETBFL)..... | 506 |
| 14.3.9 | IEBus Transmit Buffer Register (IETBR) | 507 |
| 14.3.10 | IEBus Reception Master Address Register 1 (IEMA1) | 508 |
| 14.3.11 | IEBus Reception Master Address Register 2 (IEMA2) | 508 |
| 14.3.12 | IEBus Receive Control Field Register (IERCTL)..... | 509 |
| 14.3.13 | IEBus Receive Message Length Register (IERBFL)..... | 509 |
| 14.3.14 | IEBus Receive Buffer Register (IERBR)..... | 510 |
| 14.3.15 | IEBus Lock Address Register 1 (IELA1) | 511 |
| 14.3.16 | IEBus Lock Address Register 2 (IELA2) | 511 |
| 14.3.17 | IEBus General Flag Register (IEFLG)..... | 512 |
| 14.3.18 | IEBus Transmit/Runaway Status Register (IETSR) | 515 |
| 14.3.19 | IEBus Transmit/Runaway Interrupt Enable Register (IEIET) | 518 |
| 14.3.20 | IEBus Transmit Error Flag Register (IETEF)..... | 519 |
| 14.3.21 | IEBus Receive Status Register (IERSR)..... | 522 |
| 14.3.22 | IEBus Receive Interrupt Enable Register (IEIER)..... | 524 |
| 14.3.23 | IEBus Receive Error Flag Register (IEREF) | 524 |
| 14.4 | Operation Descriptions..... | 527 |
| 14.4.1 | Master Transmit Operation | 527 |

| | | |
|------------|--|-----|
| 14.4.2 | Slave Receive Operation..... | 529 |
| 14.4.3 | Master Reception | 533 |
| 14.4.4 | Slave Transmission | 536 |
| 14.5 | Interrupt Sources..... | 540 |
| 14.6 | Usage Notes | 541 |
| 14.6.1 | Setting Module Stop Mode | 541 |
| 14.6.2 | TxRDY Flag and Underrun Error | 541 |
| 14.6.3 | RxRDY Flag and Overrun Error | 542 |
| 14.6.4 | Error Flag s in the IETEF..... | 542 |
| 14.6.5 | Error Flags in IEREF | 543 |
| 14.6.6 | Notes on Slave Transmission..... | 544 |
| 14.6.7 | Notes on DTC Specification | 545 |
| 14.6.8 | Error Handling in Transmission..... | 545 |
| 14.6.9 | Power-Down Mode Operation | 546 |
| 14.6.10 | Notes on Middle-Speed Mode | 546 |
| 14.6.11 | Notes on Register Access..... | 546 |
| Section 15 | Serial Communication Interface (SCI)..... | 547 |
| 15.1 | Features..... | 547 |
| 15.2 | Input/Output Pins | 551 |
| 15.3 | Register Descriptions | 551 |
| 15.3.1 | Receive Shift Register (RSR) | 552 |
| 15.3.2 | Receive Data Register (RDR)..... | 552 |
| 15.3.3 | Transmit Data Register (TDR)..... | 552 |
| 15.3.4 | Transmit Shift Register (TSR) | 553 |
| 15.3.5 | Serial Mode Register (SMR) | 553 |
| 15.3.6 | Serial Control Register (SCR)..... | 557 |
| 15.3.7 | Serial Status Register (SSR) | 563 |
| 15.3.8 | Smart Card Mode Register (SCMR)..... | 570 |
| 15.3.9 | Bit Rate Register (BRR) | 571 |
| 15.3.10 | Serial Expansion Mode Register (SEMR_0) | 581 |
| 15.4 | Operation in Asynchronous Mode | 585 |
| 15.4.1 | Data Transfer Format..... | 585 |
| 15.4.2 | Receive Data Sampling Timing and Reception Margin in Asynchronous Mode | 587 |
| 15.4.3 | Clock..... | 588 |
| 15.4.4 | SCI Initialization (Asynchronous Mode)..... | 589 |
| 15.4.5 | Serial Data Transmission (Asynchronous Mode) | 590 |
| 15.4.6 | Serial Data Reception (Asynchronous Mode)..... | 592 |
| 15.5 | Multiprocessor Communication Function..... | 596 |
| 15.5.1 | Multiprocessor Serial Data Transmission | 597 |

| | | |
|------------|---|-----|
| 15.5.2 | Multiprocessor Serial Data Reception..... | 599 |
| 15.6 | Operation in Clocked Synchronous Mode | 602 |
| 15.6.1 | Clock..... | 602 |
| 15.6.2 | SCI Initialization (Clocked Synchronous Mode)..... | 602 |
| 15.6.3 | Serial Data Transmission (Clocked Synchronous Mode) | 603 |
| 15.6.4 | Serial Data Reception (Clocked Synchronous Mode)..... | 606 |
| 15.6.5 | Simultaneous Serial Data Transmission and Reception (Clocked Synchronous Mode) | 608 |
| 15.7 | Operation in Smart Card Interface | 610 |
| 15.7.1 | Pin Connection Example..... | 610 |
| 15.7.2 | Data Format (Except for Block Transfer Mode)..... | 610 |
| 15.7.3 | Block Transfer Mode | 612 |
| 15.7.4 | Receive Data Sampling Timing and Reception Margin..... | 612 |
| 15.7.5 | Initialization | 613 |
| 15.7.6 | Serial Data Transmission (Except for Block Transfer Mode)..... | 614 |
| 15.7.7 | Serial Data Reception (Except for Block Transfer Mode)..... | 617 |
| 15.7.8 | Clock Output Control..... | 618 |
| 15.8 | SCI Select Function (H8S/2239 Group Only)..... | 620 |
| 15.9 | Interrupt Sources | 622 |
| 15.9.1 | Interrupts in Normal Serial Communication Interface Mode..... | 622 |
| 15.9.2 | Interrupts in Smart Card Interface Mode | 624 |
| 15.10 | Usage Notes | 625 |
| 15.10.1 | Module Stop Mode Setting | 625 |
| 15.10.2 | Break Detection and Processing (Asynchronous Mode Only)..... | 625 |
| 15.10.3 | Mark State and Break Detection (Asynchronous Mode Only) | 625 |
| 15.10.4 | Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only)..... | 625 |
| 15.10.5 | Restrictions on Use of DMAC or DTC..... | 626 |
| 15.10.6 | Operation in Case of Mode Transition | 626 |
| 15.10.7 | Switching from SCK Pin Function to Port Pin Function | 630 |
| 15.10.8 | Assignment and Selection of Registers | 631 |
| Section 16 | I ² C Bus Interface (IIC) (Option) | 633 |
| 16.1 | Features | 633 |
| 16.2 | Input/Output Pins | 636 |
| 16.3 | Register Descriptions | 636 |
| 16.3.1 | I ² C Bus Data Register (ICDR) | 637 |
| 16.3.2 | Slave Address Register (SAR)..... | 639 |
| 16.3.3 | Second Slave Address Register (SARX) | 639 |
| 16.3.4 | I ² C Bus Mode Register (ICMR)..... | 640 |

| | | |
|-------------------|--|------------|
| 16.3.5 | Serial Control Register X (SCRX)..... | 643 |
| 16.3.6 | I ² C Bus Control Register (ICCR)..... | 644 |
| 16.3.7 | I ² C Bus Status Register (ICSR)..... | 649 |
| 16.3.8 | DDC Switch Register (DDCSWR)..... | 653 |
| 16.4 | Operation | 653 |
| 16.4.1 | I ² C Bus Data Format | 653 |
| 16.4.2 | Initial Setting..... | 655 |
| 16.4.3 | Master Transmit Operation | 655 |
| 16.4.4 | Master Receive Operation..... | 659 |
| 16.4.5 | Slave Receive Operation..... | 664 |
| 16.4.6 | Slave Transmit Operation | 669 |
| 16.4.7 | IRIC Setting Timing and SCL Control | 672 |
| 16.4.8 | Operation Using the DTC | 673 |
| 16.4.9 | Noise Canceler | 674 |
| 16.4.10 | Initialization of Internal State | 674 |
| 16.5 | Interrupt Source | 676 |
| 16.6 | Usage Notes | 676 |
| 16.6.1 | Module Stop Mode Setting | 687 |
| Section 17 | A/D Converter | 689 |
| 17.1 | Features..... | 689 |
| 17.2 | Input/Output Pins | 691 |
| 17.3 | Register Descriptions | 692 |
| 17.3.1 | A/D Data Registers A to D (ADDRA to ADDR D) | 692 |
| 17.3.2 | A/D Control/Status Register (ADCSR) | 693 |
| 17.3.3 | A/D Control Register (ADCR) | 695 |
| 17.4 | Interface to Bus Master | 696 |
| 17.5 | Operation | 697 |
| 17.5.1 | Single Mode..... | 697 |
| 17.5.2 | Scan Mode | 698 |
| 17.5.3 | Input Sampling and A/D Conversion Time | 699 |
| 17.5.4 | External Trigger Input Timing | 701 |
| 17.6 | Interrupt Source | 701 |
| 17.7 | A/D Conversion Accuracy Definitions | 702 |
| 17.8 | Usage Notes | 704 |
| 17.8.1 | Module Stop Mode Setting | 704 |
| 17.8.2 | Permissible Signal Source Impedance | 704 |
| 17.8.3 | Influences on Absolute Accuracy | 704 |
| 17.8.4 | Range of Analog Power Supply and Other Pin Settings..... | 705 |
| 17.8.5 | Notes on Board Design | 705 |

| | | |
|-------------------|--|------------|
| 17.8.6 | Notes on Noise Countermeasures | 705 |
| Section 18 | D/A Converter..... | 707 |
| 18.1 | Features | 707 |
| 18.2 | Input/Output Pins | 708 |
| 18.3 | Register Description..... | 708 |
| 18.3.1 | D/A Data Registers 0 and 1 (DADR0 and DADR1)..... | 708 |
| 18.3.2 | D/A Control Register (DACR) | 709 |
| 18.4 | Operation..... | 710 |
| 18.5 | Usage Notes | 711 |
| 18.5.1 | Analog Power Supply Current in Power-Down Mode..... | 711 |
| 18.5.2 | Setting for Module Stop Mode..... | 711 |
| Section 19 | RAM | 713 |
| Section 20 | Flash Memory (F-ZTAT Version)..... | 715 |
| 20.1 | Features | 715 |
| 20.2 | Mode Transitions | 716 |
| 20.3 | Block Configuration..... | 720 |
| 20.4 | Input/Output Pins | 724 |
| 20.5 | Register Descriptions | 724 |
| 20.5.1 | Flash Memory Control Register 1 (FLMCR1)..... | 725 |
| 20.5.2 | Flash Memory Control Register 2 (FLMCR2)..... | 726 |
| 20.5.3 | Erase Block Register 1 (EBR1)..... | 726 |
| 20.5.4 | Erase Block Register 2 (EBR2)..... | 728 |
| 20.5.5 | RAM Emulation Register (RAMER)..... | 729 |
| 20.5.6 | Flash Memory Power Control Register (FLPWCR) | 731 |
| 20.5.7 | Serial Control Register X (SCRX) | 731 |
| 20.6 | On-Board Programming Modes | 732 |
| 20.6.1 | Boot Mode | 732 |
| 20.6.2 | Programming/Erasing in User Program Mode..... | 735 |
| 20.7 | Flash Memory Emulation in RAM..... | 735 |
| 20.8 | Flash Memory Programming/Erasing | 737 |
| 20.8.1 | Program/Program-Verify | 738 |
| 20.8.2 | Erase/Erase-Verify | 740 |
| 20.9 | Program/Erase Protection..... | 742 |
| 20.9.1 | Hardware Protection | 742 |
| 20.9.2 | Software Protection..... | 742 |
| 20.9.3 | Error Protection..... | 742 |
| 20.10 | Interrupt Handling When Programming/Erasing Flash Memory | 743 |

| | | |
|--|---|-----|
| 20.11 | Programmer Mode | 743 |
| 20.12 | Power-Down States for Flash Memory | 745 |
| 20.13 | Flash Memory Programming and Erasing Precautions | 745 |
| 20.14 | Note on Switching from F-ZTAT Version to Masked ROM Version | 751 |
| Section 21 Masked ROM | | 753 |
| 21.1 | Features | 753 |
| Section 22 PROM | | 755 |
| 22.1 | PROM Mode Setting | 755 |
| 22.2 | Socket Adapter and Memory Map | 755 |
| 22.3 | Programming | 759 |
| 22.3.1 | Programming and Verification | 759 |
| 22.3.2 | Programming Precautions | 763 |
| 22.3.3 | Reliability of Programmed Data | 764 |
| Section 23 Clock Pulse Generator | | 765 |
| 23.1 | Register Descriptions | 766 |
| 23.1.1 | System Clock Control Register (SCKCR) | 766 |
| 23.1.2 | Low-Power Control Register (LPWRCR) | 768 |
| 23.2 | System Clock Oscillator | 770 |
| 23.2.1 | Connecting a Crystal Resonator | 770 |
| 23.2.2 | External Clock Input | 771 |
| 23.2.3 | Notes on Switching External Clock | 777 |
| 23.3 | Duty Adjustment Circuit | 779 |
| 23.4 | Medium-Speed Clock Divider | 779 |
| 23.5 | Bus Master Clock Selection Circuit | 779 |
| 23.6 | System Clock when Using IEBus | 779 |
| 23.7 | Subclock Oscillator | 780 |
| 23.7.1 | Connecting 32.768-kHz Crystal Resonator | 780 |
| 23.7.2 | Handling Pins when Subclock Not Required | 781 |
| 23.8 | Subclock Waveform Generation Circuit | 781 |
| 23.9 | Usage Notes | 781 |
| 23.9.1 | Note on Crystal Resonator | 781 |
| 23.9.2 | Note on Board Design | 782 |
| Section 24 Power-Down Modes | | 783 |
| 24.1 | Register Description | 787 |
| 24.1.1 | Standby Control Register (SBYCR) | 787 |
| 24.1.2 | Module Stop Control Registers A to C (MSTPCRA to MSTPCRC) | 789 |

| | | |
|--------------------------------------|---|-----|
| 24.2 | Medium-Speed Mode..... | 790 |
| 24.3 | Sleep Mode | 791 |
| 24.3.1 | Transition to Sleep Mode..... | 791 |
| 24.3.2 | Exiting Sleep Mode..... | 792 |
| 24.4 | Software Standby Mode..... | 792 |
| 24.4.1 | Transition to Software Standby Mode | 792 |
| 24.4.2 | Clearing Software Standby Mode | 792 |
| 24.4.3 | Oscillation Settling Time after Clearing Software Standby Mode..... | 793 |
| 24.4.4 | Software Standby Mode Application Example | 794 |
| 24.5 | Hardware Standby Mode..... | 795 |
| 24.5.1 | Transition to Hardware Standby Mode | 795 |
| 24.5.2 | Clearing Hardware Standby Mode..... | 795 |
| 24.5.3 | Hardware Standby Mode Timing..... | 795 |
| 24.6 | Module Stop Mode..... | 796 |
| 24.7 | Watch Mode..... | 797 |
| 24.7.1 | Transition to Watch Mode | 797 |
| 24.7.2 | Exiting Watch Mode | 797 |
| 24.8 | Subsleep Mode..... | 798 |
| 24.8.1 | Transition to Subsleep Mode | 798 |
| 24.8.2 | Exiting Subsleep Mode | 798 |
| 24.9 | Subactive Mode..... | 799 |
| 24.9.1 | Transition to Subactive Mode | 799 |
| 24.9.2 | Exiting Subactive Mode..... | 799 |
| 24.10 | Direct Transitions..... | 800 |
| 24.10.1 | Direct Transitions from High-Speed Mode to Subactive Mode..... | 800 |
| 24.10.2 | Direct Transitions from Subactive Mode to High-Speed Mode..... | 800 |
| 24.11 | ϕ Clock Output Enable..... | 800 |
| 24.12 | Usage Notes | 801 |
| 24.12.1 | I/O Port Status..... | 801 |
| 24.12.2 | Current Dissipation during Oscillation Settling Wait Period | 801 |
| 24.12.3 | DTC and DMAC Module Stop | 801 |
| 24.12.4 | On-Chip Peripheral Module Interrupt..... | 801 |
| 24.12.5 | Writing to MSTPCR | 802 |
| 24.12.6 | Entering Subactive/Watch Mode and DMAC and DTC Module Stop | 802 |
| Section 25 Power Supply Circuit..... | | 803 |
| 25.1 | Overview..... | 803 |
| 25.2 | Power Supply Connection for H8S/2258 Group, H8S/2238B, and H8S/2236B (On-Chip Internal Power Supply Step-Down Circuit) | 803 |

| | | |
|---|--|-----|
| 25.3 | Power Supply Connection for H8S/2239 Group, H8S/2238R, H8S/2236R, H8S/2237 Group, and H8S/2227 Group (No Internal Power Supply Step-Down Circuit)..... | 804 |
| 25.4 | Note on Bypass Capacitor..... | 805 |
| Section 26 List of Registers..... | | 807 |
| 26.1 | Register Addresses (In Address Order)..... | 807 |
| 26.2 | Register Bits..... | 818 |
| 26.3 | Register States in Each Operating Mode..... | 830 |
| Section 27 Electrical Characteristics | | 839 |
| 27.1 | Power Supply Voltage and Operating Frequency Range..... | 839 |
| 27.2 | Electrical Characteristics of H8S/2258 Group..... | 844 |
| 27.2.1 | Absolute Maximum Ratings | 844 |
| 27.2.2 | DC Characteristics | 845 |
| 27.2.3 | AC Characteristics | 853 |
| 27.2.4 | A/D Conversion Characteristics..... | 860 |
| 27.2.5 | D/A Conversion Characteristics..... | 861 |
| 27.2.6 | Flash Memory Characteristics | 862 |
| 27.3 | Electrical Characteristics of H8S/2239 Group..... | 864 |
| 27.3.1 | Absolute Maximum Ratings | 864 |
| 27.3.2 | DC Characteristics | 865 |
| 27.3.3 | AC Characteristics | 873 |
| 27.3.4 | A/D Conversion Characteristics..... | 883 |
| 27.3.5 | D/A Conversion Characteristics..... | 884 |
| 27.3.6 | Flash Memory Characteristics | 885 |
| 27.4 | Electrical Characteristics of H8S/2238B and H8S/2236B | 887 |
| 27.4.1 | Absolute Maximum Ratings | 887 |
| 27.4.2 | DC Characteristics | 888 |
| 27.4.3 | AC Characteristics | 896 |
| 27.4.4 | A/D Conversion Characteristics..... | 904 |
| 27.4.5 | D/A Conversion Characteristics..... | 904 |
| 27.4.6 | Flash Memory Characteristics | 905 |
| 27.5 | Electrical Characteristics of H8S/2238R and H8S/2236R | 907 |
| 27.5.1 | Absolute Maximum Ratings | 907 |
| 27.5.2 | DC Characteristics | 908 |
| 27.5.3 | AC Characteristics | 915 |
| 27.5.4 | A/D Conversion Characteristics..... | 923 |
| 27.5.5 | D/A Conversion Characteristics..... | 924 |
| 27.5.6 | Flash Memory Characteristics | 925 |
| 27.6 | Electrical Characteristics of H8S/2237 Group and H8S/2227 Group | 927 |

| | | |
|--|--|-----|
| 27.6.1 | Absolute Maximum Ratings | 927 |
| 27.6.2 | DC Characteristics | 928 |
| 27.6.3 | AC Characteristics | 937 |
| 27.6.4 | A/D Conversion Characteristics..... | 944 |
| 27.6.5 | D/A Conversion Characteristics..... | 945 |
| 27.6.6 | Flash Memory Characteristics | 946 |
| 27.7 | Operating Timing | 948 |
| 27.7.1 | Clock Timing | 948 |
| 27.7.2 | Control Signal Timing | 949 |
| 27.7.3 | Bus Timing | 950 |
| 27.7.4 | Timing of On-Chip Peripheral Modules | 957 |
| 27.8 | Usage Note..... | 961 |
| Appendix A I/O Port States in Each Pin State | | 963 |
| A.1 | I/O Port State in Each Pin State | 963 |
| Appendix B Product Codes..... | | 968 |
| Appendix C Package Dimensions..... | | 973 |
| Index | | 979 |

Figures

Section 1 Overview

| | | |
|-------------|--|----|
| Figure 1.1 | Internal Block Diagram of H8S/2258 Group | 4 |
| Figure 1.2 | Internal Block Diagram of H8S/2239 Group | 5 |
| Figure 1.3 | Internal Block Diagram of H8S/2238 Group | 6 |
| Figure 1.4 | Internal Block Diagram of H8S/2237 Group | 7 |
| Figure 1.5 | Internal Block Diagram of H8S/2227 Group | 8 |
| Figure 1.6 | Pin Arrangement of H8S/2258 Group (TFP-100B, TFP-100BV, FP-100B, FP-100BV: Top View)..... | 9 |
| Figure 1.7 | Pin Arrangement of H8S/2258 Group (FP-100A, FP-100AV: Top View)..... | 10 |
| Figure 1.8 | Pin Arrangement of H8S/2239 Group (TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)..... | 11 |
| Figure 1.9 | Pin Arrangement of H8S/2239 Group (TBP-112A, TBP-112AV: Top View, Only for HD64F2239)..... | 12 |
| Figure 1.10 | Pin Arrangement of H8S/2238 Group (TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)..... | 13 |
| Figure 1.11 | Pin Arrangement of H8S/2238 Group (FP-100A, FP-100AV: Top View, Only for H8S/2238B and H8S/2236B) | 14 |
| Figure 1.12 | Pin Arrangement of H8S/2238 Group (BP-112, BP-112V, TBP-112A, TBP-112AV: Top View, Only for HD64F2238R)..... | 15 |
| Figure 1.13 | Pin Arrangement of H8S/2237 Group (TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)..... | 16 |
| Figure 1.14 | Pin Arrangement of H8S/2237 Group (FP-100A, FP-100AV: Top View)..... | 17 |
| Figure 1.15 | Pin Arrangement of H8S/2227 Group (TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)..... | 18 |
| Figure 1.16 | Pin Arrangement of H8S/2227 Group (FP-100A, FP-100AV: Top View, Only for HD6432227)..... | 19 |

Section 2 CPU

| | | |
|------------|---|----|
| Figure 2.1 | Exception Vector Table (Normal Mode)..... | 67 |
| Figure 2.2 | Stack Structure in Normal Mode..... | 67 |
| Figure 2.3 | Exception Vector Table (Advanced Mode)..... | 68 |
| Figure 2.4 | Stack Structure in Advanced Mode..... | 69 |
| Figure 2.5 | Memory Map..... | 70 |
| Figure 2.6 | CPU Registers | 71 |
| Figure 2.7 | Usage of General Registers | 72 |
| Figure 2.8 | Stack Status | 73 |
| Figure 2.9 | General Register Data Formats (1)..... | 76 |

| | | |
|-------------|--|-----|
| Figure 2.9 | General Register Data Formats (2)..... | 77 |
| Figure 2.10 | Memory Data Formats | 78 |
| Figure 2.11 | Instruction Formats (Examples)..... | 90 |
| Figure 2.12 | Branch Address Specification in Memory Indirect Mode..... | 93 |
| Figure 2.13 | State Transitions..... | 97 |
| Figure 2.14 | Flowchart for Access Methods for Registers That Include Write-Only Bits..... | 101 |

Section 3 MCU Operating Modes

| | | |
|------------|---|-----|
| Figure 3.1 | H8S/2258 Memory Map in Each Operating Mode | 109 |
| Figure 3.2 | H8S/2256 Memory Map in Each Operating Mode | 110 |
| Figure 3.3 | H8S/2239 Memory Map in Each Operating Mode | 111 |
| Figure 3.4 | H8S/2238B and H8S/2238R Memory Map in Each Operating Mode | 112 |
| Figure 3.5 | H8S/2236B and H8S/2236R Memory Map in Each Operating Mode | 113 |
| Figure 3.6 | H8S/2237 and H8S/2227 Memory Map in Each Operating Mode..... | 114 |
| Figure 3.7 | H8S/2235 and H8S/2225 Memory Map in Each Operating Mode..... | 115 |
| Figure 3.8 | H8S/2224 Memory Map in Each Operating Mode | 116 |
| Figure 3.9 | H8S/2233 and H8S/2223 Memory Map in Each Operating Mode..... | 117 |

Section 4 Exception Handling

| | | |
|------------|--|-----|
| Figure 4.1 | Reset Sequence (Mode 4)..... | 122 |
| Figure 4.2 | Stack Status after Exception Handling (Advanced Mode)..... | 125 |
| Figure 4.3 | Operation When SP Value Is Odd..... | 126 |

Section 5 Interrupt Controller

| | | |
|------------|---|-----|
| Figure 5.1 | Block Diagram of Interrupt Controller | 128 |
| Figure 5.2 | Block Diagram of IRQn Interrupts | 135 |
| Figure 5.3 | Set Timing for IRQnF..... | 136 |
| Figure 5.4 | Block Diagram of Interrupt Control Operation..... | 143 |
| Figure 5.5 | Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 0. | 146 |
| Figure 5.6 | Flowchart of Procedure Up to Interrupt Acceptance in Control Mode 2..... | 148 |
| Figure 5.7 | Interrupt Exception Handling | 149 |
| Figure 5.8 | DTC and DMAC Interrupt Control..... | 152 |
| Figure 5.9 | Contention between Interrupt Generation and Disabling..... | 155 |

Section 6 PC Break Controller (PBC)

| | | |
|------------|--|-----|
| Figure 6.1 | Block Diagram of PC Break Controller | 158 |
| Figure 6.2 | Operation in Power-Down Mode Transitions | 162 |

Section 7 Bus Controller

| | | |
|------------|---------------------------------------|-----|
| Figure 7.1 | Block Diagram of Bus Controller | 166 |
|------------|---------------------------------------|-----|

| | | |
|-------------|---|-----|
| Figure 7.2 | Overview of Area Divisions..... | 175 |
| Figure 7.3 | \overline{CSn} Signal Output Timing (n = 0 to 7)..... | 178 |
| Figure 7.4 | On-Chip Memory Access Cycle | 179 |
| Figure 7.5 | Pin States during On-Chip Memory Access..... | 179 |
| Figure 7.6 | On-Chip Peripheral Module Access Cycle | 180 |
| Figure 7.7 | Pin States during On-Chip Peripheral Module Access..... | 180 |
| Figure 7.8 | Access Sizes and Data Alignment Control (8-Bit Access Space) | 181 |
| Figure 7.9 | Access Sizes and Data Alignment Control (16-Bit Access Space) | 182 |
| Figure 7.10 | Bus Timing for 8-Bit 2-State Access Space..... | 183 |
| Figure 7.11 | Bus Timing for 8-Bit 3-State Access Space..... | 184 |
| Figure 7.12 | Bus Timing for 16-Bit 2-State Access Space (1) (Even Address Byte Access)... | 185 |
| Figure 7.13 | Bus Timing for 16-Bit 2-State Access Space (2) (Odd Address Byte Access) | 186 |
| Figure 7.14 | Bus Timing for 16-Bit 2-State Access Space (3) (Word Access) | 187 |
| Figure 7.15 | Bus Timing for 16-Bit 3-State Access Space (1) (Even Address Byte Access)... | 188 |
| Figure 7.16 | Bus Timing for 16-Bit 3-State Access Space (2) (Odd Address Byte Access) | 189 |
| Figure 7.17 | Bus Timing for 16-Bit 3-State Access Space (3) (Word Access) | 190 |
| Figure 7.18 | Example of Wait State Insertion Timing..... | 191 |
| Figure 7.19 | Example of Burst ROM Access Timing (When $AST0 = BRSTS1 = 1$) | 193 |
| Figure 7.20 | Example of Burst ROM Access Timing (When $AST0 = BRSTS1 = 0$) | 193 |
| Figure 7.21 | Example of Idle Cycle Operation (1) | 194 |
| Figure 7.22 | Example of Idle Cycle Operation (2) | 195 |
| Figure 7.23 | Relationship between Chip Select (\overline{CS}) and Read (\overline{RD}) | 196 |
| Figure 7.24 | Bus-Released State Transition Timing..... | 198 |

Section 8 DMA Controller (DMAC)

| | | |
|-------------|---|-----|
| Figure 8.1 | Block Diagram of DMAC | 204 |
| Figure 8.2 | Areas for Register Re-Setting by DTC (Channel 0A)..... | 230 |
| Figure 8.3 | Operation in Sequential Mode..... | 237 |
| Figure 8.4 | Example of Sequential Mode Setting Procedure..... | 238 |
| Figure 8.5 | Operation in Idle Mode | 239 |
| Figure 8.6 | Example of Idle Mode Setting Procedure | 240 |
| Figure 8.7 | Operation in Repeat mode..... | 242 |
| Figure 8.8 | Example of Repeat Mode Setting Procedure | 243 |
| Figure 8.9 | Data Bus in Single Address Mode | 244 |
| Figure 8.10 | Operation in Single Address Mode (when Sequential Mode Is Specified) | 246 |
| Figure 8.11 | Example of Single Address Mode Setting Procedure (when Sequential Mode Is Specified) | 247 |
| Figure 8.12 | Operation in Normal Mode | 249 |
| Figure 8.13 | Example of Normal Mode Setting Procedure | 250 |
| Figure 8.14 | Operation in Block Transfer Mode ($BLKDIR = 0$)..... | 252 |

| | | |
|-------------|---|-----|
| Figure 8.15 | Operation in Block Transfer Mode (BLKDIR = 1)..... | 253 |
| Figure 8.16 | Operation Flow in Block Transfer Mode | 254 |
| Figure 8.17 | Example of Block Transfer Mode Setting Procedure..... | 255 |
| Figure 8.18 | Example of DMA Transfer Bus Timing..... | 256 |
| Figure 8.19 | Example of Short Address Mode Transfer..... | 257 |
| Figure 8.20 | Example of Full Address Mode Transfer (Cycle Steal) | 258 |
| Figure 8.21 | Example of Full Address Mode Transfer (Burst Mode)..... | 259 |
| Figure 8.22 | Example of Full Address Mode Transfer (Block Transfer Mode) | 260 |
| Figure 8.23 | Example of $\overline{\text{DREQ}}$ Pin Falling Edge Activated Normal Mode Transfer | 261 |
| Figure 8.24 | Example of $\overline{\text{DREQ}}$ Pin Falling Edge Activated Block Transfer Mode Transfer.. | 262 |
| Figure 8.25 | Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Normal Mode Transfer..... | 263 |
| Figure 8.26 | Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Block Transfer Mode Transfer..... | 264 |
| Figure 8.27 | Example of Single Address Mode Transfer (Byte Read) | 265 |
| Figure 8.28 | Example of Single Address Mode (Word Read) Transfer..... | 266 |
| Figure 8.29 | Example of Single Address Mode Transfer (Byte Write) | 267 |
| Figure 8.30 | Example of Single Address Mode Transfer (Word Write)..... | 268 |
| Figure 8.31 | Example of $\overline{\text{DREQ}}$ Pin Falling Edge Activated Single Address Mode Transfer | 269 |
| Figure 8.32 | Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Single Address Mode Transfer..... | 270 |
| Figure 8.33 | Example of Multi-Channel Transfer | 272 |
| Figure 8.34 | Example of Procedure for Continuing Transfer on Channel Interrupted by NMI Interrupt..... | 273 |
| Figure 8.35 | Example of Procedure for Forcibly Terminating DMAC Operation..... | 274 |
| Figure 8.36 | Example of Procedure for Clearing Full Address Mode | 274 |
| Figure 8.37 | Block Diagram of Transfer End/Transfer Break Interrupt | 275 |
| Figure 8.38 | DMAC Register Update Timing | 276 |
| Figure 8.39 | Contention between DMAC Register Update and CPU Read..... | 277 |

Section 9 Data Transfer Controller (DTC)

| | | |
|-------------|--|-----|
| Figure 9.1 | Block Diagram of DTC | 282 |
| Figure 9.2 | Block Diagram of DTC Activation Source Control | 289 |
| Figure 9.3 | The Location of the DTC Register Information in the Address Space..... | 290 |
| Figure 9.4 | Correspondence between DTC Vector Address and Register Information | 290 |
| Figure 9.5 | Flowchart of DTC Operation | 293 |
| Figure 9.6 | Memory Mapping in Normal Mode | 294 |
| Figure 9.7 | Memory Mapping in Repeat Mode | 295 |
| Figure 9.8 | Memory Mapping in Block Transfer Mode | 296 |
| Figure 9.9 | Chain Transfer Operation..... | 297 |
| Figure 9.10 | DTC Operation Timing (Example in Normal Mode or Repeat Mode) | 298 |
| Figure 9.11 | DTC Operation Timing (Example of Block Transfer Mode, with Block Size of 2) | 299 |

| | | |
|-------------|--|-----|
| Figure 9.12 | DTC Operation Timing (Example of Chain Transfer) | 299 |
|-------------|--|-----|

Section 10 I/O Ports

| | | |
|-------------|-----------------------------------|-----|
| Figure 10.1 | Types of Open Drain Outputs | 318 |
|-------------|-----------------------------------|-----|

Section 11 16-Bit Timer Pulse Unit (TPU)

| | | |
|--------------|---|-----|
| Figure 11.1 | Block Diagram of TPU (H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group) | 362 |
| Figure 11.2 | Block Diagram of TPU (H8S/2227 Group) | 363 |
| Figure 11.3 | Example of Counter Operation Setting Procedure | 398 |
| Figure 11.4 | Free-Running Counter Operation..... | 399 |
| Figure 11.5 | Periodic Counter Operation..... | 400 |
| Figure 11.6 | Example of Setting Procedure for Waveform Output by Compare Match..... | 400 |
| Figure 11.7 | Example of 0 Output/1 Output Operation | 401 |
| Figure 11.8 | Example of Toggle Output Operation | 401 |
| Figure 11.9 | Example of Setting Procedure for Input Capture Operation | 402 |
| Figure 11.10 | Example of Input Capture Operation | 403 |
| Figure 11.11 | Example of Synchronous Operation Setting Procedure | 404 |
| Figure 11.12 | Example of Synchronous Operation..... | 405 |
| Figure 11.13 | Compare Match Buffer Operation..... | 406 |
| Figure 11.14 | Input Capture Buffer Operation | 406 |
| Figure 11.15 | Example of Buffer Operation Setting Procedure..... | 407 |
| Figure 11.16 | Example of Buffer Operation (1) | 408 |
| Figure 11.17 | Example of Buffer Operation (2) | 409 |
| Figure 11.18 | Cascaded Operation Setting Procedure | 410 |
| Figure 11.19 | Example of Cascaded Operation (1) | 410 |
| Figure 11.20 | Example of Cascaded Operation (2) | 411 |
| Figure 11.21 | Example of PWM Mode Setting Procedure | 413 |
| Figure 11.22 | Example of PWM Mode Operation (1)..... | 414 |
| Figure 11.23 | Example of PWM Mode Operation (2)..... | 414 |
| Figure 11.24 | Example of PWM Mode Operation (3)..... | 415 |
| Figure 11.25 | Example of Phase Counting Mode Setting Procedure | 417 |
| Figure 11.26 | Example of Phase Counting Mode 1 Operation | 417 |
| Figure 11.27 | Example of Phase Counting Mode 2 Operation | 419 |
| Figure 11.28 | Example of Phase Counting Mode 3 Operation | 420 |
| Figure 11.29 | Example of Phase Counting Mode 4 Operation | 421 |
| Figure 11.30 | Phase Counting Mode Application Example | 422 |
| Figure 11.31 | Count Timing in Internal Clock Operation | 426 |
| Figure 11.32 | Count Timing in External Clock Operation | 427 |
| Figure 11.33 | Output Compare Output Timing | 427 |

| | | |
|--------------|---|-----|
| Figure 11.34 | Input Capture Input Signal Timing | 428 |
| Figure 11.35 | Counter Clear Timing (Compare Match) | 428 |
| Figure 11.36 | Counter Clear Timing (Input Capture) | 429 |
| Figure 11.37 | Buffer Operation Timing (Compare Match) | 429 |
| Figure 11.38 | Buffer Operation Timing (Input Capture) | 430 |
| Figure 11.39 | TGI Interrupt Timing (Compare Match) | 430 |
| Figure 11.40 | TGI Interrupt Timing (Input Capture) | 431 |
| Figure 11.41 | TCIV Interrupt Setting Timing | 431 |
| Figure 11.42 | TCIU Interrupt Setting Timing | 432 |
| Figure 11.43 | Timing for Status Flag Clearing by CPU | 432 |
| Figure 11.44 | Timing for Status Flag Clearing by DTC/DMAC Activation | 433 |
| Figure 11.45 | Phase Difference, Overlap, and Pulse Width in Phase Counting Mode | 434 |
| Figure 11.46 | Contention between TCNT Write and Clear Operations | 435 |
| Figure 11.47 | Contention between TCNT Write and Increment Operations | 435 |
| Figure 11.48 | Contention between TGR Write and Compare Match | 436 |
| Figure 11.49 | Contention between Buffer Register Write and Compare Match | 437 |
| Figure 11.50 | Contention between TGR Read and Input Capture | 437 |
| Figure 11.51 | Contention between TGR Write and Input Capture | 438 |
| Figure 11.52 | Contention between Buffer Register Write and Input Capture | 439 |
| Figure 11.53 | Contention between Overflow and Counter Clearing | 439 |
| Figure 11.54 | Contention between TCNT Write and Overflow | 440 |

Section 12 8-Bit Timers

| | | |
|--------------|---|-----|
| Figure 12.1 | Block Diagram of 8-Bit Timer Module | 442 |
| Figure 12.2 | Example of Pulse Output | 453 |
| Figure 12.3 | Count Timing for Internal Clock Input | 453 |
| Figure 12.4 | Count Timing for External Clock Input | 454 |
| Figure 12.5 | Timing of CMF Setting | 454 |
| Figure 12.6 | Timing of Timer Output | 455 |
| Figure 12.7 | Timing of Compare-Match Clear | 455 |
| Figure 12.8 | Timing of Clearing by External Reset Input | 456 |
| Figure 12.9 | Timing of OVF Setting | 456 |
| Figure 12.10 | Contention between TCNT Write and Clear | 459 |
| Figure 12.11 | Contention between TCNT Write and Increment | 460 |
| Figure 12.12 | Contention between TCOR Write and Compare-Match | 460 |

Section 13 Watchdog Timer (WDT)

| | | |
|-------------|-------------------------------------|-----|
| Figure 13.1 | Block Diagram of WDT_0 (1) | 466 |
| Figure 13.1 | Block Diagram of WDT_1 (2) | 467 |
| Figure 13.2 | Watchdog Timer Mode Operation | 474 |

| | | |
|-------------|---|-----|
| Figure 13.3 | Interval Timer Mode Operation | 475 |
| Figure 13.4 | Timing of OVf Setting | 475 |
| Figure 13.5 | Timing of WOVf Setting..... | 476 |
| Figure 13.6 | Writing to TCNT, TCSR..... | 477 |
| Figure 13.7 | Writing to RSTCSR | 478 |
| Figure 13.8 | Contention between TCNT Write and Increment | 478 |

Section 14 IEBus™ Controller (IEB) [H8S/2258 Group]

| | | |
|--------------|--|-----|
| Figure 14.1 | Block Diagram of IEB | 482 |
| Figure 14.2 | Transfer Signal Format | 486 |
| Figure 14.3 | Bit Configuration of Slave Status (SSR)..... | 494 |
| Figure 14.4 | Locked Address Configuration | 495 |
| Figure 14.5 | IEBus Bit Format (Conceptual Diagram)..... | 496 |
| Figure 14.6 | Transmission Signal Format and Registers in Data Transfer | 507 |
| Figure 14.7 | Relationship between Transmission Signal Format and Registers in IEBus Data Reception | 510 |
| Figure 14.8 | Master Transmit Operation Timing..... | 529 |
| Figure 14.9 | Slave Reception Operation Timing | 532 |
| Figure 14.10 | Error Occurrence in the Broadcast Reception (DEE = 1) | 533 |
| Figure 14.11 | Master Receive Operation Timing | 536 |
| Figure 14.12 | Slave Transmit Operation Timing..... | 539 |
| Figure 14.13 | Relationships among Transfer Interrupt Sources | 540 |
| Figure 14.14 | Relationships among Receive Interrupt Sources | 540 |
| Figure 14.15 | Error Processing in Transfer | 545 |

Section 15 Serial Communication Interface (SCI)

| | | |
|-------------|---|-----|
| Figure 15.1 | Block Diagram of SCI..... | 549 |
| Figure 15.2 | Block Diagram of SCI_0 of H8S/2239 Group | 550 |
| Figure 15.3 | Example of the Internal Base Clock When the Average Transfer Rate Is Selected (1)..... | 583 |
| Figure 15.4 | Example of the Internal Base Clock When the Average Transfer Rate Is Selected (2)..... | 584 |
| Figure 15.5 | Data Format in Asynchronous Communication (Example with 8-Bit Data, Parity, Two Stop Bits)..... | 585 |
| Figure 15.6 | Receive Data Sampling Timing in Asynchronous Mode | 588 |
| Figure 15.7 | Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode)..... | 588 |
| Figure 15.8 | Sample SCI Initialization Flowchart | 589 |
| Figure 15.9 | Example of Operation in Transmission in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)..... | 590 |

| | | |
|--------------|--|-----|
| Figure 15.10 | Sample Serial Transmission Flowchart | 591 |
| Figure 15.11 | Example of SCI Operation in Reception (Example with 8-Bit Data, Parity, One Stop Bit) | 592 |
| Figure 15.12 | Sample Serial Reception Data Flowchart (1) | 594 |
| Figure 15.12 | Sample Serial Reception Data Flowchart (2) | 595 |
| Figure 15.13 | Example of Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A) | 597 |
| Figure 15.14 | Sample Multiprocessor Serial Transmission Flowchart | 598 |
| Figure 15.15 | Example of SCI Operation in Reception (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit) | 599 |
| Figure 15.16 | Sample Multiprocessor Serial Reception Flowchart (1) | 600 |
| Figure 15.16 | Sample Multiprocessor Serial Reception Flowchart (2) | 601 |
| Figure 15.17 | Data Format in Synchronous Communication (For LSB-First) | 602 |
| Figure 15.18 | Sample SCI Initialization Flowchart | 603 |
| Figure 15.19 | Sample SCI Transmission Operation in Clocked Synchronous Mode | 604 |
| Figure 15.20 | Sample Serial Transmission Flowchart | 605 |
| Figure 15.21 | Example of SCI Operation in Reception | 606 |
| Figure 15.22 | Sample Serial Reception Flowchart | 607 |
| Figure 15.23 | Sample Flowchart of Simultaneous Serial Transmit and Receive Operations | 609 |
| Figure 15.24 | Schematic Diagram of Smart Card Interface Pin Connections | 610 |
| Figure 15.25 | Normal Smart Card Interface Data Format | 611 |
| Figure 15.26 | Direct Convention (SDIR = SINV = O/\overline{E} = 0) | 611 |
| Figure 15.27 | Inverse Convention (SDIR = SINV = O/\overline{E} = 1) | 611 |
| Figure 15.28 | Receive Data Sampling Timing in Smart Card Mode (Using Clock of 372 Times the Transfer Rate) | 613 |
| Figure 15.29 | Retransfer Operation in SCI Transmit Mode | 615 |
| Figure 15.30 | TEND Flag Generation Timing in Transmission Operation | 615 |
| Figure 15.31 | Example of Transmission Processing Flow | 616 |
| Figure 15.32 | Retransfer Operation in SCI Receive Mode | 617 |
| Figure 15.33 | Example of Reception Processing Flow | 618 |
| Figure 15.34 | Timing for Fixing Clock Output Level | 618 |
| Figure 15.35 | Clock Halt and Restart Procedure | 619 |
| Figure 15.36 | Example of Communication Using SCI Select Function | 620 |
| Figure 15.37 | Summary of SCI Select Function Operation | 621 |
| Figure 15.38 | Example of Clocked Synchronous Transmission by DMAC or DTC | 626 |
| Figure 15.39 | Sample Flowchart for Mode Transition during Transmission | 627 |
| Figure 15.40 | Asynchronous Transmission Using Internal Clock | 628 |
| Figure 15.41 | Synchronous Transmission Using Internal Clock | 628 |
| Figure 15.42 | Sample Flowchart for Mode Transition during Reception | 629 |
| Figure 15.43 | Operation when Switching from SCK Pin Function to Port Pin Function | 630 |

| | | |
|--------------|--|-----|
| Figure 15.44 | Operation when Switching from SCK Pin Function to Port Pin Function (Example of Preventing Low-Level Output)..... | 631 |
|--------------|--|-----|

Section 16 I²C Bus Interface (IIC) (Option)

| | | |
|--------------|--|-----|
| Figure 16.1 | Block Diagram of I ² C Bus Interface..... | 635 |
| Figure 16.2 | I ² C Bus Interface Connections (Example: This LSI as Master) | 636 |
| Figure 16.3 | I ² C Bus Data Formats (I ² C Bus Formats) | 654 |
| Figure 16.4 | I ² C Bus Data Format (Serial Format) | 654 |
| Figure 16.5 | I ² C Bus Timing..... | 654 |
| Figure 16.6 | Flowchart for IIC Initialization (Example) | 655 |
| Figure 16.7 | Flowchart for Master Transmit Mode (Example) | 656 |
| Figure 16.8 | Example of Master Transmit Mode Operation Timing (MLS = WAIT = 0) | 658 |
| Figure 16.9 | Example of Master Transmit Mode Stop Condition Generation Timing (MLS = WAIT = 0)..... | 658 |
| Figure 16.10 | Flowchart for Master Receive Mode (Receiving Multiple Bytes) (WAIT = 1) (Example)..... | 660 |
| Figure 16.11 | Flowchart for Master Receive Mode (Receiving 1 Byte) (WAIT = 1) (Example)..... | 661 |
| Figure 16.12 | Example of Master Receive Mode Operation Timing (MLS = ACKB = 0, WAIT = 1)..... | 663 |
| Figure 16.13 | Example of Master Receive Mode Stop Condition Generation Timing (MLS = ACKB = 0, WAIT = 1)..... | 664 |
| Figure 16.14 | Flowchart for Slave Transmit Mode (Example)..... | 665 |
| Figure 16.15 | Example of Slave Receive Mode Operation Timing (1) (MLS = ACKB = 0)..... | 667 |
| Figure 16.16 | Example of Slave Receive Mode Operation Timing (2) (MLS = ACKB = 0)..... | 668 |
| Figure 16.17 | Sample Flowchart for Slave Transmit Mode | 669 |
| Figure 16.18 | Example of Slave Transmit Mode Operation Timing (MLS = 0) | 671 |
| Figure 16.19 | IRIC Setting Timing and SCL Control | 672 |
| Figure 16.20 | Block Diagram of Noise Canceler | 674 |
| Figure 16.21 | Points for Attention Concerning Reading of Master Receive Data..... | 680 |
| Figure 16.22 | Flowchart and Timing of Start Condition Instruction Issuance for Retransmission | 681 |
| Figure 16.23 | Timing of Stop Condition Issuance..... | 682 |
| Figure 16.24 | IRIC Flag Clearance in WAIT = 1 Status | 682 |
| Figure 16.25 | ICDR Read and ICCR Access Timing in Slave Transmit Mode | 683 |
| Figure 16.26 | TRS Bit Setting Timing in Slave Mode | 684 |
| Figure 16.27 | Diagram of Erroneous Operation When Arbitration Is Lost | 686 |
| Figure 16.28 | IRIC Flag Clearing Timing in Wait Operation..... | 687 |

Section 17 A/D Converter

| | | |
|--------------|---|-----|
| Figure 17.1 | Block Diagram of A/D Converter | 690 |
| Figure 17.2 | Access to ADDR (When Reading H'AA40) | 696 |
| Figure 17.3 | Example of A/D converter Operation (Single Mode, Channel 1 Selected) | 698 |
| Figure 17.4 | Example of A/D Converter Operation (Scan Mode, Channels AN0 to AN2 Selected) | 699 |
| Figure 17.5 | A/D Conversion Timing | 700 |
| Figure 17.6 | External Trigger Input Timing | 701 |
| Figure 17.7 | A/D Conversion Accuracy Definitions | 703 |
| Figure 17.8 | A/D Conversion Accuracy Definitions | 703 |
| Figure 17.9 | Example of Analog Input Circuit | 704 |
| Figure 17.10 | Example of Analog Input Protection Circuit | 706 |
| Figure 17.11 | Analog Input Pin Equivalent Circuit | 706 |

Section 18 D/A Converter

| | | |
|-------------|---------------------------------------|-----|
| Figure 18.1 | Block Diagram of D/A Converter | 707 |
| Figure 18.2 | D/A Converter Operation Example | 710 |

Section 20 Flash Memory (F-ZTAT Version)

| | | |
|--------------|--|-----|
| Figure 20.1 | Block Diagram of Flash Memory | 716 |
| Figure 20.2 | Flash Memory State Transitions | 717 |
| Figure 20.3 | Boot Mode (Example) | 718 |
| Figure 20.4 | User Program Mode (Example) | 719 |
| Figure 20.5 | Block Configuration of 384-kbyte Flash Memory | 721 |
| Figure 20.6 | Block Configuration of 256-kbyte Flash Memory | 722 |
| Figure 20.7 | Block Configuration of 128-kbyte Flash Memory | 723 |
| Figure 20.8 | Programming/Erasing Flowchart Example in User Program Mode | 735 |
| Figure 20.9 | Flowchart for Flash Memory Emulation in RAM | 736 |
| Figure 20.10 | Example of RAM Overlap Operation | 737 |
| Figure 20.11 | Program/Program-Verify Flowchart | 739 |
| Figure 20.12 | Erase/Erase-Verify Flowchart | 741 |
| Figure 20.13 | Socket Adapter Pin Correspondence Diagram | 744 |
| Figure 20.14 | Power-On/Off Timing (Boot Mode) | 748 |
| Figure 20.15 | Power-On/Off Timing (User Program Mode) | 749 |
| Figure 20.16 | Mode Transition Timing (Example: Boot Mode → User Mode ↔ User Program Mode) | 750 |

Section 21 Masked ROM

| | | |
|-------------|--|-----|
| Figure 21.1 | Block Diagram of On-Chip Masked ROM (384 kbytes) | 754 |
|-------------|--|-----|

Section 22 PROM

| | | |
|-------------|---|-----|
| Figure 22.1 | HD6472237 Socket Adapter Pin Correspondence Diagram (FP-100B, TFP-100B, TFP-100G)..... | 756 |
| Figure 22.2 | HD6472237 Socket Adapter Pin Correspondence Diagram (FP-100A) | 757 |
| Figure 22.3 | Memory Map in PROM Mode | 758 |
| Figure 22.4 | High-Speed Programming Flowchart..... | 760 |
| Figure 22.5 | PROM Programming/Verification Timing | 763 |
| Figure 22.6 | Recommended Screening Procedure..... | 764 |

Section 23 Clock Pulse Generator

| | | |
|--------------|---|-----|
| Figure 23.1 | Block Diagram of Clock Pulse Generator..... | 765 |
| Figure 23.2 | Connection of Crystal Resonator (Example)..... | 770 |
| Figure 23.3 | Crystal Resonator Equivalent Circuit..... | 771 |
| Figure 23.4 | External Clock Input (Examples)..... | 772 |
| Figure 23.5 | External Clock Input Timing..... | 777 |
| Figure 23.6 | External Clock Switching Circuit (Example)..... | 778 |
| Figure 23.7 | External Clock Switching Timing (Example)..... | 778 |
| Figure 23.8 | Connection Example of 32.768-kHz Quartz Oscillator..... | 780 |
| Figure 23.9 | Equivalence Circuit for 32.768-kHz Oscillator..... | 780 |
| Figure 23.10 | Pin Handling when Subclock Not Required..... | 781 |
| Figure 23.11 | Note on Board Design of Oscillator Circuit..... | 782 |

Section 24 Power-Down Modes

| | | |
|-------------|---|-----|
| Figure 24.1 | Mode Transition Diagram | 785 |
| Figure 24.2 | Medium-Speed Mode Transition and Clearance Timing | 791 |
| Figure 24.3 | Software Standby Mode Application Example | 794 |
| Figure 24.4 | Hardware Standby Mode Timing | 796 |

Section 25 Power Supply Circuit

| | | |
|-------------|---|-----|
| Figure 25.1 | Power Supply Connection for H8S/2258 Group, H8S/2238B, and H8S/2236B (On-Chip Internal Power Supply Step-Down Circuit) | 804 |
| Figure 25.2 | Power Supply Connection for H8S/2239 Group, H8S/2238R, H8S/2236R, H8S/2237 Group, and H8S/2227 Group (No Internal Power Supply Step-Down Circuit) | 804 |

Section 27 Electrical Characteristics

| | | |
|-------------|--|-----|
| Figure 27.1 | Power Supply Voltage and Operating Ranges (H8S/2258 Group) | 839 |
| Figure 27.2 | Power Supply Voltage and Operating Ranges (H8S/2239 Group) | 840 |
| Figure 27.3 | Power Supply Voltage and Operating Ranges (H8S/2238B and H8S/2236B) | 841 |
| Figure 27.4 | Power Supply Voltage and Operating Ranges (H8S/2238R and H8S/2236R) | 842 |

| | | |
|--------------|---|-----|
| Figure 27.5 | Power Supply Voltage and Operating Ranges (H8S/2237 Group and H8S/2227 Group)..... | 843 |
| Figure 27.6 | Output Load Circuit | 853 |
| Figure 27.7 | I ² C Bus Interface Input/Output Timing (Optional)..... | 859 |
| Figure 27.8 | Output Load Circuit | 873 |
| Figure 27.9 | Output Load Circuit | 896 |
| Figure 27.10 | System Clock Timing..... | 948 |
| Figure 27.11 | Oscillation Stabilization Timing | 948 |
| Figure 27.12 | Reset Input Timing..... | 949 |
| Figure 27.13 | Interrupt Input Timing..... | 949 |
| Figure 27.14 | Basic Bus Timing (Two-State Access) | 950 |
| Figure 27.15 | Basic Bus Timing (Three-State Access)..... | 951 |
| Figure 27.16 | Basic Bus Timing (Three-State Access with One Wait State) | 952 |
| Figure 27.17 | Burst ROM Access Timing (Two-State Access)..... | 953 |
| Figure 27.18 | Burst ROM Access Timing (One-State Access) | 954 |
| Figure 27.19 | External Bus Release Timing..... | 954 |
| Figure 27.20 | DMAC Single Address Transfer Timing (Two-State Access) | 955 |
| Figure 27.21 | DMAC Single Address Transfer Timing (Three-State Access) | 956 |
| Figure 27.22 | DMAC TEND Output Timing | 957 |
| Figure 27.23 | DMAC DREQ Input Timing..... | 957 |
| Figure 27.24 | I/O Port Input/Output Timing | 957 |
| Figure 27.25 | TPU Input/Output Timing..... | 958 |
| Figure 27.26 | TPU Clock Input Timing..... | 958 |
| Figure 27.27 | 8-Bit Timer Output Timing | 958 |
| Figure 27.28 | 8-Bit Timer Clock Input Timing | 959 |
| Figure 27.29 | 8-Bit Timer Reset Input Timing..... | 959 |
| Figure 27.30 | WDT_1 Output Timing..... | 959 |
| Figure 27.31 | SCK Clock Input Timing | 959 |
| Figure 27.32 | SCI Input/Output Timing (Clock Synchronous Mode) | 960 |
| Figure 27.33 | A/D Converter External Trigger Input Timing | 960 |
| Figure 27.34 | I ² C Bus Interface Input/Output Timing (Optional)..... | 960 |

Appendix C Package Dimensions

| | | |
|------------|---|-----|
| Figure C.1 | TFP-100B Package Dimensions..... | 973 |
| Figure C.2 | TFP-100G Package Dimensions | 974 |
| Figure C.3 | FP-100A Package Dimensions..... | 975 |
| Figure C.4 | FP-100B Package Dimensions | 976 |
| Figure C.5 | BP-112 Package Dimensions | 977 |
| Figure C.6 | TBP-112A, TBP-112AV Package Dimensions..... | 978 |

Tables

Section 1 Overview

| | | |
|-----------|--|----|
| Table 1.1 | Pin Arrangements in Each Mode of H8S/2258 Group | 20 |
| Table 1.2 | Pin Arrangements in Each Mode of H8S/2239 Group | 24 |
| Table 1.3 | Pin Arrangements in Each Mode of H8S/2238 Group | 29 |
| Table 1.4 | Pin Arrangements in Each Mode of H8S/2237 Group | 34 |
| Table 1.5 | Pin Arrangements in Each Mode of H8S/2227 Group | 39 |
| Table 1.6 | Pin Functions of H8S/2258 Group | 44 |
| Table 1.7 | Pin Functions of H8S/2239 Group and H8S/2238 Group | 50 |
| Table 1.8 | Pin Functions of H8S/2237 Group and H8S/2227 Group | 57 |

Section 2 CPU

| | | |
|------------|--|----|
| Table 2.1 | Instruction Classification..... | 79 |
| Table 2.2 | Operation Notation..... | 80 |
| Table 2.3 | Data Transfer Instructions | 81 |
| Table 2.4 | Arithmetic Operations Instructions | 82 |
| Table 2.5 | Logic Operations Instructions | 84 |
| Table 2.6 | Shift Instructions | 84 |
| Table 2.7 | Bit Manipulation Instructions..... | 85 |
| Table 2.8 | Branch Instructions | 87 |
| Table 2.9 | System Control Instructions | 88 |
| Table 2.10 | Block Data Transfer Instructions | 89 |
| Table 2.11 | Addressing Modes..... | 90 |
| Table 2.12 | Absolute Address Access Ranges | 92 |
| Table 2.13 | Effective Address Calculation..... | 94 |

Section 3 MCU Operating Modes

| | | |
|-----------|---|-----|
| Table 3.1 | MCU Operating Mode Selection..... | 103 |
| Table 3.2 | Pin Functions in Each Operating Mode..... | 108 |

Section 4 Exception Handling

| | | |
|-----------|---|-----|
| Table 4.1 | Exception Types and Priority | 119 |
| Table 4.2 | Exception Handling Vector Table | 120 |
| Table 4.3 | Reset Types | 121 |
| Table 4.4 | Status of CCR and EXR after Trace Exception Handling..... | 124 |
| Table 4.5 | Status of CCR and EXR after Trap Instruction Exception Handling | 125 |

Section 5 Interrupt Controller

| | | |
|-----------|--|-----|
| Table 5.1 | Pin Configuration | 129 |
| Table 5.2 | Interrupt Sources, Vector Addresses, and Interrupt Priorities | 137 |
| Table 5.3 | Interrupt Control Modes | 142 |
| Table 5.4 | Interrupts Selected in Each Interrupt Control Mode (1) | 143 |
| Table 5.5 | Interrupts Selected in Each Interrupt Control Mode (2) | 144 |
| Table 5.6 | Operations and Control Signal Functions in Each Interrupt Control Mode | 144 |
| Table 5.7 | Interrupt Response Times | 150 |
| Table 5.8 | Number of States in Interrupt Handling Routine Execution Status | 151 |
| Table 5.9 | Interrupt Source Selection and Clear Control | 153 |

Section 7 Bus Controller

| | | |
|-----------|--|-----|
| Table 7.1 | Pin Configuration | 167 |
| Table 7.2 | Bus Specifications for Each Area (Basic Bus Interface) | 177 |
| Table 7.3 | Data Buses Used and Valid Strobes | 182 |
| Table 7.4 | Pin States in Idle Cycle | 196 |
| Table 7.5 | Pin States in Bus Released State | 197 |

Section 8 DMA Controller (DMAC)

| | | |
|------------|--|-----|
| Table 8.1 | Pin Configuration | 205 |
| Table 8.2 | Short Address Mode and Full Address Mode (Channel 0) | 206 |
| Table 8.3 | DMAC Activation Sources | 232 |
| Table 8.4 | DMAC Transfer Modes | 234 |
| Table 8.5 | Register Functions in Sequential Mode | 236 |
| Table 8.6 | Register Functions in Idle Mode | 239 |
| Table 8.7 | Register Functions in Repeat Mode | 241 |
| Table 8.8 | Register Functions in Single Address Mode | 245 |
| Table 8.9 | Register Functions in Normal Mode | 248 |
| Table 8.10 | Register Functions in Block Transfer Mode | 251 |
| Table 8.11 | DMAC Channel Priority Order | 271 |
| Table 8.12 | Interrupt Sources and Priority Order | 275 |

Section 9 Data Transfer Controller (DTC)

| | | |
|-----------|--|-----|
| Table 9.1 | Activation Source and DTCER Clearance | 288 |
| Table 9.2 | Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs | 291 |
| Table 9.3 | Register Information in Normal Mode | 294 |
| Table 9.4 | Register Information in Repeat Mode | 295 |
| Table 9.5 | Register Information in Block Transfer Mode | 296 |
| Table 9.6 | DTC Execution Status | 300 |

| | | |
|-----------|---|-----|
| Table 9.7 | Number of States Required for Each Execution Status | 300 |
|-----------|---|-----|

Section 10 I/O Ports

| | | |
|------------|--|-----|
| Table 10.1 | Port Functions | 306 |
| Table 10.2 | Input Pull-Up MOS States in Port A | 332 |
| Table 10.3 | Input Pull-Up MOS States in Port B | 339 |
| Table 10.4 | Input Pull-Up MOS States in Port C | 342 |
| Table 10.5 | Input Pull-Up MOS States in Port D | 346 |
| Table 10.6 | Input Pull-Up MOS States in Port E | 349 |
| Table 10.7 | Examples of Ways to Handle Unused Input Pins | 358 |

Section 11 16-Bit Timer Pulse Unit (TPU)

| | | |
|-------------|---|-----|
| Table 11.1 | TPU Functions | 360 |
| Table 11.2 | Pin Configuration | 364 |
| Table 11.3 | CCLR2 to CCLR0 (Channels 0 and 3) | 368 |
| Table 11.4 | CCLR2 to CCLR0 (Channels 1, 2, 4, and 5) | 368 |
| Table 11.5 | TPSC2 to TPSC0 (Channel 0) | 369 |
| Table 11.6 | TPSC2 to TPSC0 (Channel 1) | 369 |
| Table 11.7 | TPSC2 to TPSC0 (Channel 2) | 370 |
| Table 11.8 | TPSC2 to TPSC0 (Channel 3) | 370 |
| Table 11.9 | TPSC2 to TPSC0 (Channel 4) | 371 |
| Table 11.10 | TPSC2 to TPSC0 (Channel 5) | 371 |
| Table 11.11 | MD3 to MD0 | 373 |
| Table 11.12 | TIORH_0 | 375 |
| Table 11.13 | TIORL_0 | 376 |
| Table 11.14 | TIOR_1 | 377 |
| Table 11.15 | TIOR_2 | 378 |
| Table 11.16 | TIORH_3 | 379 |
| Table 11.17 | TIORL_3 | 380 |
| Table 11.18 | TIOR_4 | 381 |
| Table 11.19 | TIOR_5 | 382 |
| Table 11.20 | TIORH_0 | 383 |
| Table 11.21 | TIORL_0 | 384 |
| Table 11.22 | TIOR_1 | 385 |
| Table 11.23 | TIOR_2 | 386 |
| Table 11.24 | TIORH_3 | 387 |
| Table 11.25 | TIORL_3 | 388 |
| Table 11.26 | TIOR_4 | 389 |
| Table 11.27 | TIOR_5 | 390 |
| Table 11.28 | Register Combinations in Buffer Operation | 405 |

| | | |
|-------------|---|-----|
| Table 11.29 | Cascaded Combinations | 409 |
| Table 11.30 | PWM Output Registers and Output Pins | 412 |
| Table 11.31 | Clock Input Pins in Phase Counting Mode | 416 |
| Table 11.32 | Up/Down-Count Conditions in Phase Counting Mode 1 | 418 |
| Table 11.33 | Up/Down-Count Conditions in Phase Counting Mode 2 | 419 |
| Table 11.34 | Up/Down-Count Conditions in Phase Counting Mode 3 | 420 |
| Table 11.35 | Up/Down-Count Conditions in Phase Counting Mode 4 | 421 |
| Table 11.36 | TPU Interrupts | 424 |

Section 12 8-Bit Timers

| | | |
|------------|--|-----|
| Table 12.1 | Pin Configuration | 443 |
| Table 12.2 | 8-Bit Timer Interrupt Sources | 458 |
| Table 12.3 | Timer Output Priorities | 461 |
| Table 12.4 | Switching of Internal Clock and TCNT Operation | 462 |

Section 13 Watchdog Timer (WDT)

| | | |
|------------|----------------------------|-----|
| Table 13.1 | Pin Configuration | 467 |
| Table 13.2 | WDT Interrupt Source | 476 |

Section 14 IEBus™ Controller (IEB) [H8S/2258 Group]

| | | |
|------------|---|-----|
| Table 14.1 | Mode Types | 483 |
| Table 14.2 | Transfer speed and Maximum Number of Transfer Bytes in Each Communications Mode | 484 |
| Table 14.3 | Contents of Message Length Bits | 489 |
| Table 14.4 | Control Bit Contents | 493 |
| Table 14.5 | Control Field for Locked Slave Unit | 494 |
| Table 14.6 | Pin Configuration | 497 |

Section 15 Serial Communication Interface (SCI)

| | | |
|------------|--|-----|
| Table 15.1 | Pin Configuration | 551 |
| Table 15.2 | The Relationships between the N Setting in BRR and Bit Rate B | 571 |
| Table 15.3 | BRR Settings for Various Bit Rates (Asynchronous Mode) | 572 |
| Table 15.4 | Maximum Bit Rate for Each Frequency (Asynchronous Mode) | 576 |
| Table 15.5 | Maximum Bit Rate with External Clock Input (Asynchronous Mode) | 577 |
| Table 15.6 | BRR Settings for Various Bit Rates (Clocked Synchronous Mode) | 578 |
| Table 15.7 | Maximum Bit Rate with External Clock Input (Clocked Synchronous Mode) | 579 |
| Table 15.8 | Examples of Bit Rate for Various BRR Settings (Smart Card Interface Mode) (When n = 0 and S = 372) | 580 |
| Table 15.9 | Maximum Bit Rate at Various Frequencies (Smart Card Interface Mode) (When S = 372) | 580 |

| | | |
|-------------|---|-----|
| Table 15.10 | Serial Transfer Formats (Asynchronous Mode)..... | 586 |
| Table 15.11 | SSR Status Flags and Receive Data Handling..... | 593 |
| Table 15.12 | Interrupt Sources of Serial Communication Interface Mode..... | 623 |
| Table 15.13 | Interrupt Sources in Smart Card Interface Mode..... | 624 |

Section 16 I²C Bus Interface (IIC) (Option)

| | | |
|------------|--|-----|
| Table 16.1 | Pin Configuration..... | 636 |
| Table 16.2 | Transfer Format..... | 640 |
| Table 16.3 | I ² C Transfer Rate..... | 642 |
| Table 16.4 | Flags and Transfer States | 648 |
| Table 16.5 | Flags and Transfer States | 673 |
| Table 16.6 | IIC Interrupt Source | 676 |
| Table 16.7 | I ² C Bus Timing (SCL and SDA Output) | 677 |
| Table 16.8 | Permissible SCL Rise Time (t_{sr}) Values | 678 |
| Table 16.9 | I ² C Bus Timing (with Maximum Influence of t_{sr}/t_{sf}) | 679 |

Section 17 A/D Converter

| | | |
|------------|---|-----|
| Table 17.1 | Pin Configuration..... | 691 |
| Table 17.2 | Analog Input Channels and Corresponding ADDR Registers..... | 692 |
| Table 17.3 | A/D Conversion Time (Single Mode)..... | 700 |
| Table 17.4 | A/D Conversion Time (Scan Mode)..... | 700 |
| Table 17.5 | A/D Converter Interrupt Source | 701 |
| Table 17.6 | Analog Pin Specifications | 706 |

Section 18 D/A Converter

| | | |
|------------|------------------------------|-----|
| Table 18.1 | Pin Configuration..... | 708 |
| Table 18.2 | D/A Conversion Control | 709 |

Section 20 Flash Memory (F-ZTAT Version)

| | | |
|------------|--|-----|
| Table 20.1 | Differences between Boot Mode and User Program Mode..... | 717 |
| Table 20.2 | Pin Configuration..... | 724 |
| Table 20.3 | Setting On-Board Programming Modes..... | 732 |
| Table 20.4 | Boot Mode Operation..... | 734 |
| Table 20.5 | System Clock Frequencies for Which Automatic Adjustment of LSI Bit Rate Is Possible..... | 734 |
| Table 20.6 | Flash Memory Operating States | 745 |
| Table 20.7 | Registers Present in F-ZTAT Version but Absent in Masked ROM Version | 751 |

Section 22 PROM

| | | |
|------------|---------------------------|-----|
| Table 22.1 | Selecting PROM Mode | 755 |
|------------|---------------------------|-----|

| | | |
|------------|---------------------------------------|-----|
| Table 22.2 | Socket Adapters..... | 758 |
| Table 22.3 | Mode Selection in PROM Mode | 759 |
| Table 22.4 | DC Characteristics in PROM Mode | 761 |
| Table 22.5 | AC Characteristics in PROM Mode | 762 |

Section 23 Clock Pulse Generator

| | | |
|------------|---|-----|
| Table 23.1 | Damping Resistance Value..... | 771 |
| Table 23.2 | Crystal Resonator Characteristics..... | 771 |
| Table 23.3 | External Clock Input Conditions (1) (H8S/2258 Group)..... | 772 |
| Table 23.3 | External Clock Input Conditions (2) (H8S/2238B, H8S/2236B) | 773 |
| Table 23.3 | External Clock Input Conditions (3) (H8S/2238R, H8S/2236R) | 773 |
| Table 23.3 | External Clock Input Conditions (4) (H8S/2237 Group, H8S/2227 Group) | 774 |
| Table 23.3 | External Clock Input Conditions (5) (H8S/2239 Group)..... | 774 |
| Table 23.4 | External Clock Input Conditions (Duty Adjustment Circuit Unused) (1) (H8S/2258 Group)..... | 775 |
| Table 23.4 | External Clock Input Conditions (Duty Adjustment Circuit Unused) (2) (H8S/2238B, H8S/2236B)..... | 775 |
| Table 23.4 | External Clock Input Conditions (Duty Adjustment Circuit Unused) (3) (H8S/2238R, H8S/2236R)..... | 776 |
| Table 23.4 | External Clock Input Conditions (Duty Adjustment Circuit Unused) (4) (H8S/2237 Group, H8S/2227 Group)..... | 776 |
| Table 23.4 | External Clock Input Conditions (Duty Adjustment Circuit Unused) (5) (H8S/2239 Group)..... | 777 |

Section 24 Power-Down Modes

| | | |
|------------|--|-----|
| Table 24.1 | LSI Internal States in Each Mode..... | 784 |
| Table 24.2 | Low Power Dissipation Mode Transition Conditions | 786 |
| Table 24.3 | Oscillation Settling Time Settings..... | 793 |
| Table 24.4 | ϕ Pin States in Respective Processes | 800 |

Section 27 Electrical Characteristics

| | | |
|------------|----------------------------------|-----|
| Table 27.1 | Absolute Maximum Ratings..... | 844 |
| Table 27.2 | DC Characteristics (1)..... | 845 |
| Table 27.2 | DC Characteristics (2)..... | 847 |
| Table 27.2 | DC Characteristics (3)..... | 849 |
| Table 27.3 | Permissible Output Current | 851 |
| Table 27.4 | Bus Driving Characteristics..... | 852 |
| Table 27.5 | Clock Timing..... | 854 |
| Table 27.6 | Control Signal Timing..... | 855 |
| Table 27.7 | Bus Timing..... | 856 |

| | | |
|-------------|---|-----|
| Table 27.8 | Timing of On-Chip Peripheral Modules..... | 857 |
| Table 27.9 | I ² C Bus Timing..... | 858 |
| Table 27.10 | A/D Conversion Characteristics | 860 |
| Table 27.11 | D/A Conversion Characteristics | 861 |
| Table 27.12 | Flash Memory Characteristics..... | 862 |
| Table 27.13 | Absolute Maximum Ratings..... | 864 |
| Table 27.14 | DC Characteristics (1)..... | 865 |
| Table 27.14 | DC Characteristics (2)..... | 867 |
| Table 27.14 | DC Characteristics (3)..... | 869 |
| Table 27.15 | Permissible Output Currents | 871 |
| Table 27.16 | Bus Driving Characteristics..... | 872 |
| Table 27.17 | Clock Timing | 874 |
| Table 27.18 | Control Signal Timing..... | 876 |
| Table 27.19 | Bus Timing..... | 877 |
| Table 27.20 | DMAC Timing | 879 |
| Table 27.21 | Timing of On-Chip Peripheral Modules..... | 880 |
| Table 27.22 | I ² C Bus Timing..... | 882 |
| Table 27.23 | A/D Conversion Characteristics | 883 |
| Table 27.24 | D/A Conversion Characteristics | 884 |
| Table 27.25 | Flash Memory Characteristics..... | 885 |
| Table 27.26 | Absolute Maximum Ratings..... | 887 |
| Table 27.27 | DC Characteristics (1)..... | 888 |
| Table 27.27 | DC Characteristics (2)..... | 890 |
| Table 27.27 | DC Characteristics (3)..... | 892 |
| Table 27.28 | Permissible Output Currents | 894 |
| Table 27.29 | Bus Drive Characteristics..... | 895 |
| Table 27.30 | Clock Timing | 897 |
| Table 27.31 | Control Signal Timing..... | 898 |
| Table 27.32 | Bus Timing..... | 899 |
| Table 27.33 | Timing of On-Chip Peripheral Modules..... | 901 |
| Table 27.34 | I ² C Bus Timing..... | 903 |
| Table 27.35 | A/D Conversion Characteristics (F-ZTAT and Masked ROM Versions) | 904 |
| Table 27.36 | D/A Conversion Characteristics (F-ZTAT and Masked ROM Versions) | 904 |
| Table 27.37 | Flash Memory Characteristics..... | 905 |
| Table 27.38 | Absolute Maximum Ratings..... | 907 |
| Table 27.39 | DC Characteristics (1)..... | 908 |
| Table 27.39 | DC Characteristics (2)..... | 910 |
| Table 27.39 | DC Characteristics (3)..... | 912 |
| Table 27.40 | Permissible Output Currents | 914 |
| Table 27.41 | Bus Driving Characteristics..... | 915 |

| | | |
|-------------|---|-----|
| Table 27.42 | Clock Timing..... | 916 |
| Table 27.43 | Control Signal Timing..... | 917 |
| Table 27.44 | Bus Timing..... | 918 |
| Table 27.45 | Timing of On-Chip Peripheral Modules..... | 920 |
| Table 27.46 | I ² C Bus Timing..... | 922 |
| Table 27.47 | A/D Conversion Characteristics..... | 923 |
| Table 27.48 | D/A Conversion Characteristics..... | 924 |
| Table 27.49 | Flash Memory Characteristics..... | 925 |
| Table 27.50 | Absolute Maximum Ratings..... | 927 |
| Table 27.51 | DC Characteristics (1)..... | 928 |
| Table 27.51 | DC Characteristics (2)..... | 930 |
| Table 27.51 | DC Characteristics (3)..... | 932 |
| Table 27.51 | DC Characteristics (4)..... | 934 |
| Table 27.52 | Permissible Output Currents..... | 936 |
| Table 27.53 | Clock Timing..... | 937 |
| Table 27.54 | Control Signal Timing..... | 939 |
| Table 27.55 | Bus Timing..... | 940 |
| Table 27.56 | Timing of On-Chip Peripheral Modules..... | 942 |
| Table 27.57 | A/D Conversion Characteristics..... | 944 |
| Table 27.58 | D/A Conversion Characteristics..... | 945 |
| Table 27.59 | Flash Memory Characteristics..... | 946 |

Appendix B Product Codes

| | | |
|-----------|---|-----|
| Table B.1 | Product Codes of H8S/2258 Group..... | 968 |
| Table B.2 | Product Codes of H8S/2239 Group..... | 969 |
| Table B.3 | Product Codes of H8S/2238 Group..... | 970 |
| Table B.4 | Product Codes of H8S/2237 Group and H8S/2227 Group..... | 972 |

Section 1 Overview

1.1 Features

- High-speed H8S/2000 central processing unit with an internal 16-bit architecture
 - Upward-compatible with H8/300 and H8/300H CPUs on an object level
 - Sixteen 16-bit general registers
 - 65 basic instructions
- Various peripheral functions
 - PC break controller
 - DMA controller (DMAC)
Supported only by the H8S/2239 Group.
 - Data transfer controller (DTC)
 - 16-bit timer-pulse unit (TPU)
H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group: Six channels
H8S/2227 Group: Three channels
 - 8-bit timer (TMR)
H8S/2258 Group, H8S/2239 Group, H8S/2238 Group: Four channels
H8S/2237 Group, H8S/2227 Group: Two channels
 - Watchdog timer (WDT)
 - Serial communication interface (SCI)
H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group: Four channels (SCI_0 to SCI_3)
H8S/2227 Group: Three channels (SCI_0, SCI_1, and SCI_3)
 - I²C bus interface (IIC)
Optional function for the H8S/2258 Group, H8S/2239 Group, and H8S/2238 Group
 - 10-bit A/D converter
 - 8-bit D/A converter
Not available in the H8S/2227 Group.
 - IEBus controller (IEB)
H8S/2258 Group: One channel

- On-chip memory

| ROM | Model | ROM | RAM | Remarks |
|----------------------|-------------|------------|-----------|---------|
| Flash memory version | HD64F2258 | 256 kbytes | 16 kbytes | |
| | HD64F2239 | 384 kbytes | 32 kbytes | |
| | HD64F2238B | 256 kbytes | 16 kbytes | |
| | HD64F2238R | 256 kbytes | 16 kbytes | |
| | HD64F2227 | 128 kbytes | 16 kbytes | |
| PROM version | HD6472237 | 128 kbytes | 16 kbytes | |
| Masked ROM version | HD6432258 | 256 kbytes | 16 kbytes | |
| | HD6432258W | 256 kbytes | 16 kbytes | |
| | HD6432256 | 128 kbytes | 8 kbytes | |
| | HD6432256W | 128 kbytes | 8 kbytes | |
| | HD6432239 | 384 kbytes | 32 kbytes | |
| | HD6432239W | 384 kbytes | 32 kbytes | |
| | HD6432238B | 256 kbytes | 16 kbytes | |
| | HD6432238BW | 256 kbytes | 16 kbytes | |
| | HD6432238R | 256 kbytes | 16 kbytes | |
| | HD6432238RW | 256 kbytes | 16 kbytes | |
| | HD6432236B | 128 kbytes | 8 kbytes | |
| | HD6432236BW | 128 kbytes | 8 kbytes | |
| | HD6432236R | 128 kbytes | 8 kbytes | |
| | HD6432236RW | 128 kbytes | 8 kbytes | |
| | HD6432237 | 128 kbytes | 16 kbytes | |
| | HD6432235 | 128 kbytes | 4 kbytes | |
| | HD6432233 | 64 kbytes | 4 kbytes | |
| | HD6432227 | 128 kbytes | 16 kbytes | |
| | HD6432225 | 128 kbytes | 4 kbytes | |
| | HD6432224 | 96 kbytes | 4 kbytes | |
| | HD6432223 | 64 kbytes | 4 kbytes | |

- General I/O ports
 - I/O pins: 72
 - Input-only pins: 10
- Supports various power-down states

- Compact package

| Package | (Code) ^{*6} | Body Size | Pin Pitch |
|-------------------------|------------------------|----------------|-----------|
| TQFP-100 | TFP-100B, TFP-100BV | 14.0 × 14.0 mm | 0.5 mm |
| TQFP-100 ^{*1} | TFP-100G, TFP-100GV | 12.0 × 12.0 mm | 0.4 mm |
| QFP-100 ^{*2} | FP-100A, FP-100AV | 14.0 × 20.0 mm | 0.65 mm |
| QFP-100 ^{*3} | FP-100B, FP-100BV | 14.0 × 14.0 mm | 0.5 mm |
| LFBGA-112 ^{*4} | BP-112, BP-112V | 10.0 × 10.0 mm | 0.8 mm |
| TFBGA-112 ^{*5} | TBP-112A, TBP-112AV | 10.0 × 10.0 mm | 0.8 mm |

Notes: 1. Not supported by the H8S/2258 Group.

2. Supported only by the H8S/2258 Group, H8S/2238B, H8S/2236B, H8S/2237 Group, and HD6432227.

3. Not supported by the HD64F2227.

4. Supported only by the HD64F2238R.

5. Supported only by the HD64F2238R and HD64F2239.

6. Package code ending in the letter V designate Pb-free Product.

1.2 Internal Block Diagram

Figures 1.1 to 1.5 show the internal block diagrams.

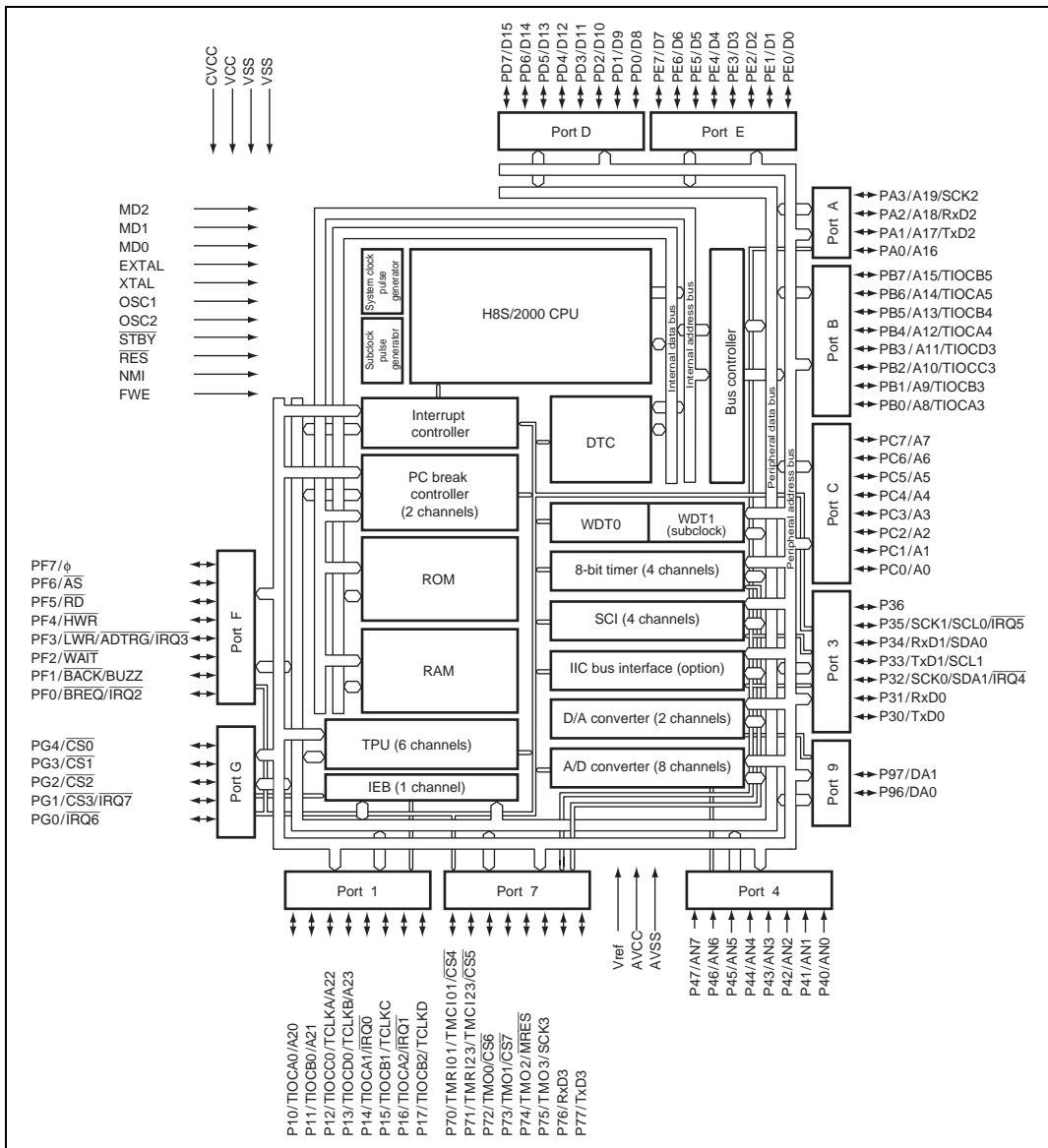


Figure 1.1 Internal Block Diagram of H8S/2258 Group

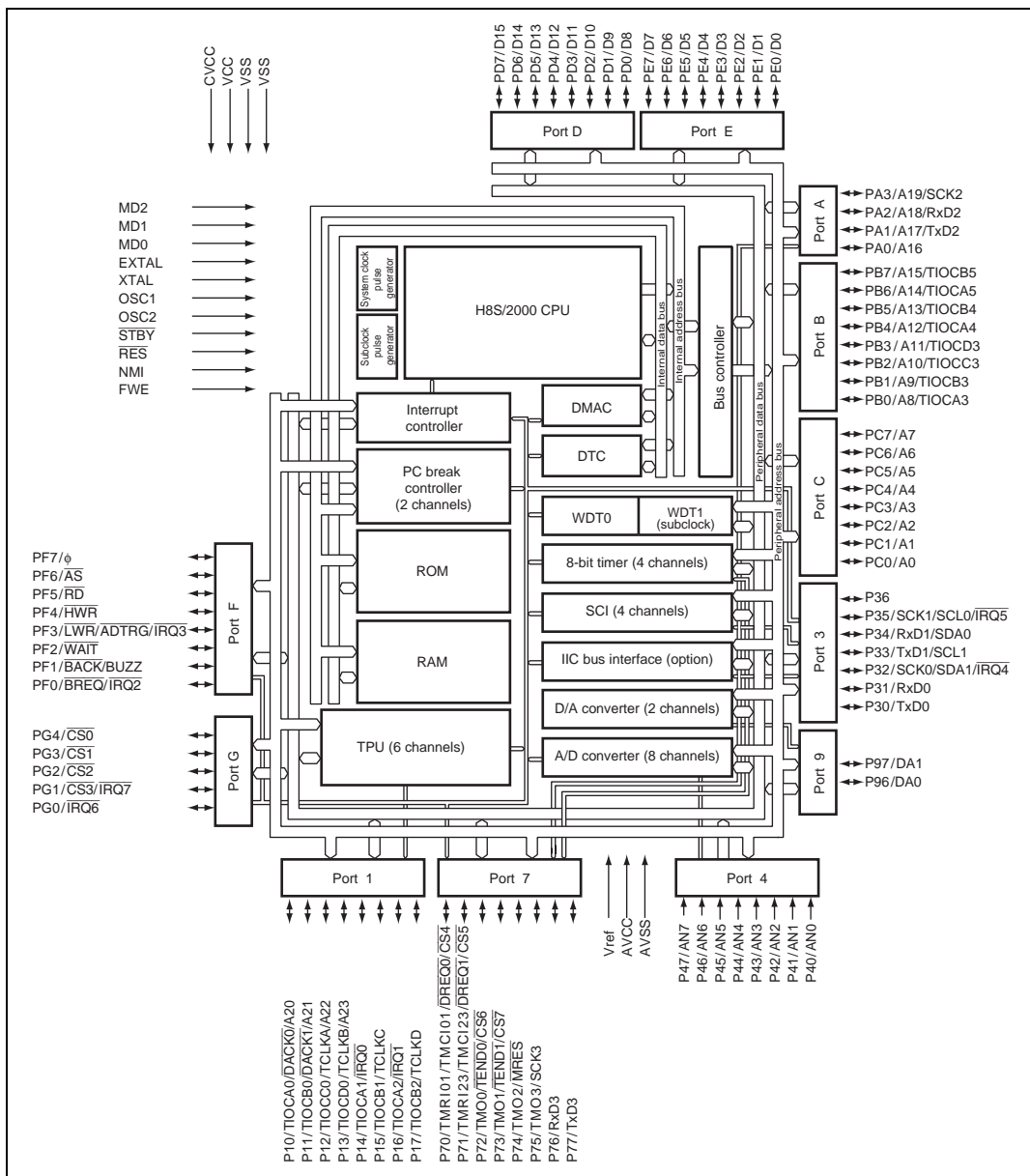


Figure 1.2 Internal Block Diagram of H8S/2239 Group

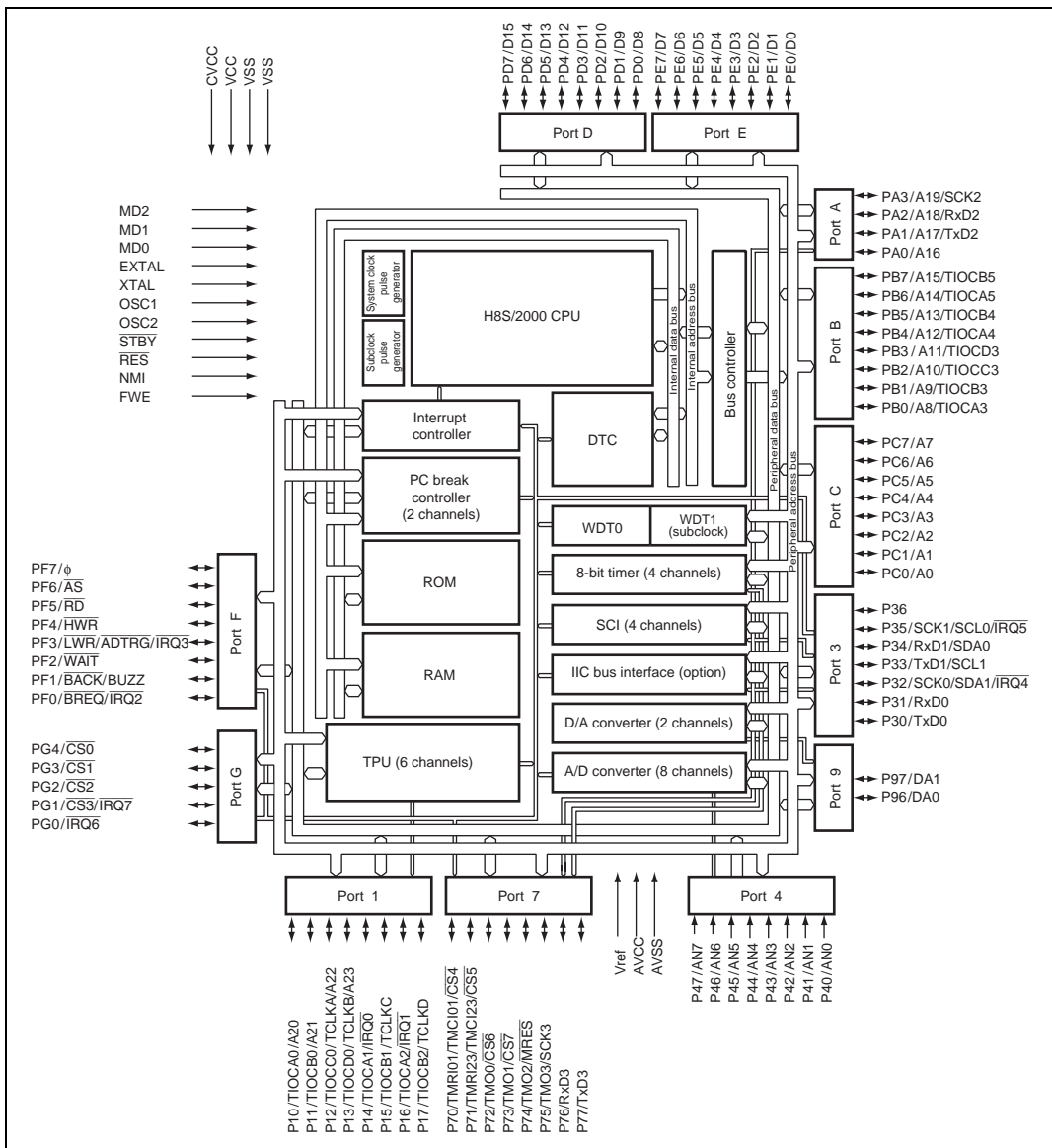


Figure 1.3 Internal Block Diagram of H8S/2238 Group

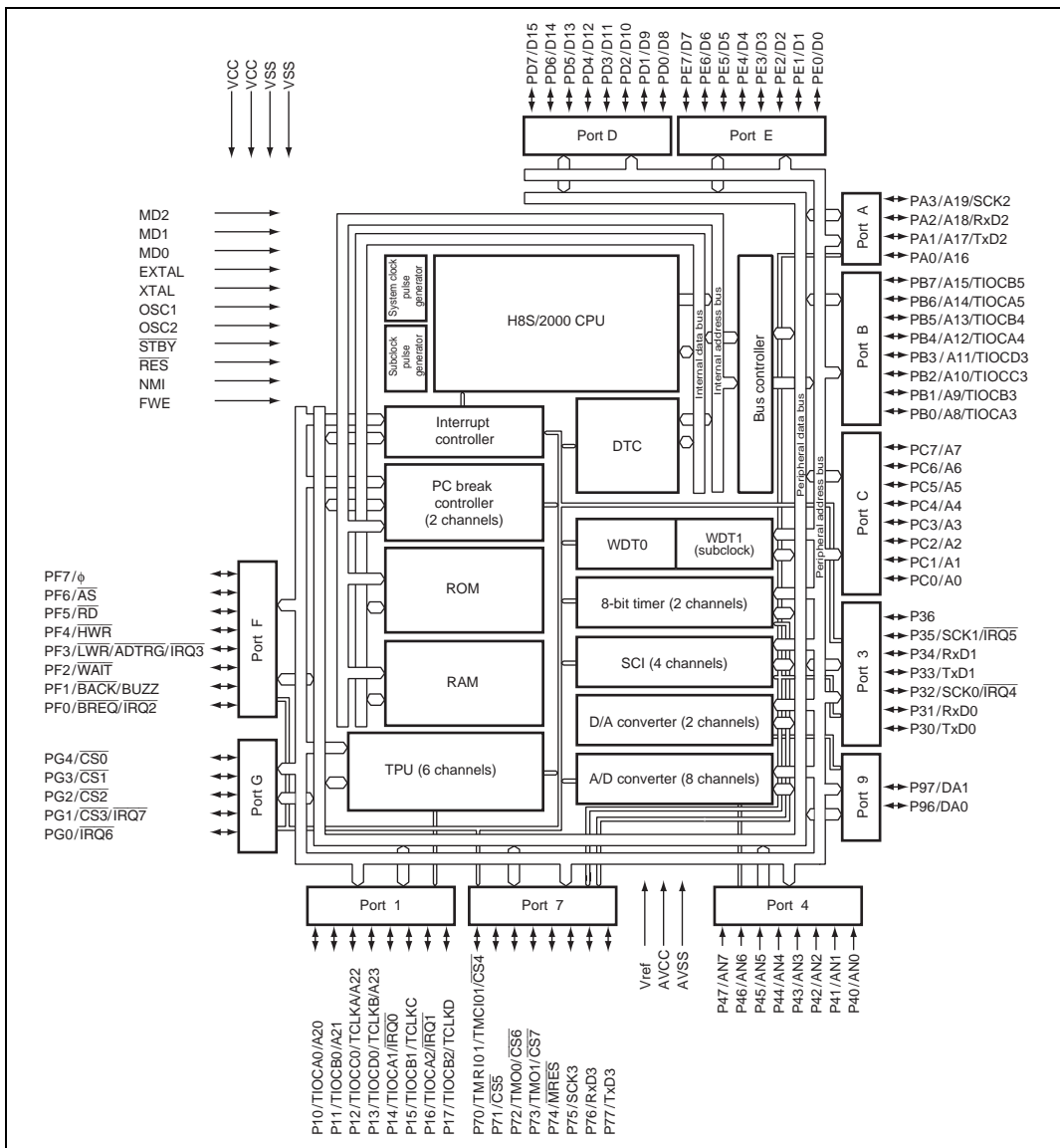


Figure 1.4 Internal Block Diagram of H8S/2237 Group

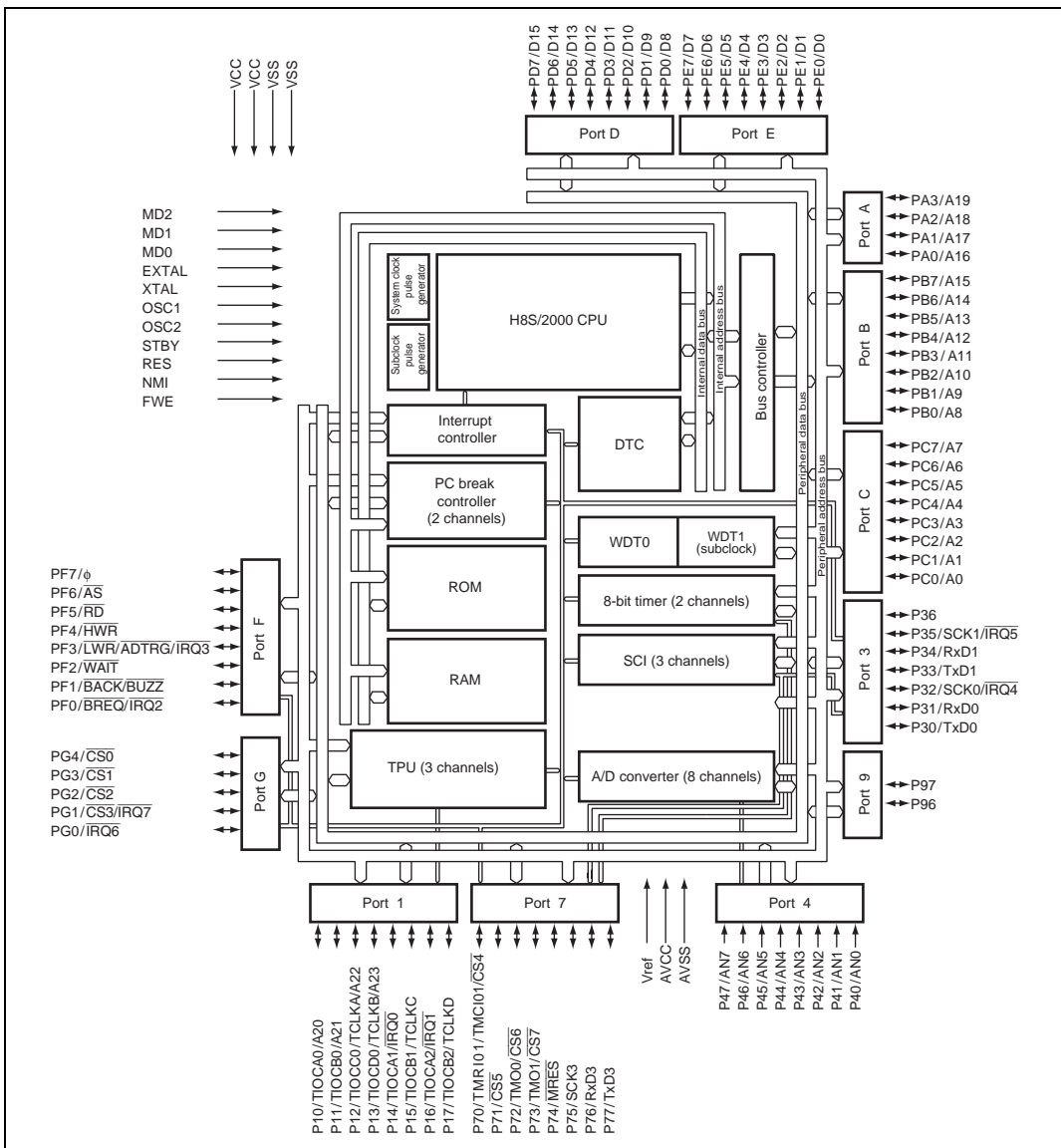


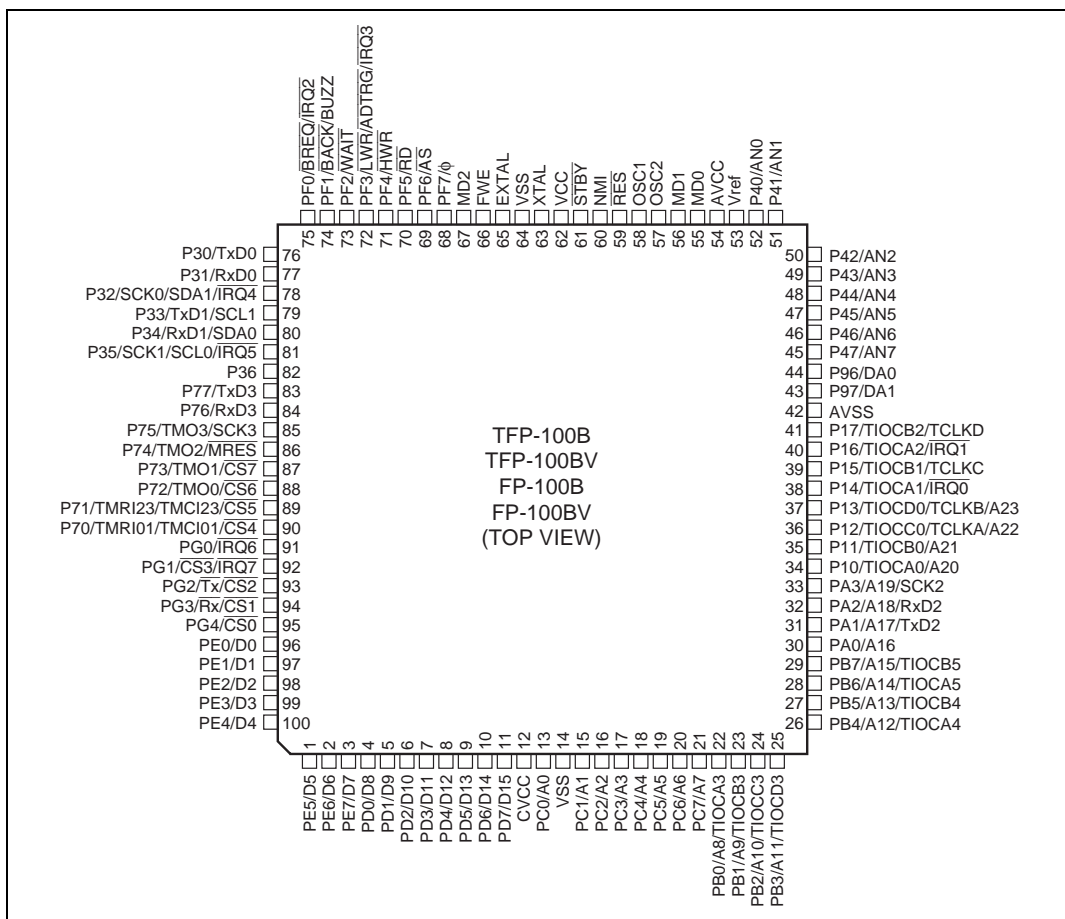
Figure 1.5 Internal Block Diagram of H8S/2227 Group

1.3 Pin Description

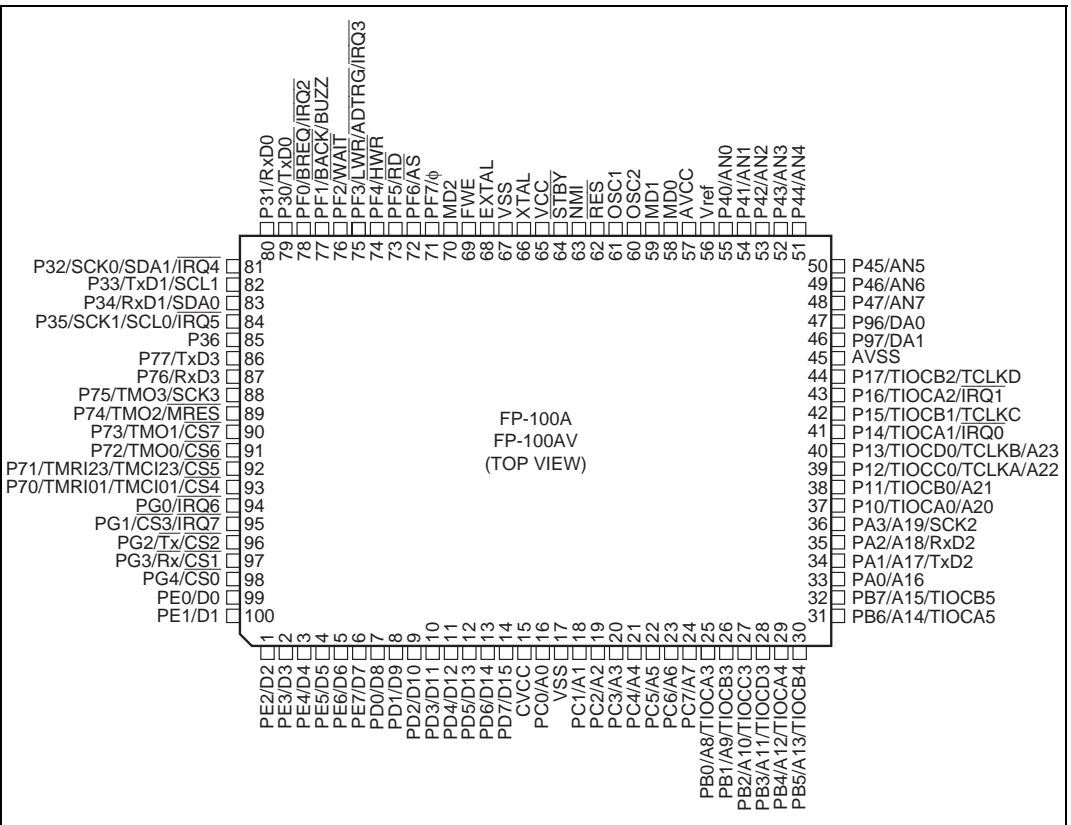
1.3.1 Pin Arrangement

(1) Pin Arrangement of H8S/2258 Group

Figures 1.6 and 1.7 show the pin arrangement of the H8S/2258 Group.



**Figure 1.6 Pin Arrangement of H8S/2258 Group
(TFP-100B, TFP-100BV, FP-100B, FP-100BV: Top View)**



**Figure 1.7 Pin Arrangement of H8S/2258 Group
(FP-100A, FP-100AV: Top View)**

(2) Pin Arrangement of H8S/2239 Group

Figures 1.8 and 1.9 show the pin arrangement of the H8S/2239 Group.

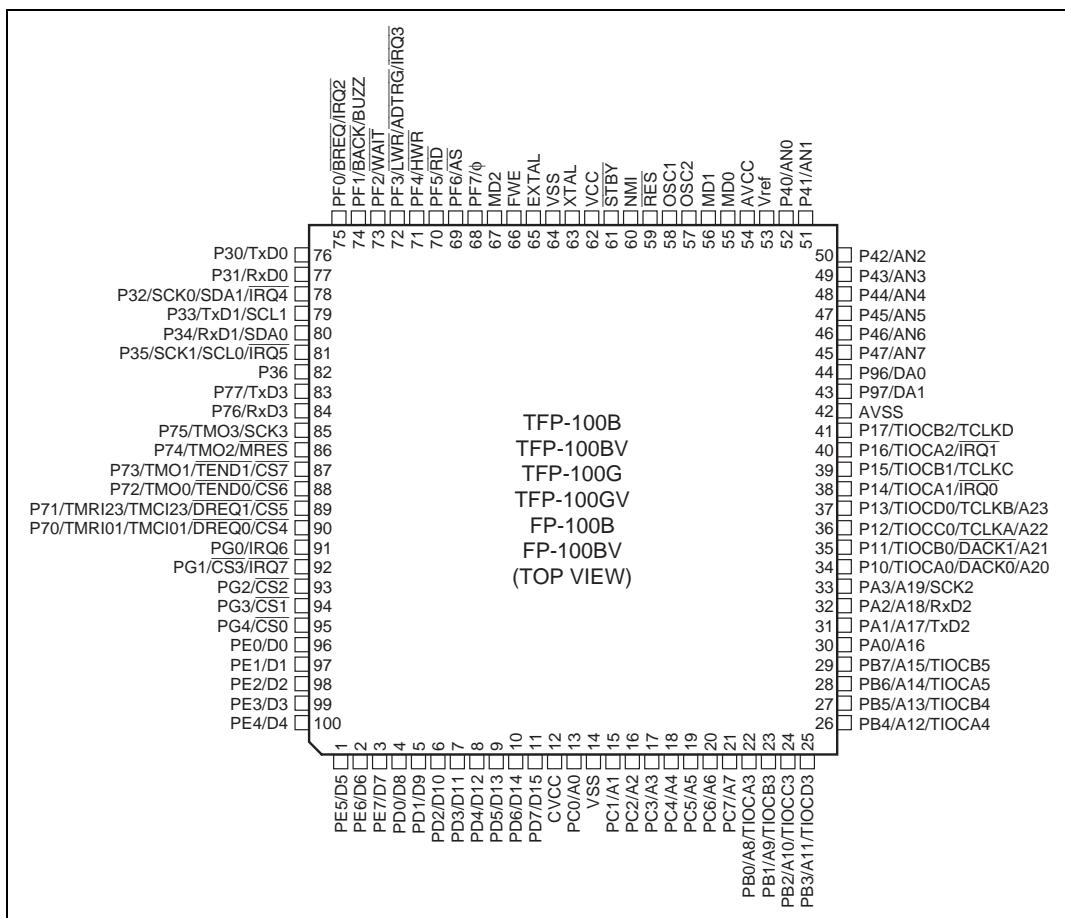


Figure 1.8 Pin Arrangement of H8S/2239 Group
(TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)

| | A | B | C | D | E | F | G | H | J | K | L |
|----|---|---|--|---|-------------------------------------|--------|--------------------------|----------------------------------|---|---|----------------------------------|
| 11 | NC (Reserve) | PF1/ BACK/ BUZZ | PF4/ HWR | PF7/ ϕ | EXTAL | XTAL | $\overline{\text{STBY}}$ | OSC1 | MD0 | P40/AN0 | NC (Reserve) |
| 10 | P30/ TxD0 | NC (Reserve) | PF2/ WAIT | PF5/ $\overline{\text{RD}}$ | FWE | VSS | VCC | OSC2 | AVCC | P41/AN1 | P42/AN2 |
| 9 | P33/ TxD1/ SCL1 | P32/ SCK0/ SDA1/ $\overline{\text{IRQ4}}$ | PF0/ $\overline{\text{BREQ}}$ / $\overline{\text{IRQ2}}$ | PF3/ LWR/ ADTRG/ $\overline{\text{IRQ3}}$ | MD2 | VCC | NMI | MD1 | NC (Reserve) | P43/AN3 | P45/AN5 |
| 8 | P36 | P35/ SCK1/ SCL0/ $\overline{\text{IRQ5}}$ | P34/ RxD1/ SDA0 | P31/ RxD0 | PF6/ $\overline{\text{AS}}$ | VSS | $\overline{\text{RES}}$ | Vref | P44/AN4 | P46/AN6 | P96/DA0 |
| 7 | P75/ TMO3/ SCK3 | P74/ TMO2/ MRES | P76/ RxD3 | P77/ TxD3 | TBP-112A TBP-112AV (TOP VIEW) | | | P47/AN7 | P97/DA1 | AVSS | AVSS |
| 6 | P72/ TMO0/ $\overline{\text{TEND0}}$ / CS6 | P71/ TMRI23/ TMC123/ $\overline{\text{DREQ1/CS5}}$ | P73/ TMO1/ $\overline{\text{TEND1}}$ / CS7 | P70/ TMRI01/ TMC101/ $\overline{\text{DREQ0/CS4}}$ | | | | P17/ TIOCB2/ TCLKD | P14/ TIOCA1/ $\overline{\text{IRQ0}}$ | P16/ TIOCA2/ $\overline{\text{IRQ1}}$ | P15/ TIOCB1/ TCLKC |
| 5 | PG0/ $\overline{\text{IRQ6}}$ | PG1/ CS3/ $\overline{\text{IRQ7}}$ | PG2/ CS2 | PG4/ CS0 | | | | P10/ TIOCA0/ DACK0/ A20 | P11/ TIOCB0/ DACK1/ A21 | P13/ TIOCD0/ TCLKB/ A23 | P12/ TIOCC0/ TCLKA/ A22 |
| 4 | PG3/ CS1 | PE0/D0 | PE2/D2 | PE7/D7 | PD5/D13 | VSS | PC5/A5 | PB6/ A14/ TIOCA5 | PA1/ A17/ TxD2 | PA2/ A18/ RxD2 | PA3/ A19/ SCK2 |
| 3 | PE1/D1 | PE3/D3 | NC (Reserve) | PD2/D10 | PD6/D14 | CVCC | PC3/A3 | PB0/ A8/ TIOCA3 | PB3/ A11/ TIOCD3 | PB7/ A15/ TIOCB5 | PA0/A16 |
| 2 | PE4/D4 | PE5/D5 | PD0/D8 | PD3/D11 | CVCC | VSS | PC2/A2 | PC6/A6 | PB1/A9/ TIOCB3 | PB4/ A12/ TIOCA4 | PB5/ A13/ TIOCB4 |
| 1 | NC (Reserve) | PE6/D6 | PD1/D9 | PD4/D12 | PD7/D15 | PC0/A0 | PC1/A1 | PC4/A4 | PC7/A7 | PB2/ A10/ TIOCC3 | NC (Reserve) |

INDEX

**Figure 1.9 Pin Arrangement of H8S/2239 Group
(TBP-112A, TBP-112AV: Top View, Only for HD64F2239)**

(3) Pin Arrangement of H8S/2238 Group

Figures 1.10 to 1.12 show the pin arrangement of the H8S/2238 Group.

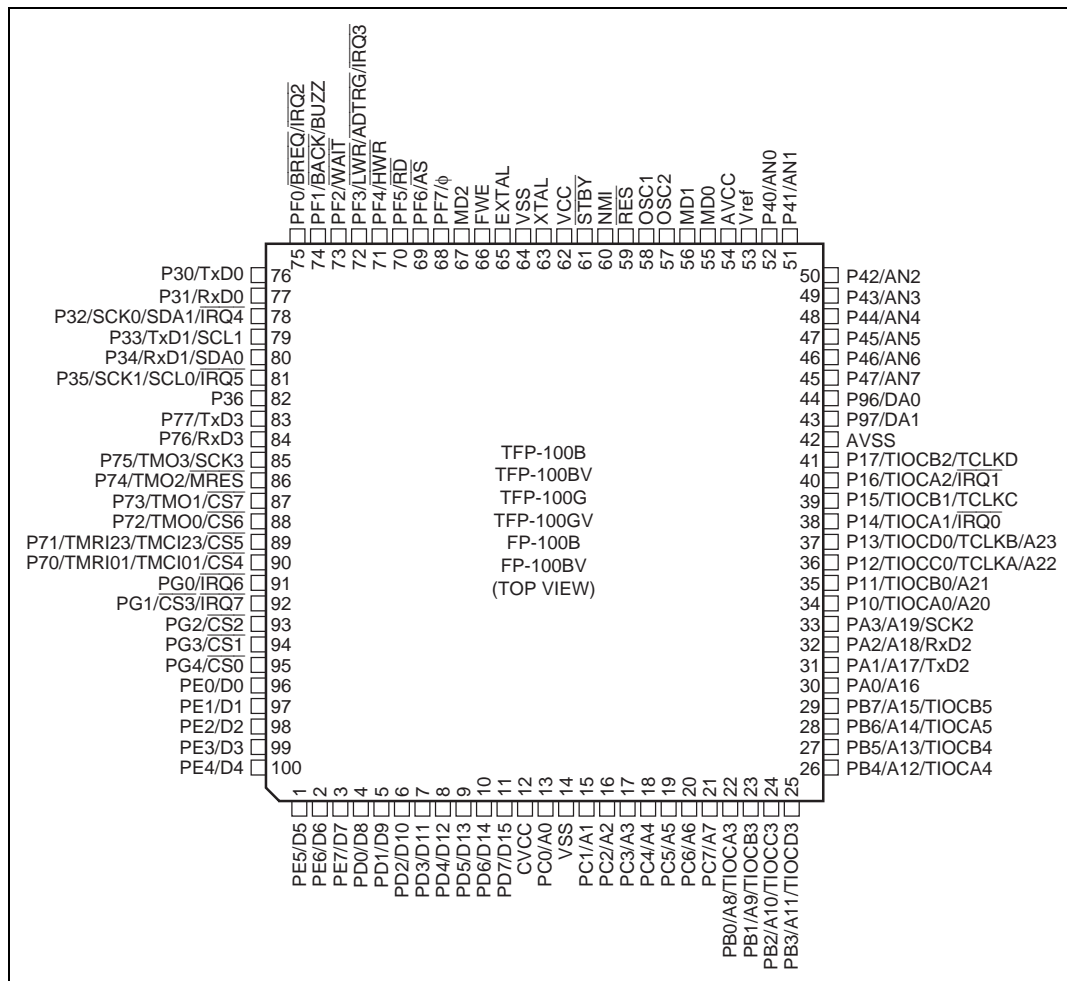
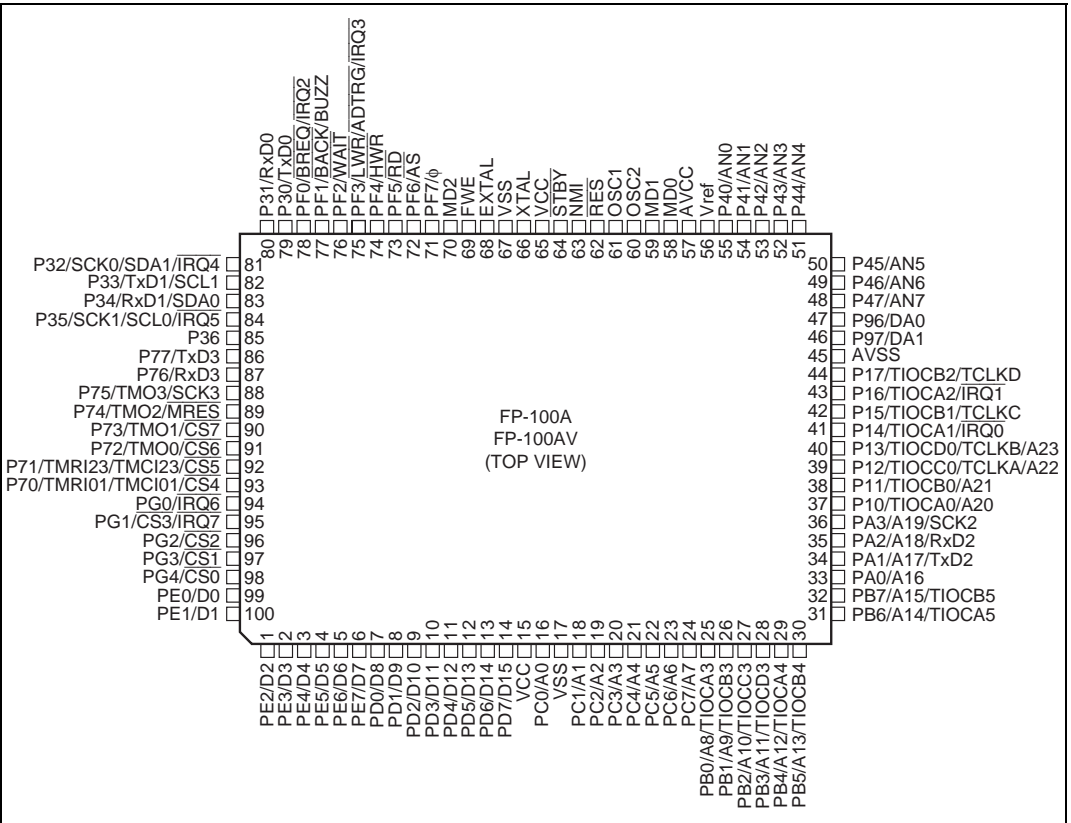


Figure 1.10 Pin Arrangement of H8S/2238 Group
(TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)



**Figure 1.11 Pin Arrangement of H8S/2238 Group
(FP-100A, FP-100AV: Top View, Only for H8S/2238B and H8S/2236B)**

| | A | B | C | D | E | F | G | H | J | K | L |
|----|-----------------------|-----------------------------------|--|-----------------------------------|--|--------|--------------------------|--------------------------|-------------------------|----------------------------------|----------------------------------|
| 11 | NC | PF1/ BACK/ BUZZ | PF4/ HWR | PF7/ ϕ | EXTAL | XTAL | $\overline{\text{STBY}}$ | OSC1 | MD0 | P40/AN0 | NC |
| 10 | P30/ TxD0 | NC | PF2/ $\overline{\text{WAIT}}$ | PF5/ $\overline{\text{RD}}$ | FWE | VSS | VCC | OSC2 | AVCC | P41/AN1 | P42/AN2 |
| 9 | P33/ TxD1/ SCL1 | P32/ SCK0/ SDA1/ IRQ4 | PF0/ $\overline{\text{BREQ}}$ / IRQ2 | PF3/L WR/ ADTRG/ IRQ3 | MD2 | VCC | NMI | MD1 | NC | P43/AN3 | P45/AN5 |
| 8 | P36 | P35/ SCK1/ SCL0/ IRQ5 | P34/ RxD1/ SDA0 | P31/ RxD0 | PF6/ $\overline{\text{AS}}$ | VSS | $\overline{\text{RES}}$ | Vref | P44/AN4 | P46/AN6 | P96/DA0 |
| 7 | P75/ TMO3/ SCK3 | P74/ TMO2/ MRES | P76/ RxD3 | P77/ TxD3 | BP-112 BP-112V TBP-112A TBP-112AV (TOP VIEW) | | | P47/AN7 | P97/DA1 | AVSS | AVSS |
| 6 | P72/ TMO0/ CS6 | P71/ TMR123/ TMC123/ CS5 | P73/ TMO1/ CS7 | P70/ TMR101/ TMC101/ CS4 | | | | P17/ TIOCB2/ TCLKD | P14/ TIOCA1/ IRQ0 | P16/ TIOCA2/ IRQ1 | P15/ TIOCB1/ TCLKC |
| 5 | PG0/ IRQ6 | PG1/ CS3/ IRQ7 | PG2/ CS2 | PG4/ CS0 | | | | P10/ TIOCA0/ A20 | P11/ TIOCB0/ A21 | P13/ TIOCD0/ TCLKB/ A23 | P12/ TIOCC0/ TCLKA/ A22 |
| 4 | PG3/ CS1 | PE0/D0 | PE2/D2 | PE7/D7 | PD5/D13 | VSS | PC5/A5 | PB6/ A14/ TIOCA5 | PA1/ A17/ TxD2 | PA2/ A18/ RxD2 | PA3/ A19/ SCK2 |
| 3 | PE1/D1 | PE3/D3 | NC | PD2/D10 | PD6/D14 | CVCC | PC3/A3 | PB0/ A8/ TIOCA3 | PB3/ A11/ TIOCD3 | PB7/ A15/ TIOCB5 | PA0/A16 |
| 2 | PE4/D4 | PE5/D5 | PD0/D8 | PD3/D11 | CVCC | VSS | PC2/A2 | PC6/A6 | PB1/A9/ TIOCB3 | PB4/ A12/ TIOCA4 | PB5/ A13/ TIOCB4 |
| 1 | NC | PE6/D6 | PD1/D9 | PD4/D12 | PD7/D15 | PC0/A0 | PC1/A1 | PC4/A4 | PC7/A7 | PB2/ A10/ TIOCC3 | NC |

INDEX

**Figure 1.12 Pin Arrangement of H8S/2238 Group
(BP-112, BP-112V, TBP-112A, TBP-112AV: Top View, Only for HD64F2238R)**

(4) Pin Arrangement of H8S/2237 Group

Figures 1.13 and 1.14 show the pin arrangement of the H8S/2237 Group.

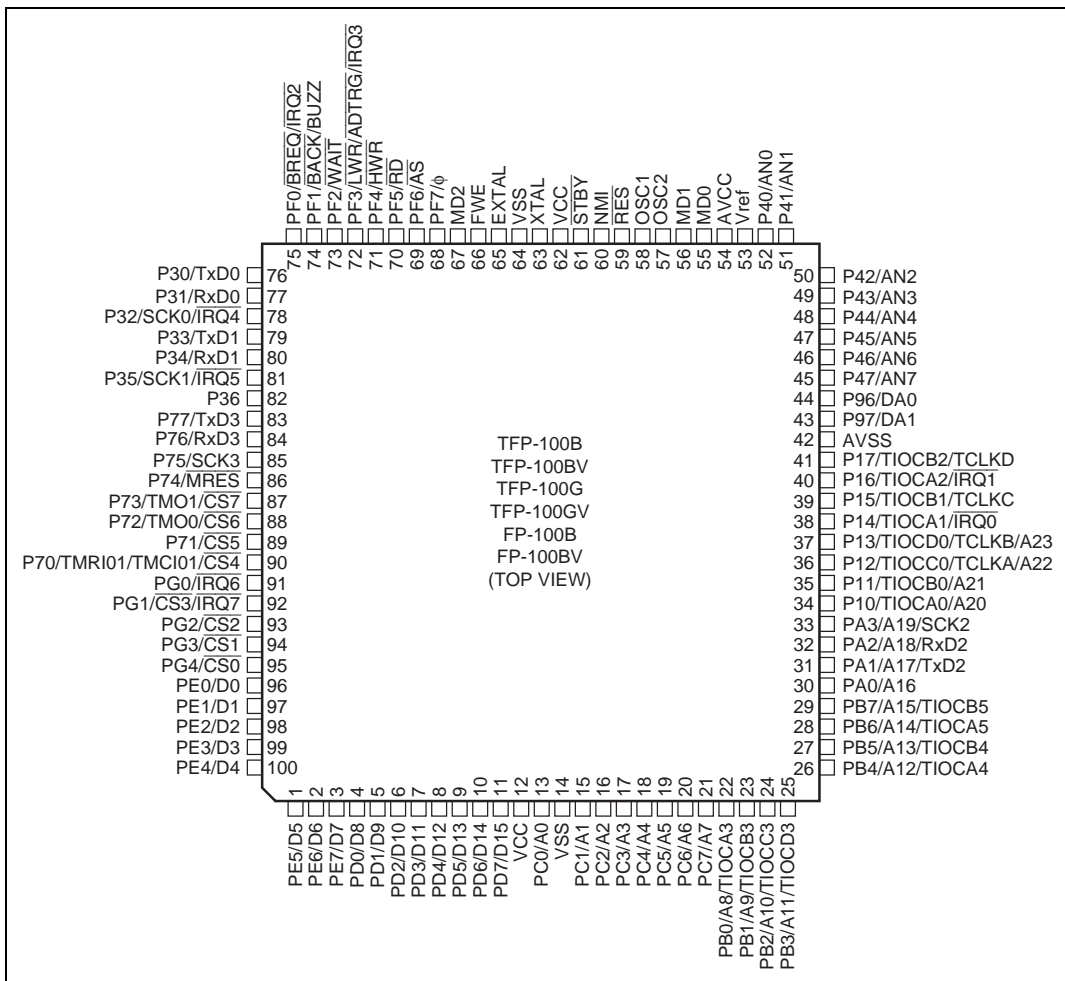


Figure 1.13 Pin Arrangement of H8S/2237 Group
(TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B, FP-100BV: Top View)

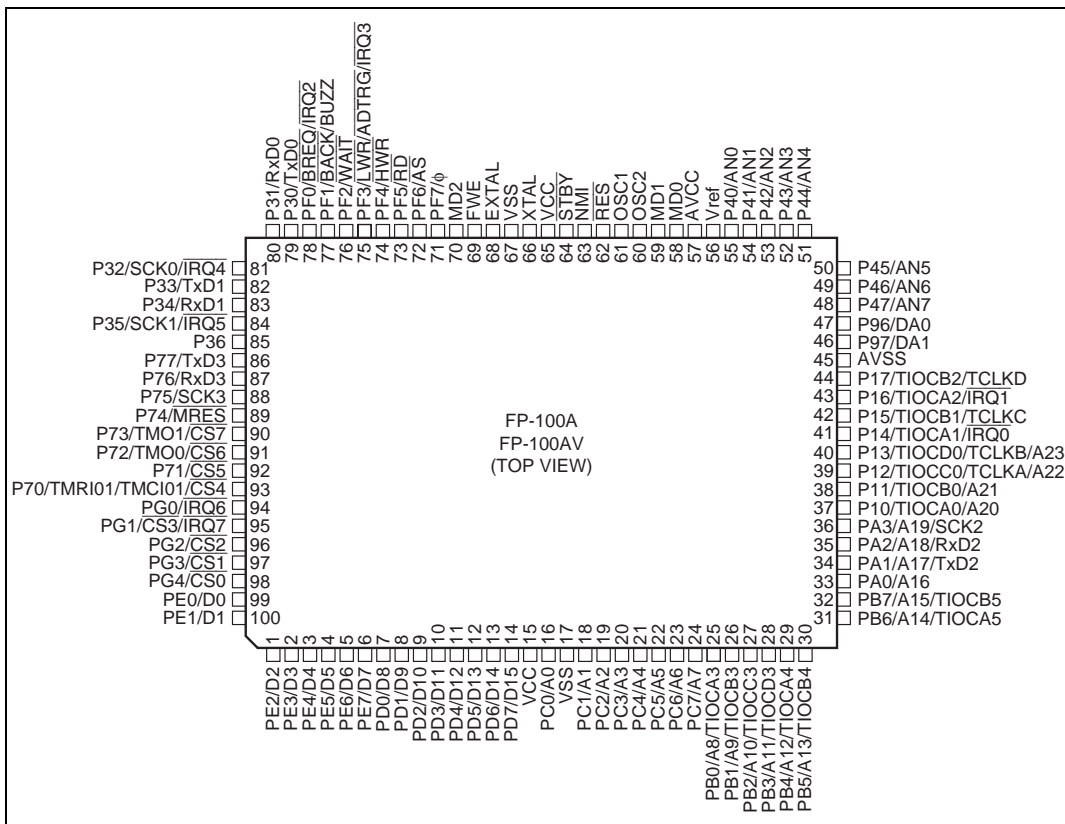


Figure 1.14 Pin Arrangement of H8S/2237 Group (FP-100A, FP-100AV: Top View)

(5) Pin Arrangement of H8S/2227 Group

Figures 1.15 and 1.16 show the pin arrangement of the H8S/2227 Group.

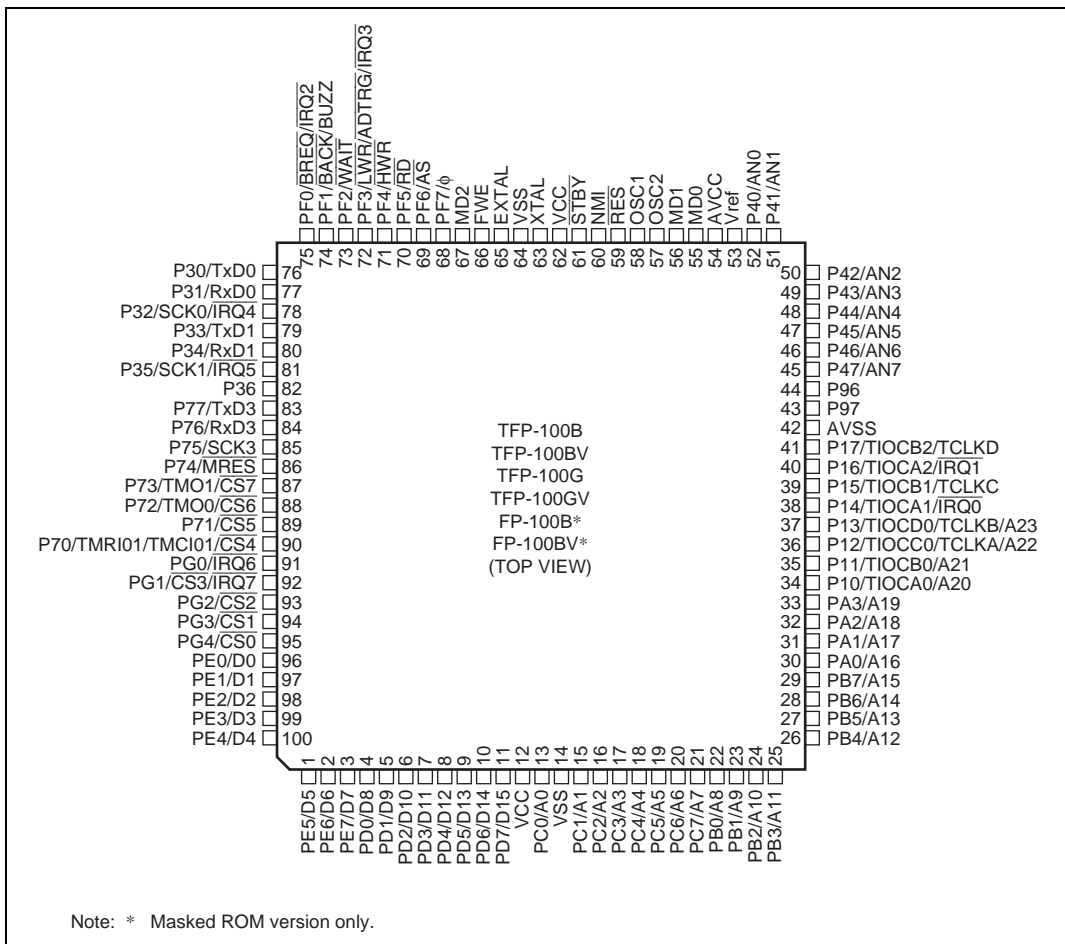
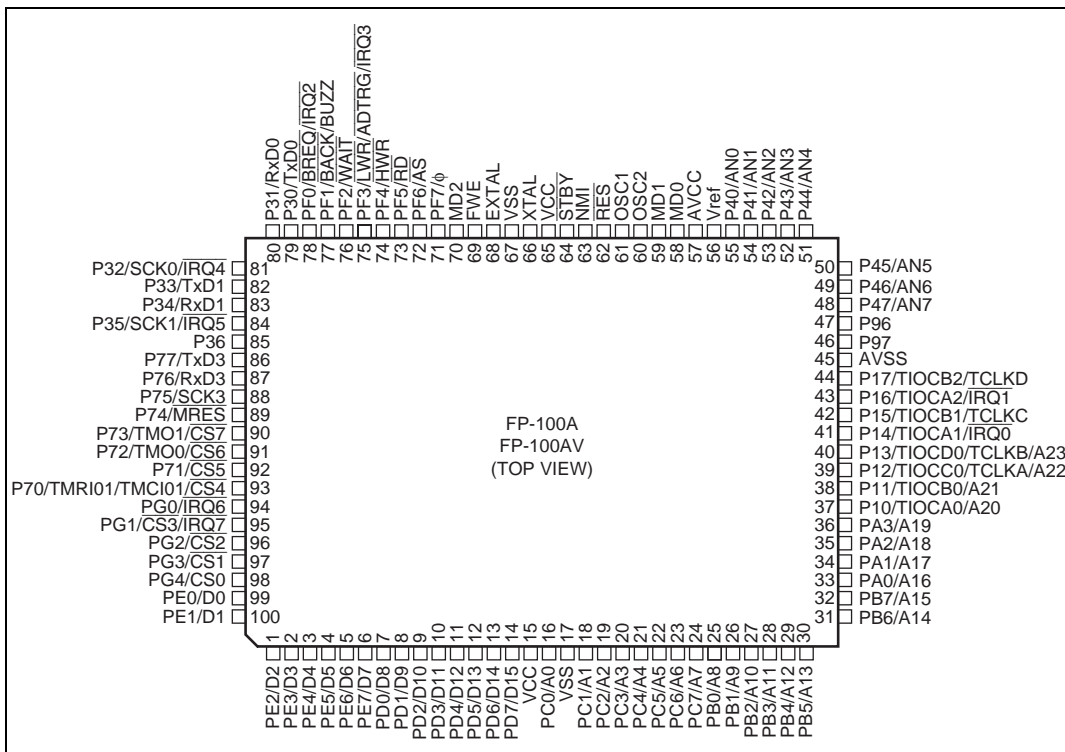


Figure 1.15 Pin Arrangement of H8S/2227 Group

(TFP-100B, TFP-100BV, TFP-100G, TFP-100GV, FP-100B*, FP-100BV*: Top View)



**Figure 1.16 Pin Arrangement of H8S/2227 Group
(FP-100A, FP-100AV: Top View, Only for HD6432227)**

1.3.2 Pin Arrangements in Each Mode

Tables 1.1 to 1.5 show the pin arrangements in each mode.

Table 1.1 Pin Arrangements in Each Mode of H8S/2258 Group

| Pin No. | | Pin Name | | | | | |
|----------|---------|----------|----------------|----------------|----------------|------------|---------------------------------|
| TFP-100B | FP-100B | FP-100A | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode* |
| 1 | 4 | | PE5/D5 | PE5/D5 | PE5/D5 | PE5 | \overline{OE} |
| 2 | 5 | | PE6/D6 | PE6/D6 | PE6/D6 | PE6 | \overline{WE} |
| 3 | 6 | | PE7/D7 | PE7/D7 | PE7/D7 | PE7 | \overline{CE} |
| 4 | 7 | | D8 | D8 | D8 | PD0 | D0 |
| 5 | 8 | | D9 | D9 | D9 | PD1 | D1 |
| 6 | 9 | | D10 | D10 | D10 | PD2 | D2 |
| 7 | 10 | | D11 | D11 | D11 | PD3 | D3 |
| 8 | 11 | | D12 | D12 | D12 | PD4 | D4 |
| 9 | 12 | | D13 | D13 | D13 | PD5 | D5 |
| 10 | 13 | | D14 | D14 | D14 | PD6 | D6 |
| 11 | 14 | | D15 | D15 | D15 | PD7 | D7 |
| 12 | 15 | | CVCC | CVCC | CVCC | CVCC | VCC |
| 13 | 16 | | A0 | A0 | PC0/A0 | PC0 | A0 |
| 14 | 17 | | VSS | VSS | VSS | VSS | VSS |
| 15 | 18 | | A1 | A1 | PC1/A1 | PC1 | A1 |
| 16 | 19 | | A2 | A2 | PC2/A2 | PC2 | A2 |
| 17 | 20 | | A3 | A3 | PC3/A3 | PC3 | A3 |
| 18 | 21 | | A4 | A4 | PC4/A4 | PC4 | A4 |
| 19 | 22 | | A5 | A5 | PC5/A5 | PC5 | A5 |
| 20 | 23 | | A6 | A6 | PC6/A6 | PC6 | A6 |
| 21 | 24 | | A7 | A7 | PC7/A7 | PC7 | A7 |
| 22 | 25 | | PB0/A8/TIOCA3 | PB0/A8/TIOCA3 | PB0/A8/TIOCA3 | PB0/TIOCA3 | A8 |
| 23 | 26 | | PB1/A9/TIOCB3 | PB1/A9/TIOCB3 | PB1/A9/TIOCB3 | PB1/TIOCB3 | A9 |
| 24 | 27 | | PB2/A10/TIOCC3 | PB2/A10/TIOCC3 | PB2/A10/TIOCC3 | PB2/TIOCC3 | A10 |
| 25 | 28 | | PB3/A11/TIOCD3 | PB3/A11/TIOCD3 | PB3/A11/TIOCD3 | PB3/TIOCD3 | A11 |

| Pin No. | | Pin Name | | | | |
|----------|---------|----------------------|----------------------|----------------------|------------------|---------------------------------|
| TFP-100B | | | | | | Flash Memory Programmable Mode* |
| FP-100B | FP-100A | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 26 | 29 | PB4/A12/TIOCA4 | PB4/A12/TIOCA4 | PB4/A12/TIOCA4 | PB4/TIOCA4 | A12 |
| 27 | 30 | PB5/A13/TIOCB4 | PB5/A13/TIOCB4 | PB5/A13/TIOCB4 | PB5/TIOCB4 | A13 |
| 28 | 31 | PB6/A14/TIOCA5 | PB6/A14/TIOCA5 | PB6/A14/TIOCA5 | PB6/TIOCA5 | A14 |
| 29 | 32 | PB7/A15/TIOCB5 | PB7/A15/TIOCB5 | PB7/A15/TIOCB5 | PB7/TIOCB5 | A15 |
| 30 | 33 | PA0/A16 | PA0/A16 | PA0/A16 | PA0 | A16 |
| 31 | 34 | PA1/A17/TxD2 | PA1/A17/TxD2 | PA1/A17/TxD2 | PA1/TxD2 | A17 |
| 32 | 35 | PA2/A18/RxD2 | PA2/A18/RxD2 | PA2/A18/RxD2 | PA2/RxD2 | A18 |
| 33 | 36 | PA3/A19/SCK2 | PA3/A19/SCK2 | PA3/A19/SCK2 | PA3/SCK2 | NC |
| 34 | 37 | P10/TIOCA0/A20 | P10/TIOCA0/A20 | P10/TIOCA0/A20 | P10/TIOCA0 | NC |
| 35 | 38 | P11/TIOCB0/A21 | P11/TIOCB0/A21 | P11/TIOCB0/A21 | P11/TIOCB0 | NC |
| 36 | 39 | P12/TIOCC0/TCLKA/A22 | P12/TIOCC0/TCLKA/A22 | P12/TIOCC0/TCLKA/A22 | P12/TIOCC0/TCLKA | NC |
| 37 | 40 | P13/TIOCD0/TCLKB/A23 | P13/TIOCD0/TCLKB/A23 | P13/TIOCD0/TCLKB/A23 | P13/TIOCD0/TCLKB | NC |
| 38 | 41 | P14/TIOCA1/IRQ0 | P14/TIOCA1/IRQ0 | P14/TIOCA1/IRQ0 | P14/TIOCA1/IRQ0 | VSS |
| 39 | 42 | P15/TIOCB1/TCLKC | P15/TIOCB1/TCLKC | P15/TIOCB1/TCLKC | P15/TIOCB1/TCLKC | NC |
| 40 | 43 | P16/TIOCA2/IRQ1 | P16/TIOCA2/IRQ1 | P16/TIOCA2/IRQ1 | P16/TIOCA2/IRQ1 | VSS |
| 41 | 44 | P17/TIOCB2/TCLKD | P17/TIOCB2/TCLKD | P17/TIOCB2/TCLKD | P17/TIOCB2/TCLKD | NC |
| 42 | 45 | AVSS | AVSS | AVSS | AVSS | VSS |
| 43 | 46 | P97/DA1 | P97/DA1 | P97/DA1 | P97/DA1 | NC |
| 44 | 47 | P96/DA0 | P96/DA0 | P96/DA0 | P96/DA0 | NC |
| 45 | 48 | P47/AN7 | P47/AN7 | P47/AN7 | P47/AN7 | NC |
| 46 | 49 | P46/AN6 | P46/AN6 | P46/AN6 | P46/AN6 | NC |
| 47 | 50 | P45/AN5 | P45/AN5 | P45/AN5 | P45/AN5 | NC |
| 48 | 51 | P44/AN4 | P44/AN4 | P44/AN4 | P44/AN4 | NC |

| Pin No. | | Pin Name | | | | |
|----------|---------|------------------------|------------------------|------------------------|--------------------|---------------------------------|
| TFP-100B | | | | | | |
| FP-100B | FP-100A | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode* |
| 49 | 52 | P43/AN3 | P43/AN3 | P43/AN3 | P43/AN3 | NC |
| 50 | 53 | P42/AN2 | P42/AN2 | P42/AN2 | P42/AN2 | NC |
| 51 | 54 | P41/AN1 | P41/AN1 | P41/AN1 | P41/AN1 | NC |
| 52 | 55 | P40/AN0 | P40/AN0 | P40/AN0 | P40/AN0 | NC |
| 53 | 56 | Vref | Vref | Vref | Vref | VCC |
| 54 | 57 | AVCC | AVCC | AVCC | AVCC | VCC |
| 55 | 58 | MD0 | MD0 | MD0 | MD0 | VSS |
| 56 | 59 | MD1 | MD1 | MD1 | MD1 | VSS |
| 57 | 60 | OSC2 | OSC2 | OSC2 | OSC2 | NC |
| 58 | 61 | OSC1 | OSC1 | OSC1 | OSC1 | VSS |
| 59 | 62 | RES | RES | RES | RES | RES |
| 60 | 63 | NMI | NMI | NMI | NMI | VCC |
| 61 | 64 | STBY | STBY | STBY | STBY | VCC |
| 62 | 65 | VCC | VCC | VCC | VCC | VCC |
| 63 | 66 | XTAL | XTAL | XTAL | XTAL | XTAL |
| 64 | 67 | VSS | VSS | VSS | VSS | VSS |
| 65 | 68 | EXTAL | EXTAL | EXTAL | EXTAL | EXTAL |
| 66 | 69 | FWE | FWE | FWE | FWE | FWE |
| 67 | 70 | MD2 | MD2 | MD2 | MD2 | VSS |
| 68 | 71 | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | NC |
| 69 | 72 | AS | AS | AS | PF6 | NC |
| 70 | 73 | RD | RD | RD | PF5 | NC |
| 71 | 74 | HWR | HWR | HWR | PF4 | NC |
| 72 | 75 | PF3/LWR/ ADTRG/IRQ3 | PF3/LWR/ ADTRG/IRQ3 | PF3/LWR/ ADTRG/IRQ3 | PF3/ADTRG/ IRQ3 | NC |
| 73 | 76 | PF2/WAIT | PF2/WAIT | PF2/WAIT | PF2 | NC |
| 74 | 77 | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BUZZ | NC |
| 75 | 78 | PF0/BREQ/ IRQ2 | PF0/BREQ/ IRQ2 | PF0/BREQ/ IRQ2 | PF0/IRQ2 | VCC |
| 76 | 79 | P30/TxD0 | P30/TxD0 | P30/TxD0 | P30/TxD0 | NC |
| 77 | 80 | P31/RxD0 | P31/RxD0 | P31/RxD0 | P31/RxD0 | NC |

| Pin No. | | Pin Name | | | | |
|----------|---------|---------------------------|---------------------------|---------------------------|------------------------|---------------------------------|
| TFP-100B | | | | | | Flash Memory Programmable Mode* |
| FP-100B | FP-100A | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 78 | 81 | P32/SCK0/ SDA1/IRQ4 | P32/SCK0/ SDA1/IRQ4 | P32/SCK0/ SDA1/IRQ4 | P32/SCK0/ SDA1/IRQ4 | NC |
| 79 | 82 | P33/TxD1/ SCL1 | P33/TxD1/ SCL1 | P33/TxD1/ SCL1 | P33/TxD1/ SCL1 | NC |
| 80 | 83 | P34/RxD1/ SDA0 | P34/RxD1/ SDA0 | P34/RxD1/ SDA0 | P34/RxD1/ SDA0 | NC |
| 81 | 84 | P35/SCK1/ SCL0/IRQ5 | P35/SCK1/ SCL0/IRQ5 | P35/SCK1/ SCL0/IRQ5 | P35/SCK1/ SCL0/IRQ5 | NC |
| 82 | 85 | P36 | P36 | P36 | P36 | NC |
| 83 | 86 | P77/TxD3 | P77/TxD3 | P77/TxD3 | P77/TxD3 | NC |
| 84 | 87 | P76/RxD3 | P76/RxD3 | P76/RxD3 | P76/RxD3 | NC |
| 85 | 88 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | NC |
| 86 | 89 | P74/TMO2/ MRES | P74/TMO2/ MRES | P74/TMO2/ MRES | P74/TMO2/ MRES | NC |
| 87 | 90 | P73/TMO1/CS7 | P73/TMO1/CS7 | P73/TMO1/CS7 | P73/TMO1 | NC |
| 88 | 91 | P72/TMO0/CS6 | P72/TMO0/CS6 | P72/TMO0/CS6 | P72/TMO0 | NC |
| 89 | 92 | P71/TMRI23/ TMC123/CS5 | P71/TMRI23/ TMC123/CS5 | P71/TMRI23/ TMC123/CS5 | P71/TMRI23/ TMC123 | NC |
| 90 | 93 | P70/TMRI01/ TMC101/CS4 | P70/TMRI01/ TMC101/CS4 | P70/TMRI01/ TMC101/CS4 | P70/TMRI01/ TMC101 | NC |
| 91 | 94 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | NC |
| 92 | 95 | PG1/CS3/IRQ7 | PG1/CS3/IRQ7 | PG1/CS3/IRQ7 | PG1/IRQ7 | NC |
| 93 | 96 | PG2/Tx/CS2 | PG2/Tx/CS2 | PG2/Tx/CS2 | PG2/Tx | NC |
| 94 | 97 | PG3/Rx/CS1 | PG3/Rx/CS1 | PG3/Rx/CS1 | PG3/Rx | NC |
| 95 | 98 | PG4/CS0 | PG4/CS0 | PG4/CS0 | PG4 | NC |
| 96 | 99 | PE0/D0 | PE0/D0 | PE0/D0 | PE0 | NC |
| 97 | 100 | PE1/D1 | PE1/D1 | PE1/D1 | PE1 | NC |
| 98 | 1 | PE2/D2 | PE2/D2 | PE2/D2 | PE2 | NC |
| 99 | 2 | PE3/D3 | PE3/D3 | PE3/D3 | PE3 | VCC |
| 100 | 3 | PE4/D4 | PE4/D4 | PE4/D4 | PE4 | VSS |

Note: * The NC should be left open.

Table 1.2 Pin Arrangements in Each Mode of H8S/2239 Group

| Pin No. | | Pin Name | | | | |
|-----------|-------------------------|--------------------|--------------------|--------------------|------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | TBP-112A ^{*1} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*2} |
| FP-100BV | TBP-112AV ^{*1} | | | | | |
| 1 | B2 | PE5/D5 | PE5/D5 | PE5/D5 | PE5 | \overline{OE} |
| 2 | B1 | PE6/D6 | PE6/D6 | PE6/D6 | PE6 | \overline{WE} |
| 3 | D4 | PE7/D7 | PE7/D7 | PE7/D7 | PE7 | \overline{CE} |
| 4 | C2 | D8 | D8 | D8 | PD0 | D0 |
| 5 | C1 | D9 | D9 | D9 | PD1 | D1 |
| 6 | D3 | D10 | D10 | D10 | PD2 | D2 |
| 7 | D2 | D11 | D11 | D11 | PD3 | D3 |
| 8 | D1 | D12 | D12 | D12 | PD4 | D4 |
| 9 | E4 | D13 | D13 | D13 | PD5 | D5 |
| 10 | E3 | D14 | D14 | D14 | PD6 | D6 |
| 11 | E1 | D15 | D15 | D15 | PD7 | D7 |
| 12 | E2, F3 | CVCC | CVCC | CVCC | CVCC | VCC |
| 13 | F1 | A0 | A0 | PC0/A0 | PC0 | A0 |
| 14 | F2, F4 | VSS | VSS | VSS | VSS | VSS |
| 15 | G1 | A1 | A1 | PC1/A1 | PC1 | A1 |
| 16 | G2 | A2 | A2 | PC2/A2 | PC2 | A2 |
| 17 | G3 | A3 | A3 | PC3/A3 | PC3 | A3 |
| 18 | H1 | A4 | A4 | PC4/A4 | PC4 | A4 |
| 19 | G4 | A5 | A5 | PC5/A5 | PC5 | A5 |
| 20 | H2 | A6 | A6 | PC6/A6 | PC6 | A6 |
| 21 | J1 | A7 | A7 | PC7/A7 | PC7 | A7 |
| 22 | H3 | PB0/A8/ TIOCA3 | PB0/A8/ TIOCA3 | PB0/A8/ TIOCA3 | PB0/TIOCA3 | A8 |
| 23 | J2 | PB1/A9/ TIOCB3 | PB1/A9/ TIOCB3 | PB1/A9/ TIOCB3 | PB1/TIOCB3 | A9 |
| 24 | K1 | PB2/A10/ TIOCC3 | PB2/A10/ TIOCC3 | PB2/A10/ TIOCC3 | PB2/TIOCC3 | A10 |
| 25 | J3 | PB3/A11/ TIOCD3 | PB3/A11/ TIOCD3 | PB3/A11/ TIOCD3 | PB3/TIOCD3 | A11 |
| 26 | K2 | PB4/A12/ TIOCA4 | PB4/A12/ TIOCA4 | PB4/A12/ TIOCA4 | PB4/TIOCA4 | A12 |

| Pin No. | | Pin Name | | | | |
|-----------|-------------------------|--------------------------|--------------------------|--------------------------|----------------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | TBP-112A ^{*1} | | | | | |
| FP-100BV | TBP-112AV ^{*1} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*2} |
| 27 | L2 | PB5/A13/ TIOCB4 | PB5/A13/ TIOCB4 | PB5/A13/ TIOCB4 | PB5/TIOCB4 | A13 |
| 28 | H4 | PB6/A14/ TIOCA5 | PB6/A14/ TIOCA5 | PB6/A14/ TIOCA5 | PB6/TIOCA5 | A14 |
| 29 | K3 | PB7/A15/ TIOCB5 | PB7/A15/ TIOCB5 | PB7/A15/ TIOCB5 | PB7/TIOCB5 | A15 |
| 30 | L3 | PA0/A16 | PA0/A16 | PA0/A16 | PA0 | A16 |
| 31 | J4 | PA1/A17/ TxD2 | PA1/A17/ TxD2 | PA1/A17/ TxD2 | PA1/TxD2 | A17 |
| 32 | K4 | PA2/A18/ RxD2 | PA2/A18/ RxD2 | PA2/A18/ RxD2 | PA2/RxD2 | A18 |
| 33 | L4 | PA3/A19/ SCK2 | PA3/A19/ SCK2 | PA3/A19/ SCK2 | PA3/SCK2 | NC |
| 34 | H5 | P10/TIOCA0/ DACK0/A20 | P10/TIOCA0/ DACK0/A20 | P10/TIOCA0/ DACK0/A20 | P10/TIOCA0/ DACK0 | NC |
| 35 | J5 | P11/TIOCB0/ DACK1/A21 | P11/TIOCB0/ DACK1/A21 | P11/TIOCB0/ DACK1/A21 | P11/TIOCB0/ DACK1 | NC |
| 36 | L5 | P12/TIOCC0/ TCLKA/A22 | P12/TIOCC0/ TCLKA/A22 | P12/TIOCC0/ TCLKA/A22 | P12/TIOCC0/ TCLKA | NC |
| 37 | K5 | P13/TIOCD0/ TCLKB/A23 | P13/TIOCD0/ TCLKB/A23 | P13/TIOCD0/ TCLKB/A23 | P13/TIOCD0/ TCLKB | NC |
| 38 | J6 | P14/TIOCA1/ IRQ0 | P14/TIOCA1/ IRQ0 | P14/TIOCA1/ IRQ0 | P14/TIOCA1/ IRQ0 | VSS |
| 39 | L6 | P15/TIOCB1/ TCLKC | P15/TIOCB1/ TCLKC | P15/TIOCB1/ TCLKC | P15/TIOCB1/ TCLKC | NC |
| 40 | K6 | P16/TIOCA2/ IRQ1 | P16/TIOCA2/ IRQ1 | P16/TIOCA2/ IRQ1 | P16/TIOCA2/ IRQ1 | VSS |
| 41 | H6 | P17/TIOCB2/ TCLKD | P17/TIOCB2/ TCLKD | P17/TIOCB2/ TCLKD | P17/TIOCB2/ TCLKD | NC |
| 42 | K7, L7 | AVSS | AVSS | AVSS | AVSS | VSS |
| 43 | J7 | P97/DA1 | P97/DA1 | P97/DA1 | P97/DA1 | NC |
| 44 | L8 | P96/DA0 | P96/DA0 | P96/DA0 | P96/DA0 | NC |
| 45 | H7 | P47/AN7 | P47/AN7 | P47/AN7 | P47/AN7 | NC |
| 46 | K8 | P46/AN6 | P46/AN6 | P46/AN6 | P46/AN6 | NC |
| 47 | L9 | P45/AN5 | P45/AN5 | P45/AN5 | P45/AN5 | NC |

| Pin No. | | Pin Name | | | | |
|-----------|-------------------------|----------------------------|----------------------------|----------------------------|--------------------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | TBP-112A* ¹ | | | | | Flash Memory Programmable Mode* ² |
| FP-100BV | TBP-112AV* ¹ | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 48 | J8 | P44/AN4 | P44/AN4 | P44/AN4 | P44/AN4 | NC |
| 49 | K9 | P43/AN3 | P43/AN3 | P43/AN3 | P43/AN3 | NC |
| 50 | L10 | P42/AN2 | P42/AN2 | P42/AN2 | P42/AN2 | NC |
| 51 | K10 | P41/AN1 | P41/AN1 | P41/AN1 | P41/AN1 | NC |
| 52 | K11 | P40/AN0 | P40/AN0 | P40/AN0 | P40/AN0 | NC |
| 53 | H8 | Vref | Vref | Vref | Vref | VCC |
| 54 | J10 | AVCC | AVCC | AVCC | AVCC | VCC |
| 55 | J11 | MD0 | MD0 | MD0 | MD0 | VSS |
| 56 | H9 | MD1 | MD1 | MD1 | MD1 | VSS |
| 57 | H10 | OSC2 | OSC2 | OSC2 | OSC2 | NC |
| 58 | H11 | OSC1 | OSC1 | OSC1 | OSC1 | VSS |
| 59 | G8 | $\overline{\text{RES}}$ | $\overline{\text{RES}}$ | $\overline{\text{RES}}$ | $\overline{\text{RES}}$ | $\overline{\text{RES}}$ |
| 60 | G9 | NMI | NMI | NMI | NMI | VCC |
| 61 | G11 | $\overline{\text{STBY}}$ | $\overline{\text{STBY}}$ | $\overline{\text{STBY}}$ | $\overline{\text{STBY}}$ | VCC |
| 62 | F9, G10 | VCC | VCC | VCC | VCC | VCC |
| 63 | F11 | XTAL | XTAL | XTAL | XTAL | XTAL |
| 64 | F8, F10 | VSS | VSS | VSS | VSS | VSS |
| 65 | E11 | EXTAL | EXTAL | EXTAL | EXTAL | EXTAL |
| 66 | E10 | FWE | FWE | FWE | FWE | FWE |
| 67 | E9 | MD2 | MD2 | MD2 | MD2 | VSS |
| 68 | D11 | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | NC |
| 69 | E8 | $\overline{\text{AS}}$ | $\overline{\text{AS}}$ | $\overline{\text{AS}}$ | PF6 | NC |
| 70 | D10 | $\overline{\text{RD}}$ | $\overline{\text{RD}}$ | $\overline{\text{RD}}$ | PF5 | NC |
| 71 | C11 | $\overline{\text{HWR}}$ | $\overline{\text{HWR}}$ | $\overline{\text{HWR}}$ | PF4 | NC |
| 72 | D9 | PF3/LWR/ ADTRG/ IRQ3 | PF3/LWR/ ADTRG/ IRQ3 | PF3/LWR/ ADTRG/ IRQ3 | PF3/ ADTRG/ IRQ3 | NC |
| 73 | C10 | PF2/WAIT | PF2/WAIT | PF2/WAIT | PF2 | NC |
| 74 | B11 | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BUZZ | NC |
| 75 | C9 | PF0/BREQ/ IRQ2 | PF0/BREQ/ IRQ2 | PF0/BREQ/ IRQ2 | PF0/IRQ2 | VCC |

| Pin No. | | Pin Name | | | | |
|-----------|-------------------------|-------------------------------------|-------------------------------------|-------------------------------------|---------------------------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | TBP-112A ^{*1} | | | | | Flash Memory Programmable Mode ^{*2} |
| FP-100BV | TBP-112AV ^{*1} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 76 | A10 | P30/TxD0 | P30/TxD0 | P30/TxD0 | P30/TxD0 | NC |
| 77 | D8 | P31/RxD0 | P31/RxD0 | P31/RxD0 | P31/RxD0 | NC |
| 78 | B9 | P32/SCK0/ SDA1/IRQ4 | P32/SCK0/ SDA1/IRQ4 | P32/SCK0/ SDA1/IRQ4 | P32/SCK0/ SDA1/IRQ4 | NC |
| 79 | A9 | P33/TxD1/ SCL1 | P33/TxD1/ SCL1 | P33/TxD1/ SCL1 | P33/TxD1/ SCL1 | NC |
| 80 | C8 | P34/RxD1/ SDA0 | P34/RxD1/ SDA0 | P34/RxD1/ SDA0 | P34/RxD1/ SDA0 | NC |
| 81 | B8 | P35/SCK1/ SCL0/IRQ5 | P35/SCK1/ SCL0/IRQ5 | P35/SCK1/ SCL0/IRQ5 | P35/SCK1/ SCL0/IRQ5 | NC |
| 82 | A8 | P36 | P36 | P36 | P36 | NC |
| 83 | D7 | P77/TxD3 | P77/TxD3 | P77/TxD3 | P77/TxD3 | NC |
| 84 | C7 | P76/RxD3 | P76/RxD3 | P76/RxD3 | P76/RxD3 | NC |
| 85 | A7 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | NC |
| 86 | B7 | P74/TMO2/ MRES | P74/TMO2/ MRES | P74/TMO2/ MRES | P74/TMO2/ MRES | NC |
| 87 | C6 | P73/TMO1/ TEND1/CS7 | P73/TMO1/ TEND1/CS7 | P73/TMO1/ TEND1/CS7 | P73/TMO1/ TEND1 | NC |
| 88 | A6 | P72/TMO0/ TEND0/CS6 | P72/TMO0/ TEND0/CS6 | P72/TMO0/ TEND0/CS6 | P72/TMO0/ TEND0 | NC |
| 89 | B6 | P71/TMRI23/ TMC123/ DREQ1/CS5 | P71/TMRI23/ TMC123/ DREQ1/CS5 | P71/TMRI23/ TMC123/ DREQ1/CS5 | P71/TMRI23/ TMC123/ DREQ1 | NC |
| 90 | D6 | P70/TMRI01/ TMC101/ DREQ0/CS4 | P70/TMRI01/ TMC101/ DREQ0/CS4 | P70/TMRI01/ TMC101/ DREQ0/CS4 | P70/TMRI01/ TMC101/ DREQ0 | NC |
| 91 | A5 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | NC |
| 92 | B5 | PG1/CS3/ IRQ7 | PG1/CS3/ IRQ7 | PG1/CS3/ IRQ7 | PG1/IRQ7 | NC |
| 93 | C5 | PG2/CS2 | PG2/CS2 | PG2/CS2 | PG2 | NC |
| 94 | A4 | PG3/CS1 | PG3/CS1 | PG3/CS1 | PG3 | NC |
| 95 | D5 | PG4/CS0 | PG4/CS0 | PG4/CS0 | PG4 | NC |
| 96 | B4 | PE0/D0 | PE0/D0 | PE0/D0 | PE0 | NC |
| 97 | A3 | PE1/D1 | PE1/D1 | PE1/D1 | PE1 | NC |

| Pin No. | | Pin Name | | | | |
|-----------|-------------------------|----------|--------|--------|--------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | TBP-112A ^{*1} | | | | | Flash Memory Programmable Mode ^{*2} |
| FP-100BV | TBP-112AV ^{*1} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 98 | C4 | PE2/D2 | PE2/D2 | PE2/D2 | PE2 | NC |
| 99 | B3 | PE3/D3 | PE3/D3 | PE3/D3 | PE3 | VCC |
| 100 | A2 | PE4/D4 | PE4/D4 | PE4/D4 | PE4 | VSS |

Notes: 1. Supported only by HD64F2239.

2. The NC should be left open.

Table 1.3 Pin Arrangements in Each Mode of H8S/2238 Group

| Pin No. | | | Pin Name | | | | |
|---|---|--|--------------------|--------------------|--------------------|----------------|--|
| TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B FP-100BV | FP-100A ^{*1} FP-100AV ^{*1} | BP-112 ^{*2} BP-112V ^{*2} TBP-112A ^{*2} TBP-112AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*4} |
| 1 | 4 | B2 | PE5/D5 | PE5/D5 | PE5/D5 | PE5 | \overline{OE} |
| 2 | 5 | B1 | PE6/D6 | PE6/D6 | PE6/D6 | PE6 | \overline{WE} |
| 3 | 6 | D4 | PE7/D7 | PE7/D7 | PE7/D7 | PE7 | \overline{CE} |
| 4 | 7 | C2 | D8 | D8 | D8 | PD0 | D0 |
| 5 | 8 | C1 | D9 | D9 | D9 | PD1 | D1 |
| 6 | 9 | D3 | D10 | D10 | D10 | PD2 | D2 |
| 7 | 10 | D2 | D11 | D11 | D11 | PD3 | D3 |
| 8 | 11 | D1 | D12 | D12 | D12 | PD4 | D4 |
| 9 | 12 | E4 | D13 | D13 | D13 | PD5 | D5 |
| 10 | 13 | E3 | D14 | D14 | D14 | PD6 | D6 |
| 11 | 14 | E1 | D15 | D15 | D15 | PD7 | D7 |
| 12 | 15 | E2, F3 | CVCC | CVCC | CVCC | CVCC | VCC |
| 13 | 16 | F1 | A0 | A0 | PC0/A0 | PC0 | A0 |
| 14 | 17 | F2, F4 | VSS | VSS | VSS | VSS | VSS |
| 15 | 18 | G1 | A1 | A1 | PC1/A1 | PC1 | A1 |
| 16 | 19 | G2 | A2 | A2 | PC2/A2 | PC2 | A2 |
| 17 | 20 | G3 | A3 | A3 | PC3/A3 | PC3 | A3 |
| 18 | 21 | H1 | A4 | A4 | PC4/A4 | PC4 | A4 |
| 19 | 22 | G4 | A5 | A5 | PC5/A5 | PC5 | A5 |
| 20 | 23 | H2 | A6 | A6 | PC6/A6 | PC6 | A6 |
| 21 | 24 | J1 | A7 | A7 | PC7/A7 | PC7 | A7 |
| 22 | 25 | H3 | PB0/A8/ TIOCA3 | PB0/A8/ TIOCA3 | PB0/A8/ TIOCA3 | PB0/ TIOCA3 | A8 |
| 23 | 26 | J2 | PB1/A9/ TIOCB3 | PB1/A9/ TIOCB3 | PB1/A9/ TIOCB3 | PB1/ TIOCB3 | A9 |
| 24 | 27 | K1 | PB2/A10/ TIOCC3 | PB2/A10/ TIOCC3 | PB2/A10/ TIOCC3 | PB2/ TIOCC3 | A10 |
| 25 | 28 | J3 | PB3/A11/ TIOCD3 | PB3/A11/ TIOCD3 | PB3/A11/ TIOCD3 | PB3/ TIOCD3 | A11 |
| 26 | 29 | K2 | PB4/A12/ TIOCA4 | PB4/A12/ TIOCA4 | PB4/A12/ TIOCA4 | PB4/ TIOCA4 | A12 |

| Pin No. | | | Pin Name | | | | |
|-----------|------------|------------|------------------------------|------------------------------|------------------------------|--------------------------|---------------------|
| TFP-100B | | | | | | | |
| TFP-100BV | | BP-112*2 | | | | | |
| TFP-100G | | BP-112V*2 | | | | | |
| TFP-100GV | | TBP-112A*2 | | | | | |
| FP-100B | FP-100A*1 | TBP- | | | | | Flash Memory |
| FP-100BV | FP-100AV*1 | 112AV*2 | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Programmable Mode*4 |
| 27 | 30 | L2 | PB5/A13/ TIOCB4 | PB5/A13/ TIOCB4 | PB5/A13/ TIOCB4 | PB5/ TIOCB4 | A13 |
| 28 | 31 | H4 | PB6/A14/ TIOCA5 | PB6/A14/ TIOCA5 | PB6/A14/ TIOCA5 | PB6/ TIOCA5 | A14 |
| 29 | 32 | K3 | PB7/A15/ TIOCB5 | PB7/A15/ TIOCB5 | PB7/A15/ TIOCB5 | PB7/ TIOCB5 | A15 |
| 30 | 33 | L3 | PA0/A16 | PA0/A16 | PA0/A16 | PA0 | A16 |
| 31 | 34 | J4 | PA1/A17/ TxD2 | PA1/A17/ TxD2 | PA1/A17/ TxD2 | PA1/TxD2 | A17 |
| 32 | 35 | K4 | PA2/A18/ RxD2 | PA2/A18/ RxD2 | PA2/A18/ RxD2 | PA2/ RxD2 | A18 |
| 33 | 36 | L4 | PA3/A19/ SCK2 | PA3/A19/ SCK2 | PA3/A19/ SCK2 | PA3/ SCK2 | NC |
| 34 | 37 | H5 | P10/ TIOCA0/ A20 | P10/ TIOCA0/ A20 | P10/ TIOCA0/ A20 | P10/ TIOCA0 | NC |
| 35 | 38 | J5 | P11/ TIOCB0/ A21 | P11/ TIOCB0/ A21 | P11/ TIOCB0/ A21 | P11/ TIOCB0 | NC |
| 36 | 39 | L5 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA | NC |
| 37 | 40 | K5 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB | NC |
| 38 | 41 | J6 | P14/ TIOCA1/ IRQ0 | P14/ TIOCA1/ IRQ0 | P14/ TIOCA1/ IRQ0 | P14/ TIOCA1/ IRQ0 | VSS |
| 39 | 42 | L6 | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | NC |
| 40 | 43 | K6 | P16/ TIOCA2/ IRQ1 | P16/ TIOCA2/ IRQ1 | P16/ TIOCA2/ IRQ1 | P16/ TIOCA2/ IRQ1 | VSS |
| 41 | 44 | H6 | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | NC |

| Pin No. | | | Pin Name | | | | |
|-----------|------------------------|------------------------|-------------|-------------|-------------|-------------|--------------------|
| TFP-100B | | | | | | | |
| TFP-100BV | | BP-112 ^{*2} | | | | | |
| TFP-100G | | BP-112V ^{*2} | | | | | |
| TFP-100GV | | TBP-112A ^{*2} | | | | | Flash Memory |
| FP-100B | FP-100A ^{*1} | TBP- | | | | | Programmable |
| FP-100BV | FP-100AV ^{*1} | 112AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Mode ^{*4} |
| 42 | 45 | K7, L7 | AVSS | AVSS | AVSS | AVSS | VSS |
| 43 | 46 | J7 | P97/DA1 | P97/DA1 | P97/DA1 | P97/DA1 | NC |
| 44 | 47 | L8 | P96/DA0 | P96/DA0 | P96/DA0 | P96/DA0 | NC |
| 45 | 48 | H7 | P47/AN7 | P47/AN7 | P47/AN7 | P47/AN7 | NC |
| 46 | 49 | K8 | P46/AN6 | P46/AN6 | P46/AN6 | P46/AN6 | NC |
| 47 | 50 | L9 | P45/AN5 | P45/AN5 | P45/AN5 | P45/AN5 | NC |
| 48 | 51 | J8 | P44/AN4 | P44/AN4 | P44/AN4 | P44/AN4 | NC |
| 49 | 52 | K9 | P43/AN3 | P43/AN3 | P43/AN3 | P43/AN3 | NC |
| 50 | 53 | L10 | P42/AN2 | P42/AN2 | P42/AN2 | P42/AN2 | NC |
| 51 | 54 | K10 | P41/AN1 | P41/AN1 | P41/AN1 | P41/AN1 | NC |
| 52 | 55 | K11 | P40/AN0 | P40/AN0 | P40/AN0 | P40/AN0 | NC |
| 53 | 56 | H8 | Vref | Vref | Vref | Vref | VCC |
| 54 | 57 | J10 | AVCC | AVCC | AVCC | AVCC | VCC |
| 55 | 58 | J11 | MD0 | MD0 | MD0 | MD0 | VSS |
| 56 | 59 | H9 | MD1 | MD1 | MD1 | MD1 | VSS |
| 57 | 60 | H10 | OSC2 | OSC2 | OSC2 | OSC2 | NC |
| 58 | 61 | H11 | OSC1 | OSC1 | OSC1 | OSC1 | VSS |
| 59 | 62 | G8 | RES | RES | RES | RES | RES |
| 60 | 63 | G9 | NMI | NMI | NMI | NMI | VCC |
| 61 | 64 | G11 | STBY | STBY | STBY | STBY | VCC |
| 62 | 65 | F9, G10 | VCC | VCC | VCC | VCC | VCC |
| 63 | 66 | F11 | XTAL | XTAL | XTAL | XTAL | XTAL |
| 64 | 67 | F8, F10 | VSS | VSS | VSS | VSS | VSS |
| 65 | 68 | E11 | EXTAL | EXTAL | EXTAL | EXTAL | EXTAL |
| 66 | 69 | E10 | FWE | FWE | FWE | FWE | FWE |
| 67 | 70 | E9 | MD2 | MD2 | MD2 | MD2 | VSS |
| 68 | 71 | D11 | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | NC |
| 69 | 72 | E8 | AS | AS | AS | PF6 | NC |
| 70 | 73 | D10 | RD | RD | RD | PF5 | NC |
| 71 | 74 | C11 | HWR | HWR | HWR | PF4 | NC |

| Pin No. | | | Pin Name | | | | |
|-----------|------------|------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|---------------------|
| TFP-100B | | | | | | | |
| TFP-100BV | | BP-112*2 | | | | | |
| TFP-100G | | BP-112V*2 | | | | | |
| TFP-100GV | | TBP-112A*2 | | | | | |
| FP-100B | FP-100A*1 | TBP- | | | | | Flash Memory |
| FP-100BV | FP-100AV*1 | 112AV*2 | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Programmable Mode*4 |
| 72 | 75 | D9 | PF3/ LWR/ ADTRG/ IRQ3 | PF3/ LWR/ ADTRG/ IRQ3 | PF3/ LWR/ ADTRG/ IRQ3 | PF3/ ADTRG/ IRQ3 | NC*3 |
| 73 | 76 | C10 | PF2/ WAIT | PF2/ WAIT | PF2/ WAIT | PF2 | NC |
| 74 | 77 | B11 | PF1/ BACK/ BUZZ | PF1/ BACK/ BUZZ | PF1/ BACK/ BUZZ | PF1/ BUZZ | NC |
| 75 | 78 | C9 | PF0/ BREQ/ IRQ2 | PF0/ BREQ/ IRQ2 | PF0/ BREQ/ IRQ2 | PF0/ IRQ2 | VCC |
| 76 | 79 | A10 | P30/ TxD0 | P30/ TxD0 | P30/ TxD0 | P30/ TxD0 | NC |
| 77 | 80 | D8 | P31/ RxD0 | P31/ RxD0 | P31/ RxD0 | P31/ RxD0 | NC |
| 78 | 81 | B9 | P32/ SCK0/ SDA1/ IRQ4 | P32/ SCK0/ SDA1/ IRQ4 | P32/ SCK0/ SDA1/ IRQ4 | P32/ SCK0/ SDA1/ IRQ4 | NC |
| 79 | 82 | A9 | P33/ TxD1/ SCL1 | P33/ TxD1/ SCL1 | P33/ TxD1/ SCL1 | P33/ TxD1/ SCL1 | NC |
| 80 | 83 | C8 | P34/ RxD1/ SDA0 | P34/ RxD1/ SDA0 | P34/ RxD1/ SDA0 | P34/ RxD1/ SDA0 | NC |
| 81 | 84 | B8 | P35/ SCK1/ SCL0/ IRQ5 | P35/ SCK1/ SCL0/ IRQ5 | P35/ SCK1/ SCL0/ IRQ5 | P35/ SCK1/ SCL0/ IRQ5 | NC |
| 82 | 85 | A8 | P36 | P36 | P36 | P36 | NC |
| 83 | 86 | D7 | P77/ TxD3 | P77/ TxD3 | P77/ TxD3 | P77/ TxD3 | NC |
| 84 | 87 | C7 | P76/ RxD3 | P76/ RxD3 | P76/ RxD3 | P76/ RxD3 | NC |

| Pin No. | | | Pin Name | | | | |
|-----------|------------------------|------------------------|-------------------------------|-------------------------------|-------------------------------|-----------------------|---------------------------------|
| TFP-100B | | | | | | | |
| TFP-100BV | | BP-112 ^{*2} | | | | | |
| TFP-100G | | BP-112V ^{*2} | | | | | |
| TFP-100GV | | TBP-112A ^{*2} | | | | | |
| FP-100B | FP-100A ^{*1} | TBP- | | | | | Flash Memory |
| FP-100BV | FP-100AV ^{*1} | 112AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Programmable Mode ^{*4} |
| 85 | 88 | A7 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | P75/TMO3/ SCK3 | NC |
| 86 | 89 | B7 | P74/TMO2/ MRES | P74/TMO2/ MRES | P74/TMO2/ MRES | P74/TMO2/ MRES | NC |
| 87 | 90 | C6 | P73/TMO1/ CS7 | P73/TMO1/ CS7 | P73/TMO1/ CS7 | P73/TMO1 | NC |
| 88 | 91 | A6 | P72/TMO0/ CS6 | P72/TMO0/ CS6 | P72/TMO0/ CS6 | P72/TMO0 | NC |
| 89 | 92 | B6 | P71/TMRI23/ TMCi23/ CS5 | P71/TMRI23/ TMCi23/ CS5 | P71/TMRI23/ TMCi23/ CS5 | P71/TMRI23/ TMCi23 | NC |
| 90 | 93 | D6 | P70/TMRI01/ TMCi01/ CS4 | P70/TMRI01/ TMCi01/ CS4 | P70/TMRI01/ TMCi01/ CS4 | P70/TMRI01/ TMCi01 | NC |
| 91 | 94 | A5 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | NC |
| 92 | 95 | B5 | PG1/CS3/ IRQ7 | PG1/CS3/ IRQ7 | PG1/CS3/ IRQ7 | PG1/IRQ7 | NC |
| 93 | 96 | C5 | PG2/CS2 | PG2/CS2 | PG2/CS2 | PG2 | NC |
| 94 | 97 | A4 | PG3/CS1 | PG3/CS1 | PG3/CS1 | PG3 | NC |
| 95 | 98 | D5 | PG4/CS0 | PG4/CS0 | PG4/CS0 | PG4 | NC |
| 96 | 99 | B4 | PE0/D0 | PE0/D0 | PE0/D0 | PE0 | NC |
| 97 | 100 | A3 | PE1/D1 | PE1/D1 | PE1/D1 | PE1 | NC |
| 98 | 1 | C4 | PE2/D2 | PE2/D2 | PE2/D2 | PE2 | NC |
| 99 | 2 | B3 | PE3/D3 | PE3/D3 | PE3/D3 | PE3 | VCC |
| 100 | 3 | A2 | PE4/D4 | PE4/D4 | PE4/D4 | PE4 | VSS |

- Notes:
1. Supported only by the H8S/2238B and H8S/2236B.
 2. Supported only by the HD64F2238R.
 3. Vcc in the H8S/2238B and H8S/2236B.
 4. The NC should be left open.

Table 1.4 Pin Arrangements in Each Mode of H8S/2237 Group

| Pin No. | | Pin Name | | | | |
|-----------|----------|--------------------|--------------------|--------------------|----------------|-----------------|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | FP-100A | Mode 4 | Mode 5 | Mode 6 | Mode 7 | PROM Mode* |
| FP-100BV | FP-100AV | | | | | |
| 1 | 4 | PE5/D5 | PE5/D5 | PE5/D5 | PE5 | NC |
| 2 | 5 | PE6/D6 | PE6/D6 | PE6/D6 | PE6 | NC |
| 3 | 6 | PE7/D7 | PE7/D7 | PE7/D7 | PE7 | NC |
| 4 | 7 | D8 | D8 | D8 | PD0 | D0 |
| 5 | 8 | D9 | D9 | D9 | PD1 | D1 |
| 6 | 9 | D10 | D10 | D10 | PD2 | D2 |
| 7 | 10 | D11 | D11 | D11 | PD3 | D3 |
| 8 | 11 | D12 | D12 | D12 | PD4 | D4 |
| 9 | 12 | D13 | D13 | D13 | PD5 | D5 |
| 10 | 13 | D14 | D14 | D14 | PD6 | D6 |
| 11 | 14 | D15 | D15 | D15 | PD7 | D7 |
| 12 | 15 | VCC | VCC | VCC | VCC | VCC |
| 13 | 16 | A0 | A0 | PC0/A0 | PC0 | A0 |
| 14 | 17 | VSS | VSS | VSS | VSS | VSS |
| 15 | 18 | A1 | A1 | PC1/A1 | PC1 | A1 |
| 16 | 19 | A2 | A2 | PC2/A2 | PC2 | A2 |
| 17 | 20 | A3 | A3 | PC3/A3 | PC3 | A3 |
| 18 | 21 | A4 | A4 | PC4/A4 | PC4 | A4 |
| 19 | 22 | A5 | A5 | PC5/A5 | PC5 | A5 |
| 20 | 23 | A6 | A6 | PC6/A6 | PC6 | A6 |
| 21 | 24 | A7 | A7 | PC7/A7 | PC7 | A7 |
| 22 | 25 | PB0/A8/ TIOCA3 | PB0/A8/ TIOCA3 | PB0/A8/ TIOCA3 | PB0/ TIOCA3 | A8 |
| 23 | 26 | PB1/A9/ TIOCB3 | PB1/A9/ TIOCB3 | PB1/A9/ TIOCB3 | PB1/ TIOCB3 | \overline{OE} |
| 24 | 27 | PB2/A10/ TIOCC3 | PB2/A10/ TIOCC3 | PB2/A10/ TIOCC3 | PB2/ TIOCC3 | A10 |

| Pin No. | | Pin Name | | | | |
|-----------|----------|------------------------------|------------------------------|------------------------------|--------------------------|-------|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | FP-100A | | | | | PROM |
| FP-100BV | FP-100AV | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Mode* |
| 25 | 28 | PB3/A11/ TIOCD3 | PB3/A11/ TIOCD3 | PB3/A11/ TIOCD3 | PB3/ TIOCD3 | A11 |
| 26 | 29 | PB4/A12/ TIOCA4 | PB4/A12/ TIOCA4 | PB4/A12/ TIOCA4 | PB4/ TIOCA4 | A12 |
| 27 | 30 | PB5/A13/ TIOCB4 | PB5/A13/ TIOCB4 | PB5/A13/ TIOCB4 | PB5/ TIOCB4 | A13 |
| 28 | 31 | PB6/A14/ TIOCA5 | PB6/A14/ TIOCA5 | PB6/A14/ TIOCA5 | PB6/ TIOCA5 | A14 |
| 29 | 32 | PB7/A15/ TIOCB5 | PB7/A15/ TIOCB5 | PB7/A15/ TIOCB5 | PB7/ TIOCB5 | A15 |
| 30 | 33 | PA0/A16 | PA0/A16 | PA0/A16 | PA0 | A16 |
| 31 | 34 | PA1/A17/ TxD2 | PA1/A17/ TxD2 | PA1/A17/ TxD2 | PA1/TxD2 | VCC |
| 32 | 35 | PA2/A18/ RxD2 | PA2/A18/ RxD2 | PA2/A18/ RxD2 | PA2/RxD2 | VCC |
| 33 | 36 | PA3/A19/ SCK2 | PA3/A19/ SCK2 | PA3/A19/ SCK2 | PA3/SCK2 | NC |
| 34 | 37 | P10/ TIOCA0/A20 | P10/ TIOCA0/A20 | P10/ TIOCA0/A20 | P10/ TIOCA0 | NC |
| 35 | 38 | P11/ TIOCB0/A21 | P11/ TIOCB0/A21 | P11/ TIOCB0/A21 | P11/ TIOCB0 | NC |
| 36 | 39 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA | NC |
| 37 | 40 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB | NC |
| 38 | 41 | P14/ TIOCA1/ IRQ0 | P14/ TIOCA1/ IRQ0 | P14/ TIOCA1/ IRQ0 | P14/ TIOCA1/ IRQ0 | NC |
| 39 | 42 | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | NC |

| Pin No. | | Pin Name | | | | |
|--|---------------------|--------------------------|--------------------------|--------------------------|--------------------------|---------------|
| TFP-100B TFP-100BV TFP-100G TFP-100GV | | | | | | |
| FP-100B FP-100BV | FP-100A FP-100AV | Mode 4 | Mode 5 | Mode 6 | Mode 7 | PROM Mode* |
| 40 | 43 | P16/ TIOCA2/ IRQ1 | P16/ TIOCA2/ IRQ1 | P16/ TIOCA2/ IRQ1 | P16/ TIOCA2/ IRQ1 | NC |
| 41 | 44 | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | NC |
| 42 | 45 | AVSS | AVSS | AVSS | AVSS | VSS |
| 43 | 46 | P97/DA1 | P97/DA1 | P97/DA1 | P97/DA1 | NC |
| 44 | 47 | P96/DA0 | P96/DA0 | P96/DA0 | P96/DA0 | NC |
| 45 | 48 | P47/AN7 | P47/AN7 | P47/AN7 | P47/AN7 | NC |
| 46 | 49 | P46/AN6 | P46/AN6 | P46/AN6 | P46/AN6 | NC |
| 47 | 50 | P45/AN5 | P45/AN5 | P45/AN5 | P45/AN5 | NC |
| 48 | 51 | P44/AN4 | P44/AN4 | P44/AN4 | P44/AN4 | NC |
| 49 | 52 | P43/AN3 | P43/AN3 | P43/AN3 | P43/AN3 | NC |
| 50 | 53 | P42/AN2 | P42/AN2 | P42/AN2 | P42/AN2 | NC |
| 51 | 54 | P41/AN1 | P41/AN1 | P41/AN1 | P41/AN1 | NC |
| 52 | 55 | P40/AN0 | P40/AN0 | P40/AN0 | P40/AN0 | NC |
| 53 | 56 | Vref | Vref | Vref | Vref | VCC |
| 54 | 57 | AVCC | AVCC | AVCC | AVCC | VCC |
| 55 | 58 | MD0 | MD0 | MD0 | MD0 | VSS |
| 56 | 59 | MD1 | MD1 | MD1 | MD1 | VSS |
| 57 | 60 | OSC2 | OSC2 | OSC2 | OSC2 | NC |
| 58 | 61 | OSC1 | OSC1 | OSC1 | OSC1 | NC |
| 59 | 62 | RES | RES | RES | RES | VPP |
| 60 | 63 | NMI | NMI | NMI | NMI | A9 |
| 61 | 64 | STBY | STBY | STBY | STBY | VSS |
| 62 | 65 | VCC | VCC | VCC | VCC | VCC |
| 63 | 66 | XTAL | XTAL | XTAL | XTAL | NC |
| 64 | 67 | VSS | VSS | VSS | VSS | VSS |
| 65 | 68 | EXTAL | EXTAL | EXTAL | EXTAL | NC |

| Pin No. | | Pin Name | | | | |
|-----------|----------|---|---|---|--------------------------------|-----------------|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B | FP-100A | | | | | PROM |
| FP-100BV | FP-100AV | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Mode* |
| 66 | 69 | FWE | FWE | FWE | FWE | NC |
| 67 | 70 | MD2 | MD2 | MD2 | MD2 | VSS |
| 68 | 71 | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | NC |
| 69 | 72 | \overline{AS} | \overline{AS} | \overline{AS} | PF6 | NC |
| 70 | 73 | \overline{RD} | \overline{RD} | \overline{RD} | PF5 | NC |
| 71 | 74 | \overline{HWR} | \overline{HWR} | \overline{HWR} | PF4 | NC |
| 72 | 75 | PF3/LWR/ ADTRG/ IRQ3 | PF3/LWR/ ADTRG/ IRQ3 | PF3/LWR/ ADTRG/ IRQ3 | PF3/ADTRG/ IRQ3 | NC |
| 73 | 76 | PF2/ \overline{WAIT} | PF2/ \overline{WAIT} | PF2/ \overline{WAIT} | PF2 | \overline{CE} |
| 74 | 77 | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BUZZ | PGM |
| 75 | 78 | PF0/ \overline{BREQ} / $\overline{IRQ2}$ | PF0/ \overline{BREQ} / $\overline{IRQ2}$ | PF0/ \overline{BREQ} / $\overline{IRQ2}$ | PF0/ $\overline{IRQ2}$ | NC |
| 76 | 79 | P30/TxD0 | P30/TxD0 | P30/TxD0 | P30/TxD0 | NC |
| 77 | 80 | P31/RxD0 | P31/RxD0 | P31/RxD0 | P31/RxD0 | NC |
| 78 | 81 | P32/SCK0/ IRQ4 | P32/SCK0/ IRQ4 | P32/SCK0/ IRQ4 | P32/SCK0/ IRQ4 | NC |
| 79 | 82 | P33/TxD1 | P33/TxD1 | P33/TxD1 | P33/TxD1 | NC |
| 80 | 83 | P34/RxD1 | P34/RxD1 | P34/RxD1 | P34/RxD1 | NC |
| 81 | 84 | P35/SCK1/ $\overline{IRQ5}$ | P35/SCK1/ $\overline{IRQ5}$ | P35/SCK1/ $\overline{IRQ5}$ | P35/SCK1/ $\overline{IRQ5}$ | NC |
| 82 | 85 | P36 | P36 | P36 | P36 | NC |
| 83 | 86 | P77/TxD3 | P77/TxD3 | P77/TxD3 | P77/TxD3 | NC |
| 84 | 87 | P76/RxD3 | P76/RxD3 | P76/RxD3 | P76/RxD3 | NC |
| 85 | 88 | P75/SCK3 | P75/SCK3 | P75/SCK3 | P75/SCK3 | NC |
| 86 | 89 | P74/ \overline{MRES} | P74/ \overline{MRES} | P74/ \overline{MRES} | P74/ \overline{MRES} | NC |
| 87 | 90 | P73/TMO1/ CS7 | P73/TMO1/ CS7 | P73/TMO1/ CS7 | P73/TMO1 | NC |

| Pin No. | | Pin Name | | | | |
|---|--------|-----------------------------------|-----------------------------------|-----------------------------------|---------------------------|----|
| TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B FP-100BV | | | | | | |
| FP-100A FP-100AV | Mode 4 | Mode 5 | Mode 6 | Mode 7 | PROM Mode* | |
| 88 | 91 | P72/TMO0/ CS6 | P72/TMO0/ CS6 | P72/TMO0/ CS6 | P72/TMO0 | NC |
| 89 | 92 | P71/CS5 | P71/CS5 | P71/CS5 | P71 | NC |
| 90 | 93 | P70/ TMRi01/ TMCi01/ CS4 | P70/ TMRi01/ TMCi01/ CS4 | P70/ TMRi01/ TMCi01/ CS4 | P70/ TMRi01/ TMCi01 | NC |
| 91 | 94 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | NC |
| 92 | 95 | PG1/CS3/ IRQ7 | PG1/CS3/ IRQ7 | PG1/CS3/ IRQ7 | PG1/IRQ7 | NC |
| 93 | 96 | PG2/CS2 | PG2/CS2 | PG2/CS2 | PG2 | NC |
| 94 | 97 | PG3/CS1 | PG3/CS1 | PG3/CS1 | PG3 | NC |
| 95 | 98 | PG4/CS0 | PG4/CS0 | PG4/CS0 | PG4 | NC |
| 96 | 99 | PE0/D0 | PE0/D0 | PE0/D0 | PE0 | NC |
| 97 | 100 | PE1/D1 | PE1/D1 | PE1/D1 | PE1 | NC |
| 98 | 1 | PE2/D2 | PE2/D2 | PE2/D2 | PE2 | NC |
| 99 | 2 | PE3/D3 | PE3/D3 | PE3/D3 | PE3 | NC |
| 100 | 3 | PE4/D4 | PE4/D4 | PE4/D4 | PE4 | NC |

Note: * The NC should be left open.

Table 1.5 Pin Arrangements in Each Mode of H8S/2227 Group

| Pin No. | | Pin Name | | | | |
|------------------------|------------------------|----------|---------|---------|--------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B ^{*1} | FP-100A ^{*2} | | | | | |
| FP-100BV ^{*1} | FP-100AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*3} |
| 1 | 4 | PE5/D5 | PE5/D5 | PE5/D5 | PE5 | \overline{OE} |
| 2 | 5 | PE6/D6 | PE6/D6 | PE6/D6 | PE6 | \overline{WE} |
| 3 | 6 | PE7/D7 | PE7/D7 | PE7/D7 | PE7 | \overline{CE} |
| 4 | 7 | D8 | D8 | D8 | PD0 | D0 |
| 5 | 8 | D9 | D9 | D9 | PD1 | D1 |
| 6 | 9 | D10 | D10 | D10 | PD2 | D2 |
| 7 | 10 | D11 | D11 | D11 | PD3 | D3 |
| 8 | 11 | D12 | D12 | D12 | PD4 | D4 |
| 9 | 12 | D13 | D13 | D13 | PD5 | D5 |
| 10 | 13 | D14 | D14 | D14 | PD6 | D6 |
| 11 | 14 | D15 | D15 | D15 | PD7 | D7 |
| 12 | 15 | VCC | VCC | VCC | VCC | VCC |
| 13 | 16 | A0 | A0 | PC0/A0 | PC0 | A0 |
| 14 | 17 | VSS | VSS | VSS | VSS | VSS |
| 15 | 18 | A1 | A1 | PC1/A1 | PC1 | A1 |
| 16 | 19 | A2 | A2 | PC2/A2 | PC2 | A2 |
| 17 | 20 | A3 | A3 | PC3/A3 | PC3 | A3 |
| 18 | 21 | A4 | A4 | PC4/A4 | PC4 | A4 |
| 19 | 22 | A5 | A5 | PC5/A5 | PC5 | A5 |
| 20 | 23 | A6 | A6 | PC6/A6 | PC6 | A6 |
| 21 | 24 | A7 | A7 | PC7/A7 | PC7 | A7 |
| 22 | 25 | PB0/A8 | PB0/A8 | PB0/A8 | PB0 | A8 |
| 23 | 26 | PB1/A9 | PB1/A9 | PB1/A9 | PB1 | A9 |
| 24 | 27 | PB2/A10 | PB2/A10 | PB2/A10 | PB2 | A10 |
| 25 | 28 | PB3/A11 | PB3/A11 | PB3/A11 | PB3 | A11 |

| Pin No. | | Pin Name | | | | |
|------------------------|------------------------|---|---|---|---|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B ^{*1} | FP-100A ^{*2} | | | | | |
| FP-100BV ^{*1} | FP-100AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*3} |
| 26 | 29 | PB4/A12 | PB4/A12 | PB4/A12 | PB4 | A12 |
| 27 | 30 | PB5/A13 | PB5/A13 | PB5/A13 | PB5 | A13 |
| 28 | 31 | PB6/A14 | PB6/A14 | PB6/A14 | PB6 | A14 |
| 29 | 32 | PB7/A15 | PB7/A15 | PB7/A15 | PB7 | A15 |
| 30 | 33 | PA0/A16 | PA0/A16 | PA0/A16 | PA0 | A16 |
| 31 | 34 | PA1/A17 | PA1/A17 | PA1/A17 | PA1 | A17 |
| 32 | 35 | PA2/A18 | PA2/A18 | PA2/A18 | PA2 | A18 |
| 33 | 36 | PA3/A19 | PA3/A19 | PA3/A19 | PA3 | NC |
| 34 | 37 | P10/ TIOCA0/ A20 | P10/ TIOCA0/ A20 | P10/ TIOCA0/ A20 | P10/ TIOCA0 | NC |
| 35 | 38 | P11/ TIOCB0/ A21 | P11/ TIOCB0/ A21 | P11/ TIOCB0/ A21 | P11/ TIOCB0 | NC |
| 36 | 39 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA/A22 | P12/ TIOCC0/ TCLKA | NC |
| 37 | 40 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB/A23 | P13/ TIOCD0/ TCLKB | NC |
| 38 | 41 | P14/ TIOCA1/ $\overline{\text{IRQ0}}$ | P14/ TIOCA1/ $\overline{\text{IRQ0}}$ | P14/ TIOCA1/ $\overline{\text{IRQ0}}$ | P14/ TIOCA1/ $\overline{\text{IRQ0}}$ | VSS |
| 39 | 42 | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | P15/ TIOCB1/ TCLKC | NC |
| 40 | 43 | P16/ TIOCA2/ $\overline{\text{IRQ1}}$ | P16/ TIOCA2/ $\overline{\text{IRQ1}}$ | P16/ TIOCA2/ $\overline{\text{IRQ1}}$ | P16/ TIOCA2/ $\overline{\text{IRQ1}}$ | VSS |
| 41 | 44 | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | P17/ TIOCB2/ TCLKD | NC |
| 42 | 45 | AVSS | AVSS | AVSS | AVSS | VSS |

| Pin No. | | Pin Name | | | | |
|------------------------|------------------------|-------------|-------------|-------------|-------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B ^{*1} | FP-100A ^{*2} | | | | | Flash Memory Programmable Mode ^{*3} |
| FP-100BV ^{*1} | FP-100AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 43 | 46 | P97 | P97 | P97 | P97 | NC |
| 44 | 47 | P96 | P96 | P96 | P96 | NC |
| 45 | 48 | P47/AN7 | P47/AN7 | P47/AN7 | P47/AN7 | NC |
| 46 | 49 | P46/AN6 | P46/AN6 | P46/AN6 | P46/AN6 | NC |
| 47 | 50 | P45/AN5 | P45/AN5 | P45/AN5 | P45/AN5 | NC |
| 48 | 51 | P44/AN4 | P44/AN4 | P44/AN4 | P44/AN4 | NC |
| 49 | 52 | P43/AN3 | P43/AN3 | P43/AN3 | P43/AN3 | NC |
| 50 | 53 | P42/AN2 | P42/AN2 | P42/AN2 | P42/AN2 | NC |
| 51 | 54 | P41/AN1 | P41/AN1 | P41/AN1 | P41/AN1 | NC |
| 52 | 55 | P40/AN0 | P40/AN0 | P40/AN0 | P40/AN0 | NC |
| 53 | 56 | Vref | Vref | Vref | Vref | VCC |
| 54 | 57 | AVCC | AVCC | AVCC | AVCC | VCC |
| 55 | 58 | MD0 | MD0 | MD0 | MD0 | VSS |
| 56 | 59 | MD1 | MD1 | MD1 | MD1 | VSS |
| 57 | 60 | OSC2 | OSC2 | OSC2 | OSC2 | NC |
| 58 | 61 | OSC1 | OSC1 | OSC1 | OSC1 | VSS |
| 59 | 62 | RES | RES | RES | RES | RES |
| 60 | 63 | NMI | NMI | NMI | NMI | VCC |
| 61 | 64 | STBY | STBY | STBY | STBY | VCC |
| 62 | 65 | VCC | VCC | VCC | VCC | VCC |
| 63 | 66 | XTAL | XTAL | XTAL | XTAL | XTAL |
| 64 | 67 | VSS | VSS | VSS | VSS | VSS |
| 65 | 68 | EXTAL | EXTAL | EXTAL | EXTAL | EXTAL |
| 66 | 69 | FWE | FWE | FWE | FWE | FWE |
| 67 | 70 | MD2 | MD2 | MD2 | MD2 | VSS |
| 68 | 71 | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | PF7/ ϕ | NC |
| 69 | 72 | AS | AS | AS | PF6 | NC |
| 70 | 73 | RD | RD | RD | PF5 | NC |

| Pin No. | | Pin Name | | | | |
|------------------------|------------------------|-----------------------------------|-----------------------------------|-----------------------------------|---------------------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B ^{*1} | FP-100A ^{*2} | | | | | |
| FP-100BV ^{*1} | FP-100AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | Flash Memory Programmable Mode ^{*3} |
| 71 | 74 | HWR | HWR | HWR | PF4 | NC |
| 72 | 75 | PF3/LWR/ ADTRG/ IRQ3 | PF3/LWR/ ADTRG/ IRQ3 | PF3/LWR/ ADTRG/ IRQ3 | PF3/ ADTRG/ IRQ3 | VCC |
| 73 | 76 | PF2/WAIT | PF2/WAIT | PF2/WAIT | PF2 | NC |
| 74 | 77 | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BACK/ BUZZ | PF1/BUZZ | NC |
| 75 | 78 | PF0/BREQ/ IRQ2 | PF0/BREQ/ IRQ2 | PF0/BREQ/ IRQ2 | PF0/IRQ2 | VCC |
| 76 | 79 | P30/TxD0 | P30/TxD0 | P30/TxD0 | P30/TxD0 | NC |
| 77 | 80 | P31/RxD0 | P31/RxD0 | P31/RxD0 | P31/RxD0 | NC |
| 78 | 81 | P32/SCK0/ IRQ4 | P32/SCK0/ IRQ4 | P32/SCK0/ IRQ4 | P32/SCK0/ IRQ4 | NC |
| 79 | 82 | P33/TxD1 | P33/TxD1 | P33/TxD1 | P33/TxD1 | NC |
| 80 | 83 | P34/RxD1 | P34/RxD1 | P34/RxD1 | P34/RxD1 | NC |
| 81 | 84 | P35/SCK1/ IRQ5 | P35/SCK1/ IRQ5 | P35/SCK1/ IRQ5 | P35/SCK1/ IRQ5 | NC |
| 82 | 85 | P36 | P36 | P36 | P36 | NC |
| 83 | 86 | P77/TxD3 | P77/TxD3 | P77/TxD3 | P77/TxD3 | NC |
| 84 | 87 | P76/RxD3 | P76/RxD3 | P76/RxD3 | P76/RxD3 | NC |
| 85 | 88 | P75/SCK3 | P75/SCK3 | P75/SCK3 | P75/SCK3 | NC |
| 86 | 89 | P74/MRES | P74/MRES | P74/MRES | P74/MRES | NC |
| 87 | 90 | P73/TMO1/ CS7 | P73/TMO1/ CS7 | P73/TMO1/ CS7 | P73/TMO1 | NC |
| 88 | 91 | P72/TMO0/ CS6 | P72/TMO0/ CS6 | P72/TMO0/ CS6 | P72/TMO0 | NC |
| 89 | 92 | P71/CS5 | P71/CS5 | P71/CS5 | P71 | NC |
| 90 | 93 | P70/ TMRI01/ TMCI01/ CS4 | P70/ TMRI01/ TMCI01/ CS4 | P70/ TMRI01/ TMCI01/ CS4 | P70/ TMRI01/ TMCI01 | NC |
| 91 | 94 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | PG0/IRQ6 | NC |

| Pin No. | | Pin Name | | | | |
|------------------------|------------------------|--|--|--|-------------------------------|--|
| TFP-100B | | | | | | |
| TFP-100BV | | | | | | |
| TFP-100G | | | | | | |
| TFP-100GV | | | | | | |
| FP-100B ^{*1} | FP-100A ^{*2} | | | | | Flash Memory Programmable Mode ^{*3} |
| FP-100BV ^{*1} | FP-100AV ^{*2} | Mode 4 | Mode 5 | Mode 6 | Mode 7 | |
| 92 | 95 | PG1/ $\overline{\text{CS3}}$ / $\overline{\text{IRQ7}}$ | PG1/ $\overline{\text{CS3}}$ / $\overline{\text{IRQ7}}$ | PG1/ $\overline{\text{CS3}}$ / $\overline{\text{IRQ7}}$ | PG1/ $\overline{\text{IRQ7}}$ | NC |
| 93 | 96 | PG2/ $\overline{\text{CS2}}$ | PG2/ $\overline{\text{CS2}}$ | PG2/ $\overline{\text{CS2}}$ | PG2 | NC |
| 94 | 97 | PG3/ $\overline{\text{CS1}}$ | PG3/ $\overline{\text{CS1}}$ | PG3/ $\overline{\text{CS1}}$ | PG3 | NC |
| 95 | 98 | PG4/ $\overline{\text{CS0}}$ | PG4/ $\overline{\text{CS0}}$ | PG4/ $\overline{\text{CS0}}$ | PG4 | NC |
| 96 | 99 | PE0/D0 | PE0/D0 | PE0/D0 | PE0 | NC |
| 97 | 100 | PE1/D1 | PE1/D1 | PE1/D1 | PE1 | NC |
| 98 | 1 | PE2/D2 | PE2/D2 | PE2/D2 | PE2 | NC |
| 99 | 2 | PE3/D3 | PE3/D3 | PE3/D3 | PE3 | VCC |
| 100 | 3 | PE4/D4 | PE4/D4 | PE4/D4 | PE4 | VSS |

Notes: 1. Supported only by masked ROM version.

2. Supported only by the HD6432227.

3. The NC should be left open.

1.3.3 Pin Functions

Table 1.6 lists the pin functions of the H8S/2258 Group. Table 1.7 lists the pin functions of the H8S/2239 Group and H8S/2238 Group. Table 1.8 lists the pin functions of the H8S/2237 Group and H8S/2227 Group.

Table 1.6 Pin Functions of H8S/2258 Group

| Type | Symbol | Pin No. | | I/O | Function |
|--------------|--------|--|---------------------|-------|--|
| | | TFP-100B TFP-100BV FP-100B FP-100BV | FP-100A FP-100AV | | |
| Power supply | VCC | 62 | 65 | Input | For connection to the power supply. Connect all V_{cc} pins to the system power supply. |
| | CVCC | 12 | 15 | Input | Connect a 0.1- μ F stabilization capacitance between this pin and ground. Permanent damage on the chip may result if the absolute maximum rating of CV_{cc} 4.3 V is exceeded. Must not connect the 5 V external power supply to this pin. See section 25, Power Supply Circuit, for connection examples. |
| | VSS | 14 64 | 17 67 | Input | For connection to the power supply (0 V). Connect all VSS pins to the system power supply (0 V). |
| Clock | XTAL | 63 | 66 | Input | For connection to a crystal resonator. For examples of crystal resonator connection and external clock input, see section 23, Clock Pulse Generator. |
| | EXTAL | 65 | 68 | Input | For connection to a crystal resonator. This pin can be also used for external clock input. For examples of crystal resonator connection and external clock input, see section 23, Clock Pulse Generator. |
| | OSC1 | 58 | 61 | Input | Connects to a 32.768 kHz crystal resonator. See section 23, Clock Pulse Generator, for typical connection diagrams for a crystal resonator. |

| Type | Symbol | Pin No. | | I/O | Function |
|------------------------|----------------------------|--|---------------------|--------|--|
| | | TFP-100B TFP-100BV FP-100B FP-100BV | FP-100A FP-100AV | | |
| Clock | OSC2 | 57 | 60 | Input | Connects to a 32.768 kHz crystal resonator. See section 23, Clock Pulse Generator, for typical connection diagrams for a crystal resonator. |
| | ϕ | 68 | 71 | Output | Supplies the system clock to external devices. |
| Operating mode control | MD2 | 67 | 70 | Input | Sets the operating mode. Inputs at these pins should not be changed during operation. Except for mode changing, be sure to fix the levels of the mode pins (MD2 to MD0) by pulling them down or pulling them up until the power turns off. |
| | MD1 | 56 | 59 | | |
| | MD0 | 55 | 58 | | |
| System control | $\overline{\text{RES}}^*$ | 59 | 62 | Input | Reset input pin. When this pin is low, the chip enters the power-on reset state. |
| | $\overline{\text{MRES}}$ | 86 | 89 | Input | When this pin is low, the chip enters the manual reset state. |
| | $\overline{\text{STBY}}^*$ | 61 | 64 | Input | When this pin is low, a transition is made to hardware standby mode. |
| | $\overline{\text{BREQ}}$ | 75 | 78 | Input | Used by an external bus master to request the bus mastership to this LSI. |
| | $\overline{\text{BACK}}$ | 74 | 77 | Output | Indicates that the bus mastership has been granted to an external bus master. |
| | FWE | 66 | 69 | Input | Enables/disables programming the flash memory. |
| Interrupts | NMI^* | 60 | 63 | Input | Nonmaskable interrupt pin. If this pin is not used, it should be fixed high. |
| | $\overline{\text{IRQ7}}$ | 92 | 95 | Input | These pins request a maskable interrupt. |
| | $\overline{\text{IRQ6}}$ | 91 | 94 | | |
| | $\overline{\text{IRQ5}}$ | 81 | 84 | | |
| | $\overline{\text{IRQ4}}$ | 78 | 81 | | |
| | $\overline{\text{IRQ3}}$ | 72 | 75 | | |
| | $\overline{\text{IRQ2}}$ | 75 | 78 | | |
| | $\overline{\text{IRQ1}}$ | 40 | 43 | | |
| | $\overline{\text{IRQ0}}$ | 38 | 41 | | |
| Address bus | A23 to A0 | 37 to 15, 13 | 40 to 18, 16 | Output | Outputs Address. |

| | | Pin No. | | | |
|-------------------------------|-------------------|----------------------------------|---------------------|------------------|---|
| | | TFP-100B TFP-100BV FP-100B | FP-100A FP-100AV | | |
| Type | Symbol | FP-100BV | FP-100AV | I/O | Function |
| Data bus | D15 to D0 | 100 to 96, 11 to 1 | 100, 99, 14 to 1 | Input/ output | Used as the bidirectional data bus. |
| Bus control | $\overline{CS7}$ | 87 | 90 | Output | Select signals for areas 7 to 0. |
| | $\overline{CS6}$ | 88 | 91 | | |
| | $\overline{CS5}$ | 89 | 92 | | |
| | $\overline{CS4}$ | 90 | 93 | | |
| | $\overline{CS3}$ | 92 | 95 | | |
| | $\overline{CS2}$ | 93 | 96 | | |
| | $\overline{CS1}$ | 94 | 97 | | |
| | $\overline{CS0}$ | 95 | 98 | | |
| | \overline{AS} | 69 | 72 | Output | When this pin is low, it indicates valid address output on the address bus. |
| | \overline{RD} | 70 | 73 | Output | When this pin is low, it indicates that the external address space is being read. |
| | \overline{HWR} | 71 | 74 | Output | Strobe signal: Writes to the external address bus to indicate valid data on the upper data bus (D15 to D8). |
| | \overline{LWR} | 72 | 75 | Output | Strobe signal: Writes to the external bus to indicate valid data on the lower data bus (D7 to D0). |
| | \overline{WAIT} | 73 | 76 | Input | Requests insertion of wait states in bus cycle when accesses to the external three-state address. |
| 16-bit timer-pulse unit (TPU) | TCLKD | 41 | 44 | Input | These pins input an external clock. |
| | TCLKC | 39 | 42 | | |
| | TCLKB | 37 | 40 | | |
| | TCLKA | 36 | 39 | | |
| | TIOCA0 | 34 | 37 | Input/ Output | Pins for the TGRA_0 to TGRD_0 input capture input, output compare output, or PWM output. |
| | TIOCB0 | 35 | 38 | | |
| | TIOCC0 | 36 | 39 | | |
| | TIOCD0 | 37 | 40 | | |
| | TIOCA1 | 38 | 41 | Input/ Output | Pins for the TGRA_1 and TGRB_1 input capture input, output compare output, or PWM output. |
| | TIOCB1 | 39 | 42 | | |
| | TIOCA2 | 40 | 43 | Input/ Output | Pins for the TGRA_2 and TGRB_2 input capture input, output compare output, or PWM output. |
| | TIOCB2 | 41 | 44 | | |

| | | Pin No. | | | |
|---|--------------|----------------------------------|----------|------------------|--|
| | | TFP-100B TFP-100BV FP-100B | FP-100A | | |
| Type | Symbol | FP-100BV | FP-100AV | I/O | Function |
| 16-bit timer- pulse unit (TPU) | TIOCA3 | 22 | 25 | Input/ Output | Pins for the TGRA_3 to TGRD_3 input capture input, output compare output, or PWM output. |
| | TIOCB3 | 23 | 26 | | |
| | TIOCC3 | 24 | 27 | | |
| | TIOCD3 | 25 | 28 | | |
| | TIOCA4 | 26 | 29 | Input/ Output | Pins for the TGRA_4 and TGRB_4 input capture input, output compare output, or PWM output. |
| | TIOCB4 | 27 | 30 | | |
| | TIOCA5 | 28 | 31 | Input/ Output | Pins for the TGRA_5 and TGRB_5 input capture input, output compare output, or PWM output. |
| | TIOCB5 | 29 | 32 | | |
| | | | | | |
| | | | | | |
| 8-bit timer | TMO3 to TMO0 | 88 to 85 | 91 to 88 | Output | Compare-match output pins. |
| | TMCI23 | 89 | 92 | Input | Pins for external clock input to the counter. |
| | TMCI01 | 90 | 93 | | |
| | TMRI23 | 89 | 92 | Input | Counter reset input pins. |
| | TMRI01 | 90 | 93 | | |
| Watchdog timer (WDT) | BUZZ | 74 | 77 | Output | This pin outputs the pulse that is divided by watchdog timer. |
| Serial communi- cation interface (SCI)/ smart card interface | TxD3 | 83 | 86 | Output | Data output pins. |
| | TxD2 | 31 | 34 | | |
| | TxD1 | 79 | 82 | | |
| | TxD0 | 76 | 79 | | |
| | RxD3 | 84 | 87 | Input | Data input pins. |
| | RxD2 | 32 | 35 | | |
| | RxD1 | 80 | 83 | | |
| | RxD0 | 77 | 80 | | |
| | SCK3 | 85 | 88 | Input/ Output | Clock input/output pins. SCK1 outputs NMOS push/pull. |
| | SCK2 | 33 | 36 | | |
| | SCK1 | 81 | 84 | | |
| | SCK0 | 78 | 81 | | |
| I ² C bus interface (IIC) (optional) | SCL1 | 79 | 82 | Input/ Output | I ² C clock input/output pins. These pins drive bus. The output of SCL0 is NMOS open drain. |
| | SCL0 | 81 | 84 | | |
| | | | | | |
| | SDA1 | 78 | 81 | Input/ Output | I ² C data input/output pins. These pins drive bus. The output of SDA0 is NMOS open drain. |
| | SDA0 | 80 | 83 | | |

| Type | Symbol | Pin No. | | I/O | Function |
|---------------------------------------|------------------------|--|---------------------|------------------|--|
| | | TFP-100B TFP-100BV FP-100B FP-100BV | FP-100A FP-100AV | | |
| IEBus controller (IEB) | $\overline{\text{Tx}}$ | 93 | 96 | Output | IEB transmit data output pin. |
| | $\overline{\text{Rx}}$ | 94 | 97 | Input | IEB receive data input pin |
| A/D converter | AN7 to AN0 | 52 to 45 | 55 to 48 | Input | Analog input pins for the A/D converter. |
| | ADTRG | 72 | 75 | Input | Pin for input of an external trigger to start A/D conversion. |
| D/A converter | DA1 | 43 | 46 | Output | Analog output pins for the D/A converter. |
| | DA0 | 44 | 47 | | |
| A/D converter, D/A converter | AVCC | 54 | 57 | Input | Power supply pin for the A/D converter and D/A converter. If none of the A/D converter and D/A converter is used, connect this pin to the system power supply (+5 V). |
| | AVSS | 42 | 45 | Input | Ground pin for the A/D converter and D/A converter. Connect this pin to the system power supply (0 V). |
| | Vref | 53 | 56 | Input | Reference voltage input pin for the A/D converter and D/A converter. If neither the A/D converter nor D/A converter is used, connect this pin to the system power supply (+5 V). |
| I/O ports | P17 to P10 | 41 to 34 | 44 to 37 | Input/ Output | 8-bit I/O pins. |
| | P36 to P30 | 82 to 76 | 85 to 79 | Input/ Output | 7-bit I/O pins. P34 and P35 output NMOS push/pull. |
| | P47 to P40 | 52 to 45 | 55 to 48 | Input | 8-bit input pins. |
| | P77 to P70 | 90 to 83 | 93 to 86 | Input/ Output | 8-bit I/O pins. |
| | P97 P96 | 43 44 | 46 47 | Input | 2-bit input pins. |
| | PA3 to PA0 | 33 to 30 | 36 to 33 | Input/ Output | 4-bit I/O pins. |
| | PB7 to PB0 | 29 to 22 | 32 to 25 | Input/ Output | 8-bit I/O pins. |

| | | Pin No. | | | |
|-----------|------------|----------------------------------|---------------------|------------------|-----------------|
| | | TFP-100B TFP-100BV FP-100B | FP-100A FP-100AV | | |
| Type | Symbol | FP-100BV | FP-100AV | I/O | Function |
| I/O ports | PC7 to PC0 | 21 to 15, 13 | 24 to 18, 16 | Input/ Output | 8-bit I/O pins. |
| | PD7 to PD0 | 11 to 4 | 14 to 7 | Input/ Output | 8-bit I/O pins. |
| | PE7 to PE0 | 100 to 96, 3 to 1 | 100, 99, 6 to 1 | Input/ Output | 8-bit I/O pins. |
| | PF7 to PF0 | 75 to 68 | 78 to 71 | Input/ Output | 8-bit I/O pins. |
| | PG4 to PG0 | 95 to 91 | 98 to 94 | Input/ Output | 5-bit I/O pins. |

Note: * Measures should be taken to deal with noise, which can cause operation errors otherwise.

Table 1.7 Pin Functions of H8S/2239 Group and H8S/2238 Group

| | | Pin No. | | | | |
|--------------|--------|-----------|------------------------|-------------------------|-------|---|
| Type | Symbol | TFP-100B | | | | Function |
| | | TFP-100BV | | | | |
| | | TFP-100G | | BP-112 ^{*1} | | |
| | | TFP-100GV | | BP-112V ^{*1} | | |
| | | FP-100B | FP-100A ^{*3} | TBP-112A ^{*4} | | |
| | | FP-100BV | FP-100AV ^{*3} | TBP-112AV ^{*4} | I/O | |
| Power supply | VCC | 62 | 65 | F9, G10 | Input | For connection to the power supply. Connect all V _{cc} pins to the system power supply. |
| | CVCC | 12 | 15 | E2, F3 | Input | With a 5-V external power supply (H8S/2238B used), connect a 0.1-μF stabilization capacitance between this pin and ground. Permanent damage on the chip may result if the absolute maximum rating of CV _{cc} 4.3 V is exceeded. Must not connect the 5 V external power supply to this pin. With a 3-V external power supply (H8S/2239, H8S/2238R, and H8S/2236R used), connect this pin to the system power supply. See section 25, Power Supply Circuit, for connection examples. |
| | VSS | 14 64 | 17 67 | F3, F2 F10, F8 | Input | For connection to the power supply (0 V). Connect all VSS pins to the system power supply (0 V). |
| Clock | XTAL | 63 | 66 | F11 | Input | For connection to a crystal resonator. For examples of crystal resonator connection and external clock input, see section 23, Clock Pulse Generator. |
| | EXTAL | 65 | 68 | E11 | Input | For connection to a crystal resonator. This pin can be also used for external clock input. For examples of crystal resonator connection and external clock input, see section 23, Clock Pulse Generator. |
| | OSC1 | 58 | 61 | H11 | Input | Connects to a 32.768 kHz crystal resonator. See section 23, Clock Pulse Generator, for typical connection diagrams for a crystal resonator. |

| | | Pin No. | | | | Function |
|------------------------|--------------------------------------|---|---|--|--------|--|
| Type | Symbol | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B FP-100BV | FP-100A ^{*3} FP-100AV ^{*3} | BP-112 ^{*1} BP-112V ^{*1} TBP-112A ^{*4} TBP-112AV ^{*4} | I/O | |
| Clock | OSC2 | 57 | 60 | H10 | Input | Connects to a 32.768 kHz crystal resonator. See section 23, Clock Pulse Generator, for typical connection diagrams for a crystal resonator. |
| | ϕ | 68 | 71 | D11 | Output | Supplies the system clock to external devices. |
| Operating mode control | MD2 | 67 | 70 | E9 | Input | Sets the operating mode. Inputs at these pins should not be changed during operation. Except for mode changing, be sure to fix the levels of the mode pins (MD2 to MD0) by pulling them down or pulling them up until the power turns off. |
| | MD1 | 56 | 59 | H9 | | |
| | MD0 | 55 | 58 | J11 | | |
| System control | $\overline{\text{RES}}^{\text{*5}}$ | 59 | 62 | G8 | Input | Reset input pin. When this pin is low, the chip enters the power-on reset state. |
| | $\overline{\text{MRES}}$ | 86 | 89 | B7 | Input | When this pin is low, the chip enters the manual reset state. |
| | $\overline{\text{STBY}}^{\text{*5}}$ | 61 | 64 | G11 | Input | When this pin is low, a transition is made to hardware standby mode. |
| | $\overline{\text{BREQ}}$ | 75 | 78 | C9 | Input | Used by an external bus master to request the bus mastership to this LSI. |
| | $\overline{\text{BACK}}$ | 74 | 77 | B11 | Output | Indicates that the bus mastership has been granted to an external bus master. |
| | FWE | 66 | 69 | E10 | Input | Enables/disables programming the flash memory. |
| Interrupts | $\text{NMI}^{\text{*5}}$ | 60 | 63 | G9 | Input | Nonmaskable interrupt pin. If this pin is not used, it should be fixed high. |
| | $\overline{\text{IRQ7}}$ | 92 | 95 | B5 | Input | These pins request a maskable interrupt. |
| | $\overline{\text{IRQ6}}$ | 91 | 94 | A5 | | |
| | $\overline{\text{IRQ5}}$ | 81 | 84 | B8 | | |
| | $\overline{\text{IRQ4}}$ | 78 | 81 | B9 | | |
| | $\overline{\text{IRQ3}}$ | 72 | 75 | D9 | | |
| | $\overline{\text{IRQ2}}$ | 75 | 78 | C9 | | |
| | $\overline{\text{IRQ1}}$ | 40 | 43 | K6 | | |
| | $\overline{\text{IRQ0}}$ | 38 | 41 | J6 | | |

| | | Pin No. | | | | |
|-------------|-------------------|--------------------|------------------|--|--------------|---|
| | | TFP-100B | | | | |
| | | TFP-100BV | | | | |
| | | TFP-100G | | | BP-112*1 | |
| | | TFP-100GV | | | BP-112V*1 | |
| | | FP-100B | FP-100A*3 | TBP-112A*4 | | |
| Type | Symbol | FP-100BV | FP-100AV*3 | TBP-112AV*4 | I/O | Function |
| Address bus | A23 to A0 | 37 to 15, 13 | 40 to 18, 16 | L5, L4, L3, L2, K5, K4, K3, K2, K1, J5, J4, J3, J2, J1, H5, H4, H3, H2, H1, G4, G3, G2, G1, F1 | Output | Outputs Address. |
| Data bus | D15 to D0 | 100 to 96, 11 to 1 | 100, 99, 14 to 1 | E4, E3, E1, D4, D3, D2, D1, C4, C2, C1, B4, B3, B2, B1, A3, A2 | Input/output | Used as the bidirectional data bus. |
| Bus control | $\overline{CS7}$ | 87 | 90 | C6 | Output | Select signals for areas 7 to 0. |
| | $\overline{CS6}$ | 88 | 91 | A6 | | |
| | $\overline{CS5}$ | 89 | 92 | B6 | | |
| | $\overline{CS4}$ | 90 | 93 | D6 | | |
| | $\overline{CS3}$ | 92 | 95 | B5 | | |
| | $\overline{CS2}$ | 93 | 96 | C5 | | |
| | $\overline{CS1}$ | 94 | 97 | A4 | | |
| | $\overline{CS0}$ | 95 | 98 | D5 | | |
| | \overline{AS} | 69 | 72 | E8 | Output | When this pin is low, it indicates valid address output on the address bus. |
| | \overline{RD} | 70 | 73 | D10 | Output | When this pin is low, it indicates that the external address space is being read. |
| | \overline{HWR} | 71 | 74 | C11 | Output | Strobe signal: Writes to the external address bus to indicate valid data on the upper data bus (D15 to D8). |
| | \overline{LWR} | 72 | 75 | D9 | Output | Strobe signal: Writes to the external bus to indicate valid data on the lower data bus (D7 to D0). |
| | \overline{WAIT} | 73 | 76 | C10 | Input | Requests insertion of wait states in bus cycle when accesses to the external three-state address. |

| | | Pin No. | | | | | |
|-------------------------------------|--------|-----------|------------------------|-------------------------|--------|---|--|
| Type | Symbol | TFP-100B | | | | | |
| | | TFP-100BV | | | | | |
| | | TFP-100G | | BP-112 ^{*1} | | | |
| | | TFP-100GV | | BP-112V ^{*1} | | | |
| | | FP-100B | FP-100A ^{*3} | TBP-112A ^{*4} | | | |
| | | FP-100BV | FP-100AV ^{*3} | TBP-112AV ^{*4} | I/O | Function | |
| DMA controller (DMAC) ^{*2} | DREQ1 | 89 | — | B6 | Input | Request DMAC activation. | |
| | DREQ0 | 90 | | D6 | | (Supported only by the H8S/2239 Group.) | |
| | TEND1 | 87 | — | C6 | Output | Indicate that the DMAC has ended transmitting data. | |
| | TEND0 | 88 | | A6 | | (Supported only by the H8S/2239 Group.) | |
| | DACK1 | 35 | — | J5 | Output | These pins function as single address transmitting acknowledge of DMAC. | |
| | DACK0 | 34 | | H5 | | (Supported only by the H8S/2239 Group.) | |
| 16-bit timer-pulse unit (TPU) | TCLKD | 41 | 44 | H6 | Input | These pins input an external clock. | |
| | TCLKC | 39 | 42 | L6 | | | |
| | TCLKB | 37 | 40 | K5 | | | |
| | TCLKA | 36 | 39 | L5 | | | |
| | TIOCA0 | 34 | 37 | H5 | Input/ | Pins for the TGRA_0 to TGRD_0 input capture input, output compare output, or PWM output. | |
| | TIOCB0 | 35 | 38 | J5 | Output | | |
| | TIOCC0 | 36 | 39 | L5 | | | |
| | TIOCD0 | 37 | 40 | K5 | | | |
| | TIOCA1 | 38 | 41 | J6 | Input/ | Pins for the TGRA_1 and TGRB_1 input capture input, output compare output, or PWM output. | |
| | TIOCB1 | 39 | 42 | L6 | Output | | |
| | TIOCA2 | 40 | 43 | K6 | Input/ | Pins for the TGRA_2 and TGRB_2 input capture input, output compare output, or PWM output. | |
| | TIOCB2 | 41 | 44 | H6 | Output | | |
| | TIOCA3 | 22 | 25 | H3 | Input/ | Pins for the TGRA_3 to TGRD_3 input capture input, output compare output, or PWM output. | |
| | TIOCB3 | 23 | 26 | J2 | Output | | |
| | TIOCC3 | 24 | 27 | K1 | | | |
| | TIOCD3 | 25 | 28 | J3 | | | |
| | TIOCA4 | 26 | 29 | K2 | Input/ | Pins for the TGRA_4 and TGRB_4 input capture input, output compare output, or PWM output. | |
| | TIOCB4 | 27 | 30 | L2 | Output | | |
| | TIOCA5 | 28 | 31 | H4 | Input/ | Pins for the TGRA_5 and TGRB_5 input capture input, output compare output, or PWM output. | |
| | TIOCB5 | 29 | 32 | K3 | Output | | |

| Type | Symbol | Pin No. | | | I/O | Function |
|---|--------------|--|----------|--|--------------|--|
| | | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B FP-100A ^{*3} FP-100BV FP-100AV ^{*3} | | BP-112 ^{*1} BP-112V ^{*1} TBP-112A ^{*4} TBP-112AV ^{*4} | | |
| 8-bit timer | TMO3 to TMO0 | 88 to 85 | 91 to 88 | A7, A6, B7, C6 | Output | Compare-match output pins. |
| | TMCI23 | 89 | 92 | B6 | Input | Pins for external clock input to the counter. |
| | TMCI01 | 90 | 93 | D6 | | |
| | TMRI23 | 89 | 92 | B6 | Input | Counter reset input pins. |
| | TMRI01 | 90 | 93 | D6 | | |
| Watchdog timer (WDT) | BUZZ | 74 | 77 | B11 | Output | This pin outputs the pulse that is divided by watchdog timer. |
| Serial communication interface (SCI)/smart card interface | TxD3 | 83 | 86 | D7 | Output | Data output pins. |
| | TxD2 | 31 | 34 | J4 | | |
| | TxD1 | 79 | 82 | A9 | | |
| | TxD0 | 76 | 79 | A10 | | |
| | RxD3 | 84 | 87 | C7 | Input | Data input pins. |
| | RxD2 | 32 | 35 | K4 | | |
| | RxD1 | 80 | 83 | C8 | | |
| | RxD0 | 77 | 80 | D8 | | |
| | SCK3 | 85 | 88 | A7 | Input/Output | Clock input/output pins. SCK1 outputs NMOS push/pull. |
| | SCK2 | 33 | 36 | L4 | | |
| | SCK1 | 81 | 84 | B8 | | |
| | SCK0 | 78 | 81 | B9 | | |
| I ² C bus interface (IIC) (optional) | SCL1 | 79 | 82 | A9 | Input/Output | I ² C clock input/output pins. These pins drive bus. The output of SCL0 is NMOS open drain. |
| | SCL0 | 81 | 84 | B8 | | |
| | SDA1 | 78 | 81 | B9 | Input/Output | I ² C data input/output pins. These pins drive bus. The output of SDA0 is NMOS open drain. |
| | SDA0 | 80 | 83 | C8 | | |
| A/D converter | AN7 to AN0 | 52 to 45 | 55 to 48 | L10, L9, K11, K10, K9, K8, J8, H7 | Input | Analog input pins for the A/D converter. |
| | ADTRG | 72 | 75 | D9 | Input | Pin for input of an external trigger to start A/D conversion. |

| | | Pin No. | | | | Function |
|---------------------------------|------------|---|---|--|--------------|--|
| Type | Symbol | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B FP-100BV | FP-100A ^{*3} FP-100AV ^{*3} | BP-112 ^{*1} BP-112V ^{*1} TBP-112A ^{*4} TBP-112AV ^{*4} | I/O | |
| D/A converter | DA1 | 43 | 46 | J7 | Output | Analog output pins for the D/A converter. |
| | DA0 | 44 | 47 | L8 | | |
| A/D converter, D/A converter | AVCC | 54 | 57 | J10 | Input | Power supply pin for the A/D converter and D/A converter. If none of the A/D converter and D/A converter is used, connect this pin to the system power supply (+3 V). |
| | AVSS | 42 | 45 | K7, L7 | Input | Ground pin for the A/D converter and D/A converter. Connect this pin to the system power supply (0 V). |
| | Vref | 53 | 56 | H8 | Input | Reference voltage input pin for the A/D converter and D/A converter. If neither the A/D converter nor D/A converter is used, connect this pin to the system power supply (+3 V). |
| I/O ports | P17 to P10 | 41 to 34 | 44 to 37 | L6, L5, K6, K5, J6, J5, H6, H5 | Input/Output | 8-bit I/O pins. |
| | P36 to P30 | 82 to 76 | 85 to 79 | D8, C8, B9, B8, A10, A9, A8 | Input/Output | 7-bit I/O pins. P34 and P35 output NMOS push/pull. |
| | P47 to P40 | 52 to 45 | 55 to 48 | L10, L9, K11, K10, K9, K8, H7, J8 | Input | 8-bit input pins. |
| | P77 to P70 | 90 to 83 | 93 to 86 | D7, D6, C7, C6, B7, B6, A7, A6 | Input/Output | 8-bit I/O pins. |
| | P97 P96 | 43 44 | 46 47 | J7 L8 | Input | 2-bit input pins. |
| | PA3 to PA0 | 33 to 30 | 36 to 33 | L4, L3, K3, J4 | Input/Output | 4-bit I/O pins. |
| | PB7 to PB0 | 29 to 22 | 32 to 25 | L2, K3, K2, K1, J3, J2, H4, H3 | Input/Output | 8-bit I/O pins. |

| | | Pin No. | | | | |
|-----------|------------|----------------------|------------------|-------------------------------------|---------------|-----------------|
| | | TFP-100B | | | | |
| | | TFP-100BV | | | | |
| | | TFP-100G | | BP-112*1 | | |
| | | TFP-100GV | | BP-112V*1 | | |
| | | FP-100B | FP-100A*3 | TBP-112A*4 | | |
| Type | Symbol | FP-100BV | FP-100AV*3 | TBP-112AV*4 | I/O | Function |
| I/O ports | PC7 to PC0 | 21 to 15, 1324 to 16 | 18, 16 | J1, H2, H1, G4, G3, G2, G1, F1 | Input/ Output | 8-bit I/O pins. |
| | PD7 to PD0 | 11 to 4 | 14 to 7 | E4, E3, E1, D3, D2, D1, C2, C1 | Input/ Output | 8-bit I/O pins. |
| | PE7 to PE0 | 100 to 96, to 1 | 3100, 99, 6 to 1 | D4, C4, B4, B3, B2, B1, A3, A2 | Input/ Output | 8-bit I/O pins. |
| | PF7 to PF0 | 75 to 68 | 78 to 71 | E8, D11, D10, D9, C11, C10, C9, B11 | Input/ Output | 8-bit I/O pins. |
| | PG4 to PG0 | 95 to 91 | 98 to 94 | D5, C5, B5, A5, A4 | Input/ Output | 5-bit I/O pins. |

- Notes:
1. Supported only by the HD64F2238R.
 2. Supported only by the H8S/2239 Group.
 3. Supported only by the H8S/2238B and H8S/2236B.
 4. Supported only by the HD64F2238R and HD64F2239.
 5. Measures should be taken to deal with noise, which can cause operation errors otherwise.

Table 1.8 Pin Functions of H8S/2237 Group and H8S/2227 Group

| Type | Symbol | Pin No. | | I/O | Function |
|------------------------|-------------------------------|---|---|--------|--|
| | | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B ^{*1} FP-100BV ^{*1} | FP-100A ^{*2} FP-100AV ^{*2} | | |
| Power supply | VCC | 12 62 | 15 65 | Input | For connection to the power supply. Connect all VCC pins to the system power supply. |
| | VSS | 14 64 | 17 67 | Input | For connection to the power supply (0 V). Connect all VSS pins to the system power supply (0 V). |
| Clock | XTAL | 63 | 66 | Input | For connection to a crystal resonator. For examples of crystal resonator connection and external clock input, see section 23, Clock Pulse Generator. |
| | EXTAL | 65 | 68 | Input | For connection to a crystal resonator. This pin can be also used for external clock input. For examples of crystal resonator connection and external clock input, see section 23, Clock Pulse Generator. |
| | OSC1 | 58 | 61 | Input | Connects to a 32.768 kHz crystal resonator. See section 23, Clock Pulse Generator, for typical connection diagrams for a crystal resonator. |
| | OSC2 | 57 | 60 | Input | Connects to a 32.768 kHz crystal resonator. See section 23, Clock Pulse Generator, for typical connection diagrams for a crystal resonator. |
| | ϕ | 68 | 71 | Output | Supplies the system clock to external devices. |
| Operating mode control | MD2 | 67 | 70 | Input | Sets the operating mode. Inputs at these pins should not be changed during operation. Except for mode changing, be sure to fix the levels of the mode pins (MD2 to MD0) by pulling them down or pulling them up until the power turns off. |
| | MD1 | 56 | 59 | | |
| | MD0 | 55 | 58 | | |
| System control | $\overline{\text{RES}}^{*3}$ | 59 | 62 | Input | Reset input pin. When this pin is low, the chip enters in the power-on reset state. |
| | $\overline{\text{MRES}}$ | 86 | 89 | Input | When this pin is low, the chip enters in the manual reset state. |
| | $\overline{\text{STBY}}^{*3}$ | 61 | 64 | Input | When this pin is low, a transition is made to hardware standby mode. |

| Type | Symbol | Pin No. | | I/O | Function |
|----------------|-------------------|---|---|--------------|---|
| | | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B ^{*1} FP-100BV ^{*1} | FP-100A ^{*2} FP-100AV ^{*2} | | |
| System control | BREQ | 75 | 78 | Input | Used by an external bus master to request the bus mastership to this LSI. |
| | BACK | 74 | 77 | Output | Indicates that the bus mastership has been granted to an external bus master. |
| | FEW | 66 | 69 | Input | Enables/disables programming the flash memory. |
| Interrupts | NMI ^{*3} | 60 | 63 | Input | Nonmaskable interrupt pin. If this pin is not used, it should be fixed-high. |
| | IRQ7 | 92 | 95 | Input | These pins request a maskable interrupt. |
| | IRQ6 | 91 | 94 | | |
| | IRQ5 | 81 | 84 | | |
| | IRQ4 | 78 | 81 | | |
| | IRQ3 | 72 | 75 | | |
| | IRQ2 | 75 | 78 | | |
| | IRQ1 | 40 | 43 | | |
| | IRQ0 | 38 | 41 | | |
| Address bus | A23 to A0 | 37 to 15, 13 | 40 to 18, 16 | Output | Outputs Address. |
| Data bus | D15 to D0 | 100 to 96, 11 to 1 | 100, 99, 14 to 1 | Input/output | Used as the bidirectional data bus. |
| Bus control | CS7 | 87 | 90 | Output | Select signals for areas 7 to 0. |
| | CS6 | 88 | 91 | | |
| | CS5 | 89 | 92 | | |
| | CS4 | 90 | 93 | | |
| | CS3 | 92 | 95 | | |
| | CS2 | 93 | 96 | | |
| | CS1 | 94 | 97 | | |
| | CS0 | 95 | 98 | | |
| | AS | 69 | 72 | Output | When this pin is low, it indicates valid address output on the address bus. |
| | RD | 70 | 73 | Output | When this pin is low, it indicates that the external address space is being read. |
| | HWR | 71 | 74 | Output | Strobe signal: Writes to the external address bus to indicate valid data on the upper data bus (D15 to D8). |

| Type | Symbol | Pin No. | | I/O | Function |
|--------------------------------------|--------|---|---|------------------|---|
| | | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B ^{*1} FP-100BV ^{*1} | FP-100A ^{*2} FP-100AV ^{*2} | | |
| Bus control | LWR | 72 | 75 | Output | Strobe signal: Writes to the external bus to indicate valid data on the lower data bus (D7 to D0). |
| | WAIT | 73 | 76 | Input | Requests insertion of wait states in bus cycle when accesses to the external three state address. |
| 16-bit timer- pulse unit (TPU) | TCLKD | 41 | 44 | Input | These pins input an external clock. |
| | TCLKC | 39 | 42 | | |
| | TCLKB | 37 | 40 | | |
| | TCLKA | 36 | 39 | | |
| | TIOCA0 | 34 | 37 | Input/ Output | Pins for the TGRA_0 to TGRD_0 input capture input, output compare output, or PWM output. |
| | TIOCB0 | 35 | 38 | | |
| | TIOCC0 | 36 | 39 | | |
| | TIOCD0 | 37 | 40 | | |
| | TIOCA1 | 38 | 41 | Input/ Output | Pins for the TGRA_1 and TGRB_1 input capture input, output compare output, or PWM output. |
| | TIOCB1 | 39 | 42 | | |
| | TIOCA2 | 40 | 43 | Input/ Output | Pins for the TGRA_2 and TGRB_2 input capture input, output compare output, or PWM output. |
| | TIOCB2 | 41 | 44 | | |
| | TIOCA3 | 22 | 25 | Input/ Output | Pins for the TGRA_3 to TGRD_3 input capture input, output compare output, or PWM output. (Not available in the H8S/2227 Group.) |
| | TIOCB3 | 23 | 26 | | |
| | TIOCC3 | 24 | 27 | | |
| | TIOCD3 | 25 | 28 | | |
| | TIOCA4 | 26 | 29 | Input/ Output | Pins for the TGRA_4 and TGRB_4 input capture input, output compare output, or PWM output. (Not available in the H8S/2227 Group.) |
| | TIOCB4 | 27 | 30 | | |
| | TIOCA5 | 28 | 31 | Input/ Output | Pins for the TGRA_5 and TGRB_5 input capture input, output compare output, or PWM output. (Not available in the H8S/2227 Group.) |
| | TIOCB5 | 29 | 32 | | |
| 8-bit timer | TMO1 | 87 | 90 | Output | Compare-match output pins. |
| | TMO0 | 88 | 91 | | |
| | TMCI01 | 90 | 93 | Input | Pin for external clock input to the counter. |
| | TMRI01 | 90 | 93 | Input | Counter reset input pin. |
| Watchdog timerBUZZ (WDT) | | 74 | 77 | Output | This pin outputs the pulse that is divided by watchdog timer. |

| Type | Symbol | Pin No. | | I/O | Function |
|--|------------|---|---|------------------|---|
| | | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B ^{*1} FP-100BV ^{*1} | FP-100A ^{*2} FP-100AV ^{*2} | | |
| Serial communication interface (SCI)/ smart card interface | TxD3 | 83 | 86 | Output | Data output pins. (TxD2 is not available in the H8S/2227 Group.) |
| | TxD2 | 31 | 34 | | |
| | TxD1 | 79 | 82 | | |
| | TxD0 | 76 | 79 | | |
| | RxD3 | 84 | 87 | Input | Data input pins. (RxD2 is not available in the H8S/2227 Group.) |
| | RxD2 | 32 | 35 | | |
| | RxD1 | 80 | 83 | | |
| | RxD0 | 77 | 80 | | |
| | SCK3 | 85 | 88 | Input/ Output | Clock input/output pins. (SCK2 is not available in the H8S/2227 Group.) |
| | SCK2 | 33 | 36 | | |
| | SCK1 | 81 | 84 | | |
| | SCK0 | 78 | 81 | | |
| A/D converter | AN7 to AN0 | 52 to 45 | 55 to 48 | Input | Analog input pins for the A/D converter. |
| | ADTRG | 72 | 75 | Input | Pin for input of an external trigger to start A/D conversion. |
| D/A converter | DA1 | 43 | 46 | Output | Analog output pins for the D/A converter. (Not available in the H8S/2227 Group.) |
| | DA0 | 44 | 47 | | |
| A/D converter, D/A converter | AVCC | 54 | 57 | Input | Power supply pin for the A/D converter and D/A converter. If none of the A/D converter and D/A converter is used, connect this pin to the system power supply. |
| | AVSS | 42 | 45 | Input | Ground pin for the A/D converter and D/A converter. Connect this pin to the system power supply (0 V). |
| | Vref | 53 | 56 | Input | Reference voltage input pin for the A/D converter and D/A converter. If neither the A/D converter nor D/A converter is used, connect this pin to the system power supply. |
| I/O ports | P17 to P10 | 41 to 34 | 44 to 37 | Input/ Output | 8-bit I/O pins. |
| | P36 to P30 | 82 to 76 | 85 to 79 | Input/ Output | 7-bit I/O pins. |

| Type | Symbol | Pin No. | | I/O | Function |
|-----------|------------|---|---|------------------|-------------------|
| | | TFP-100B TFP-100BV TFP-100G TFP-100GV FP-100B ^{*1} FP-100BV ^{*1} | FP-100A ^{*2} FP-100AV ^{*2} | | |
| I/O ports | P47 to P40 | 52 to 45 | 55 to 48 | Input | 8-bit input pins. |
| | P77 to P70 | 90 to 83 | 93 to 86 | Input/ Output | 8-bit I/O pins. |
| | P97 P96 | 43 44 | 46 47 | Input | 2-bit input pins. |
| | PA3 to PA0 | 33 to 30 | 36 to 33 | Input/ Output | 4-bit I/O pins. |
| | PB7 to PB0 | 29 to 22 | 32 to 25 | Input/ Output | 8-bit I/O pins. |
| | PC7 to PC0 | 21 to 15, 13 | 24 to 18, 16 | Input/ Output | 8-bit I/O pins. |
| | PD7 to PD0 | 11 to 4 | 14 to 7 | Input/ Output | 8-bit I/O pins. |
| | PE7 to PE0 | 100 to 96, 3 to 1 | 100, 99, 6 to 1 | Input/ Output | 8-bit I/O pins. |
| | PF7 to PF0 | 75 to 68 | 78 to 71 | Input/ Output | 8-bit I/O pins. |
| | PG4 to PG0 | 95 to 91 | 98 to 94 | Input/ Output | 5-bit I/O pins. |

- Notes: 1. In H8S/2227 Group, supported only by masked ROM version.
 2. In H8S/2227 Group, supported only by the HD6432227.
 3. Measures should be taken to deal with noise, which can cause operation errors otherwise.

Section 2 CPU

The H8S/2000 CPU is a high-speed central processing unit with an internal 32-bit architecture that is upward-compatible with the H8/300 and H8/300H CPUs. The H8S/2000 CPU has sixteen 16-bit general registers, can address a 16-Mbyte linear address space, and is ideal for realtime control.

This section describes the H8S/2000 CPU. The usable modes and address spaces differ depending on the product. For details on each product, refer to section 3, MCU Operating Modes.

2.1 Features

- Upward-compatible with H8/300 and H8/300H CPU
 - Can execute H8/300 and H8/300H CPU object programs
- General-register architecture
 - Sixteen 16-bit general registers also usable as sixteen 8-bit registers or eight 32-bit registers
- Sixty-five basic instructions
 - 8/16/32-bit arithmetic and logic instructions
 - Multiply and divide instructions
 - Powerful bit-manipulation instructions
- Eight addressing modes
 - Register direct [Rn]
 - Register indirect [@ERn]
 - Register indirect with displacement [@(d:16,ERn) or @(d:32,ERn)]
 - Register indirect with post-increment or pre-decrement [@ERn+ or @-ERn]
 - Absolute address [@aa:8, @aa:16, @aa:24, or @aa:32]
 - Immediate [#xx:8, #xx:16, or #xx:32]
 - Program-counter relative [@(d:8,PC) or @(d:16,PC)]
 - Memory indirect [@@aa:8]
- 16-Mbyte address space
 - Program: 16 Mbytes
 - Data: 16 Mbytes

- High-speed operation
 - All frequently-used instructions execute in one or two states
 - 8/16/32-bit register-register add/subtract : 1 state
 - 8×8 -bit register-register multiply : 12 states
 - $16 \div 8$ -bit register-register divide : 12 states
 - 16×16 -bit register-register multiply : 20 states
 - $32 \div 16$ -bit register-register divide : 20 states
- Two CPU operating modes
 - Normal mode*
 - Advanced mode
- Power-down state
 - Transition to power-down state by a SLEEP instruction
 - CPU clock speed selection

Note: * Normal mode is not available in this LSI.

2.1.1 Differences between H8S/2600 CPU and H8S/2000 CPU

The differences between the H8S/2600 CPU and the H8S/2000 CPU are shown below.

- Register configuration
 - The MAC register is supported by the H8S/2600 CPU only.
- Basic instructions
 - The four instructions MAC, CLRMAC, LDMAC, and STMAC are supported by the H8S/2600 CPU only.
- The number of execution states of the MULXU and MULXS instructions;

| Instruction | Mnemonic | Execution States | |
|-------------|-----------------|------------------|----------|
| | | H8S/2600 | H8S/2000 |
| MULXU | MULXU.B Rs, Rd | 3 | 12 |
| | MULXU.W Rs, ERd | 4 | 20 |
| MULXS | MULXS.B Rs, Rd | 4 | 13 |
| | MULXS.W Rs, ERd | 5 | 21 |

In addition, there are differences in address space, CCR and EXR register functions, and power-down modes, etc., depending on the model.

2.1.2 Differences from H8/300 CPU

In comparison to the H8/300 CPU, the H8S/2000 CPU has the following enhancements:

- More general registers and control registers
 - Eight 16-bit expanded registers, and one 8-bit and two 32-bit control registers, have been added.
- Expanded address space
 - Normal mode supports the same 64-kbyte address space as the H8/300 CPU.
 - Advanced mode supports a maximum 16-Mbyte address space.
- Enhanced addressing
 - The addressing modes have been enhanced to make effective use of the 16-Mbyte address space.
- Enhanced instructions
 - Addressing modes of bit-manipulation instructions have been enhanced.
 - Signed multiply and divide instructions have been added.
 - Two-bit shift instructions have been added.
 - Instructions for saving and restoring multiple registers have been added.
 - A test and set instruction has been added.
- Higher speed
 - Basic instructions execute twice as fast.

2.1.3 Differences from H8/300H CPU

In comparison to the H8/300H CPU, the H8S/2000 CPU has the following enhancements:

- Additional control register
 - One 8-bit control registers have been added.
- Enhanced instructions
 - Addressing modes of bit-manipulation instructions have been enhanced.
 - Two-bit shift instructions have been added.
 - Instructions for saving and restoring multiple registers have been added.
 - A test and set instruction has been added.
- Higher speed
 - Basic instructions execute twice as fast.

2.2 CPU Operating Modes

The H8S/2000 CPU has two operating modes: normal and advanced. Normal mode supports a maximum 64-kbyte address space. Advanced mode supports a maximum 16-Mbyte total address space. The mode is selected by the mode pins.

2.2.1 Normal Mode

In normal mode, the exception vector table and stack have the same structure as the H8/300 CPU.

- Address Space

Linear access is provided to a maximum address space of 64 kbytes.

- Extended Registers (En)

The extended registers (E0 to E7) can be used as 16-bit registers, or as the upper 16-bit segments of 32-bit registers. When En is used as a 16-bit register it can contain any value, even when the corresponding general register (Rn) is used as an address register. If the general register is referenced in the register indirect addressing mode with pre-decrement (@-Rn) or post-increment (@Rn+) and a carry or borrow occurs, however, the value in the corresponding extended register (En) will be affected.

- Instruction Set

All instructions and addressing modes can be used. Only the lower 16 bits of effective addresses (EA) are valid.

- Exception Vector Table and Memory Indirect Branch Addresses

In normal mode the top area starting at H'0000 is allocated to the exception vector table. One branch address is stored per 16 bits. Figure 2.1 shows the structure of the exception vector table in normal mode. For details of the exception vector table, see section 4, Exception Handling.

The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instructions uses an 8-bit absolute address included in the instruction code to specify a memory operand that contains a branch address. In normal mode the operand is a 16-bit word operand, providing a 16-bit branch address. Branch addresses can be stored in the top area from H'0000 to H'00FF. Note that this area is also used for the exception vector table.

- Stack Structure

In normal mode, when the program counter (PC) is pushed onto the stack in a subroutine call, and the PC, condition-code register (CCR) and extended control register (EXR) are pushed onto the stack in exception handling, they are stored as shown in figure 2.2. EXR is not pushed onto the stack in interrupt control mode 0. For details, see section 4, Exception Handling.

Note: Normal mode is not available in this LSI.

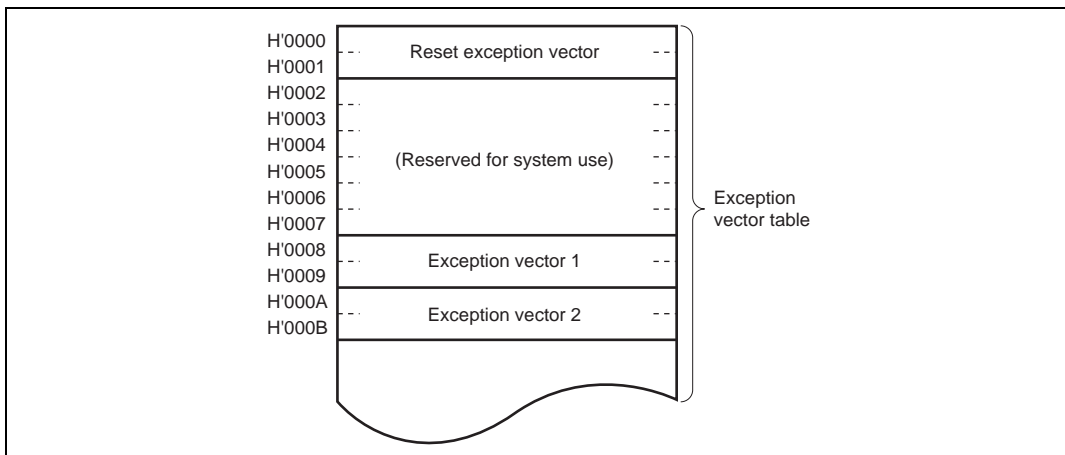


Figure 2.1 Exception Vector Table (Normal Mode)

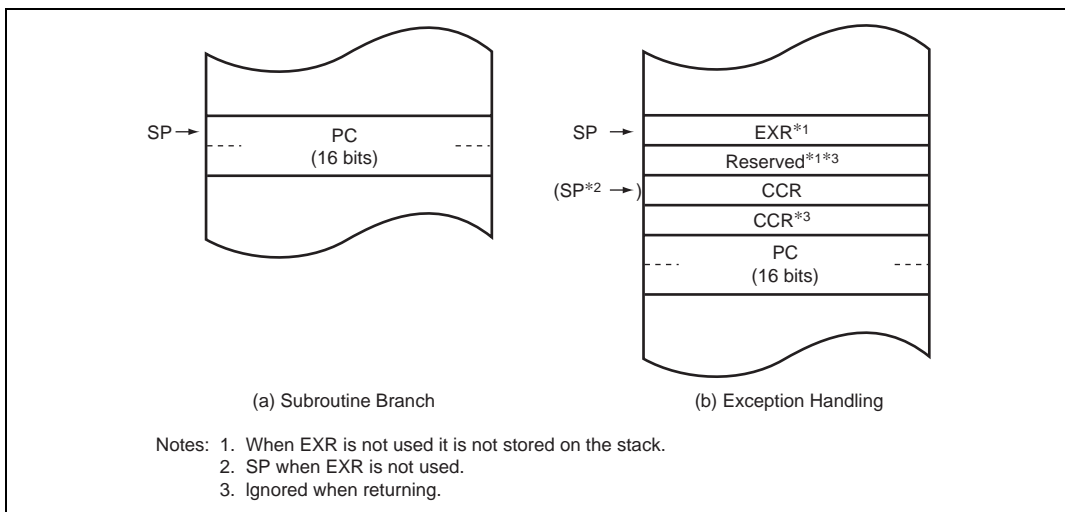


Figure 2.2 Stack Structure in Normal Mode

2.2.2 Advanced Mode

- Address Space

Linear access is provided to a maximum 16-Mbyte address space.

- Extended Registers (En)

The extended registers (E0 to E7) can be used as 16-bit registers, or as the upper 16-bit segments of 32-bit registers or address registers.

- Instruction Set

All instructions and addressing modes can be used.

- Exception Vector Table and Memory Indirect Branch Addresses

In advanced mode, the top area starting at H'00000000 is allocated to the exception vector table in units of 32 bits. In each 32 bits, the upper 8 bits are ignored and a branch address is stored in the lower 24 bits (figure 2.3). For details of the exception vector table, see section 4, Exception Handling.

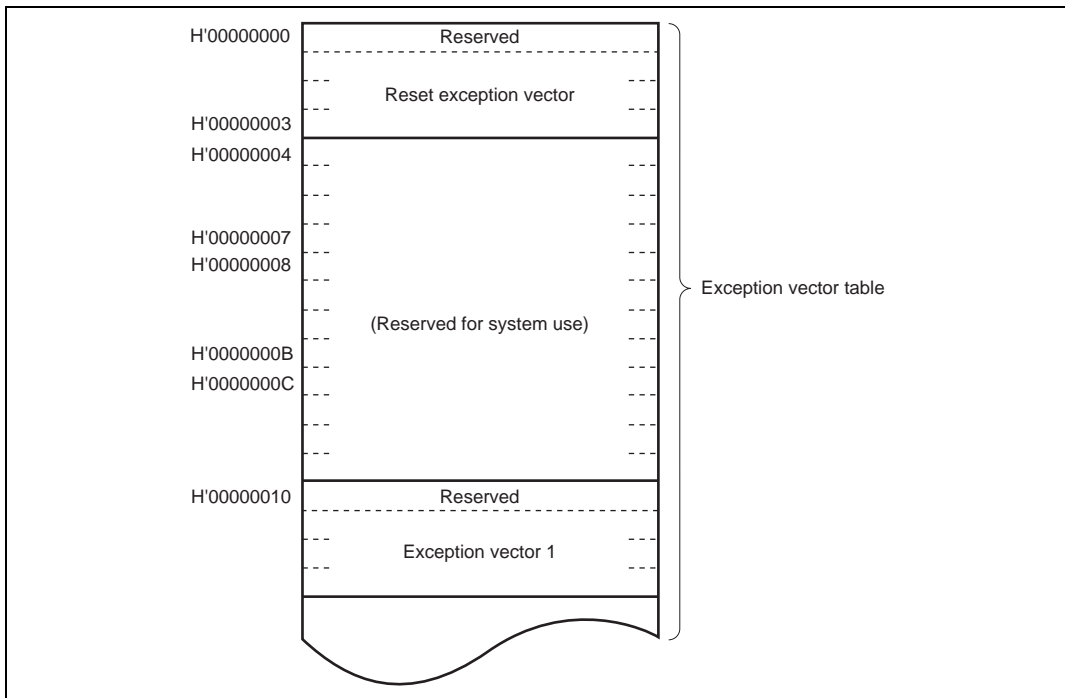


Figure 2.3 Exception Vector Table (Advanced Mode)

The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instructions uses an 8-bit absolute address included in the instruction code to specify a memory operand that contains a branch address. In advanced mode, the operand is a 32-bit longword operand, providing a 32-bit branch address. The upper 8 bits of these 32 bits is a reserved area that is regarded as H'00. Branch addresses can be stored in the area from H'00000000 to H'000000FF. Note that the first part of this range is also the exception vector table.

- Stack Structure

In advanced mode, when the program counter (PC) is pushed onto the stack in a subroutine call, and the PC, condition-code register (CCR), and extended control register (EXR) are pushed onto the stack in exception handling, they are stored as shown in figure 2.4. When EXR is invalid, it is not pushed onto the stack. For details, see section 4, Exception Handling.

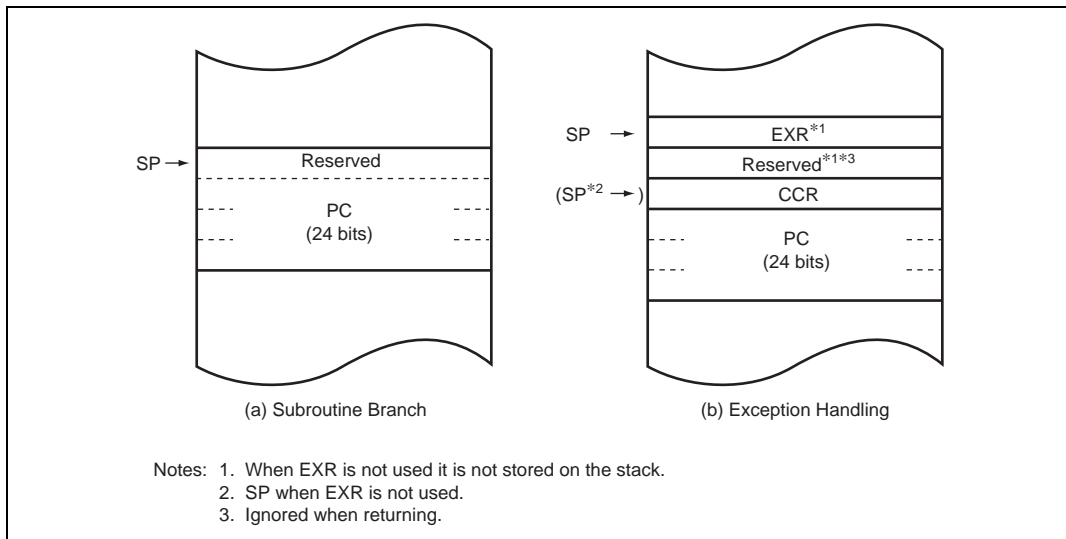


Figure 2.4 Stack Structure in Advanced Mode

2.3 Address Space

Figure 2.5 shows a memory map of the H8S/2000 CPU. The H8S/2000 CPU provides linear access to a maximum 64-kbyte address space in normal mode, and a maximum 16-Mbyte (architecturally 4-Gbyte) address space in advanced mode. The usable modes and address spaces differ depending on the product. For details on each product, refer to section 3, MCU Operating Modes.

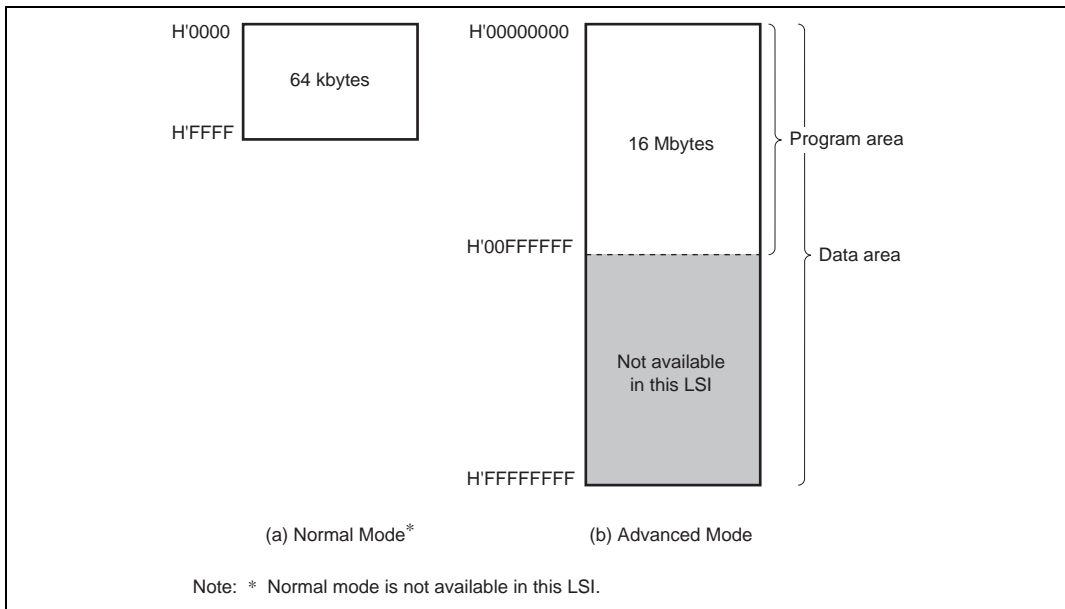


Figure 2.5 Memory Map

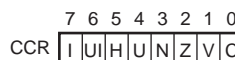
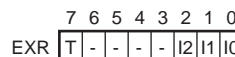
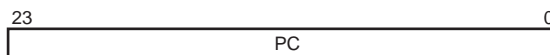
2.4 Register Configuration

The H8S/2000 CPU has the internal registers shown in figure 2.6. There are two types of registers: general registers and control registers. Control registers are a 24-bit program counter (PC), an 8-bit extended control register (EXR), and an 8-bit condition code register (CCR).

General Registers (Rn) and Extended Registers (En)

| | 15 | 0 7 | 0 7 | 0 |
|----------|----|-----|-----|---|
| ER0 | E0 | R0H | R0L | |
| ER1 | E1 | R1H | R1L | |
| ER2 | E2 | R2H | R2L | |
| ER3 | E3 | R3H | R3L | |
| ER4 | E4 | R4H | R4L | |
| ER5 | E5 | R5H | R5L | |
| ER6 | E6 | R6H | R6L | |
| ER7 (SP) | E7 | R7H | R7L | |

Control Registers (CR)



Legend:

SP: Stack pointer
PC: Program counter
EXR: Extended control register

T: Trace bit
I2 to I0: Interrupt mask bits
CCR: Condition-code register
I: Interrupt mask bit
UI: User bit or interrupt mask bit*

H: Half-carry flag
U: User bit
N: Negative flag
Z: Zero flag
V: Overflow flag
C: Carry flag

Note: * The interrupt mask bit is not available in this LSI.

Figure 2.6 CPU Registers

2.4.1 General Registers

The H8S/2000 CPU has eight 32-bit general registers. These general registers are all functionally alike and can be used as both address registers and data registers. When a general register is used as a data register, it can be accessed as a 32-bit, 16-bit, or 8-bit register. Figure 2.7 illustrates the usage of the general registers.

When the general registers are used as 32-bit registers or address registers, they are designated by the letters ER (ER0 to ER7).

The ER registers divide into 16-bit general registers designated by the letters E (E0 to E7) and R (R0 to R7). These registers are functionally equivalent, providing a maximum sixteen 16-bit registers. The E registers (E0 to E7) are also referred to as extended registers.

The R registers divide into 8-bit general registers designated by the letters RH (R0H to R7H) and RL (R0L to R7L). These registers are functionally equivalent, providing a maximum sixteen 8-bit registers.

The usage of each register can be selected independently.

General register ER7 has the function of stack pointer (SP) in addition to its general-register function, and is used implicitly in exception handling and subroutine calls. Figure 2.8 shows the stack.

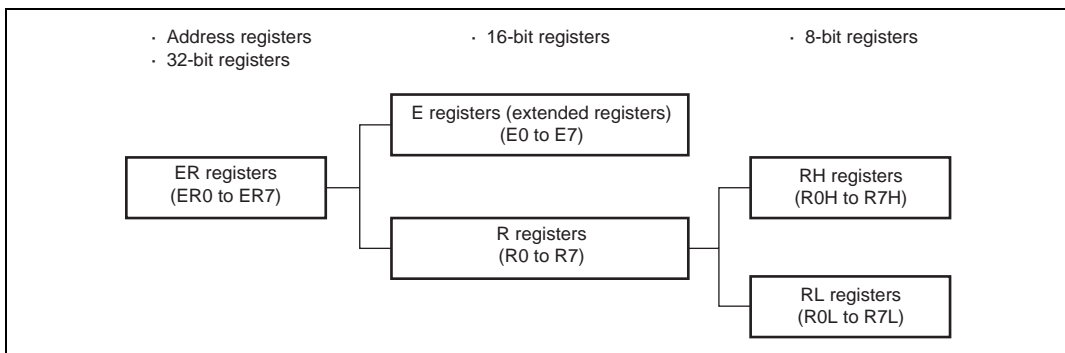


Figure 2.7 Usage of General Registers

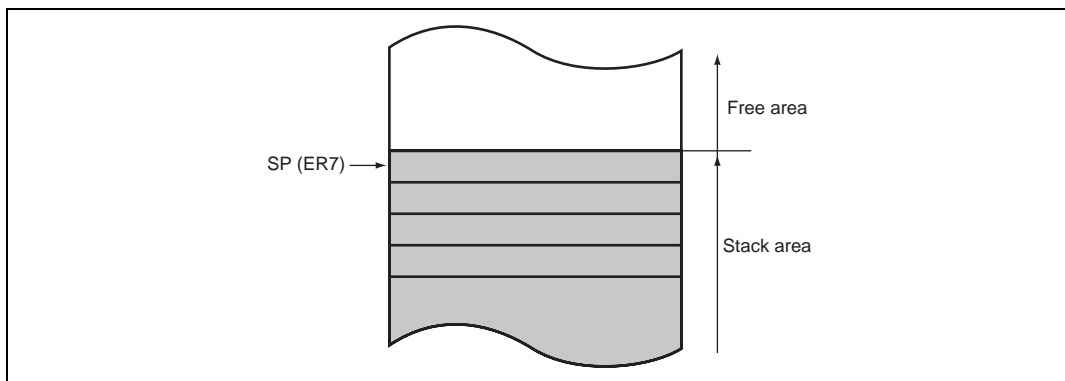


Figure 2.8 Stack Status

2.4.2 Program Counter (PC)

This 24-bit counter indicates the address of the next instruction the CPU will execute. The length of all CPU instructions is 2 bytes (one word), so the least significant PC bit is ignored (When an instruction is fetched, the least significant PC bit is regarded as 0).

2.4.3 Extended Control Register (EXR)

EXR is an 8-bit register that manipulates the LDC, STC, ANDC, ORC, and XORC instructions. When these instructions except for the STC instruction is executed, all interrupts including NMI will be masked for three states after execution is completed.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | T | 0 | R/W | Trace Bit When this bit is set to 1, a trace exception is generated each time an instruction is executed. When this bit is cleared to 0, instructions are executed in sequence. |
| 6 to 3 | — | All 1 | — | Reserved These bits are always read as 1. |
| 2 | I2 | 1 | R/W | These bits designate the interrupt mask level (0 to 7). For details, refer to section 5, Interrupt Controller. |
| 1 | I1 | 1 | R/W | |
| 0 | I0 | 1 | R/W | |

2.4.4 Condition-Code Register (CCR)

This 8-bit register contains internal CPU status information, including an interrupt mask bit (I) and half-carry (H), negative (N), zero (Z), overflow (V), and carry (C) flags.

Operations can be performed on the CCR bits by the LDC, STC, ANDC, ORC, and XORC instructions. The N, Z, V, and C flags are used as branching conditions for conditional branch (Bcc) instructions.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | I | 1 | R/W | Interrupt Mask Bit Masks interrupts other than NMI when set to 1. NMI is accepted regardless of the I bit setting. The I bit is set to 1 by hardware at the start of an exception-handling sequence. For details, refer to section 5, Interrupt Controller. |
| 6 | UI | undefined | R/W | User Bit or Interrupt Mask Bit Can be written and read by software using the LDC, STC, ANDC, ORC, and XORC instructions. This bit cannot be used as an interrupt mask bit in this LSI. |
| 5 | H | undefined | R/W | Half-Carry Flag When the ADD.B, ADDX.B, SUB.B, SUBX.B, CMP.B, or NEG.B instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and cleared to 0 otherwise. When the ADD.W, SUB.W, CMP.W, or NEG.W instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 11, and cleared to 0 otherwise. When the ADD.L, SUB.L, CMP.L, or NEG.L instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 27, and cleared to 0 otherwise. |
| 4 | U | undefined | R/W | User Bit Can be written and read by software using the LDC, STC, ANDC, ORC, and XORC instructions. |
| 3 | N | undefined | R/W | Negative Flag Stores the value of the most significant bit of data as a sign bit. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | Z | undefined | R/W | Zero Flag Set to 1 to indicate zero data, and cleared to 0 to indicate non-zero data. |
| 1 | V | undefined | R/W | Overflow Flag Set to 1 when an arithmetic overflow occurs, and cleared to 0 at other times. |
| 0 | C | undefined | R/W | Carry Flag Set to 1 when a carry occurs, and cleared to 0 otherwise. Used by: <ul style="list-style-type: none"> • Add instructions, to indicate a carry • Subtract instructions, to indicate a borrow • Shift and rotate instructions, to indicate a carry The carry flag is also used as a bit accumulator by bit manipulation instructions. |

2.4.5 Initial Values of CPU Registers

Reset exception handling loads the CPU's program counter (PC) from the vector table, clears the trace bit in EXR to 0, and sets the interrupt mask bits in CCR and EXR to 1. The other CCR bits and the general registers are not initialized. In particular, the stack pointer (ER7) is not initialized. The stack pointer should therefore be initialized by an MOV.L instruction executed immediately after a reset.

2.5 Data Formats

The H8S/2000 CPU can process 1-bit, 4-bit (BCD), 8-bit (byte), 16-bit (word), and 32-bit (longword) data. Bit-manipulation instructions operate on 1-bit data by accessing bit n (n = 0, 1, 2,..., 7) of byte operand data. The DAA and DAS decimal-adjust instructions treat byte data as two digits of 4-bit BCD data.

2.5.1 General Register Data Formats

Figure 2.9 shows the data formats in general registers.

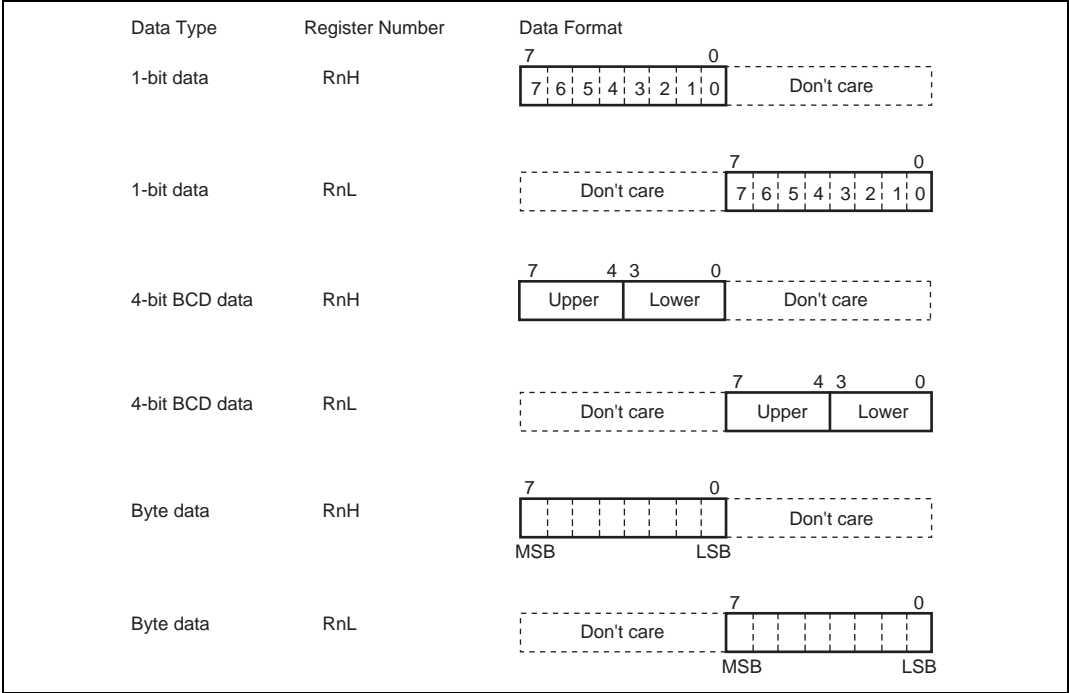


Figure 2.9 General Register Data Formats (1)

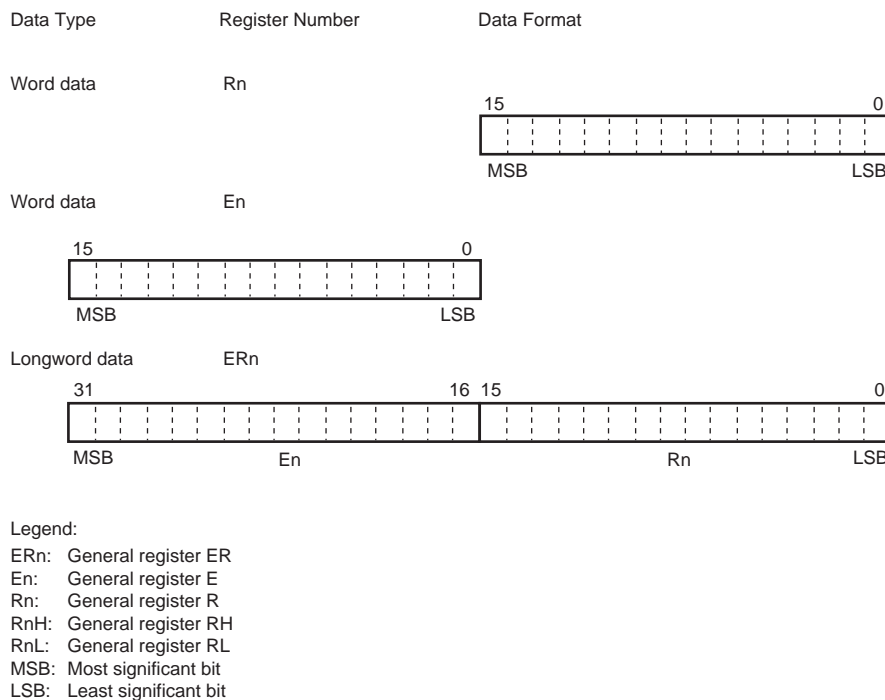


Figure 2.9 General Register Data Formats (2)

2.5.2 Memory Data Formats

Figure 2.10 shows the data formats in memory. The H8S/2000 CPU can access word data and longword data in memory, but word or longword data must begin at an even address. If an attempt is made to access word or longword data at an odd address, no address error occurs but the least significant bit of the address is regarded as 0, so the access starts at the preceding address. This also applies to instruction fetches.

When ER7 is used as an address register to access the stack, the operand size should be word or longword.

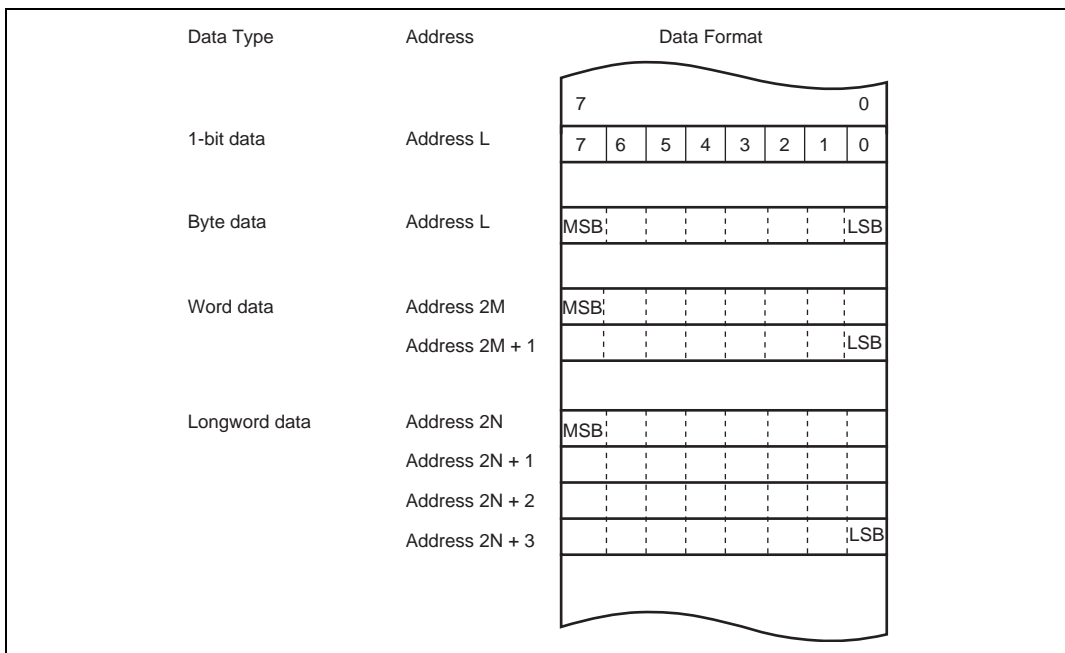


Figure 2.10 Memory Data Formats

2.6 Instruction Set

The H8S/2000 CPU has 65 types of instructions. The instructions are classified by function in table 2.1.

Table 2.1 Instruction Classification

| Function | Instructions | Size | Types |
|-----------------------|---|-------|-------|
| Data transfer | MOV | B/W/L | 5 |
| | POP ^{*1} , PUSH ^{*1} | W/L | |
| | LDM ^{*5} , STM ^{*5} | L | |
| | MOVFP ^{*3} , MOVTP ^{*3} | B | |
| Arithmetic operations | ADD, SUB, CMP, NEG | B/W/L | 19 |
| | ADDX, SUBX, DAA, DAS | B | |
| | INC, DEC | B/W/L | |
| | ADDS, SUBS | L | |
| | MULXU, DIVXU, MULXS, DIVXS | B/W | |
| | EXTU, EXTS | W/L | |
| | TAS ^{*4} | B | |
| Logic operations | AND, OR, XOR, NOT | B/W/L | 4 |
| Shift | SHAL, SHAR, SHLL, SHLR, ROTL, ROTR, ROTXL, ROTXR | B/W/L | 8 |
| Bit manipulation | BSET, BCLR, BNOT, BTST, BLD, BILD, BST, BIST, BAND, BIAND, BOR, BIOR, BXOR, BIXOR | B | 14 |
| Branch | Bcc ^{*2} , JMP, BSR, JSR, RTS | — | 5 |
| System control | TRAPA, RTE, SLEEP, LDC, STC, ANDC, ORC, XORC, NOP | — | 9 |
| Block data transfer | EEPMOV | — | 1 |

Total: 65

Legend: B: Byte

W: Word

L: Longword

- Notes: 1. POP.W Rn and PUSH.W Rn are identical to MOV.W @SP+, Rn and MOV.W Rn, @-SP. POP.L ERn and PUSH.L ERn are identical to MOV.L @SP+, ERn and MOV.L ERn, @-SP.
2. Bcc is the general name for conditional branch instructions.
3. Cannot be used in this LSI.
4. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS instruction.
5. Only register ER0 to ER6 should be used when using the STM/LDM instruction.

2.6.1 Table of Instructions Classified by Function

Tables 2.3 to 2.10 summarize the instructions in each functional category. The notation used in tables 2.3 to 2.10 is defined below.

Table 2.2 Operation Notation

| Symbol | Description |
|----------------|------------------------------------|
| Rd | General register (destination)* |
| Rs | General register (source)* |
| Rn | General register* |
| ERn | General register (32-bit register) |
| (EAd) | Destination operand |
| (EAs) | Source operand |
| EXR | Extended control register |
| CCR | Condition-code register |
| N | N (negative) flag in CCR |
| Z | Z (zero) flag in CCR |
| V | V (overflow) flag in CCR |
| C | C (carry) flag in CCR |
| PC | Program counter |
| SP | Stack pointer |
| #IMM | Immediate data |
| disp | Displacement |
| + | Addition |
| − | Subtraction |
| × | Multiplication |
| ÷ | Division |
| ^ | Logical AND |
| ∨ | Logical OR |
| ⊕ | Logical XOR |
| → | Move |
| ¬ | NOT (logical complement) |
| :8/:16/:24/:32 | 8-, 16-, 24-, or 32-bit length |

Note: * General registers include 8-bit registers (R0H to R7H, R0L to R7L), 16-bit registers (R0 to R7, E0 to E7), and 32-bit registers (ER0 to ER7).

Table 2.3 Data Transfer Instructions

| Instruction | Size^{*1} | Function |
|--------------------|--------------------------|---|
| MOV | B/W/L | (EAs) → Rd, Rs → (EAd) Moves data between two general registers or between a general register and memory, or moves immediate data to a general register. |
| MOVFPPE | B | Cannot be used in this LSI. |
| MOVTPE | B | Cannot be used in this LSI. |
| POP | W/L | @SP+ → Rn Pops a general register from the stack. POP.W Rn is identical to MOV.W @SP+, Rn. POP.L ERn is identical to MOV.L @SP+, ERn. |
| PUSH | W/L | Rn → @-SP Pushes a general register onto the stack. PUSH.W Rn is identical to MOV.W Rn, @-SP. PUSH.L ERn is identical to MOV.L ERn, @-SP. |
| LDM ^{*2} | L | @SP+ → Rn (register list) Pops two or more general registers from the stack. |
| STM ^{*2} | L | Rn (register list) → @-SP Pushes two or more general registers onto the stack. |

Notes: 1. Refers to the operand size.

B: Byte

W: Word

L: Longword

2. Only register ER0 to ER6 should be used when using the STM/LDM instruction.

Table 2.4 Arithmetic Operations Instructions

| Instruction | Size^{*1} | Function |
|--------------------|--------------------------|---|
| ADD SUB | B/W/L | $Rd \pm Rs \rightarrow Rd$, $Rd \pm \#IMM \rightarrow Rd$ Performs addition or subtraction on data in two general registers, or on immediate data and data in a general register. (Immediate byte data cannot be subtracted from byte data in a general register. Use the SUBX or ADD instruction.) |
| ADDX SUBX | B | $Rd \pm Rs \pm C \rightarrow Rd$, $Rd \pm \#IMM \pm C \rightarrow Rd$ Performs addition or subtraction with carry on byte data in two general registers, or on immediate data and data in a general register. |
| INC DEC | B/W/L | $Rd \pm 1 \rightarrow Rd$, $Rd \pm 2 \rightarrow Rd$ Increments or decrements a general register by 1 or 2. (Byte operands can be incremented or decremented by 1 only.) |
| ADDS SUBS | L | $Rd \pm 1 \rightarrow Rd$, $Rd \pm 2 \rightarrow Rd$, $Rd \pm 4 \rightarrow Rd$ Adds or subtracts the value 1, 2, or 4 to or from data in a 32-bit register. |
| DAA DAS | B | Rd decimal adjust $\rightarrow Rd$ Decimal-adjusts an addition or subtraction result in a general register by referring to the CCR to produce 4-bit BCD data. |
| MULXU | B/W | $Rd \times Rs \rightarrow Rd$ Performs unsigned multiplication on data in two general registers: either 8 bits \times 8 bits \rightarrow 16 bits or 16 bits \times 16 bits \rightarrow 32 bits. |
| MULXS | B/W | $Rd \times Rs \rightarrow Rd$ Performs signed multiplication on data in two general registers: either 8 bits \times 8 bits \rightarrow 16 bits or 16 bits \times 16 bits \rightarrow 32 bits. |
| DIVXU | B/W | $Rd \div Rs \rightarrow Rd$ Performs unsigned division on data in two general registers: either 16 bits \div 8 bits \rightarrow 8-bit quotient and 8-bit remainder or 32 bits \div 16 bits \rightarrow 16-bit quotient and 16-bit remainder. |

| Instruction | Size ^{*1} | Function |
|-------------------|--------------------|---|
| DIVXS | B/W | $Rd \div Rs \rightarrow Rd$ Performs signed division on data in two general registers: either 16 bits \div 8 bits \rightarrow 8-bit quotient and 8-bit remainder or 32 bits \div 16 bits \rightarrow 16-bit quotient and 16-bit remainder. |
| CMP | B/W/L | $Rd - Rs, Rd - \#IMM$ Compares data in a general register with data in another general register or with immediate data, and sets CCR bits according to the result. |
| NEG | B/W/L | $0 - Rd \rightarrow Rd$ Takes the two's complement (arithmetic complement) of data in a general register. |
| EXTU | W/L | $Rd \text{ (zero extension)} \rightarrow Rd$ Extends the lower 8 bits of a 16-bit register to word size, or the lower 16 bits of a 32-bit register to longword size, by padding with zeros on the left. |
| EXTS | W/L | $Rd \text{ (sign extension)} \rightarrow Rd$ Extends the lower 8 bits of a 16-bit register to word size, or the lower 16 bits of a 32-bit register to longword size, by extending the sign bit. |
| TAS ^{*2} | B | $@ERd - 0, 1 \rightarrow (<\text{bit } 7> \text{ of } @ERd)$ Tests memory contents, and sets the most significant bit (bit 7) to 1. |

Notes: 1. Refers to the operand size.

B: Byte

W: Word

L: Longword

2. Only register ER0, ER1, ER4, or ER5 should be used when using the TAS instruction.

Table 2.5 Logic Operations Instructions

| Instruction | Size* | Function |
|--------------------|--------------|---|
| AND | B/W/L | $Rd \wedge Rs \rightarrow Rd$, $Rd \wedge \#IMM \rightarrow Rd$ Performs a logical AND operation on a general register and another general register or immediate data. |
| OR | B/W/L | $Rd \vee Rs \rightarrow Rd$, $Rd \vee \#IMM \rightarrow Rd$ Performs a logical OR operation on a general register and another general register or immediate data. |
| XOR | B/W/L | $Rd \oplus Rs \rightarrow Rd$, $Rd \oplus \#IMM \rightarrow Rd$ Performs a logical exclusive OR operation on a general register and another general register or immediate data. |
| NOT | B/W/L | $\neg Rd \rightarrow Rd$ Takes the one's complement of general register contents. |

Note: * Refers to the operand size.

B: Byte

W: Word

L: Longword

Table 2.6 Shift Instructions

| Instruction | Size* | Function |
|--------------------|--------------|---|
| SHAL | B/W/L | $Rd \text{ (shift)} \rightarrow Rd$ |
| SHAR | | Performs an arithmetic shift on general register contents. 1-bit or 2-bit shifts are possible. |
| SHLL | B/W/L | $Rd \text{ (shift)} \rightarrow Rd$ |
| SHLR | | Performs a logical shift on general register contents. 1-bit or 2-bit shifts are possible. |
| ROTL | B/W/L | $Rd \text{ (rotate)} \rightarrow Rd$ |
| ROTR | | Rotates general register contents. 1-bit or 2-bit rotations are possible. |
| ROTXL | B/W/L | $Rd \text{ (rotate)} \rightarrow Rd$ |
| ROTXR | | Rotates general register contents through the carry flag. 1-bit or 2-bit rotations are possible. |

Note: * Refers to the operand size.

B: Byte

W: Word

L: Longword

Table 2.7 Bit Manipulation Instructions

| Instruction | Size* | Function |
|--------------------|--------------|---|
| BSET | B | $1 \rightarrow \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle$ Sets a specified bit in a general register or memory operand to 1. The bit number is specified by 3-bit immediate data or the lower three bits of a general register. |
| BCLR | B | $0 \rightarrow \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle$ Clears a specified bit in a general register or memory operand to 0. The bit number is specified by 3-bit immediate data or the lower three bits of a general register. |
| BNOT | B | $\neg \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle \rightarrow \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle$ Inverts a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data or the lower three bits of a general register. |
| BTST | B | $\neg \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle \rightarrow Z$ Tests a specified bit in a general register or memory operand and sets or clears the Z flag accordingly. The bit number is specified by 3-bit immediate data or the lower three bits of a general register. |
| BAND | B | $C \wedge \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle \rightarrow C$ ANDs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag. |
| BIAND | B | $C \wedge [\neg \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle] \rightarrow C$ ANDs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data. |
| BOR | B | $C \vee \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle \rightarrow C$ ORs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag. |
| BIOR | B | $C \vee [\neg \langle \text{bit-No.} \rangle \text{ of } \langle \text{EAd} \rangle] \rightarrow C$ ORs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data. |

| Instruction | Size* | Function |
|-------------|-------|---|
| BXOR | B | $C \oplus (\text{<bit-No.> of <EAd>}) \rightarrow C$ XORs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag. |
| BIXOR | B | $C \oplus [\neg (\text{<bit-No.> of <EAd>})] \rightarrow C$ XORs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data. |
| BLD | B | $(\text{<bit-No.> of <EAd>}) \rightarrow C$ Transfers a specified bit in a general register or memory operand to the carry flag. |
| BILD | B | $\neg (\text{<bit-No.> of <EAd>}) \rightarrow C$ Transfers the inverse of a specified bit in a general register or memory operand to the carry flag. The bit number is specified by 3-bit immediate data. |
| BST | B | $C \rightarrow (\text{<bit-No.> of <EAd>})$ Transfers the carry flag value to a specified bit in a general register or memory operand. |
| BIST | B | $\neg C \rightarrow (\text{<bit-No.> of <EAd>})$ Transfers the inverse of the carry flag value to a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data. |

Note: * Refers to the operand size.

B: Byte

Table 2.8 Branch Instructions

| Instruction | Size | Function |
|-------------|-------------------------------|--|
| Bcc | — | Branches to a specified address if a specified condition is true. The branching conditions are listed below. |
| Mnemonic | Description | Condition |
| BRA(BT) | Always (true) | Always |
| BRN(BF) | Never (false) | Never |
| BHI | High | $C \vee Z = 0$ |
| BLS | Low or same | $C \vee Z = 1$ |
| BCC(BHS) | Carry clear (high or same) | $C = 0$ |
| BCS(BLO) | Carry set (low) | $C = 1$ |
| BNE | Not equal | $Z = 0$ |
| BEQ | Equal | $Z = 1$ |
| BVC | Overflow clear | $V = 0$ |
| BVS | Overflow set | $V = 1$ |
| BPL | Plus | $N = 0$ |
| BMI | Minus | $N = 1$ |
| BGE | Greater or equal | $N \oplus V = 0$ |
| BLT | Less than | $N \oplus V = 1$ |
| BGT | Greater than | $Z \vee (N \oplus V) = 0$ |
| BLE | Less or equal | $Z \vee (N \oplus V) = 1$ |
| JMP | — | Branches unconditionally to a specified address. |
| BSR | — | Branches to a subroutine at a specified address. |
| JSR | — | Branches to a subroutine at a specified address. |
| RTS | — | Returns from a subroutine |

Table 2.9 System Control Instructions

| Instruction | Size* | Function |
|--------------------|--------------|---|
| TRAPA | — | Starts trap-instruction exception handling. |
| RTE | — | Returns from an exception-handling routine. |
| SLEEP | — | Causes a transition to a power-down state. |
| LDC | B/W | (EAs) → CCR, (EAs) → EXR Moves the source operand contents or immediate data to CCR or EXR. Although CCR and EXR are 8-bit registers, word-size transfers are performed between them and memory. The upper 8 bits are valid. |
| STC | B/W | CCR → (EAd), EXR → (EAd) Transfers CCR or EXR contents to a general register or memory. Although CCR and EXR are 8-bit registers, word-size transfers are performed between them and memory. The upper 8 bits are valid. |
| ANDC | B | CCR ∧ #IMM → CCR, EXR ∧ #IMM → EXR Logically ANDs the CCR or EXR contents with immediate data. |
| ORC | B | CCR ∨ #IMM → CCR, EXR ∨ #IMM → EXR Logically ORs the CCR or EXR contents with immediate data. |
| XORC | B | CCR ⊕ #IMM → CCR, EXR ⊕ #IMM → EXR Logically XORs the CCR or EXR contents with immediate data. |
| NOP | — | PC + 2 → PC Only increments the program counter. |

Note: * Refers to the operand size.

B: Byte

W: Word

Table 2.10 Block Data Transfer Instructions

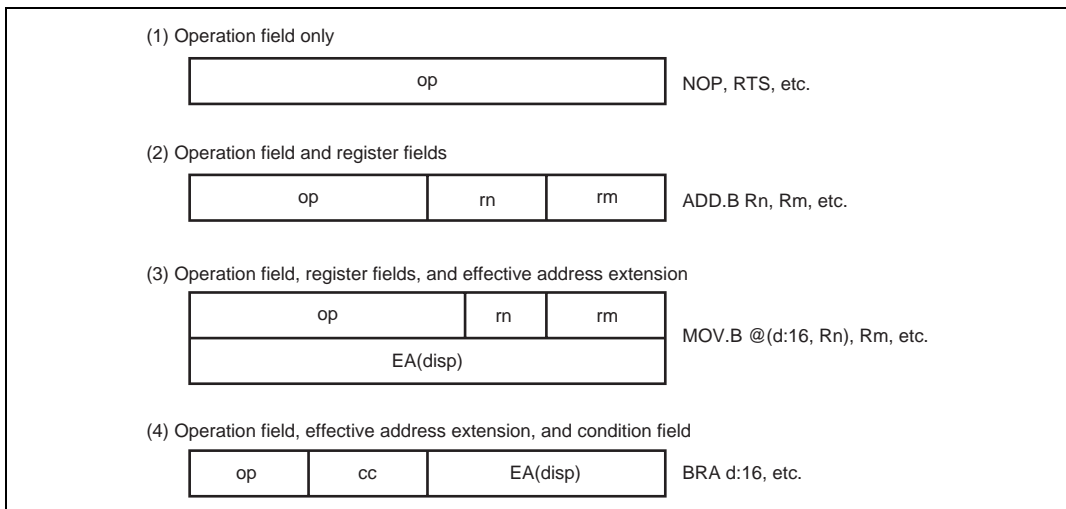
| Instruction | Size | Function |
|-------------|------|---|
| EEPMOV.B | — | if R4L \neq 0 then Repeat @ER5+ \rightarrow @ER6+ R4L-1 \rightarrow R4L Until R4L = 0 else next; |
| EEPMOV.W | — | if R4 \neq 0 then Repeat @ER5+ \rightarrow @ER6+ R4-1 \rightarrow R4 Until R4 = 0 else next; Transfers a data block. Starting from the address set in ER5, transfers data for the number of bytes set in R4L or R4 to the address location set in ER6. Execution of the next instruction begins as soon as the transfer is completed. |

2.6.2 Basic Instruction Formats

The H8S/2000 CPU instructions consist of 2-byte (1-word) units. An instruction consists of an operation field (op), a register field (r), an effective address extension (EA), and a condition field (cc).

Figure 2.11 shows examples of instruction formats.

- **Operation Field**
Indicates the function of the instruction, the addressing mode, and the operation to be carried out on the operand. The operation field always includes the first four bits of the instruction. Some instructions have two operation fields.
- **Register Field**
Specifies a general register. Address registers are specified by 3 bits, and data registers by 3 bits or 4 bits. Some instructions have two register fields. Some have no register field.
- **Effective Address Extension**
8, 16, or 32 bits specifying immediate data, an absolute address, or a displacement.
- **Condition Field**
Specifies the branching condition of Bcc instructions.

**Figure 2.11 Instruction Formats (Examples)**

2.7 Addressing Modes and Effective Address Calculation

The H8S/2000 CPU supports the eight addressing modes listed in table 2.11. Each instruction uses a subset of these addressing modes. Arithmetic and logic instructions can use the register direct and immediate modes. Data transfer instructions can use all addressing modes except program-counter relative and memory indirect. Bit manipulation instructions use register direct, register indirect, or the absolute addressing mode to specify an operand, and register direct (BSET, BCLR, BNOT, and BTST instructions) or immediate (3-bit) addressing mode to specify a bit number in the operand.

Table 2.11 Addressing Modes

| No. | Addressing Mode | Symbol |
|-----|---|----------------------------|
| 1 | Register direct | Rn |
| 2 | Register indirect | @ERn |
| 3 | Register indirect with displacement | @(d:16,ERn)/@(d:32,ERn) |
| 4 | Register indirect with post-increment Register indirect with pre-decrement | @ERn+ @-ERn |
| 5 | Absolute address | @aa:8/@aa:16/@aa:24/@aa:32 |
| 6 | Immediate | #xx:8/#xx:16/#xx:32 |
| 7 | Program-counter relative | @(d:8,PC)/@(d:16,PC) |
| 8 | Memory indirect | @ @aa:8 |

2.7.1 Register Direct—Rn

The register field of the instruction specifies an 8-, 16-, or 32-bit general register containing the operand. R0H to R7H and R0L to R7L can be specified as 8-bit registers. R0 to R7 and E0 to E7 can be specified as 16-bit registers. ER0 to ER7 can be specified as 32-bit registers.

2.7.2 Register Indirect—@ERn

The register field of the instruction code specifies an address register (ERn) which contains the address of the operand on memory. If the address is a program instruction address, the lower 24 bits are valid and the upper 8 bits are all assumed to be 0 (H'00).

2.7.3 Register Indirect with Displacement—@(d:16, ERn) or @(d:32, ERn)

A 16-bit or 32-bit displacement contained in the instruction is added to an address register (ERn) specified by the register field of the instruction, and the sum gives the address of a memory operand. A 16-bit displacement is sign-extended when added.

2.7.4 Register Indirect with Post-Increment—@ERn+ or Register Indirect with Pre-Decrement—@-ERn

Register indirect with post-increment—@ERn+: The register field of the instruction code specifies an address register (ERn) which contains the address of a memory operand. After the operand is accessed, 1, 2, or 4 is added to the address register contents and the sum is stored in the address register. The value added is 1 for byte access, 2 for word transfer instruction, or 4 for longword transfer instruction. For the word or longword transfer instructions, the register value should be even.

Register indirect with pre-decrement—@-ERn: The value 1, 2, or 4 is subtracted from an address register (ERn) specified by the register field in the instruction code, and the result is the address of a memory operand. The result is also stored in the address register. The value subtracted is 1 for byte access, 2 for word transfer instruction, or 4 for longword transfer instruction. For the word or longword transfer instructions, the register value should be even.

2.7.5 Absolute Address—@aa:8, @aa:16, @aa:24, or @aa:32

The instruction code contains the absolute address of a memory operand. The absolute address may be 8 bits long (@aa:8), 16 bits long (@aa:16), 24 bits long (@aa:24), or 32 bits long (@aa:32). Table 2.12 indicates the accessible absolute address ranges.

To access data, the absolute address should be 8 bits (@aa:8), 16 bits (@aa:16), or 32 bits (@aa:32) long. For an 8-bit absolute address, the upper 24 bits are all assumed to be 1 (H'FFFF). For a 16-bit absolute address the upper 16 bits are a sign extension. A 32-bit absolute address can access the entire address space.

A 24-bit absolute address (@aa:24) indicates the address of a program instruction. The upper 8 bits are all assumed to be 0 (H'00).

Table 2.12 Absolute Address Access Ranges

| Absolute Address | | Normal Mode* | Advanced Mode |
|-----------------------------|------------------|------------------|---|
| Data address | 8 bits (@aa:8) | H'FF00 to H'FFFF | H'FFFF00 to H'FFFFFF |
| | 16 bits (@aa:16) | H'0000 to H'FFFF | H'000000 to H'007FFF, H'FF8000 to H'FFFFFF |
| | 32 bits (@aa:32) | | H'000000 to H'FFFFFF |
| Program instruction address | 24 bits (@aa:24) | | |

Note: * Normal mode is not available in this LSI.

2.7.6 Immediate—#xx:8, #xx:16, or #xx:32

The instruction contains 8-bit (#xx:8), 16-bit (#xx:16), or 32-bit (#xx:32) immediate data as an operand.

The ADDS, SUBS, INC and DEC instructions contain immediate data implicitly. Some bit manipulation instructions contain 3-bit immediate data in the instruction code, specifying a bit number. The TRAPA instruction contains 2-bit immediate data in its instruction code, specifying a vector address.

2.7.7 Program-Counter Relative—@(d:8, PC) or @(d:16, PC)

This mode is used in the Bcc and BSR instructions. An 8-bit or 16-bit displacement contained in the instruction is sign-extended and added to the 24-bit PC contents to generate a branch address. Only the lower 24 bits of this branch address are valid; the upper 8 bits are all assumed to be 0 (H'00). The PC value to which the displacement is added is the address of the first byte of the next instruction, so the possible branching range is –126 to +128 bytes (–63 to +64 words) or –32766 to +32768 bytes (–16383 to +16384 words) from the branch instruction. The resulting value should be an even number.

2.7.8 Memory Indirect—@@aa:8

This mode can be used by the JMP and JSR instructions. The instruction code contains an 8-bit absolute address specifying a memory operand. This memory operand contains a branch address. The upper bits of the absolute address are all assumed to be 0, so the address range is 0 to 255 (H'0000 to H'00FF in normal mode*, H'000000 to H'0000FF in advanced mode). In normal mode, the memory operand is a word operand and the branch address is 16 bits long. In advanced mode, the memory operand is a longword operand, the first byte of which is assumed to be 0 (H'00).

Note that the first part of the address range is also the exception vector area. For further details, refer to section 4, Exception Handling.

If an odd address is specified in word or longword memory access, or as a branch address, the least significant bit is regarded as 0, causing data to be accessed or instruction code to be fetched at the address preceding the specified address (For further information, see section 2.5.2, Memory Data Formats).

Note: * Normal mode is not available in this LSI.

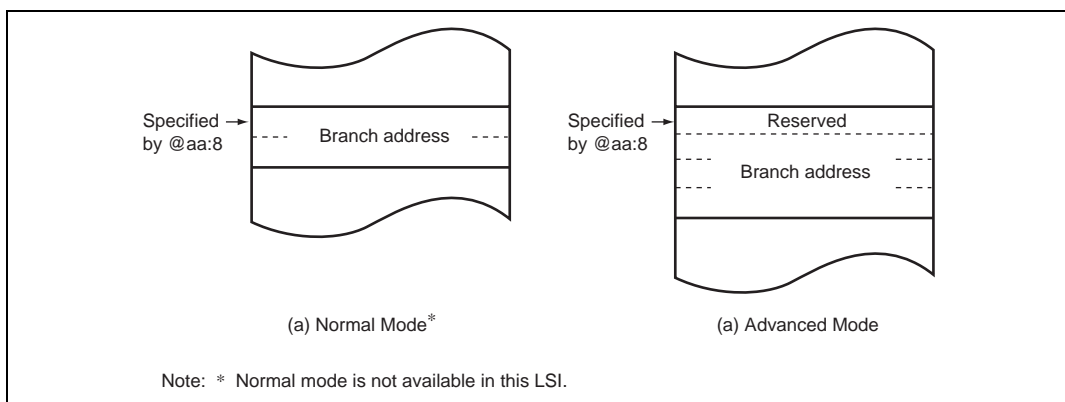


Figure 2.12 Branch Address Specification in Memory Indirect Mode

2.7.9 Effective Address Calculation

Table 2.13 indicates how effective addresses are calculated in each addressing mode. In normal mode the upper 8 bits of the effective address are ignored in order to generate a 16-bit address.

Table 2.13 Effective Address Calculation

| No | Addressing Mode and Instruction Format | Effective Address Calculation | Effective Address (EA) |
|----|---|--|---|
| 1 | Register direct(Rn) <div><div>op</div><div>rm</div><div>m</div></div> | | Operand is general register contents. |
| 2 | Register indirect(@ERn) <div><div>op</div><div>r</div><div></div></div> | <div><div>31</div><div>0</div><div>General register contents</div></div> | <div><div>31</div><div>24</div><div>23</div><div>0</div><div>Don't care</div><div></div></div> |
| 3 | Register indirect with displacement @(d:16,ERn) or @(d:32,ERn) <div><div>op</div><div>r</div><div></div><div>disp</div></div> | <div><div>31</div><div>0</div><div>General register contents</div></div> <div><div>31</div><div>0</div><div>Sign extension</div><div>disp</div></div> | <div><div>31</div><div>24</div><div>23</div><div>0</div><div>Don't care</div><div></div></div> |
| 4 | Register indirect with post-increment or pre-decrement - Register indirect with post-increment @ERn+ <div><div>op</div><div>r</div><div></div></div> - Register indirect with pre-decrement @-ERn <div><div>op</div><div>r</div><div></div></div> | <div><div>31</div><div>0</div><div>General register contents</div></div> <div><div>31</div><div>0</div><div>General register contents</div></div> <div><div>1</div><div>2</div><div>or 4</div></div> | <div><div>31</div><div>24</div><div>23</div><div>0</div><div>Don't care</div><div></div></div> <div><div>31</div><div>24</div><div>23</div><div>0</div><div>Don't care</div><div></div></div> |

| Operand Size | Offset |
|--------------|--------|
| Byte | 1 |
| Word | 2 |
| Longword | 4 |

| No | Addressing Mode and Instruction Format | Effective Address Calculation | Effective Address (EA) |
|----|--|-------------------------------|----------------------------|
| 5 | Absolute address @aa:8 | | |
| | @aa:16 | | |
| | @aa:24 | | |
| | @aa:32 | | |
| 6 | Immediate #xx:8/#xx:16/#xx:32 | | Operand is immediate data. |
| 7 | Program-counter relative @(d:8,PC)/@(d:16,PC) | | |
| 8 | Memory indirect @aa:8 Normal Mode* | | |
| | Advanced extended modes | | |

Note: * Normal mode is not available in this LSI.

2.8 Processing States

The H8S/2000 CPU has five main processing states: the reset state, exception handling state, program execution state, bus-released state, and power-down state. Figure 2.13 indicates the state transitions.

- **Reset State**

In this state, the CPU and all on-chip peripheral modules are initialized and not operating. When the $\overline{\text{RES}}$ input goes low, all current processing stops and the CPU enters the reset state. All interrupts are masked in the reset state. Reset exception handling starts when the $\overline{\text{RES}}$ signal changes from low to high. For details, refer to section 4, Exception Handling.

The reset state can also be entered by a watchdog timer overflow.

- **Exception-Handling State**

The exception-handling state is a transient state that occurs when the CPU alters the normal processing flow due to an exception source, such as a reset, trace, interrupt, or trap instruction. The CPU fetches a start address (vector) from the exception vector table and branches to that address. For further details, refer to section 4, Exception Handling.

- **Program Execution State**

In this state, the CPU executes program instructions in sequence.

- **Bus-Released State**

In a product which has a DMA controller (DMAC)* or data transfer controller (DTC), the bus-released state occurs when the bus has been released in response to a bus request from a bus master other than the CPU.

While the bus is released, the CPU halts operations.

- **Power-down State**

This is a power-down state in which the CPU stops operating. The program stop state occurs when a SLEEP instruction is executed or the CPU enters hardware standby mode. For further details, refer to section 24, Power-Down Modes.

Note: * Supported only by the H8S/2239 Group.

Rev. 6.00 Mar. 18, 2010 Page 97 of 982
REJ09B0054-0600

2.9 Usage Notes

2.9.1 TAS Instruction

Only register ER0, ER1, ER4, or ER5 should be used when using the TAS instruction. The TAS instruction is not generated by the Renesas Technology H8S and H8/300 Series C/C++ compilers. If the TAS instruction is used as a user-defined intrinsic function, ensure that only register ER0, ER1, ER4, or ER5 is used.

2.9.2 STM/LDM Instruction

With the STM or LDM instruction, the ER7 register is used as the stack pointer, and thus cannot be used as a register that allows save (STM) or restore (LDM) operation.

With a single STM or LDM instruction, two to four registers can be saved or restored. The available registers are as follows:

For two registers: ER0 and ER1, ER2 and ER3, or ER4 and ER5

For three registers: ER0 to ER2, or ER4 to ER6

For four registers: ER0 to ER3

For the Renesas Technology H8S or H8/300 Series C/C++ Compiler, the STM/LDM instruction including ER7 is not created.

2.9.3 Bit Manipulation Instructions

When a register that includes write-only bits is manipulated by a bit manipulation instruction, there are cases where the bits manipulated are not manipulated correctly or bits unrelated to the bits manipulated are changed.

When a register containing write-only bits is read, the value read is either a fixed value or an undefined value. This means that the bit manipulation instructions that use the value of bits read in their operation (BNOT, BTST, BAND, BIAND, BOR, BIOR, BXOR, BIXOR, BLD, and BILD) will not perform correct bit operations.

Also, bit manipulation instructions that perform a write operation on the data read after the calculation (BSET, BCLR, BNOT, BST, and BIST) may change bits unrelated to the bits manipulated. Thus extreme care is required when performing bit manipulation instructions on registers that include write-only bits.

The BSET, BCLR, BNOT, BST, and BIST instructions perform their operations in the following order.

1. Read the data in byte units
2. Perform the bit manipulation operation according to the instruction on the data read
3. Write the data back in byte units

Example: Using the BCLR instruction to clear only bit 4 in the port 1 P1DDR register.

The P1DDR register consists of 8 write-only bits and sets the I/O direction of the port 1 pins. Reading this register is invalid. When read, the values returned are undefined.

Here we present an example in which P14 is specified to be an input port using the BCLR instruction. Currently, P17 to P14 are set to be output pins and P13 to P10 are set to be input pins. At this point, the value of P1DDR is H'F0.

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|-------|--------|--------|--------|--------|-------|-------|-------|-------|
| I/O | Output | Output | Output | Output | Input | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

To switch P14 from the Output pin to the input pin function, the value of P1DDR bit 4 must be changed from 1 to 0 (H'F0 → H'E0). Here we assume that the BCLR instruction is used to clear P1DDR bit 4.

BCLR #4,@P1DDR

However if a bit manipulation instruction of the type shown above is used on P1DDR, which is a write-only register, the following problem may occur.

Although the first thing that happens is that data is read from P1DDR in byte units, the value read at this time is undefined. An undefined value is a value that is either 0 or 1 in the register but reads out as an arbitrary value whose relationship to the actual value is unknown. Since the P1DDR bits are all write-only bits, every bit reads out as an undefined value. Although the actual value of P1DDR at this point is H'F0, assume that bit 3 becomes a 1 here, and the value read out is H'F8.

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|------------|--------|--------|--------|--------|-------|-------|-------|-------|
| I/O | Output | Output | Output | Output | Input | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read value | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |

The bit manipulation operation is performed on this value that was read. In this example, bit 4 will be cleared for H'F8.

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|------------------------|--------|--------|--------|--------|-------|-------|-------|-------|
| I/O | Output | Output | Output | Output | Input | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| After bit manipulation | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

After the bit manipulation operation, this data will be written to P1DDR, and the BCLR instruction completes.

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|-------------|--------|--------|--------|-------|--------|-------|-------|-------|
| I/O | Output | Output | Output | Input | Output | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |
| Write value | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 |

Although the instruction was expected to write H'E0 back to P1DDR, it actually wrote H'E8, and P13, which was expected to be an input pin, is changed to function as an output pin. While this section described the case where P13 was read out as a 1, since the values read are undefined when P17 to P10 are read, when this bit manipulation instruction completes, bits that were 0 may be changed to 1, and bits that were 1 may be changed to 0. To avoid this sort of problem, see section 2.9.4, Access Methods for Registers with Write-Only Bits for methods for modifying registers that include write-only bits.

Also note that it is possible to use the BCLR instruction to clear to 0 flags in internal I/O registers. In this case, if it is clear from the interrupt handler or other information that the corresponding flag is set to 1, then there is no need to read the value of the corresponding flag in advance.

2.9.4 Access Methods for Registers with Write-Only Bits

Undefined values will be read out if a data transfer instruction is executed for a register that includes write-only bits, or if a bit manipulation instruction is executed for a register that includes write-only bits. To avoid reading undefined values, use methods such as those shown below to access registers that include write-only bits.

The basic method for writing to a register that includes write-only bits is to create a work area in internal RAM or other memory area and first write the data to that area. Then, perform the desired access operation for that memory and finally write that data to the register that includes write-only bits.

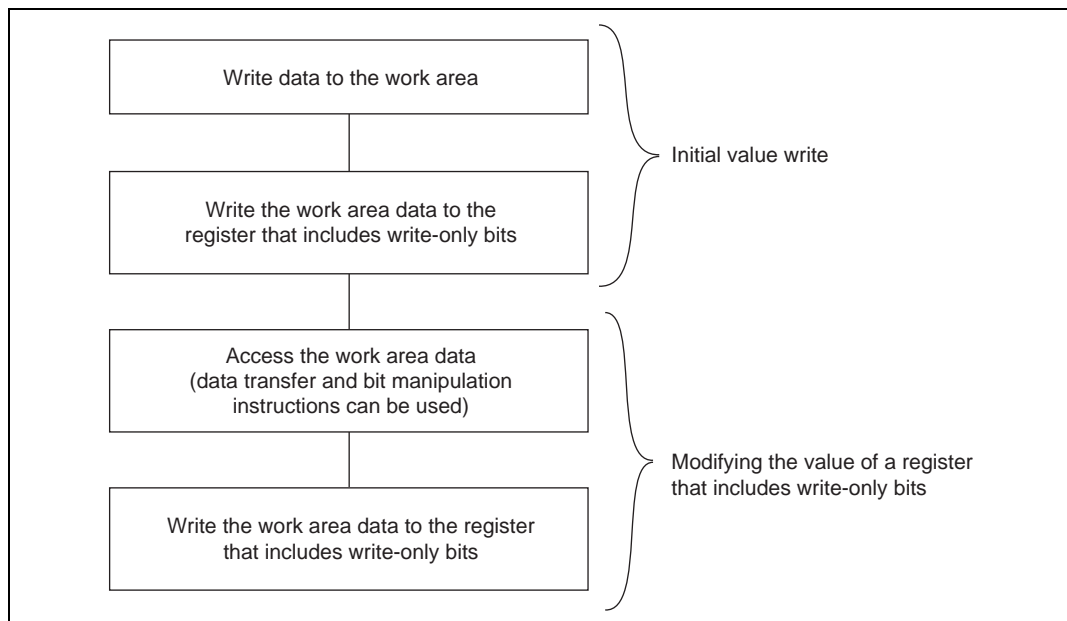


Figure 2.14 Flowchart for Access Methods for Registers That Include Write-Only Bits

Example: To clear only bit 4 in the port 1 P1DDR

The P1DDR register consists of 8 write-only bits and sets the I/O direction of the port 1 pins. Reading this register is invalid. When read, the values returned are undefined.

Here we present an example in which P14 is specified to be an input port using the BCLR instruction. First, we write the initial value H'F0 written to P1DDR to the work area in RAM (RAM0).

```

MOV.B  #H'F0,  R0L
MOV.B  R0L,    @PAM0
MOV.B  R0L,    @P1DDR
  
```

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|-------|--------|--------|--------|--------|-------|-------|-------|-------|
| I/O | Output | Output | Output | Output | Input | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

| | | | | | | | | |
|------|---|---|---|---|---|---|---|---|
| RAM0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
|------|---|---|---|---|---|---|---|---|

To switch P14 from being an output pin to being an input pin, we must change the value of P1DDR bit 4 from 1 to 0 (H'F0 → H'E0). Here, we execute a BCLR instruction for RAM0.

BCLR #4, @RAM0

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|-------|--------|--------|--------|--------|-------|-------|-------|-------|
| I/O | Output | Output | Output | Output | Input | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |

| | | | | | | | | |
|------|---|---|---|---|---|---|---|---|
| RAM0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
|------|---|---|---|---|---|---|---|---|

Since RAM0 can be read and written, when the bit manipulation instruction is executed, only bit 4 in RAM0 is cleared. Then we write this RAM0 value to P1DDR.

MOV.B @RAM0, R0L

MOV.B R0L, @P1DDR

| | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 |
|-------|--------|--------|--------|-------|-------|-------|-------|-------|
| I/O | Output | Output | Output | Input | Input | Input | Input | Input |
| P1DDR | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |

| | | | | | | | | |
|------|---|---|---|---|---|---|---|---|
| RAM0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
|------|---|---|---|---|---|---|---|---|

If this procedure is used to write registers that include write-only bits, programs can be written without depending on the type of the instructions used.

Section 3 MCU Operating Modes

3.1 Operating Mode Selection

The LSI supports four operating modes (modes 7 to 4). These operating modes are used to switch the pin functions. The operating mode is determined by the setting of the mode pins (MD2 to MD0). Modes 6 to 4 are external extended modes used to access external memory or peripheral devices. In the external extended modes each area can be specified as an 8-bit or 16-bit address space using the bus controller after program execution starts. In addition, the 16-bit bus mode is used if any of the areas is configured as 16-bit address space. The 8-bit bus mode is used if all areas are configured as 8-bit address space.

Mode 7 does not use external address space. Do not change the mode pin setting during operation.

Table 3.1 MCU Operating Mode Selection

| MCU Operating Mode | MD2 | MD1 | MD0 | CPU Operating Mode | Description | On-chip ROM | External Data Bus | |
|--------------------------|-----|-----|-----|-----------------------|-------------------------------------|----------------|-------------------|------------------|
| | | | | | | | Initial Value | Maximum Value |
| 4 | 1 | 0 | 0 | Advanced mode | On-chip ROM disabled, extended mode | Disabled | 16 bits | 16 bits |
| 5 | 1 | 0 | 1 | Advanced mode | On-chip ROM disabled, extended mode | Disabled | 8 bits | 16 bits |
| 6 | 1 | 1 | 0 | Advanced mode | On-chip ROM enabled, extended mode | Enabled | 8 bits | 16 bits |
| 7 | 1 | 1 | 1 | Advanced mode | Single-chip mode | Enabled | — | — |

3.2 Register Descriptions

The following registers are related to the operating mode.

- Mode control register (MDCR)
- System control register (SYSCR)

3.2.1 Mode Control Register (MDCR)

MDCR is used to monitor the current operating mode of this LSI.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | — | 1 | — | Reserved This bit is always read as 1 and cannot be modified. |
| 6 to 3 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 2 | MDS2 | —* | R | Mode Select 2 to 0 |
| 1 | MDS1 | —* | R | These bits indicate the input levels at pins MD2 to MD0 (the current operating mode). Bits MDS2 to MDS0 correspond to MD2 to MD0. MDS2 to MDS0 are read-only bits and they cannot be written to. The mode pin (MD2 to MD0) input levels are latched into these bits when MDCR is read. |
| 0 | MDS0 | —* | R | |
| | | | | These latches are canceled by a power-on reset, but maintained at manual reset. |

Note: * Determined by the MD2 to MD0 pin settings.

3.2.2 System Control Register (SYSCR)

SYSCR is used to select the interrupt control mode and the detected edge for NMI, select the MRES input pin enable or disable, and enables or disables on-chip RAM.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | — | 0 | R/W | Reserved The write value should always be 0. |
| 6 | — | 0 | — | Reserved This bit is always read as 0 and cannot be modified. |
| 5 | INTM1 | 0 | R/W | These bits select the control mode of the interrupt controller. For details of the interrupt control modes, see section 5.5.1, Interrupt Control Modes and Interrupt Operation. 00: Interrupt control mode 0 (Interrupt is controlled by I bit) 01: Setting prohibited 10: Interrupt control mode 2 (Interrupt is controlled by I2 to I0 bits and IPR) 11: Setting prohibited |
| 4 | INTM0 | 0 | R/W | |
| 3 | NMIEG | 0 | R/W | |
| | | | | |
| | | | | NMI Edge Select Selects the valid edge of the NMI interrupt input. 0: An interrupt is requested at the falling edge of NMI input 1: An interrupt is requested at the rising edge of NMI input |
| 2 | MRESE | 0 | R/W | Manual Reset Select Enables or disables the $\overline{\text{MRES}}$ pin input. 0: The $\overline{\text{MRES}}$ pin input (manual reset) is disabled 1: The $\overline{\text{MRES}}$ pin input (manual reset) is enabled The $\overline{\text{MRES}}$ input pin can be used |
| 1 | — | 0 | — | Reserved This bit is always read as 0 and cannot be modified. |
| 0 | RAME | 1 | R/W | RAM Enable Enables or disables the on-chip RAM. The RAME bit is initialized when the reset status is released. 0: On-chip RAM is disabled 1: On-chip RAM is enabled |

3.3 Operating Mode Descriptions

3.3.1 Mode 4

The CPU can access a 16-Mbyte address space in advanced mode. The on-chip ROM is disabled.

Pins P13 to P10, and ports A, B, and C function as an address bus, ports D and E function as a data bus, and part of port F carries bus control signals.

Pins P13 to P11 function as input ports immediately after a reset. Pin 10 and ports A and B function as address (A20 to A8) outputs immediately after a reset. Address (A23 to A21) output can be enabled or disabled by bits AE3 to AE0 in the pin function control register (PFCR) regardless of the corresponding data direction register (DDR) values. Pins for which address output is disabled among pins P13 to P10 and in ports A and B become port outputs when the corresponding DDR bits are set to 1.

Port C always has an address (A7 to A0) output function.

The initial bus mode after a reset is 16 bits, with 16-bit access to all areas. However, note that if 8-bit access is designated by the bus controller for all areas, the bus mode switches to 8 bits.

3.3.2 Mode 5

The CPU can access a 16-Mbyte address space in advanced mode. The on-chip ROM is disabled.

Pins P13 to P10, and ports A, B, and C function as an address bus, ports D and E function as a data bus, and part of port F carries bus control signals.

Pins P13 to P11 function as input ports immediately after a reset. Pin 10 and ports A and B function as address (A20 to A8) outputs immediately after a reset. Address (A23 to A21) output can be enabled or disabled by bits AE3 to AE0 in the pin function control register (PFCR) regardless of the corresponding data direction register (DDR) values. Pins for which address output is disabled among pins P13 to P10 and in ports A and B become port outputs when the corresponding DDR bits are set to 1.

Port C always has an address (A7 to A0) output function.

The initial bus mode after a reset is 8 bits, with 8-bit access to all areas. However, note that if 16-bit access is designated by the bus controller for any area, the bus mode switches to 16 bits and port E becomes a data bus.

3.3.3 Mode 6

The CPU can access a 16-Mbyte address space in advanced mode. The on-chip ROM is enabled.

Pins P13 to P10, and ports A, B, and C function as input ports immediately after a reset. Address (A23 to A8) output can be enabled or disabled by bits AE3 to AE0 in the pin function control register (PFCR) regardless of the corresponding data direction register (DDR) values. Pins for which address output is disabled among pins P13 to P10 and in ports A and B become port outputs when the corresponding DDR bits are set to 1.

Port C is an input port immediately after a reset. Addresses A7 to A0 are output by setting the corresponding DDR bits to 1.

Ports D and E function as a data bus, and part of port F carries data bus signals.

The initial bus mode after a reset is 8 bits, with 8-bit access to all areas. However, note that if 16-bit access is designated by the bus controller for any area, the bus mode switches to 16 bits and port E becomes a data bus.

3.3.4 Mode 7

The CPU can access a 16-Mbyte address space in advanced mode. The on-chip ROM is enabled, but external addresses cannot be accessed.

All I/O ports are available for use as input-output ports.

3.3.5 Pin Functions

The pin functions of ports 1, and A to F vary depending on the operating mode. Table 3.2 shows their functions in each operating mode.

Table 3.2 Pin Functions in Each Operating Mode

| Port | | Mode 4 | Mode 5 | Mode 6 | Mode 7 |
|--------|------------|--------|--------|--------|--------|
| Port 1 | P13 to P11 | P*/A | P*/A | P*/A | P |
| | P10 | P/A* | P/A* | P*/A | P |
| Port A | PA3 to PA0 | P/A* | P/A* | P*/A | P |
| Port B | | P/A* | P/A* | P*/A | P |
| Port C | | A | A | P*/A | P |
| Port D | | D | D | D | P |
| Port E | | P/D* | P*/D | P*/D | P |
| Port F | PF7 | P/C* | P/C* | P/C* | P*/C |
| | PF6 to PF4 | C | C | C | P |
| | PF3 | P/C* | P*/C | P*/C | P |
| | PF2 to PF0 | P*/C | P*/C | P*/C | P |

Legend:

P: I/O port

A: Address bus output

D: Data bus I/O

C: Control signals, clock I/O

*: After reset

3.4 Memory Map in Each Operating Mode

Figures 3.1 to 3.9 show the memory map in each operating mode.

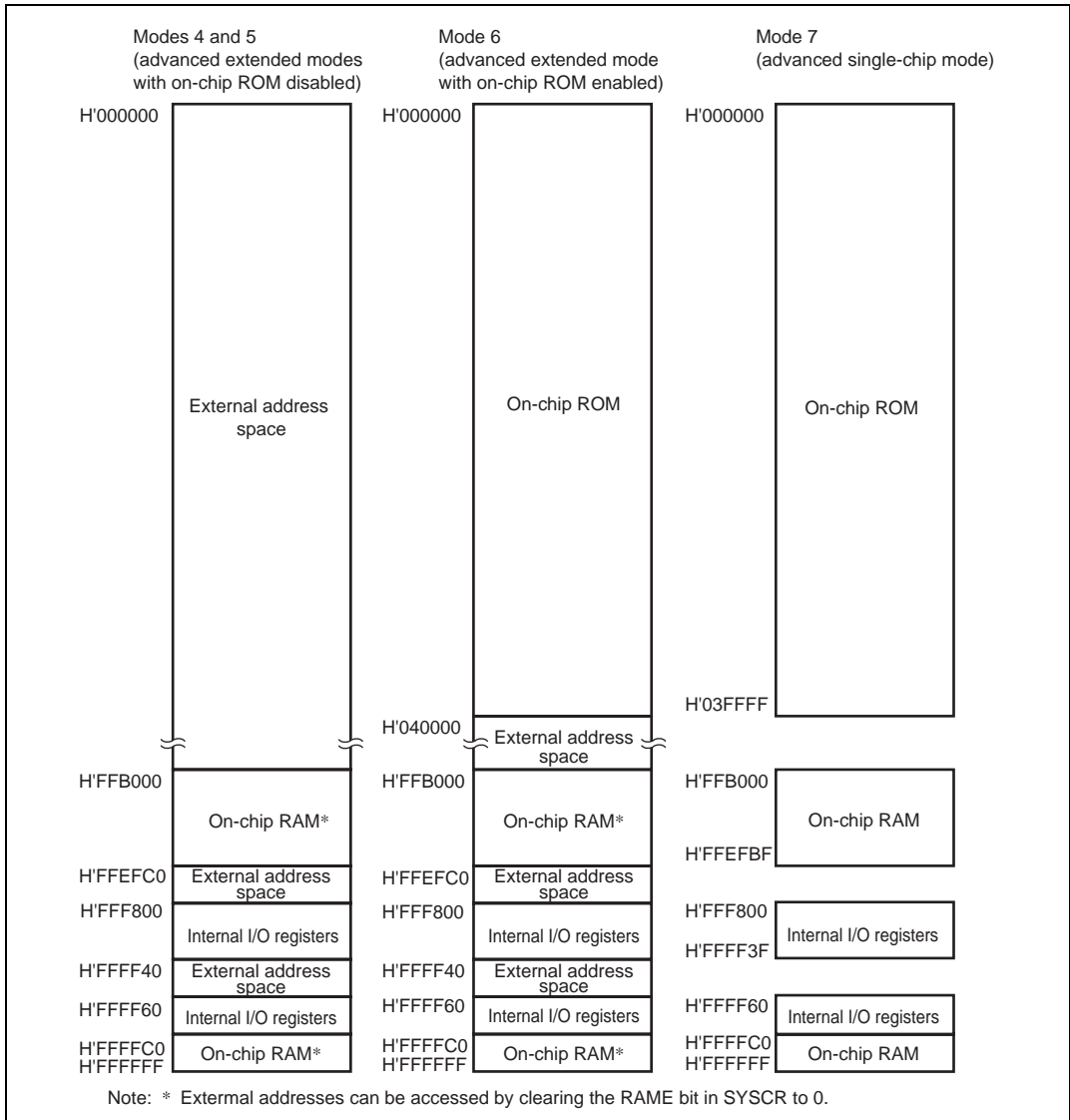


Figure 3.1 H8S/2258 Memory Map in Each Operating Mode

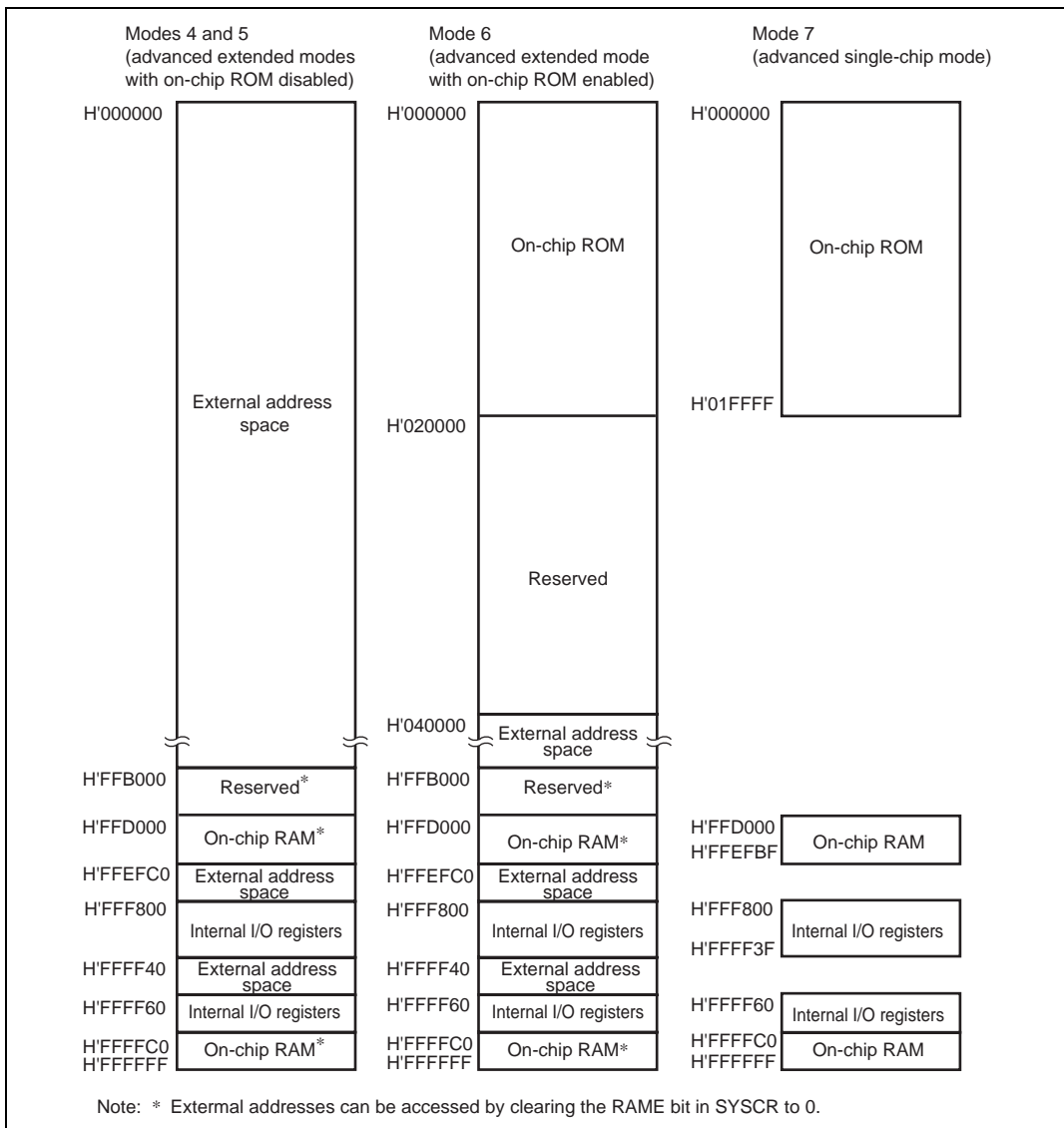


Figure 3.2 H8S/2256 Memory Map in Each Operating Mode

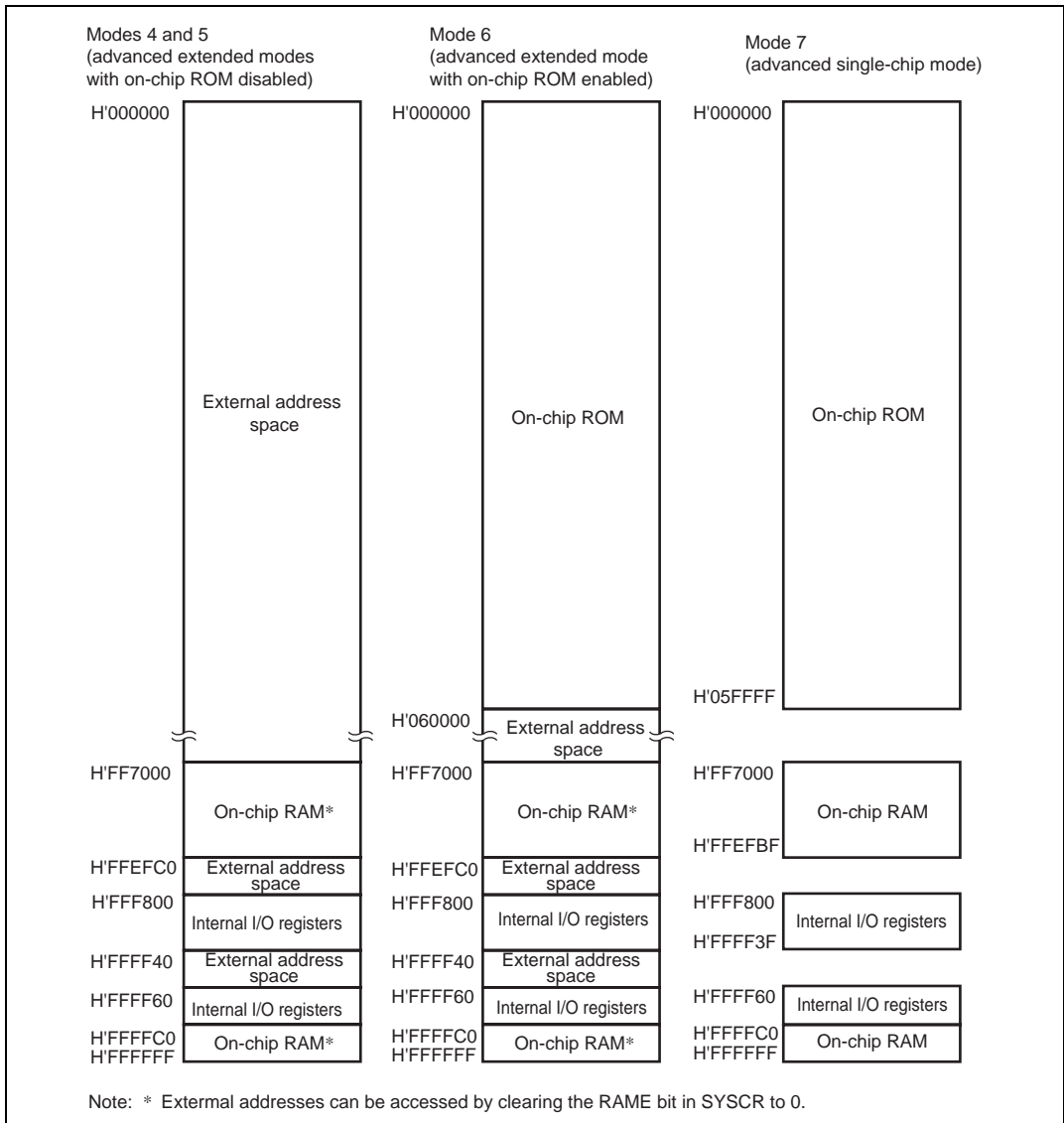


Figure 3.3 H8S/2239 Memory Map in Each Operating Mode

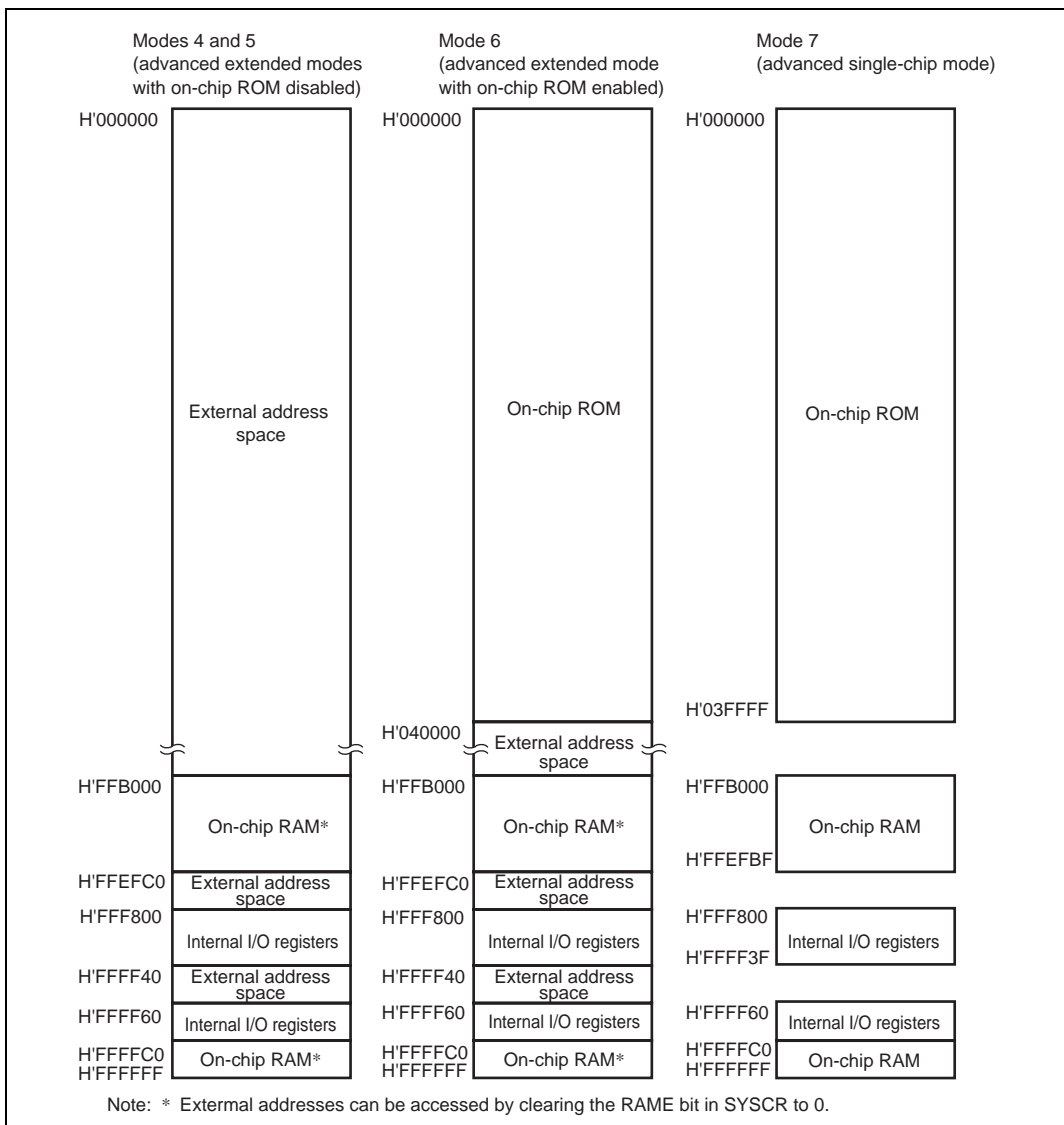


Figure 3.4 H8S/2238B and H8S/2238R Memory Map in Each Operating Mode

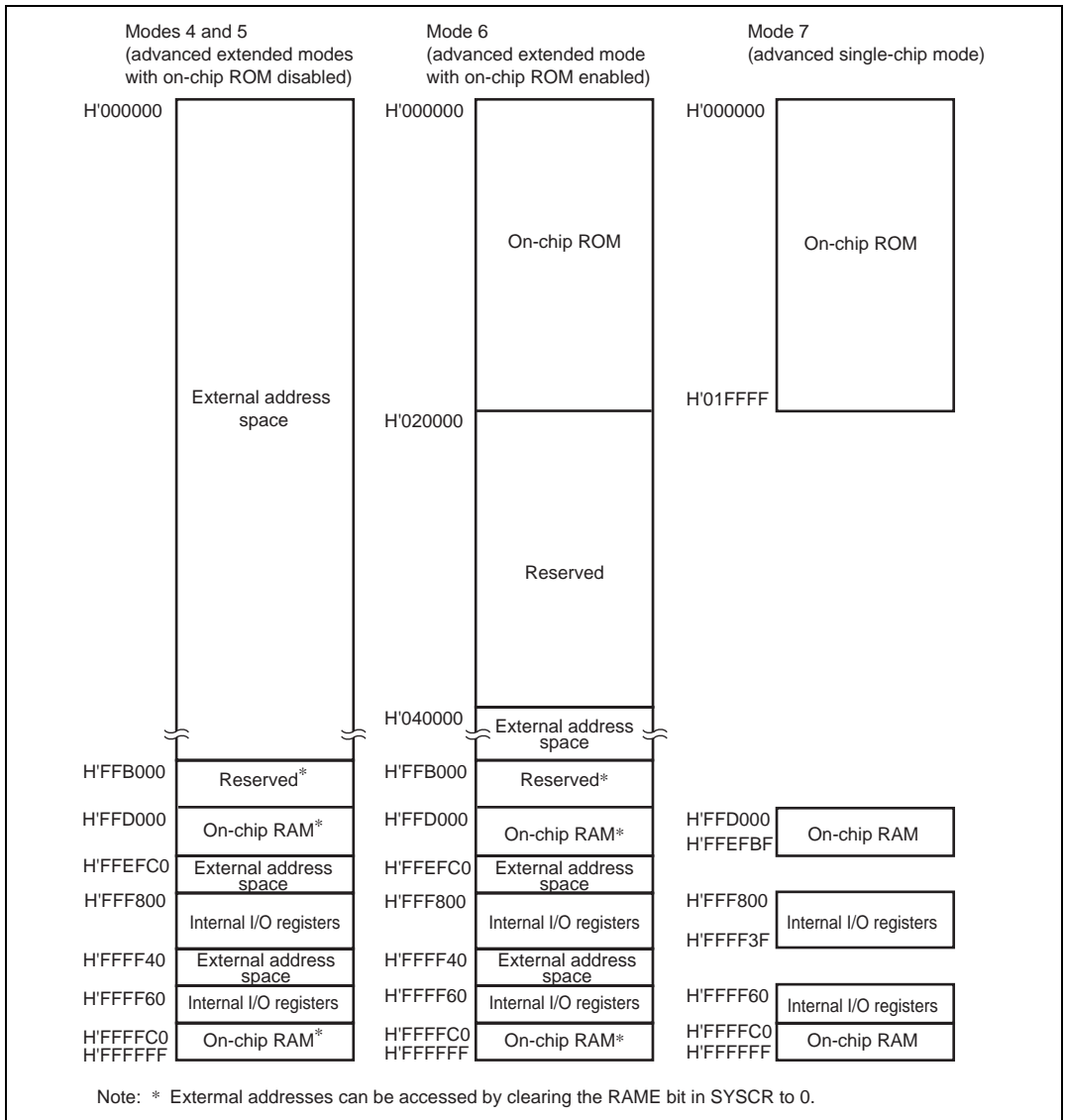


Figure 3.5 H8S/2236B and H8S/2236R Memory Map in Each Operating Mode

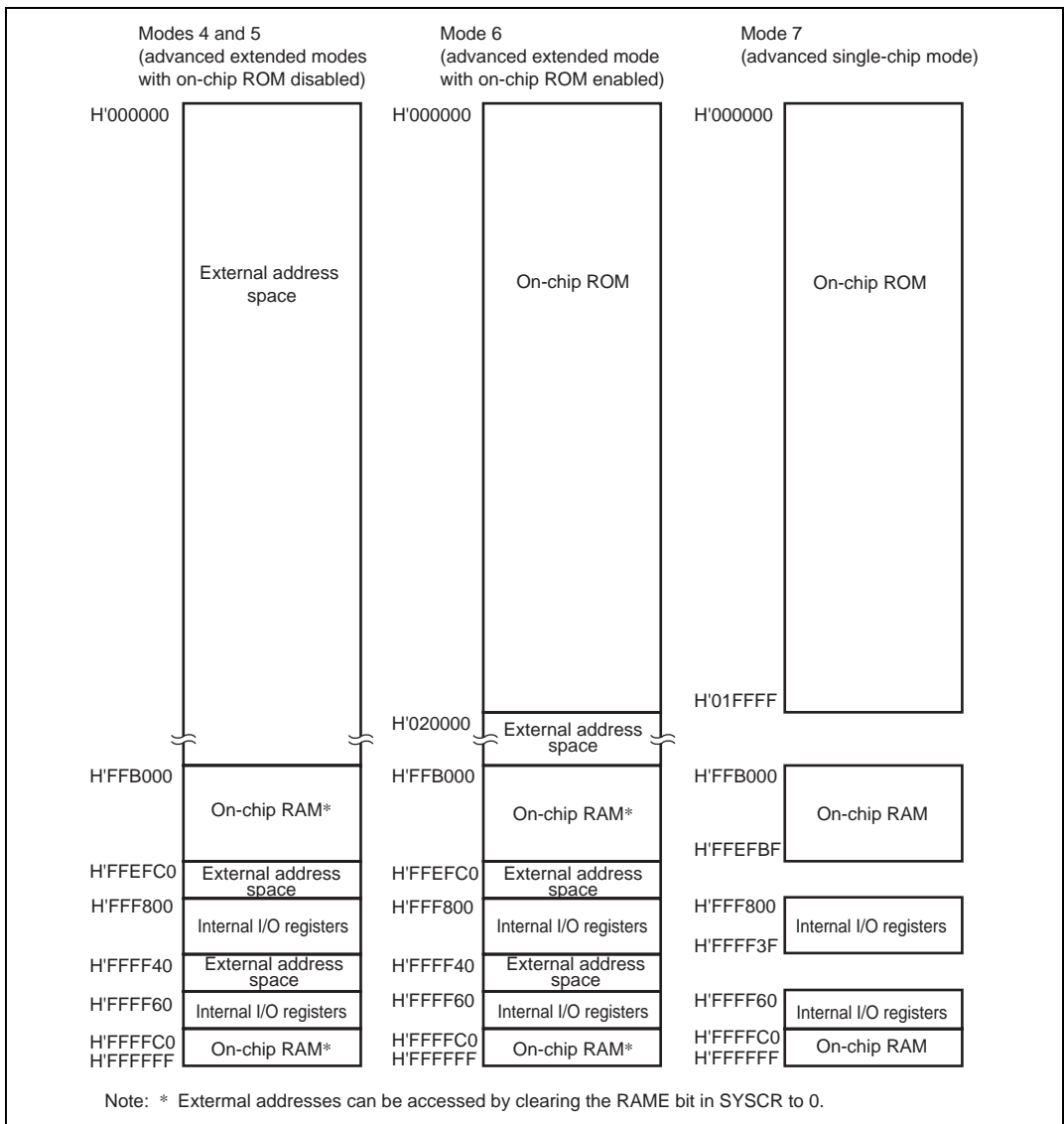


Figure 3.6 H8S/2237 and H8S/2227 Memory Map in Each Operating Mode

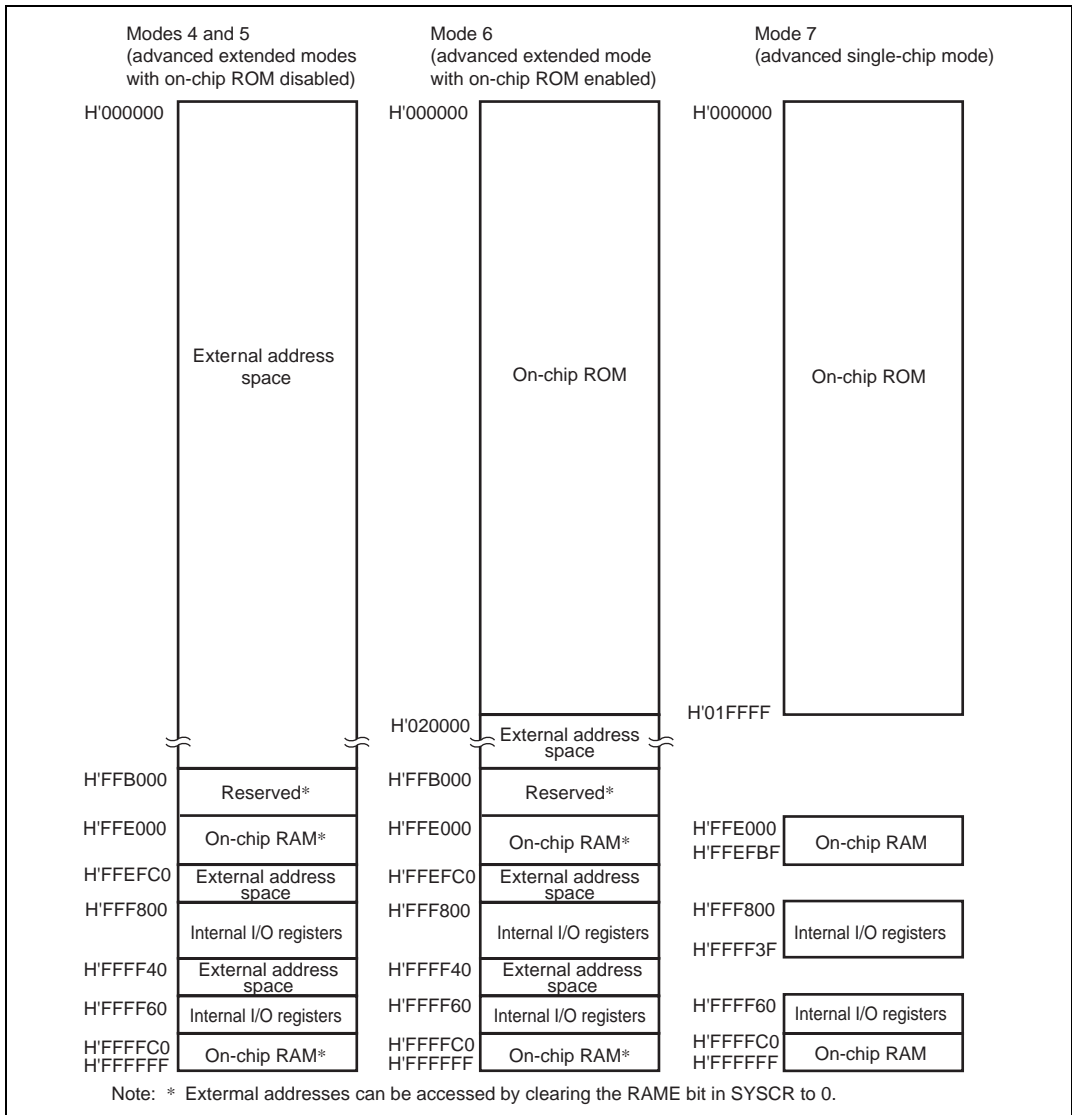


Figure 3.7 H8S/2235 and H8S/2225 Memory Map in Each Operating Mode

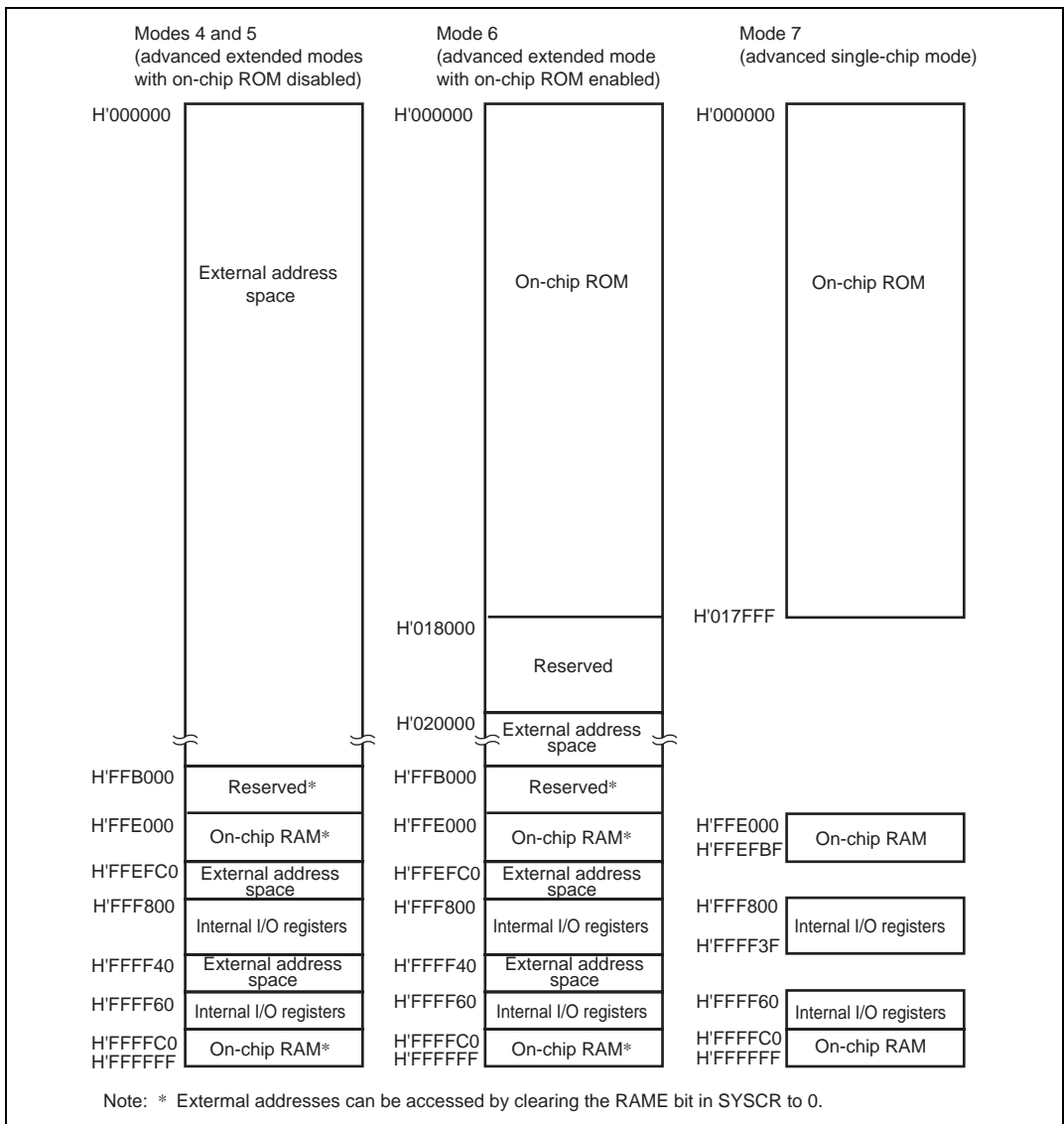


Figure 3.8 H8S/2224 Memory Map in Each Operating Mode

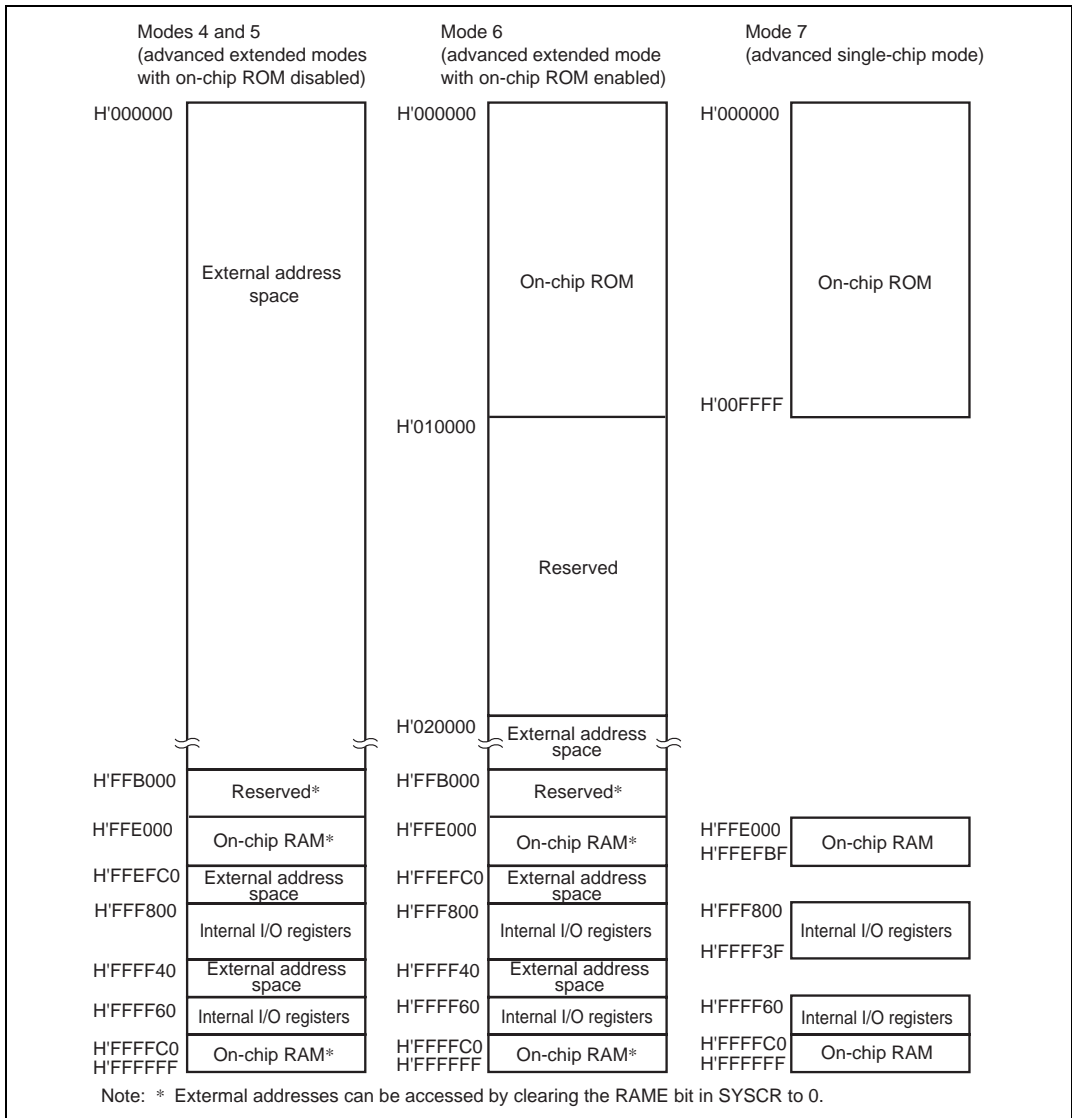


Figure 3.9 H8S/2233 and H8S/2223 Memory Map in Each Operating Mode


Section 4 Exception Handling

4.1 Exception Handling Types and Priority

As table 4.1 indicates, exception handling may be caused by a reset, trace, trap instruction, or interrupt. Exception handling is prioritized as shown in table 4.1. If two or more exceptions occur simultaneously, they are accepted and processed in order of priority. Trap instruction exception handling requests are accepted at all times in program execution state.

Exception sources, the stack structure, and operation of the CPU vary depending on the interrupt control mode set by the INTM1 and INTM0 bits in SYSCR.

Table 4.1 Exception Types and Priority

| Priority | Exception Type | Start of Exception Handling |
|---|--------------------------|--|
| High  Low | Reset | Starts immediately after a low-to-high transition at the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin, or when the watchdog timer overflows. The CPU enters the power-on reset state when the $\overline{\text{RES}}$ pin is low. The CPU enters the manual reset state when the $\overline{\text{MRES}}$ pin is low. |
| | Trace | Starts when execution of the current instruction or exception handling ends, if the trace (T) bit in the EXR is set to 1. Traces are enabled only in interrupt control mode 2. Trace exception handling is not executed after execution of an RTE instruction. |
| | Interrupt | Starts when execution of the current instruction or exception handling ends, if an interrupt request has been issued. Interrupt detection is not performed on completion of ANDC, ORC, XORC, or LDC instruction execution, or on completion of reset exception handling. |
| | Trap instruction (TRAPA) | Started by execution of a trap instruction (TRAPA). Trap instruction exception handling requests are accepted at all times in program execution state. |

4.2 Exception Sources and Exception Vector Table

Different vector addresses are assigned to different exception sources. Table 4.2 lists the exception sources and their vector addresses.

Table 4.2 Exception Handling Vector Table

| Exception Source | | Vector Number | Vector Address Advanced Mode ^{*1} |
|----------------------------------|------|---------------|---|
| Power-on reset | | 0 | H'0000 to H'0003 |
| Manual reset | | 1 | H'0004 to H'0007 |
| Reserved for system use | | 2 | H'0008 to H'000B |
| | | 3 | H'000C to H'000F |
| | | 4 | H'0010 to H'0013 |
| Trace | | 5 | H'0014 to H'0017 |
| Direct transitions ^{*3} | | 6 | H'0018 to H'001B |
| External interrupt (NMI) | | 7 | H'001C to H'001F |
| Trap instruction (four sources) | | 8 | H'0020 to H'0023 |
| | | 9 | H'0024 to H'0027 |
| | | 10 | H'0028 to H'002B |
| | | 11 | H'002C to H'002F |
| Reserved for system use | | 12 | H'0030 to H'0033 |
| | | 13 | H'0034 to H'0037 |
| | | 14 | H'0038 to H'003B |
| | | 15 | H'003C to H'003F |
| External interrupt | IRQ0 | 16 | H'0040 to H'0043 |
| | IRQ1 | 17 | H'0044 to H'0047 |
| | IRQ2 | 18 | H'0048 to H'004B |
| | IRQ3 | 19 | H'004C to H'004F |
| | IRQ4 | 20 | H'0050 to H'0053 |
| | IRQ5 | 21 | H'0054 to H'0057 |
| | IRQ6 | 22 | H'0058 to H'005B |
| | IRQ7 | 23 | H'005C to H'005F |
| Internal interrupt ^{*2} | | 24 | H'0060 to H'0063 |
| | | 123 | H'01EC to H'01EF |

Notes: 1. Lower 16 bits of the address.

2. For details of internal interrupt vectors, see section 5.4.3, Interrupt Exception Handling Vector Table.

3. For details on direct transitions, see section 24.10, Direct Transitions.

4.3 Reset

A reset has the highest exception priority.

When the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin goes low, all processing halts and this LSI enters the reset. A reset initializes the internal state of the CPU and the registers of on-chip peripheral modules. The interrupt control mode is 0 immediately after reset.

When the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin goes high from the low state, this LSI starts reset exception handling.

The chip can also be reset by overflow of the watchdog timer. For details see section 13, Watchdog Timer (WDT).

4.3.1 Reset Types

The power-on reset and the manual reset are available as the reset.

Table 4.3 lists the reset types. When the power is supplied, select the power-on reset.

Both the power-on reset and the manual reset initialize the internal state of the CPU. The power-on reset initializes all registers in on-chip peripheral modules. The manual reset initializes the registers in on-chip peripheral modules except the bus controller and the I/O ports. The state of the bus controller and the I/O ports are maintained.

At the manual reset, the on-chip peripheral modules are initialized. Thus, the ports that are used as I/O pins for the on-chip peripheral modules are changed to the ports controlled by the DDR and the DR.

Table 4.3 Reset Types

| Reset | Condition to Enter Reset | | Internal State | |
|----------------|-----------------------------|-------------------------|----------------|---|
| | $\overline{\text{MRES}}$ | $\overline{\text{RES}}$ | CPU | Internal Peripheral Modules |
| Power-on reset | × | Low | Initialized | Initialized |
| Manual reset | Low | High | Initialized | Initialized except the bus controller and the I/O ports |

Legend: ×:Don't care

The power-on reset and the manual reset are also available for the reset by the watchdog timer.

To enable the $\overline{\text{MRES}}$ pin, set the MRESE bit in SYSCR to 1.

4.3.2 Reset Exception Handling

When the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin goes low, this LSI enters the reset. To ensure that this LSI is reset, hold the $\overline{\text{RES}}$ pin low for at least 20 ms at power-up. To reset the chip during operation, hold the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin low for at least 20 states. When the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin goes high after being held low for the necessary time, this LSI starts reset exception handling as follows.

1. The internal state of the CPU and the registers of the on-chip peripheral modules are initialized, the T bit in EXR is cleared to 0, and the I bits in EXR and CCR are set to 1.
2. The reset exception handling vector address is read and transferred to the PC, and program execution starts from the address indicated by the PC.

Figures 4.1 shows an example of the reset sequence.

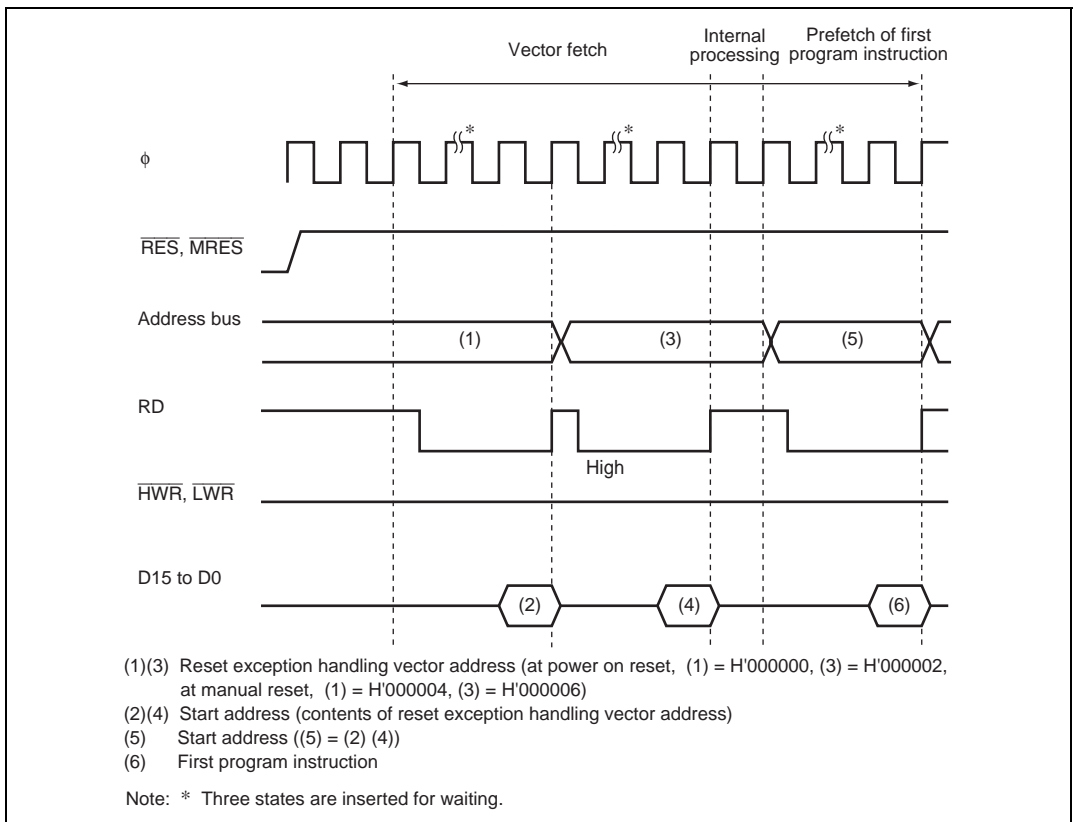


Figure 4.1 Reset Sequence (Mode 4)

4.3.3 Interrupts after Reset

If an interrupt is accepted after a reset and before the stack pointer (SP) is initialized, the PC and CCR will not be saved correctly, leading to a program crash. To prevent this, all interrupt requests, including NMI, are disabled immediately after a reset. Since the first instruction of a program is always executed immediately after the reset state ends, make sure that this instruction initializes the stack pointer (example: MOV.L #xx,SP).

4.3.4 State of On-Chip Peripheral Modules after Reset Release

After reset release, MSTPCRA is initialized to H'3F, MSTPCRB and MSTPCRC are initialized to H'FF, and all modules except the DMAC* and DTC enter module stop mode. Consequently, on-chip peripheral module registers cannot be read or written to. Register reading and writing is enabled when the module stop mode is exited.

Note: * Supported only by the H8S/2239 Group.

4.4 Traces

Traces are enabled in interrupt control mode 2. Trace mode is not activated in interrupt control mode 0, irrespective of the state of the T bit. For details of interrupt control modes, see section 5, Interrupt Controller.

If the T bit in EXR is set to 1, trace mode is activated. In trace mode, a trace exception occurs on completion of each instruction. Trace mode is not affected by interrupt masking. Table 4.4 shows the state of CCR and EXR after execution of trace exception handling. Trace mode is canceled by clearing the T bit in EXR to 0. Interrupts are accepted even within the trace exception handling routine.

The T bit saved on the stack retains its value of 1, and when control is returned from the trace exception handling routine by the RTE instruction, trace mode resumes. Trace exception handling is not carried out after execution of the RTE instruction.

Table 4.4 Status of CCR and EXR after Trace Exception Handling

| Interrupt Control Mode | CCR | | EXR | |
|------------------------|--|----|----------|---|
| | I | UI | I2 to I0 | T |
| 0 | Trace exception handling cannot be used. | | | |
| 2 | 1 | — | — | 0 |

Legend:

- 1: Set to 1
- 0: Cleared to 0
- : Retains value prior to execution

4.5 Interrupts

Interrupts are controlled by the interrupt controller. The interrupt control has two interrupt control modes and can assign interrupts other than NMI to eight priority/mask levels to enable multiplexed interrupt control. For details, refer to section 5, Interrupt Controller.

Interrupt exception handling is conducted as follows:

1. The values in the program counter (PC), condition code register (CCR), and extended control register (EXR) are saved to the stack.
2. The interrupt mask bit is updated and the T bit is cleared to 0.
3. A vector address corresponding to the interrupt source is generated, the start address is loaded from the vector table to the PC, and program execution begins from that address.

4.6 Trap Instruction

Trap instruction exception handling starts when a TRAPA instruction is executed. Trap instruction exception handling can be executed at all times in the program execution state.

Trap instruction exception handling is conducted as follows:

1. The values in the program counter (PC), condition code register (CCR), and extended control register (EXR) are saved to the stack.
2. The interrupt mask bit is updated and the T bit is cleared to 0.
3. A vector address corresponding to the interrupt source is generated, the start address is loaded from the vector table to the PC, and program execution starts from that address.

The TRAPA instruction fetches a start address from a vector table entry corresponding to a vector number from 0 to 3, as specified in the instruction code.

Table 4.5 shows the status of CCR and EXR after execution of trap instruction exception handling.

Table 4.5 Status of CCR and EXR after Trap Instruction Exception Handling

| Interrupt Control Mode | CCR | | EXR | |
|------------------------|-----|----|----------|---|
| | I | UI | I2 to I0 | T |
| 0 | 1 | — | — | — |
| 2 | 1 | — | — | 0 |

Legend:

- 1: Set to 1
- 0: Cleared to 0
- : Retains value prior to execution

4.7 Stack Status after Exception Handling

Figures 4.2 shows the stack after completion of trap instruction exception handling and interrupt exception handling.

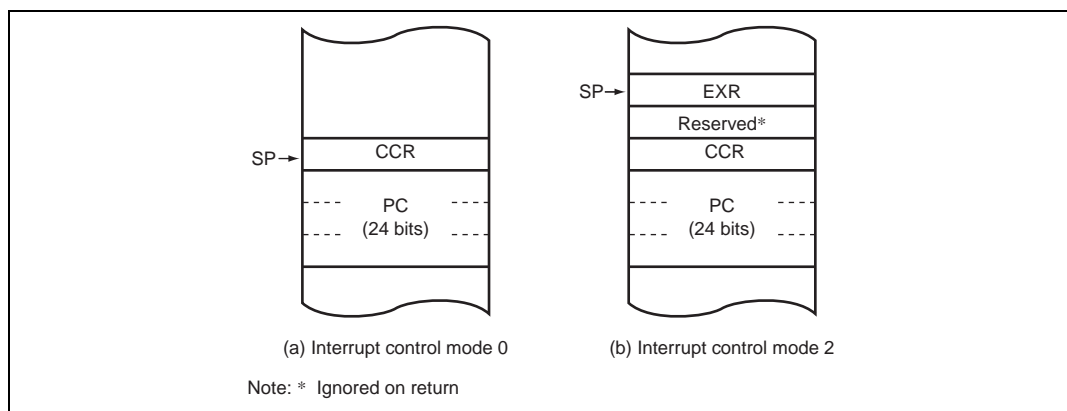


Figure 4.2 Stack Status after Exception Handling (Advanced Mode)

4.8 Usage Note

When accessing word data or longword data, this LSI assumes that the lowest address bit is 0. The stack should always be accessed by word transfer instruction or longword transfer instruction, and the value of the stack pointer (SP, ER7) should always be kept even. Use the following instructions to save registers:

```
PUSH.W   Rn      (or MOV.W Rn, @-SP)
PUSH.L   ERn     (or MOV.L ERn, @-SP)
```

Use the following instructions to restore registers:

```
POP.W    Rn      (or MOV.W @SP+, Rn)
POP.L    ERn     (or MOV.L @SP+, ERn)
```

Setting SP to an odd value may lead to a malfunction. Figure 4.3 shows an example of what happens when the SP value is odd.

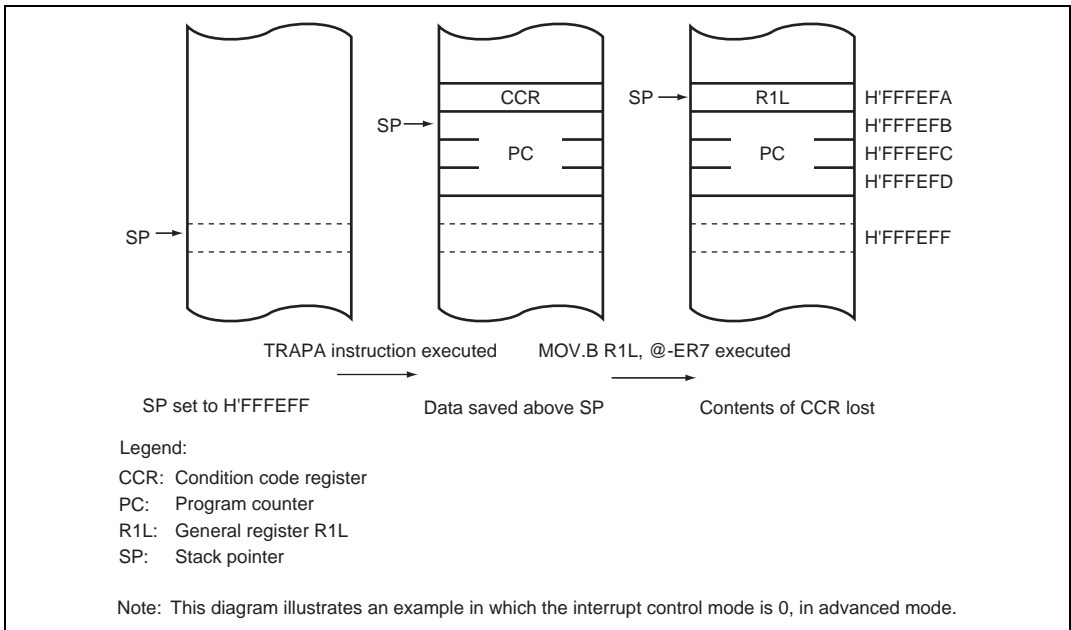


Figure 4.3 Operation When SP Value Is Odd

Section 5 Interrupt Controller

5.1 Features

This LSI controls interrupts with the interrupt controller. The interrupt controller has the following features:

- Two interrupt control modes
 - Any of two interrupt control modes can be set by means of the INTM1 and INTM0 bits in the system control register (SYSCR).
- Priorities settable with IPR
 - An interrupt priority register (IPR) is provided for setting interrupt priorities. Eight priority levels can be set for each module for all interrupts except NMI. NMI is assigned the highest priority level of 8, also accepted (using nesting) during interrupt processing. Additionally accepted during state 12 if Opcode = H'57F3.
- Independent vector addresses
 - All interrupt sources are assigned independent vector addresses, making it unnecessary for the source to be identified in the interrupt handling routine.
- Nine external interrupts
 - NMI is the highest-priority interrupt, and is accepted at all times. Rising edge or falling edge can be selected for NMI. Falling edge, rising edge, or both edge detection, or level sensing, can be independently selected for IRQ7 to IRQ0.
- DTC and DMAC* control
 - The DTC and DMAC* can be activated by an interrupt request.

Note: * Supported only by the H8S/2239 Group.

A block diagram of the interrupt controller is shown in figure 5.1.

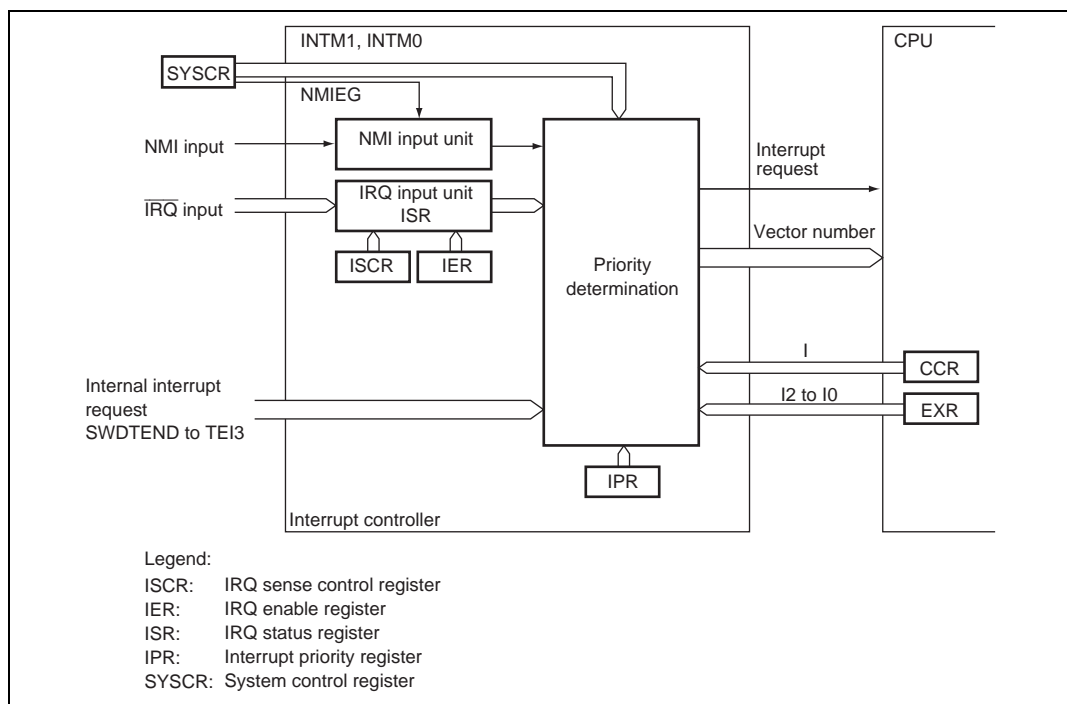


Figure 5.1 Block Diagram of Interrupt Controller

5.2 Input/Output Pins

Table 5.1 summarizes the pins of the interrupt controller.

Table 5.1 Pin Configuration

| Name | I/O | Function |
|--------------------------|-------|--|
| NMI | Input | Nonmaskable external interrupt. Rising or falling edge can be selected. |
| $\overline{\text{IRQ7}}$ | Input | Maskable external interrupts. Rising, falling, or both edges, or level sensing can be selected. |
| $\overline{\text{IRQ6}}$ | Input | |
| $\overline{\text{IRQ5}}$ | Input | |
| $\overline{\text{IRQ4}}$ | Input | |
| $\overline{\text{IRQ3}}$ | Input | |
| $\overline{\text{IRQ2}}$ | Input | |
| $\overline{\text{IRQ1}}$ | Input | |
| $\overline{\text{IRQ0}}$ | Input | |

5.3 Register Descriptions

The interrupt controller has the following registers. For the system control register, see section 3.2.2, System Control Register (SYSCR).

- System control register (SYSCR)
- IRQ sense control register H (ISCRH)
- IRQ sense control register L (ISCRL)
- IRQ enable register (IER)
- IRQ status register (ISR)
- Interrupt priority register A (IPRA)
- Interrupt priority register B (IPRB)
- Interrupt priority register C (IPRC)
- Interrupt priority register D (IPRD)
- Interrupt priority register E (IPRE)
- Interrupt priority register F (IPRF)
- Interrupt priority register G (IPRG)
- Interrupt priority register H (IPRH)
- Interrupt priority register I (IPRI)

- Interrupt priority register J (IPRJ)
- Interrupt priority register K (IPRK)
- Interrupt priority register L (IPRL)
- Interrupt priority register O (IPRO)

5.3.1 Interrupt Priority Registers A to L, and O (IPRA to IPRL, IPRO)

The IPR registers are thirteen 8-bit readable/writable registers that set priorities (levels 7 to 0) for interrupt sources other than NMI. The correspondence between interrupt sources and IPR settings is shown in table 5.2. Setting a value in the range from H'0 to H'7 in the 3-bit groups of bits 0 to 2 and 4 to 6 sets the priority of the corresponding interrupt.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | — | 0 | — | Reserved This bit is always read as 0, and cannot be modified. |
| 6 | IPR6 | 1 | R/W | Sets the priority of the corresponding interrupt source |
| 5 | IPR5 | 1 | R/W | 000: Priority level 0 (Lowest) |
| 4 | IPR4 | 1 | R/W | 001: Priority level 1 010: Priority level 2 011: Priority level 3 100: Priority level 4 101: Priority level 5 110: Priority level 6 111: Priority level 7 (Highest) |
| 3 | — | 0 | — | Reserved This bit is always read as 0, and cannot be modified. |
| 2 | IPR2 | 1 | R/W | Sets the priority of the corresponding interrupt source. |
| 1 | IPR1 | 1 | R/W | 000: Priority level 0 (Lowest) |
| 0 | IPR0 | 1 | R/W | 001: Priority level 1 010: Priority level 2 011: Priority level 3 100: Priority level 4 101: Priority level 5 110: Priority level 6 111: Priority level 7 (Highest) |

5.3.2 IRQ Enable Register (IER)

IER controls the enabling and disabling of interrupt requests $\overline{\text{IRQn}}$ ($n = 7$ to 0).

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | IRQ7E | 0 | R/W | IRQ7 Enable The IRQ7 interrupt request is enabled when this bit is 1. |
| 6 | IRQ6E | 0 | R/W | IRQ6 Enable The IRQ6 interrupt request is enabled when this bit is 1. |
| 5 | IRQ5E | 0 | R/W | IRQ5 Enable The IRQ5 interrupt request is enabled when this bit is 1. |
| 4 | IRQ4E | 0 | R/W | IRQ4 Enable The IRQ4 interrupt request is enabled when this bit is 1. |
| 3 | IRQ3E | 0 | R/W | IRQ3 Enable The IRQ3 interrupt request is enabled when this bit is 1. |
| 2 | IRQ2E | 0 | R/W | IRQ2 Enable The IRQ2 interrupt request is enabled when this bit is 1. |
| 1 | IRQ1E | 0 | R/W | IRQ1 Enable The IRQ1 interrupt request is enabled when this bit is 1. |
| 0 | IRQ0E | 0 | R/W | IRQ0 Enable The IRQ0 interrupt request is enabled when this bit is 1. |

5.3.3 IRQ Sense Control Registers H and L (ISCRH and ISCRL)

The ISCR registers select the source that generates an interrupt request at pins $\overline{\text{IRQn}}$ ($n = 7$ to 0). Specifiable sources are the falling edge, rising edge, or both edge detection, and level sensing.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | IRQ7SCB | 0 | R/W | IRQ7 Sense Control B |
| 14 | IRQ7SCA | 0 | R/W | IRQ7 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ7}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ7}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ7}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ7}}$ input |
| 13 | IRQ6SCB | 0 | R/W | IRQ6 Sense Control B |
| 12 | IRQ6SCA | 0 | R/W | IRQ6 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ6}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ6}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ6}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ6}}$ input |
| 11 | IRQ5SCB | 0 | R/W | IRQ5 Sense Control B |
| 10 | IRQ5SCA | 0 | R/W | IRQ5 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ5}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ5}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ5}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ5}}$ input |
| 9 | IRQ4SCB | 0 | R/W | IRQ4 Sense Control B |
| 8 | IRQ4SCA | 0 | R/W | IRQ4 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ4}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ4}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ4}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ4}}$ input |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | IRQ3SCB | 0 | R/W | IRQ3 Sense Control B |
| 6 | IRQ3SCA | 0 | R/W | IRQ3 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ3}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ3}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ3}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ3}}$ input |
| 5 | IRQ2SCB | 0 | R/W | IRQ2 Sense Control B |
| 4 | IRQ2SCA | 0 | R/W | IRQ2 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ2}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ2}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ2}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ2}}$ input |
| 3 | IRQ1SCB | 0 | R/W | IRQ1 Sense Control B |
| 2 | IRQ1SCA | 0 | R/W | IRQ1 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ1}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ1}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ1}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ1}}$ input |
| 1 | IRQ0SCB | 0 | R/W | IRQ0 Sense Control B |
| 0 | IRQ0SCA | 0 | R/W | IRQ0 Sense Control A |
| | | | | 00: Interrupt request generated at $\overline{\text{IRQ0}}$ input level low |
| | | | | 01: Interrupt request generated at falling edge of $\overline{\text{IRQ0}}$ input |
| | | | | 10: Interrupt request generated at rising edge of $\overline{\text{IRQ0}}$ input |
| | | | | 11: Interrupt request generated at both falling and rising edges of $\overline{\text{IRQ0}}$ input |

5.3.4 IRQ Status Register (ISR)

ISR indicates the status of IRQ_n (n = 7 to 0) interrupt requests.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|------|---|
| 7 | IRQ7F | 0 | R/W* | IRQ7 to IRQ0 Flags |
| 6 | IRQ6F | 0 | R/W* | Indicates the status of IRQ7 to IRQ0 interrupt requests. |
| 5 | IRQ5F | 0 | R/W* | [Setting condition] |
| 4 | IRQ4F | 0 | R/W* | When the interrupt source selected by the ISCRH, or |
| 3 | IRQ3F | 0 | R/W* | ISCRL occurs |
| 2 | IRQ2F | 0 | R/W* | [Clearing conditions] |
| 1 | IRQ1F | 0 | R/W* | <ul style="list-style-type: none">• Cleared by reading IRQ_nF flag when IRQ_nF = 1, then writing 0 to IRQ_nF flag• When interrupt exception handling is executed when low-level detection is set and $\overline{\text{IRQ}}_n$ input is high level• When IRQ_n interrupt exception handling is executed when falling, rising, or both-edge detection is set• When the DTC is activated by an IRQ_n interrupt, and the DISEL bit in MRB of the DTC is cleared to 0 |
| 0 | IRQ0F | 0 | R/W* | |
| | | | | |
| | | | | |

Note: * Only 0 can be written to this bit to clear the flag.

5.4 Interrupt Sources

5.4.1 External Interrupts

There are nine external interrupts: NMI and IRQ7 to IRQ0. These interrupts can be used to restore this LSI from software standby mode.

NMI Interrupt: NMI is the highest-priority interrupt, and is always accepted by the CPU regardless of the interrupt control mode or the status of the CPU interrupt mask bits. The NMIEG bit in SYSCR can be used to select whether an interrupt is requested at a rising edge or a falling edge on the NMI pin.

IRQn Interrupts (n = 7 to 0): IRQn interrupts are requested by an input signal at $\overline{\text{IRQn}}$ pins. IRQn interrupts have the following features:

- Using ISCR, it is possible to select whether an interrupt is generated by a low level, falling edge, rising edge, or both edges, at $\overline{\text{IRQn}}$ pins.
- Enabling or disabling of IRQn interrupt requests can be selected with IER.
- The interrupt priority level can be set with IPR.
- The status of IRQn interrupt requests is indicated in ISR. ISR flags can be cleared to 0 by software.

A block diagram of IRQn interrupts is shown in figure 5.2.

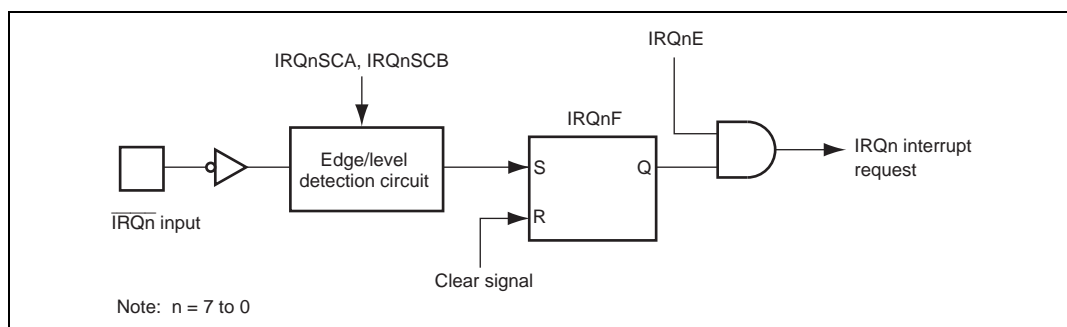


Figure 5.2 Block Diagram of IRQn Interrupts

The set timing for IRQnF is shown in figure 5.3.

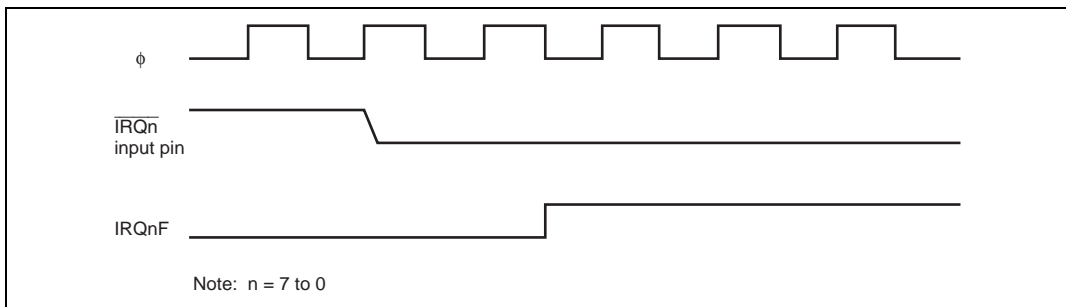


Figure 5.3 Set Timing for IRQnF

The detection of IRQn interrupts does not depend on whether the relevant pin has been set for input or output. However, when a pin is used as an external interrupt input pin, do not clear the corresponding DDR to 0; and use the pin as an I/O pin for another function. IRQnF interrupt request flag is set to 1 when the setting condition is satisfied, regardless of IER settings. Accordingly, refer to only necessary flags.

5.4.2 Internal Interrupts

Internal interrupts that are requested from the on-chip peripheral modules have the following features.

- For each on-chip peripheral module, there are flags that indicate the interrupt request status, and enable bits that select enabling or disabling of these interrupts, and they are masked independently. If the enable bit is set to 1 for a particular interrupt source, an interrupt request is issued to the interrupt controller.
- The interrupt priority level can be set with IPR.
- TPU and SCI interrupt requests can activate the DMAC* or DTC. When the DMAC* or DTC is activated by the interrupt request, the interrupt control mode and CPU interrupt mask bits are disregarded.

Note: * Supported only by the H8S/2239 Group.

5.4.3 Interrupt Exception Handling Vector Table

Table 5.2 shows interrupt exception handling sources, vector addresses, and interrupt priorities. For default priorities, the lower the vector number, the higher the priority.

Priorities among modules can be set by means of the IPR. Modules set at the same priority will conform to their default priorities. Priorities within a module are fixed.

Rev. 6.00 Mar. 18, 2010 Page 137 of 982
REJ09B0054-0600

| Interrupt Source | Origin of Interrupt Source | Vector Number | Vector Address* ¹ | IPR* ² | Priority |
|-----------------------------|---|---------------|------------------------------|-------------------|------------------------------|
| | | | Advanced Mode | | |
| TPU channel 0 | TGI0D (TGR0D input capture/compare-match) | 35 | H'008C | IPRF6 to IPRF4 | <div>↑</div> <div>High</div> |
| | TCI0V (overflow 0) | 36 | H'0090 | | |
| | Reserved | 37 | H'0094 | | |
| | | 38 | H'0098 | | |
| | | 39 | H'009C | | |
| TPU channel 1 | TGI1A (TGR1A input capture/compare-match) | 40 | H'00A0 | IPRF2 to IPRF0 | |
| | TGI1B (TGR1B input capture/compare-match) | 41 | H'00A4 | | |
| | TCI1V (overflow 1) | 42 | H'00A8 | | |
| | TCI1U (underflow 1) | 43 | H'00AC | | |
| TPU channel 2 | TGI2A (TGR2A input capture/compare-match) | 44 | H'00B0 | IPRG6 to IPRG4 | |
| | TGI2B (TGR2B input capture/compare-match) | 45 | H'00B4 | | |
| | TCI2V (overflow 2) | 46 | H'00B8 | | |
| | TCI2U (underflow 2) | 47 | H'00BC | | |
| TPU channel 3* ³ | TGI3A (TGR3A input capture/compare-match) | 48 | H'00C0 | IPRG2 to IPRG0 | |
| | TGI3B (TGR3B input capture/compare-match) | 49 | H'00C4 | | |
| | TGI3C (TGR3C input capture/compare-match) | 50 | H'00C8 | | |
| | TGI3D (TGR3D input capture/compare-match) | 51 | H'00CC | | |
| | TCI3V (overflow 3) | 52 | H'00D0 | | Low |

| Interrupt Source | Origin of Interrupt Source | Vector Number | Vector Address* ¹ | IPR* ² | Priority |
|-----------------------------|---|---------------|------------------------------|-------------------|----------------------|
| | | | Advanced Mode | | |
| — | Reserved | 53 | H'00D4 | IPRG2 to IPRG0 | High <div>↑</div> |
| | | 54 | H'00D8 | | |
| | | 55 | H'00DC | | |
| TPU channel 4* ³ | TGI4A (TGR4A input capture/compare-match) | 56 | H'00E0 | IPRH6 to IPRH4 | |
| | TGI4B (TGR4B input capture/compare-match) | 57 | H'00E4 | | |
| | TCI4V (overflow 4) | 58 | H'00E8 | | |
| | TCI4U (underflow 4) | 59 | H'00EC | | |
| TPU channel 5* ³ | TGI5A (TGR5A input capture/compare-match) | 60 | H'00F0 | IPRH2 to IPRH0 | |
| | TGI5B (TGR5B input capture/compare-match) | 61 | H'00F4 | | |
| | TCI5V (overflow 5) | 62 | H'00F8 | | |
| | TCI5U (underflow 5) | 63 | H'00FC | | |
| 8-bit timer channel 0 | CMIA0 (compare-match A0) | 64 | H'0100 | IPRI6 to IPRI4 | |
| | CMIB0 (compare-match B0) | 65 | H'0104 | | |
| | OVI0 (overflow 0) | 66 | H'0108 | | |
| — | Reserved | 67 | H'010C | | |
| 8-bit timer channel 1 | CMIA1 (compare-match A1) | 68 | H'0110 | IPRI2 to IPRI0 | |
| | CMIB1 (compare-match B1) | 69 | H'0114 | | |
| | OVI1 (overflow 1) | 70 | H'0118 | | |
| — | Reserved | 71 | H'011C | | Low |

| Interrupt Source | Origin of Interrupt Source | Vector Number | Vector Address* ¹ | IPR* ² | Priority |
|-------------------------------------|--|---------------|------------------------------|-------------------|----------|
| | | | Advanced Mode | | |
| DMAC* ⁵ | DEND0A (completion of channel 0/channel 0A transfer) | 72 | H'0120 | IPRJ6 to IPRJ4 | ↑ |
| | DEND0B (completion of channel 0B transfer) | 73 | H'0124 | | |
| | DEND1A (completion of channel 1/channel 1A transfer) | 74 | H'0128 | | |
| | DEND1B (completion of channel 1B transfer) | 75 | H'012C | | |
| SCI channel 0 | ERI0 (receive error 0) | 80 | H'0140 | IPRJ2 to IPRJ0 | |
| | RXI0 (receive completion 0) | 81 | H'0144 | | |
| | TXI0 (transmit data empty 0) | 82 | H'0148 | | |
| | TEI0 (transmit end 0) | 83 | H'014C | | |
| SCI channel 1 | ERI1 (receive error 1) | 84 | H'0150 | IPRK6 to IPRK4 | |
| | RXI1 (receive completion 1) | 85 | H'0154 | | |
| | TXI1 (transmit data empty 1) | 86 | H'0158 | | |
| | TEI1 (transmit end 1) | 87 | H'015C | | |
| SCI channel 2* ³ | ERI2 (receive error 2) | 88 | H'0160 | IPRK2 to IPRK0 | |
| | RXI2 (receive completion 2) | 89 | H'0164 | | |
| | TXI2 (transmit data empty 2) | 90 | H'0168 | | |
| | TEI2 (transmit end 2) | 91 | H'016C | | |
| 8-bit timer channel 2* ⁴ | CMA2 (compare-match A2) | 92 | H'0170 | IPRL6 to IPRL4 | |
| | CMIB2 (compare-match B2) | 93 | H'0174 | | |
| | OVI2 (overflow 2) | 94 | H'0178 | | |
| — | Reserved | 95 | H'017C | | Low |

| Interrupt Source | Origin of Interrupt Source | Vector Number | Vector Address* ¹ | IPR* ² | Priority |
|-------------------------------------|----------------------------|---------------|------------------------------|-------------------|---|
| | | | Advanced Mode | | |
| 8-bit timer channel 3* ⁴ | CMIA3 (compare-match A3) | 96 | H'0180 | IPRL6 to IPRL4 | High < |

- Notes:
1. Lower 16 bits of the start address.
 2. IPR6 to IPR4, and IPR2 to IPR0 bits are reserved, because these bits have no corresponding interruption. These bits are always read as 0 and cannot be modified.
 3. Not available in the H8S/2227 Group.
 4. Not available in the H8S/2237 Group and H8S/2227 Group.
 5. Supported only by the H8S/2239 Group.
 6. Supported only by the H8S/2258 Group.

5.5 Operation

5.5.1 Interrupt Control Modes and Interrupt Operation

Interrupt operations in this LSI differ depending on the interrupt control mode.

NMI interrupts are accepted at all times except in the reset state and the hardware standby state. In the case of IRQ interrupts and on-chip peripheral module interrupts, an enable bit is provided for each interrupt. Clearing an enable bit to 0 disables the corresponding interrupt request. Interrupt sources for which the enable bits are set to 1 are controlled by the interrupt controller.

Table 5.3 shows the interrupt control modes.

The interrupt controller performs interrupt control according to the interrupt control mode set by the INTM1 and INTM0 bits in SYSCR, the priorities set in IPR, and the masking state indicated by the I bit in the CPU's CCR, and bits I2 to I0 in EXR.

Table 5.3 Interrupt Control Modes

| Interrupt Control Mode | SYSCR | | Priority Setting Registers | Interrupt Mask Bits | Description |
|---------------------------|-------|-------|-------------------------------|------------------------|--|
| | INTM1 | INTM0 | | | |
| 0 | 0 | 0 | — | I | Interrupt mask control is performed by the I bit. |
| — | | 1 | — | — | Setting prohibited |
| 2 | 1 | 0 | IPR | I2 to I0 | 8-level interrupt mask control is performed by bits I2 to I0. 8 priority levels can be set with IPR. |
| — | | 1 | — | — | Setting prohibited |

Figure 5.4 shows the block diagram of the priority decision circuits.

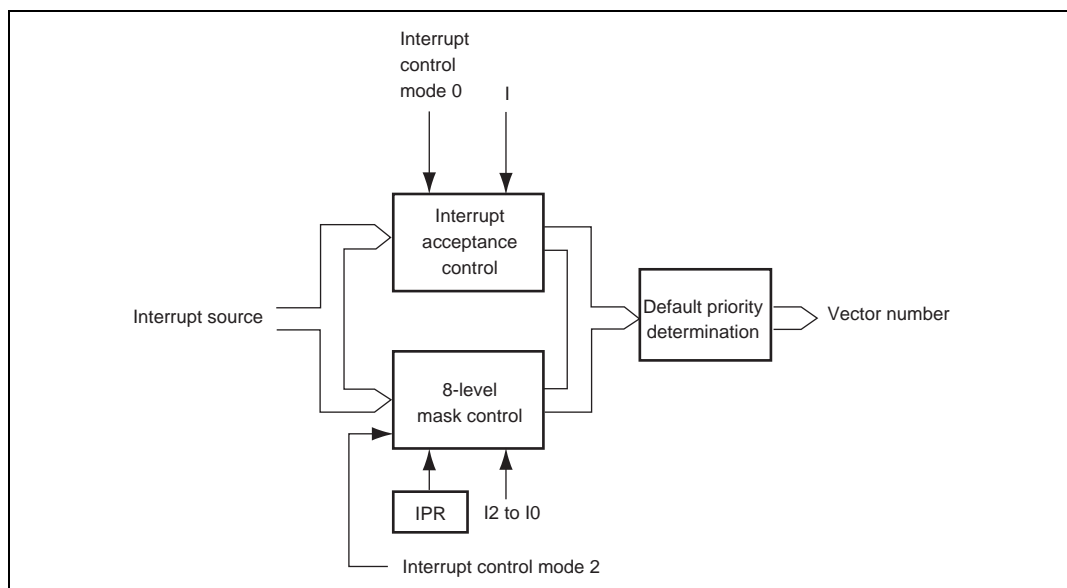


Figure 5.4 Block Diagram of Interrupt Control Operation

Interrupt Acceptance Control: In interrupt control mode 0, interrupt acceptance is controlled by the I bit in CCR.

Table 5.4 shows the interrupts selected in each interrupt control mode.

Table 5.4 Interrupts Selected in Each Interrupt Control Mode (1)

| Interrupt Control Mode | Interrupt Mask Bits | |
|------------------------|---------------------|---------------------|
| | I | Selected Interrupts |
| 0 | 0 | All interrupts |
| | 1 | NMI interrupts |
| 2 | × | All interrupts |

Legend: ×: Don't care

8-Level Control: In interrupt control mode 2, 8-level mask level determination is performed for the selected interrupts in interrupt acceptance control according to the interrupt priority level (IPR).

The interrupt source selected is the interrupt with the highest priority level, and whose priority level set in IPR is higher than the mask level.

Table 5.5 Interrupts Selected in Each Interrupt Control Mode (2)

| Interrupt Control Mode | Selected Interrupts |
|------------------------|--|
| 0 | All interrupts |
| 2 | Highest-priority-level (IPR) interrupt whose priority level is greater than the mask level (IPR > I2 to I0). |

Default Priority Determination: When an interrupt is selected by 8-level control, its priority is determined and a vector number is generated.

If the same value is set for IPR, acceptance of multiple interrupts is enabled, and so only the interrupt source with the highest priority according to the preset default priorities is selected and has a vector number generated.

Interrupt sources with a lower priority than the accepted interrupt source are held pending.

Table 5.6 shows operations and control signal functions in each interrupt control mode.

Table 5.6 Operations and Control Signal Functions in Each Interrupt Control Mode

| Interrupt Control Mode | Setting | | Interrupt Acceptance Control | 8-Level Control | | | Default Priority Determination | T (Trace) |
|------------------------|---------|-------|------------------------------|-----------------|----------|-----|--------------------------------|-----------|
| | INTM1 | INTM0 | | I | I2 to I0 | IPR | | |
| 0 | 0 | 0 | O | IM | X | — | —*2 | — |
| 2 | 1 | 0 | X | —*1 | O | IM | PR | T |

Legend:

O: Interrupt operation control performed.

X: No operation (All interrupts enabled).

IM: Used as interrupt mask bit.

PR: Sets priority.

—: Not used.

Notes: 1. Set to 1 when interrupt is accepted.

2. Keep the initial setting.

5.5.2 Interrupt Control Mode 0

Enabling and disabling of IRQ interrupts, IRQ interrupts and on-chip peripheral module interrupts can be set by means of the I bit in the CPU's CCR. Interrupts are enabled when the I bit is cleared to 0, and disabled when set to 1.

Figure 5.5 shows a flowchart of the interrupt acceptance operation in this case.

1. If an interrupt source occurs when the corresponding interrupt enable bit is set to 1, an interrupt request is sent to the interrupt controller.
2. If the I bit is set to 1, only an NMI interrupt is accepted, and other interrupt requests are held pending. If the I bit is cleared, an interrupt request is accepted.
3. Interrupt requests are sent to the interrupt controller, the highest-ranked interrupt according to the priority system is accepted, and other interrupt requests are held pending.
4. When the CPU accepts an interrupt request, it starts interrupt exception handling after execution of the current instruction has been completed.
5. The PC and CCR are saved to the stack area by interrupt exception handling. The PC saved on the stack shows the address of the first instruction to be executed after returning from the interrupt handling routine.
6. Next, the I bit in CCR is set to 1. This masks all interrupts except NMI.
7. The CPU generates a vector address for the accepted interrupt and starts execution of the interrupt handling routine at the address indicated by the contents of the vector address in the vector table.

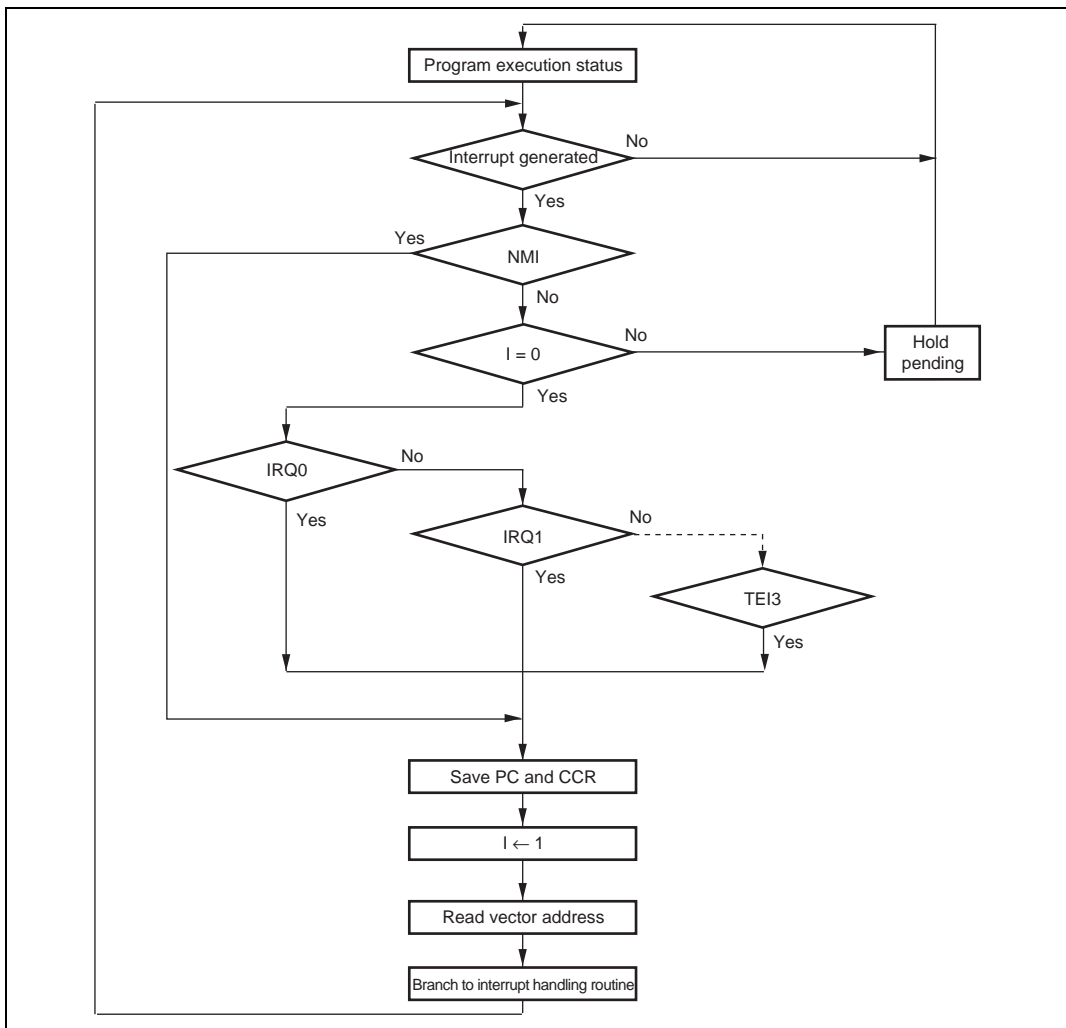


Figure 5.5 Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 0

5.5.3 Interrupt Control Mode 2

Eight-level masking is implemented for IRQ interrupts, and on-chip peripheral module interrupts by comparing the interrupt mask level set by bits I2 to I0 of EXR in the CPU with IPR.

Figure 5.6 shows a flowchart of the interrupt acceptance operation in this case.

1. If an interrupt source occurs when the corresponding interrupt enable bit is set to 1, an interrupt request is sent to the interrupt controller.
2. When interrupt requests are sent to the interrupt controller, the interrupt with the highest priority according to the interrupt priority levels set in IPR is selected, and lower-priority interrupt requests are held pending. If a number of interrupt requests with the same priority are generated at the same time, the interrupt request with the highest priority according to the priority system shown in table 5.2 is selected.
3. Next, the priority of the selected interrupt request is compared with the interrupt mask level set in EXR. An interrupt request with a priority no higher than the mask level set at that time is held pending, and only an interrupt request with a priority higher than the interrupt mask level is accepted.
4. When the CPU accepts an interrupt request, it starts interrupt exception handling after execution of the current instruction has been completed.
5. The PC, CCR, and EXR are saved to the stack area by interrupt exception handling. The PC saved on the stack shows the address of the first instruction to be executed after returning from the interrupt handling routine.
6. The T bit in EXR is cleared to 0. The interrupt mask level is rewritten with the priority level of the accepted interrupt.
If the accepted interrupt is NMI, the interrupt mask level is set to H'7.
7. The CPU generates a vector address for the accepted interrupt and starts execution of the interrupt handling routine at the address indicated by the contents of the vector address in the vector table.

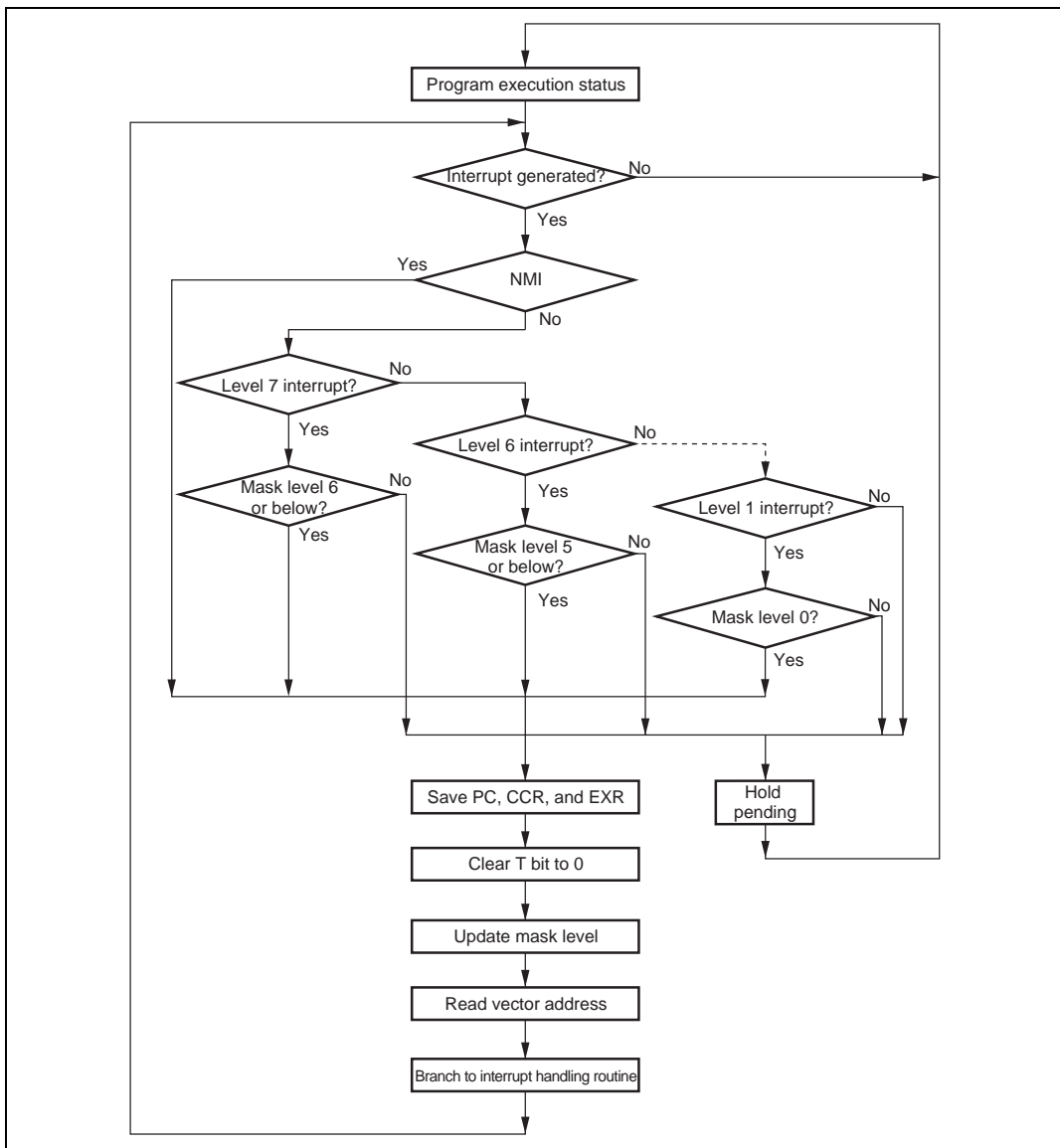


Figure 5.6 Flowchart of Procedure Up to Interrupt Acceptance in Control Mode 2

5.5.4 Interrupt Exception Handling Sequence

Figure 5.7 shows the interrupt exception handling sequence. The example shown is for the case where interrupt control mode 0 is set in advanced mode, and the program area and stack area are in on-chip memory.

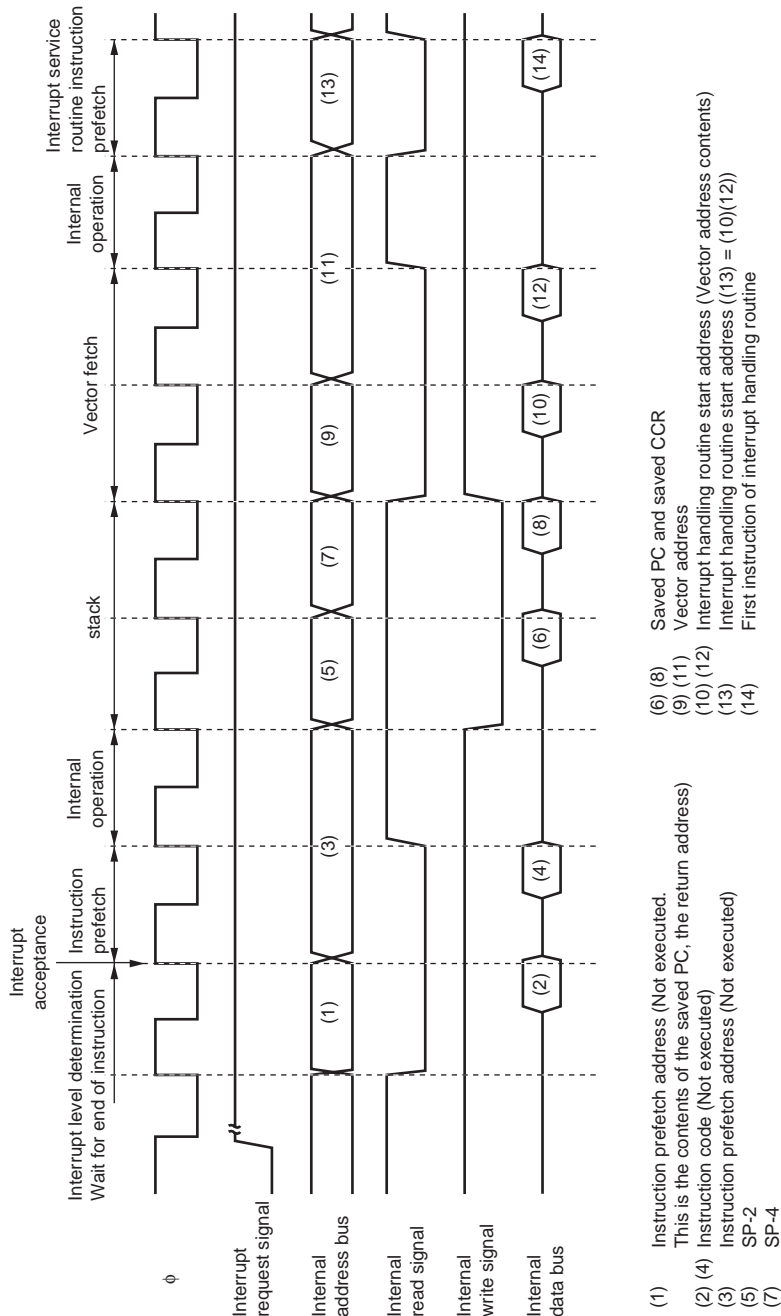


Figure 5.7 Interrupt Exception Handling

5.5.5 Interrupt Response Times

This LSI is capable of fast word transfer to on-chip memory, has the program area in on-chip ROM and the stack area in on-chip RAM, enabling high-speed processing.

Table 5.7 shows interrupt response times—the interval between generation of an interrupt request and execution of the first instruction in the interrupt handling routine. The execution status symbols used in table 5.7 are explained in table 5.8.

Table 5.7 Interrupt Response Times

| No. | Execution Status | Normal Mode* ⁵ | | Advanced Mode | |
|------------------------------|--|----------------------------|----------------------------|----------------------------|----------------------------|
| | | INTM1 = 0 | INTM1 = 1 | INTM1 = 0 | INTM1 = 1 |
| 1 | Interrupt priority determination* ¹ | 3 | 3 | 3 | 3 |
| 2 | Number of wait states until executing instruction ends* ² | 1 to 19 + 2·S _I | 1 to 19 + 2·S _I | 1 to 19 + 2·S _I | 1 to 19 + 2·S _I |
| 3 | PC, CCR, EXR stack save | 2·S _K | 3·S _K | 2·S _K | 3·S _K |
| 4 | Vector fetch | S _I | S _I | 2·S _I | 2·S _I |
| 5 | Instruction fetch* ³ | 2·S _I | 2·S _I | 2·S _I | 2·S _I |
| 6 | Internal processing* ⁴ | 2 | 2 | 2 | 2 |
| Total (using on-chip memory) | | 11 to 31 | 12 to 32 | 12 to 32 | 13 to 33 |

Notes: 1. Two states in case of internal interrupt.

2. Refers to MULXS and DIVXS instructions.

3. Prefetch after interrupt acceptance and interrupt handling routine prefetch.

4. Internal processing after interrupt acceptance and internal processing after vector fetch.

5. Not available in this LSI.

Table 5.8 Number of States in Interrupt Handling Routine Execution Status

| | | | Object of Access | | | |
|---------------------|-------|-----------------|------------------|----------------|----------------|----------------|
| | | | External Device | | | |
| | | | 8 Bit Bus | | 16 Bit Bus | |
| | | | 2-State Access | 3-State Access | 2-State Access | 3-State Access |
| Symbol | | Internal Memory | | | | |
| Instruction fetch | S_i | 1 | 4 | 6 + 2 m | 2 | 3 + m |
| Branch address read | S_j | | | | | |
| Stack manipulation | S_k | | | | | |

Legend:

m: Number of wait states in an external device access.

5.5.6 DTC and DMAC* Activation by Interrupt

The DTC and DMAC* can be started by interrupts. The following settings are required for this operation.

1. Interrupt request to the CPU
2. Start request to the DTC
3. Start request to the DMAC*
4. Multiple specification of items 1 to 3.

See section 8, DMA Controller (DMAC)*, and section 9, Data Transfer Controller (DTC) for more information on the interrupts that can start the DTC and DMAC*.

Figure 5.8 shows the block diagram of the DTC, DMAC*, and interrupt controller circuits.

Note: * Supported only by the H8S/2239 Group.

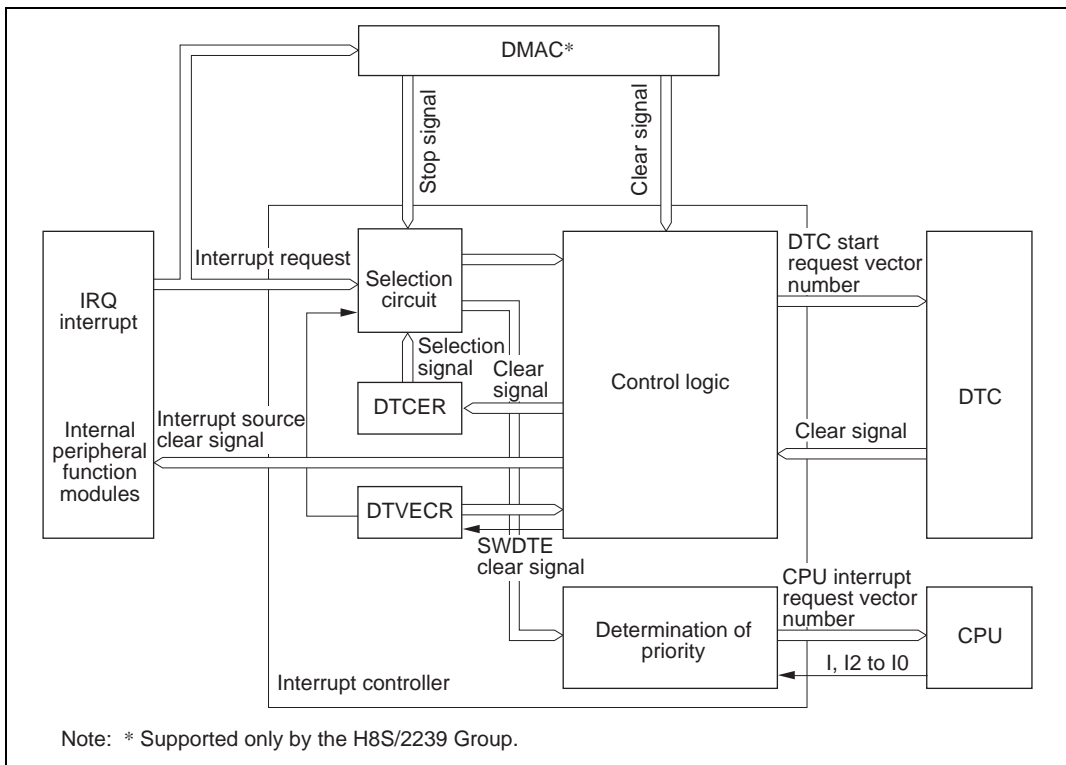


Figure 5.8 DTC and DMAC* Interrupt Control

(1) Interrupt Source Selection

The DMAC* startup sources are directly input to each channel. The startup source for each DMAC* channel is selected by the DMACR DTF3 to DTF0 bits. Whether or not the selected startup source is managed by the DMAC* can be selected with the DMABCR DTA bit. If the DTA bit is set to 1, the interrupt source that has become the DMAC* startup source will not be either a DTC startup source or a CPU interrupt source.

Interrupt sources other than the interrupt managed by the DMAC* are selected to be DTC startup sources or CPU interrupt requests by the DTC DTCERA to DTCERF DTCE bits.

After a DTC data transfer, a CPU interrupt can be requested by clearing the DTCE bit to 0 by specifying that with the DTC MRB DIESEL bit.

Note that when the DTC has performed the stipulated number of data transfers and the transfer counter has become 0, the DTCE bit can be cleared to 0 and a CPU interrupt can be requested.

Note: * Supported only by the H8S/2239 Group.

(2) Determination of priority

The DTC startup source is selected according to the default priority. This is not influenced by the mask level or the priority level. See section 9.4, Location of Register Information and DTC Vector Table, for details on these priorities.

The startup sources are directly input to each channel in the DMAC*.

Note: * Supported only by the H8S/2239 Group.

(3) Operating Sequence

When the same interrupt is selected as both the DTC startup source and a CPU interrupt source, the DTC data transfer is performed and then the CPU interrupt exception handling is performed.

When the same interrupt is selected as both the DMAC* startup source and either the DTC startup source or a CPU interrupt source, the operations are performed independently. They are performed according to the operating states and the bus priorities.

Table 5.9 shows the interrupt source selection and the interrupt source clear control according to the settings of the DMAC* DMABCR DTA bit, the DTC DTCERA to DTCERF DTCE bits, and the DTC MRB DISEL bit.

Note: * Supported only by the H8S/2239 Group.

Table 5.9 Interrupt Source Selection and Clear Control

| Settings | | | Interrupt source selection and clear control | | |
|--------------------|------|-------|--|-----|-----|
| DMAC* ¹ | DTC | | | | |
| DTA | DTCE | DISEL | DMAC* ¹ | DTC | CPU |
| 0 | 0 | * | ○ | × | ⊙ |
| | 1 | 0 | ○ | ⊙ | × |
| | | 1 | ○ | ○ | ⊙ |
| 1 | * | * | ⊙ | × | × |

Legend:

- ⊙: The corresponding interrupt is used. The interrupt source is cleared.
(The CPU must clear the source flag in the interrupt handler.)
- : The corresponding interrupt is used. The interrupt source is not cleared.
- ×: The corresponding interrupt is not used.
- *: Don't care

Note: 1. Supported only by the H8S/2239 Group.

(4) Usage Notes

The SCI and A/D converter interrupt sources are cleared when the DMAC* or DTC reads or writes the stipulated register. This does not depend on the DTA, DTCE, and DISSEL bits.

Note: * Supported only by the H8S/2239 Group.

5.6 Usage Notes

5.6.1 Contention between Interrupt Generation and Disabling

When an interrupt enable bit is cleared to 0 to disable interrupt requests, the disabling becomes effective after execution of the instruction.

When an interrupt enable bit is cleared to 0 by an instruction such as BCLR or MOV, and if an interrupt is generated during execution of the instruction, the interrupt concerned will still be enabled on completion of the instruction, and so interrupt exception handling for that interrupt will be executed on completion of the instruction. However, if there is an interrupt request of higher priority than that interrupt, interrupt exception handling will be executed for the higher-priority interrupt, and the lower-priority interrupt will be ignored.

The same also applies when an interrupt source flag is cleared to 0.

Figure 5.9 shows an example in which the CMIEA bit in the TCR register of the 8-bit timer is cleared to 0.

The above contention will not occur if an enable bit or interrupt source flag is cleared to 0 while the interrupt is masked.

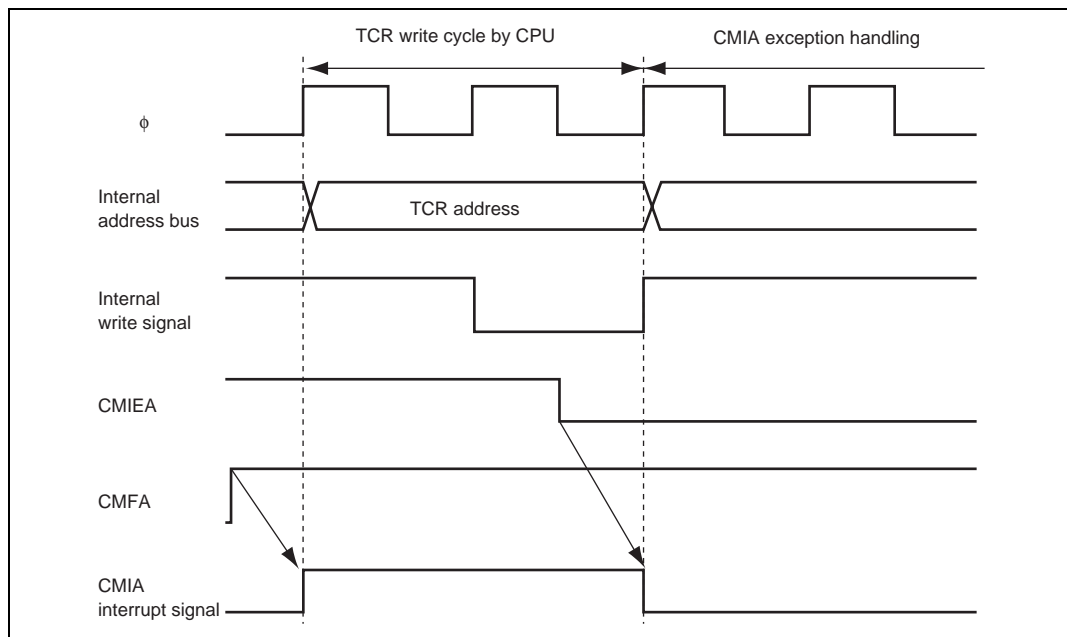


Figure 5.9 Contention between Interrupt Generation and Disabling

5.6.2 Instructions that Disable Interrupts

The instructions that disable interrupts are LDC, ANDC, ORC, and XORC. After any of these instructions are executed, all interrupts including NMI are disabled and the next instruction is always executed. When the I bit is set by one of these instructions, the new value becomes valid two states after execution of the instruction ends.

5.6.3 When Interrupts are Disabled

There are times when interrupt acceptance is disabled by the interrupt controller.

The interrupt controller disables interrupt acceptance for a 3-state period after the CPU has updated the mask level with an LDC, ANDC, ORC, or XORC instruction.

5.6.4 Interrupts during Execution of EEPMOV Instruction

Interrupt operation differs between the EEPMOV.B instruction and the EEPMOV.W instruction.

With the EEPMOV.B instruction, an interrupt request (including NMI) issued during the transfer is not accepted until the move is completed.

With the EEPMOV.W instruction, if an interrupt request is issued during the transfer, interrupt exception handling starts at a break in the transfer cycle. The PC value saved on the stack in this case is the address of the next instruction.

Therefore, if an interrupt is generated during execution of an EEPMOV.W instruction, the following coding should be used.

```
L1: EEPMOV.W
MOV.W      R4, R4
BNE        L1
```

5.6.5 IRQ Interrupt

When operating by clock input, acceptance of input to an IRQ is synchronized with the clock. In software standby mode, watch mode, subactive mode and subsleep mode, the input is accepted asynchronously. For details on the input conditions, see Operating Timing in section 27, Electrical Characteristics.

5.6.6 NMI Interrupts Usage Notes

The NMI interrupt is part of the exception processing performed cooperatively by the LSI's internal interrupt controller and the CPU when the system is operating normally under the specified electrical conditions. No operations, including NMI interrupts, are guaranteed when operation is not normal (runaway status) due to software problems or abnormal input to the LSI's pins. In such cases, the LSI may be restored to the normal program execution state by applying an external reset.

Section 6 PC Break Controller (PBC)

The PC break controller (PBC) provides functions that simplify program debugging. Using these functions, it is easy to create a self-monitoring debugger, enabling programs to be debugged with the chip alone, without using an in-circuit emulator. A block diagram of the PC break controller is shown in figure 6.1.

6.1 Features

- Two break channels (A and B)
- 24-bit break address
 - Bit masking possible
- Four types of break compare conditions
 - Instruction fetch
 - Data read
 - Data write
 - Data read/write
- Bus master
 - Either CPU or CPU/DTC can be selected
- The timing of PC break exception handling after the occurrence of a break condition is as follows:
 - Immediately before execution of the instruction fetched at the set address (instruction fetch)
 - Immediately after execution of the instruction that accesses data at the set address (data access)
- Module stop mode can be set

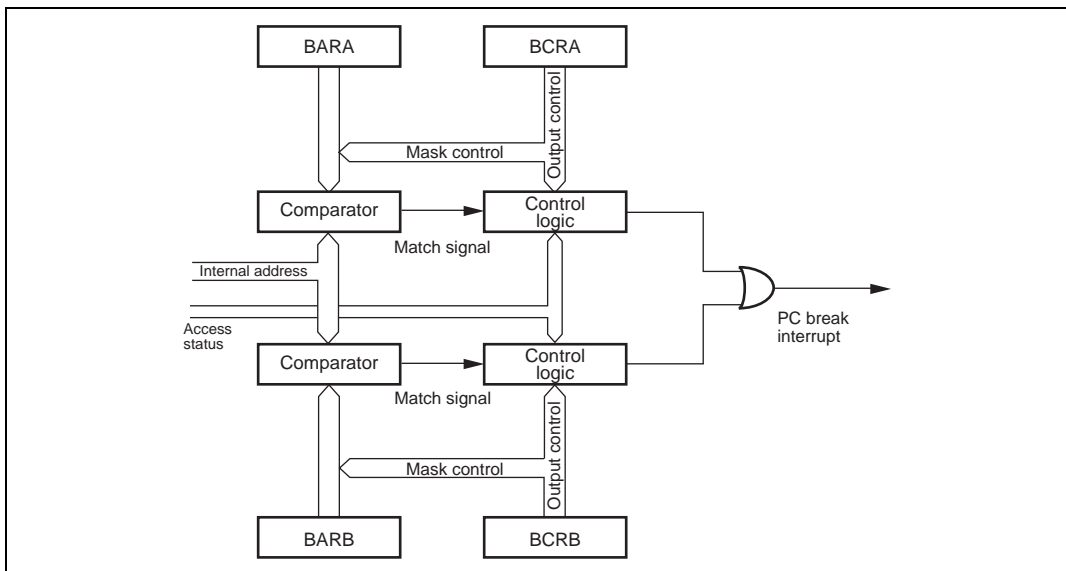


Figure 6.1 Block Diagram of PC Break Controller

6.2 Register Descriptions

The PC break controller has the following registers.

- Break address register A (BARA)
- Break address register B (BARB)
- Break control register A (BCRA)
- Break control register B (BCRB)

6.2.1 Break Address Register A (BARA)

BARA is a 32-bit readable/writable register that specifies the channel A break address.

| Bit | Bit Name | Initial Value | R/W | Description |
|----------|---------------|---------------|-----|---|
| 31 to 24 | — | Undefined | — | Reserved These bits are read as an undefined value and cannot be modified. |
| 23 to 0 | BAA23 to BAA0 | All 0 | R/W | Break Address 23 to 0 These bits set the channel A PC break address. |

6.2.2 Break Address Register B (BARB)

BARB is the channel B break address register. The bit configuration is the same as for BARA.

6.2.3 Break Control Register A (BCRA)

BCRA controls channel A PC breaks.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|---|
| 7 | CMFA | 0 | R/(W) ^{*1} | Condition Match Flag A [Setting condition] When a condition set for channel A is satisfied [Clearing condition] When 0 is written to CMFA after reading ^{*2} CMFA = 1 |
| 6 | CDA | 0 | R/W | CPU Cycle/DTC Cycle Select A Selects the channel A break condition bus master. 0: CPU 1: CPU, DTC, or DMAC ^{*3} |
| 5 | BAMRA2 | 0 | R/W | Break Address Mask Register A2 to A0 These bits specify which bits of the break address set in BARA are to be masked. 000: BAA23 to 0 (All bits are unmasked) 001: BAA23 to 1 (Lowest bit is masked) 010: BAA23 to 2 (Lower 2 bits are masked) 011: BAA23 to 3 (Lower 3 bits are masked) 100: BAA23 to 4 (Lower 4 bits are masked) 101: BAA23 to 8 (Lower 8 bits are masked) 110: BAA23 to 12 (Lower 12 bits are masked) 111: BAA23 to 16 (Lower 16 bits are masked) |
| 4 | BAMRA1 | 0 | R/W | |
| 3 | BAMRA0 | 0 | R/W | |
| 2 | CSELA1 | 0 | R/W | |
| 1 | CSELA0 | 0 | R/W | |
| | | | | Break Condition Select Selects break condition of channel A. 00: Instruction fetch 01: Data read cycle 10: Data write cycle 11: Data read/write cycle |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | BIEA | 0 | R/W | Break Interrupt Enable When this bit is 1, the PC break interrupt request of channel A is enabled. |

- Notes:
1. Only a 0 can be written to this bit to clear the flag.
 2. Read the state wherein CMFA = 1 twice or more, when the CMFA is polled after inhibiting the PC break interruption.
 3. Supported only by the H8S/2239 Group.

6.2.4 Break Control Register B (BCRB)

BCRB is the channel B break control register. The bit configuration is the same as for BCRA.

6.3 Operation

The operation flow from break condition setting to PC break interrupt exception handling is shown in section 6.3.1, PC Break Interrupt Due to Instruction Fetch, and section 6.3.2, PC Break Interrupt Due to Data Access, taking the example of channel A.

6.3.1 PC Break Interrupt Due to Instruction Fetch

1. Set the break address in BARA.
For a PC break caused by an instruction fetch, set the address of the first instruction byte as the break address.
2. Set the break conditions in BCRA.
Set bit 6 (CDA) to 0 to select the CPU because the bus master must be the CPU for a PC break caused by an instruction fetch. Set the address bits to be masked to bits 5 to 3 (BAMRA2 to 0). Set bits 2 and 1 (CSELA1 and 0) to 00 to specify an instruction fetch as the break condition. Set bit 0 (BIEA) to 1 to enable break interrupts.
3. When the instruction at the set address is fetched, a PC break request is generated immediately before execution of the fetched instruction, and the condition match flag (CMFA) is set.
4. After priority determination by the interrupt controller, PC break interrupt exception handling is started.

6.3.2 PC Break Interrupt Due to Data Access

1. Set the break address in BARA.

For a PC break caused by a data access, set the target ROM, RAM, I/O, or external address space address as the break address. Stack operations and branch address reads are included in data accesses.

2. Set the break conditions in BCRA.

Select the bus master with bit 6 (CDA). Set the address bits to be masked to bits 5 to 3 (BAMRA2 to 0). Set bits 2 and 1 (CSELA1 and 0) to 01, 10, or 11 to specify data access as the break condition. Set bit 0 (BIEA) to 1 to enable break interrupts.

3. After execution of the instruction that performs a data access on the set address, a PC break request is generated and the condition match flag (CMFA) is set.
4. After priority determination by the interrupt controller, PC break interrupt exception handling is started.

6.3.3 Notes on PC Break Interrupt Handling

- When a PC break interrupt is generated at the transfer address of an EEPMOV.B instruction PC break exception handling is executed after all data transfers have been completed and the EEPMOV.B instruction has ended.
- When a PC break interrupt is generated at a DTC transfer address PC break exception handling is executed after the DTC has completed the specified number of data transfers, or after data for which the DISEL bit is set to 1 has been transferred.

6.3.4 Operation in Transitions to Power-Down Modes

The operation when a PC break interrupt is set for an instruction fetch at the address after a SLEEP instruction is shown below.

- When the SLEEP instruction causes a transition from high-speed (medium-speed) mode to sleep mode, or from subactive mode to subsleep mode:
After execution of the SLEEP instruction, a transition is not made to sleep mode or subsleep mode, and PC break interrupt handling is executed. After execution of PC break interrupt handling, the instruction at the address after the SLEEP instruction is executed (figure 6.2 (A)).
- When the SLEEP instruction causes a transition from high speed mode to subactive mode (figure 6.2 (B)).
- When the SLEEP instruction causes a transition from subactive mode to high speed (medium speed) mode (figure 6.2 (C)).

- When the SLEEP instruction causes a transition to software standby mode or watch mode:
After execution of the SLEEP instruction, a transition is made to the respective mode, and PC break interrupt handling is not executed. However, the CMFA or CMFB flag is set (figure 6.2 (D)).

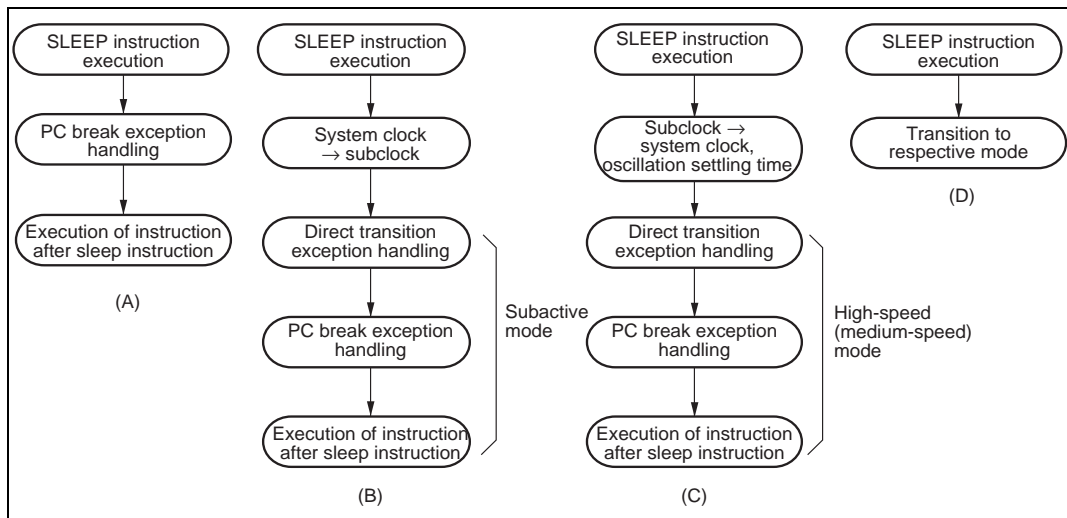


Figure 6.2 Operation in Power-Down Mode Transitions

6.3.5 When Instruction Execution Is Delayed by One State

While the break interrupt enable bit is set to 1, instruction execution is one state later than usual.

- For 1-word branch instructions (Bcc d:8, BSR, JSR, JMP, TRAPA, RTE, and RTS) in on-chip ROM or RAM.
- When break interruption by instruction fetch is set, the set address indicates on-chip ROM or RAM space, and that address is used for data access, the instruction that executes the data access is one state later than in normal operation.
- When break interruption by instruction fetch is set and a break interrupt is generated, if the executing instruction immediately preceding the set instruction has one of the addressing modes shown below, and that address indicates on-chip ROM or RAM, the instruction will be one state later than in normal operation.

Addressing modes: @ERn, @(d:16,ERn), @(d:32,ERn), @-ERn/ERn+, @aa:8, @aa:24, @aa:32, @(d:8,PC), @(d:16,PC), @@aa:8

- When break interruption by instruction fetch is set and a break interrupt is generated, if the executing instruction immediately preceding the set instruction is NOP or SLEEP, or has #xx,

Rn as its addressing mode, and that instruction is located in on-chip ROM or RAM, the instruction will be one state later than in normal operation.

6.4 Usage Notes

6.4.1 Module Stop Mode Setting

PBC operation can be disabled or enabled using the module stop control register. The initial setting is for PBC operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 24, Power-Down Modes.

6.4.2 PC Break Interrupts

The PC break interrupt is shared by channels A and B. The channel from which the request was issued must be determined by the interrupt handler.

6.4.3 CMFA and CMFB

The CMFA and CMFB flags are not automatically cleared to 0, so 0 must be written to CMFA or CMFB after first reading the flag while it is set to 1. If the flag is left set to 1, another interrupt will be requested after interrupt handling ends.

6.4.4 PC Break Interrupt when DTC and DMAC* Is Bus Master

A PC break interrupt generated when the DTC and DMAC* is the bus master is accepted after the bus has been transferred to the CPU by the bus controller.

Note: * Supported only by the H8S/2239 Group.

6.4.5 PC Break Set for Instruction Fetch at Address Following BSR, JSR, JMP, TRAPA, RTE, and RTS Instruction

Even if the instruction at the address following a BSR, JSR, JMP, TRAPA, RTE, or RTS instruction is fetched, it is not executed, and so a PC break interrupt is not generated by the instruction fetch at the next address.

6.4.6 I Bit Set by LDC, ANDC, ORC, and XORC Instruction

When the I bit is set by an LDC, ANDC, ORC, and XORC instruction, a PC break interrupt becomes valid two states after the end of the executing instruction. If a PC break interrupt is set for the instruction following one of these instructions, since interrupts, including NMI, are disabled for a 3-state period in the case of LDC, ANDC, ORC, and XOR, the next instruction is always executed. For details, see section 5, Interrupt Controller.

6.4.7 PC Break Set for Instruction Fetch at Address Following Bcc Instruction

When a PC break is set for an instruction fetch at an address following a Bcc instruction:

A PC break interrupt is generated if the instruction at the next address is executed in accordance with the branch condition, and is not generated if the instruction at the next address is not executed.

6.4.8 PC Break Set for Instruction Fetch at Branch Destination Address of Bcc Instruction

A PC break interrupt is generated if the instruction at the branch destination is executed in accordance with the branch condition, and is not generated if the instruction at the branch destination is not executed.

Section 7 Bus Controller

This LSI has a built-in bus controller (BSC) that manages the external address space divided into eight areas. The bus controller also has a bus arbitration function, and controls the operation of the internal bus masters: the CPU, DMA controller (DMAC)*, and data transfer controller (DTC).

Note: * Supported only by the H8S/2239 Group.

7.1 Features

- Manages external address space in area units
 - Manages the external space as 8 areas of 2-Mbytes
 - Bus specifications can be set independently for each area
 - Burst ROM interface can be set
- Basic bus interface
 - Chip select ($\overline{CS7}$ to $\overline{CS0}$) can be output for areas 7 to 0
 - 8-bit access or 16-bit access can be selected for each area
 - 2-state access or 3-state access can be selected for each area
 - Program wait states can be inserted for each area
- Burst ROM interface
 - Burst ROM interface can be selected for area 0
 - One or two states can be selected for the burst cycle
- Idle cycle insertion
 - Idle cycle can be inserted between consecutive read accesses to different areas
 - Idle cycle can be inserted before a write access to an external area immediately after a read access to an external area
- Bus arbitration
 - The on-chip bus arbiter arbitrates bus mastership among CPU, DMAC*, and DTC.
- Other features
 - External bus release function

Note: * Supported only by the H8S/2239 Group.

Figure 7.1 shows a block diagram of the bus controller.

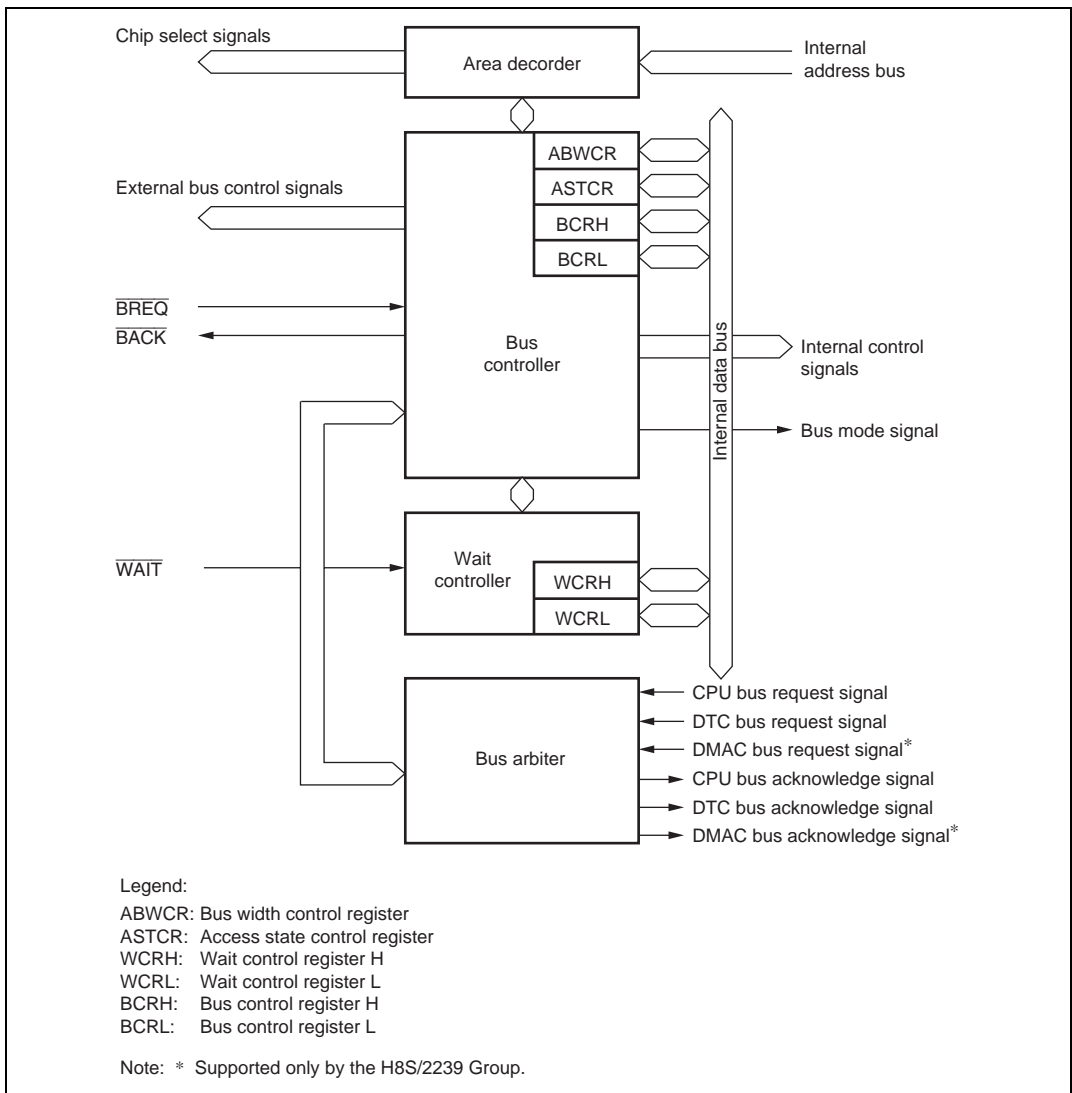


Figure 7.1 Block Diagram of Bus Controller

7.2 Input/Output Pins

Table 7.1 summarizes the pins of the bus controller.

Table 7.1 Pin Configuration

| Name | Symbol | I/O | Function |
|-------------------------|--------------------------------------|--------|---|
| Address strobe | \overline{AS} | Output | Strobe signal indicating that address output on address bus is enabled. |
| Read | \overline{RD} | Output | Strobe signal indicating that external space is being read. |
| High write | \overline{HWR} | Output | Strobe signal indicating that external space is to be written, and upper half (D15 to D8) of data bus is enabled. |
| Low write | \overline{LWR} | Output | Strobe signal indicating that external space is to be written, and lower half (D7 to D0) of data bus is enabled. |
| Chip select 7 to 0 | $\overline{CS7}$ to $\overline{CS0}$ | Output | Strobe signal indicating that areas 7 to 0 are selected. |
| Wait | \overline{WAIT} | Input | Wait request signal when accessing external 3-state access space. |
| Bus request | \overline{BREQ} | Input | Request signal that releases bus to external device. |
| Bus request acknowledge | \overline{BACK} | Output | Acknowledge signal indicating that bus has been released. |

7.3 Register Descriptions

The following shows the registers of the bus controller.

- Bus width control register (ABWCR)
- Access state control register (ASTCR)
- Wait control register H (WCRH)
- Wait control register L (WCRL)
- Bus control register H (BCRH)
- Bus control register L (BCRL)
- Pin function control register (PFCR)

7.3.1 Bus Width Control Register (ABWCR)

ABWCR designates each area for either 8-bit access or 16-bit access.

ABWCR sets the data bus width for the external memory space. The bus width for on-chip memory and internal I/O registers is fixed regardless of the settings in ABWCR.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | ABW7 | 1/0* | R/W | Area 7 to 0 Bus Width Control |
| 6 | ABW6 | 1/0* | R/W | These bits select whether the corresponding area is to be designated for 8-bit access or 16-bit access. 0: Area n is designated for 16-bit access 1: Area n is designated for 8-bit access Note: n = 7 to 0 |
| 5 | ABW5 | 1/0* | R/W | |
| 4 | ABW4 | 1/0* | R/W | |
| 3 | ABW3 | 1/0* | R/W | |
| 2 | ABW2 | 1/0* | R/W | |
| 1 | ABW1 | 1/0* | R/W | |
| 0 | ABW0 | 1/0* | R/W | |

Note: * In modes 5 to 7, initial value of each bit is 1. In mode 4, initial value of each bit is 0.

7.3.2 Access State Control Register (ASTCR)

ASTCR designates each area as either a 2-state access space or a 3-state access space.

ASTCR sets the number of access states for the external memory space. The number of access states for on-chip memory and internal I/O registers is fixed regardless of the settings in ASTCR.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | AST7 | 1 | R/W | Area 7 to 0 Access State Control |
| 6 | AST6 | 1 | R/W | These bits select whether the corresponding area is to be designated as a 2-state access space or a 3-state access space. Wait state insertion is enabled or disabled at the same time. 0: Area n is designated for 2-state access Wait state insertion in area n external space is disabled 1: Area n is designated for 3-state access Wait state insertion in area n external space is enabled Note: n = 7 to 0 |
| 5 | AST5 | 1 | R/W | |
| 4 | AST4 | 1 | R/W | |
| 3 | AST3 | 1 | R/W | |
| 2 | AST2 | 1 | R/W | |
| 1 | AST1 | 1 | R/W | |
| 0 | AST0 | 1 | R/W | |

7.3.3 Wait Control Registers H and L (WCRH, WCRL)

WCRH and WCRL select the number of program wait states for each area.

Program waits are not inserted in the case of on-chip memory or internal I/O registers.

- WCRH

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | W71 | 1 | R/W | Area 7 Wait Control 1 and 0 |
| 6 | W70 | 1 | R/W | <p>These bits select the number of program wait states when area 7 in external space is accessed while the AST7 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 7 is accessed</p> <p>01: 1 program wait state inserted when external space area 7 is accessed</p> <p>10: 2 program wait states inserted when external space area 7 is accessed</p> <p>11: 3 program wait states inserted when external space area 7 is accessed</p> |
| 5 | W61 | 1 | R/W | Area 6 Wait Control 1 and 0 |
| 4 | W60 | 1 | R/W | <p>These bits select the number of program wait states when area 6 in external space is accessed while the AST6 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 6 is accessed</p> <p>01: 1 program wait state inserted when external space area 6 is accessed</p> <p>10: 2 program wait states inserted when external space area 6 is accessed</p> <p>11: 3 program wait states inserted when external space area 6 is accessed</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | W51 | 1 | R/W | Area 5 Wait Control 1 and 0 |
| 2 | W50 | 1 | R/W | <p>These bits select the number of program wait states when area 5 in external space is accessed while the AST5 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 5 is accessed</p> <p>01: 1 program wait state inserted when external space area 5 is accessed</p> <p>10: 2 program wait states inserted when external space area 5 is accessed</p> <p>11: 3 program wait states inserted when external space area 5 is accessed</p> |
| 1 | W41 | 1 | R/W | Area 4 Wait Control 1 and 0 |
| 0 | W40 | 1 | R/W | <p>These bits select the number of program wait states when area 4 in external space is accessed while the AST4 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 4 is accessed</p> <p>01: 1 program wait state inserted when external space area 4 is accessed</p> <p>10: 2 program wait states inserted when external space area 4 is accessed</p> <p>11: 3 program wait states inserted when external space area 4 is accessed</p> |

- WCRL

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | W31 | 1 | R/W | Area 3 Wait Control 1 and 0 |
| 6 | W30 | 1 | R/W | <p>These bits select the number of program wait states when area 3 in external space is accessed while the AST3 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 3 is accessed</p> <p>01: 1 program wait state inserted when external space area 3 is accessed</p> <p>10: 2 program wait states inserted when external space area 3 is accessed</p> <p>11: 3 program wait states inserted when external space area 3 is accessed</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | W21 | 1 | R/W | Area 2 Wait Control 1 and 0 |
| 4 | W20 | 1 | R/W | <p>These bits select the number of program wait states when area 2 in external space is accessed while the AST2 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 2 is accessed</p> <p>01: 1 program wait state inserted when external space area 2 is accessed</p> <p>10: 2 program wait states inserted when external space area 2 is accessed</p> <p>11: 3 program wait states inserted when external space area 2 is accessed</p> |
| 3 | W11 | 1 | R/W | Area 1 Wait Control 1 and 0 |
| 2 | W10 | 1 | R/W | <p>These bits select the number of program wait states when area 1 in external space is accessed while the AST1 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 1 is accessed</p> <p>01: 1 program wait state inserted when external space area 1 is accessed</p> <p>10: 2 program wait states inserted when external space area 1 is accessed</p> <p>11: 3 program wait states inserted when external space area 1 is accessed</p> |
| 1 | W01 | 1 | R/W | Area 0 Wait Control 1 and 0 |
| 0 | W00 | 1 | R/W | <p>These bits select the number of program wait states when area 0 in external space is accessed while the AST0 bit in ASTCR is set to 1.</p> <p>00: Program wait not inserted when external space area 0 is accessed</p> <p>01: 1 program wait state inserted when external space area 0 is accessed</p> <p>10: 2 program wait states inserted when external space area 0 is accessed</p> <p>11: 3 program wait states inserted when external space area 0 is accessed</p> |

7.3.4 Bus Control Register H (BCRH)

BCRH selects enabling or disabling of idle cycle insertion, and the memory interface for area 0.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | ICIS1 | 1 | R/W | Idle Cycle Insert 1 Selects whether or not one idle cycle state is to be inserted between bus cycles when successive external read cycles are performed in different areas. 0: Idle cycle not inserted in case of successive external read cycles in different areas 1: Idle cycle inserted in case of successive external read cycles in different areas |
| 6 | ICIS0 | 1 | R/W | Idle Cycle Insert 0 Selects whether or not one idle cycle state is to be inserted between bus cycles when successive external read and write cycles are performed. 0: Idle cycle not inserted in case of successive external read and write cycles 1: Idle cycle inserted in case of successive external read and write cycles |
| 5 | BRSTRM | 0 | R/W | Burst ROM enable Selects whether area 0 is used as a burst ROM interface. 0: Area 0 is basic bus interface 1: Area 0 is burst ROM interface |
| 4 | BRSTS1 | 1 | R/W | Burst Cycle Select 1 Selects the number of burst cycles for the burst ROM interface. 0: Burst cycle comprises 1 state 1: Burst cycle comprises 2 states |
| 3 | BRSTS0 | 0 | R/W | Burst Cycle Select 0 Selects the number of words that can be accessed in a burst ROM interface burst access. 0: Max. 4 words in burst access 1: Max. 8 words in burst access |
| 2 to 0 | — | All 0 | R/W | Reserved The write value should always be 0. |

7.3.5 Bus Control Register L (BCRL)

BCRL performs selection of the external bus-released state protocol, and enabling or disabling of WAIT pin input.

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 7 | BRLE | 0 | R/W | Bus release enable Enables or disables external bus release. 0: External bus release is disabled. $\overline{\text{BREQ}}$ and $\overline{\text{BACK}}$ can be used as I/O ports 1: External bus release is enabled |
| 6 | — | 0 | R/W | Reserved The write value should always be 0. |
| 5 | — | 0 | — | Reserved This bit is always read as 0 and cannot be modified. |
| 4 | — | 0 | R/W | Reserved The write value should always be 0. |
| 3 | — | 1 | R/W | Reserved The write value should always be 1. |
| 2, 1 | — | All 0 | R/W | Reserved The write value should always be 0. |
| 0 | WAITE | 0 | R/W | WAIT pin enable Selects enabling or disabling of wait input by the $\overline{\text{WAIT}}$ pin. 0: Wait input by $\overline{\text{WAIT}}$ pin disabled. $\overline{\text{WAIT}}$ pin can be used as I/O port 1: Wait input by $\overline{\text{WAIT}}$ pin enabled |

7.3.6 Pin Function Control Register (PFCR)

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 7, 6 | — | All 0 | R/W | Reserved The write value should always be 0. |
| 5 | BUZZE | 0 | R/W | BUZZ Output Enable: This bit selects enabling or disabling of BUZZ output from pin PF1. WDT_1 input clock that is selected by PSS, and CKS2 to CKS0 bits is output as BUZZ signal. 0: PF1 input/output pin 1: BUZZ output pin |
| 4 | — | 0 | R/W | Reserved The write value should always be 0. |
| 3 | AE3 | 1/0* | R/W | Address Output Enable 3 to 0 |
| 2 | AE2 | 1/0* | R/W | These bits select enabling or disabling of address outputs A23 to A8 in ROMless extended mode and modes with ROM. |
| 1 | AE1 | 0 | R/W | |
| 0 | AE0 | 1/0* | R/W | When a pin is enabled for address output, the address is output regardless of the corresponding DDR setting. When a pin is disabled for address output, it becomes an output port when the corresponding DDR bit is set to 1. 0000: A23 to A8 output disabled 0001: A8 output enabled; A23 to A9 output disabled 0010: A9, A8 output enabled; A23 to A10 output disabled 0011: A10 to A8 output enabled; A23 to A11 output disabled 0100: A11 to A8 output enabled; A23 to A12 output disabled 0101: A12 to A8 output enabled; A23 to A13 output disabled 0110: A13 to A8 output enabled; A23 to A14 output disabled 0111: A14 to A8 output enabled; A23 to A15 output disabled 1000: A15 to A8 output enabled; A23 to A16 output disabled 1001: A16 to A8 output enabled; A23 to A17 output disabled 1010: A17 to A8 output enabled; A23 to A18 output disabled 1011: A18 to A8 output enabled; A23 to A19 output disabled 1100: A19 to A8 output enabled; A23 to A20 output disabled 1101: A20 to A8 output enabled; A23 to A21 output disabled 1110: A21 to A8 output enabled; A23, A22 output disabled 1111: A23 to A8 output enabled |

Note: * In modes 4 and 5, initial value of each bit is 1. In modes 6 and 7, initial value of each bit is 0.

7.4 Bus Control

7.4.1 Area Divisions

In advanced mode, the bus controller partitions the 16 Mbytes address space into eight areas, 7 to 0, in 2-Mbyte units, and performs bus control for external space in area units. In normal mode*, it controls a 64-kbyte address space comprising part of area 0.

Figure 7.2 shows an outline of the memory map.

Chip select signals ($\overline{CS7}$ to $\overline{CS0}$) can be output for each area.

Note: * Not available in this LSI.

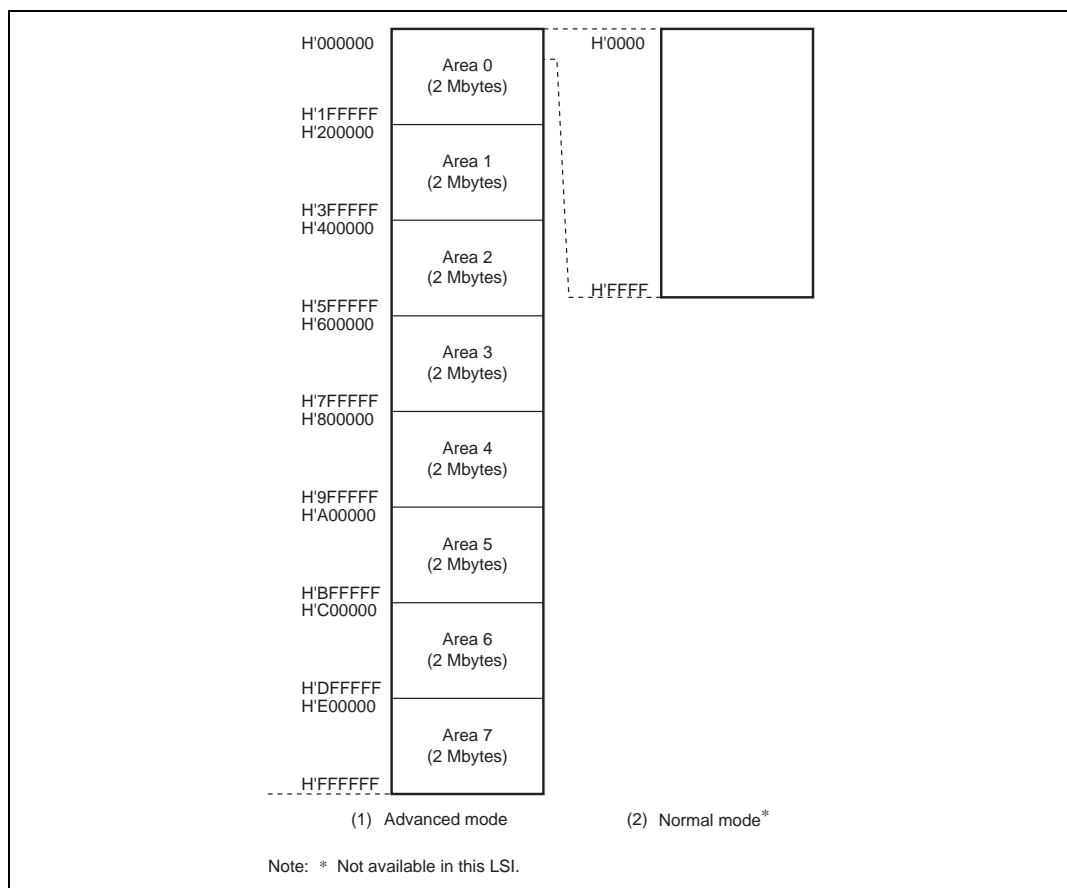


Figure 7.2 Overview of Area Divisions

7.4.2 Bus Specifications

The external space bus specifications consist of three elements: bus width, number of access states, and number of program wait states.

The bus width and number of access states for on-chip memory and internal I/O registers are fixed, and are not affected by the bus controller.

(1) Bus Width: A bus width of 8 or 16 bits can be selected with ABWCR. An area for which an 8-bit bus is selected functions as an 8-bit access space, and an area for which a 16-bit bus is selected functions as a 16-bit access space.

If all areas are designated for 8-bit access, 8-bit bus mode is set; if any area is designated for 16-bit access, 16-bit bus mode is set. When the burst ROM interface is designated, 16-bit bus mode is always set.

(2) Number of Access States: Two or three access states can be selected with ASTCR.

An area for which 2-state access is selected functions as a 2-state access space, and an area for which 3-state access is selected functions as a 3-state access space.

With the burst ROM interface, the number of access states may be determined without regard to ASTCR.

When 2-state access space is designated, wait insertion is disabled.

(3) Number of Program Wait States: When 3-state access space is designated by ASTCR, the number of program wait states to be inserted automatically is selected with WCRH and WCRL.

From 0 to 3 program wait states can be selected.

Table 7.2 Bus Specifications for Each Area (Basic Bus Interface)

| ABWCR | ASTCR | WCRH, WCRL | | Bus Specifications (Basic Bus Interface) | | |
|-------|-------|------------|-----|--|-------------------------|-------------------------------|
| ABWn | ASTn | Wn1 | Wn0 | Bus Width | Number of Access States | Number of Program Wait States |
| 0 | 0 | — | — | 16 | 2 | 0 |
| | 1 | 0 | 0 | | 3 | 0 |
| | | | 1 | | | 1 |
| | | 1 | 0 | | | 2 |
| | | | 1 | | | 3 |
| 1 | 0 | — | — | 8 | 2 | 0 |
| | 1 | 0 | 0 | | 3 | 0 |
| | | | 1 | | | 1 |
| | | 1 | 0 | | | 2 |
| | | | 1 | | | 3 |

7.4.3 Bus Interface for Each Area

The initial state of each area is basic bus interface, 3-state access space. The initial bus width is selected according to the operating mode. The bus specifications described here cover basic items only, and the sections on each memory interface (7.6, Basic Bus Interface and 7.7, Burst ROM Interface) should be referred to for further details.

- (1) **Area 0:** Area 0 includes on-chip ROM, and in ROM-disabled extended mode, all of area 0 is external space. In ROM-enabled extended mode, the space excluding on-chip ROM is external space.

When area 0 external space is accessed, the $\overline{\text{CS0}}$ signal can be output.

Either basic bus interface or burst ROM interface can be selected for area 0.

- (2) **Areas 6 to 1:** In external extended mode, all of areas 6 to 1 is external space. When area 6 to 1 external space is accessed, the $\overline{\text{CS6}}$ to $\overline{\text{CS1}}$ pin signals respectively can be output. Only the basic bus interface can be used for areas 6 to 1.

- (3) **Area 7:** Area 7 includes the on-chip RAM and internal I/O registers. In external extended mode, the space excluding the on-chip RAM and internal I/O registers, is external space. The on-chip RAM is enabled when the RAME bit in the system control register (SYSCR) is set to 1; when the RAME bit is cleared to 0, the on-chip RAM is disabled and the corresponding space becomes external space.

When area 7 external space is accessed, the $\overline{\text{CS7}}$ signal can be output.

Only the basic bus interface can be used for the area 7.

7.4.4 Chip Select Signals

This LSI can output chip select signals ($\overline{CS7}$ to $\overline{CS0}$) to areas 7 to 0, the signal being driven low when the corresponding external space area is accessed. Figure 7.3 shows an example of \overline{CSn} ($n = 7$ to 0) output timing. Enabling or disabling of the \overline{CSn} signal is performed by setting the data direction register (DDR) for the port corresponding to the particular \overline{CSn} pin.

In ROM-disabled extended mode, the $\overline{CS0}$ pin is placed in the output state after a power-on reset. Pins $\overline{CS7}$ to $\overline{CS1}$ are placed in the input state after a power-on reset, and so the corresponding DDR should be set to 1 when outputting signals $\overline{CS7}$ to $\overline{CS1}$.

In ROM-enabled extended mode, pins $\overline{CS7}$ to $\overline{CS0}$ are all placed in the input state after a power-on reset, and so the corresponding DDR should be set to 1 when outputting signals $\overline{CS7}$ to $\overline{CS0}$. For details, see section 10, I/O Ports.

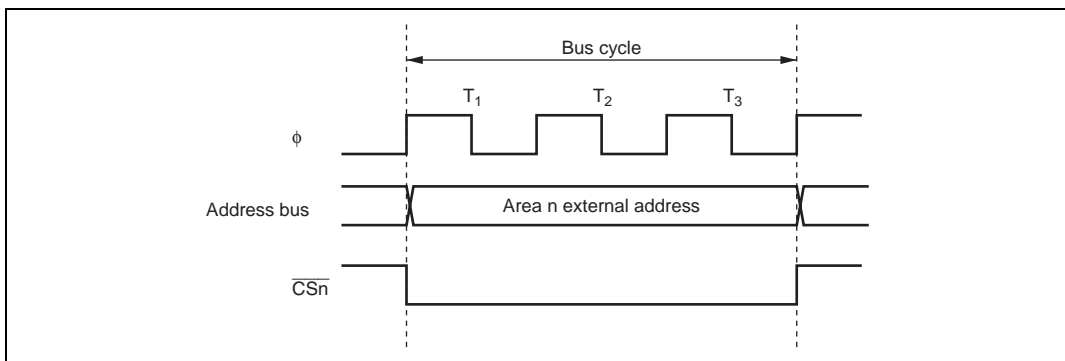


Figure 7.3 \overline{CSn} Signal Output Timing ($n = 0$ to 7)

7.5 Basic Timing

The CPU is driven by a system clock (ϕ), denoted by the symbol ϕ . The period from one rising edge of ϕ to the next is referred to as a “state”. The memory cycle or bus cycle consists of one, two, or three states. Different methods are used to access on-chip memory, on-chip peripheral modules, and the external address space.

7.5.1 On-Chip Memory (ROM, RAM) Access Timing

On-chip memory is accessed in one state. The data bus is 16 bits wide, permitting both byte and word transfer instruction. Figure 7.4 shows the on-chip memory access cycle. Figure 7.5 shows the pin states.

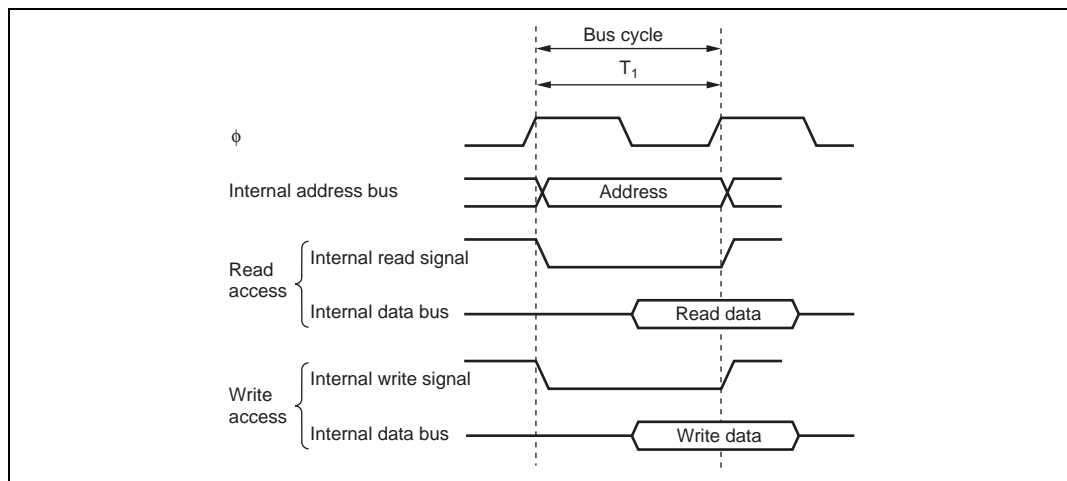


Figure 7.4 On-Chip Memory Access Cycle

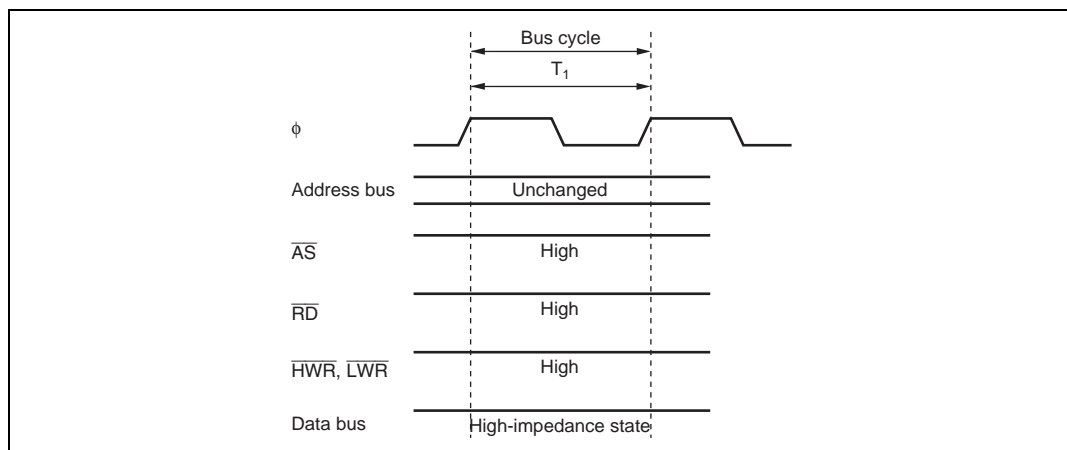


Figure 7.5 Pin States during On-Chip Memory Access

7.5.2 On-Chip Peripheral Module Access Timing

The on-chip peripheral modules are accessed in two states. The data bus is either 8 bits or 16 bits wide, depending on the particular internal I/O register being accessed. Figure 7.6 shows the access timing for the on-chip peripheral modules. Figure 7.7 shows the pin states.

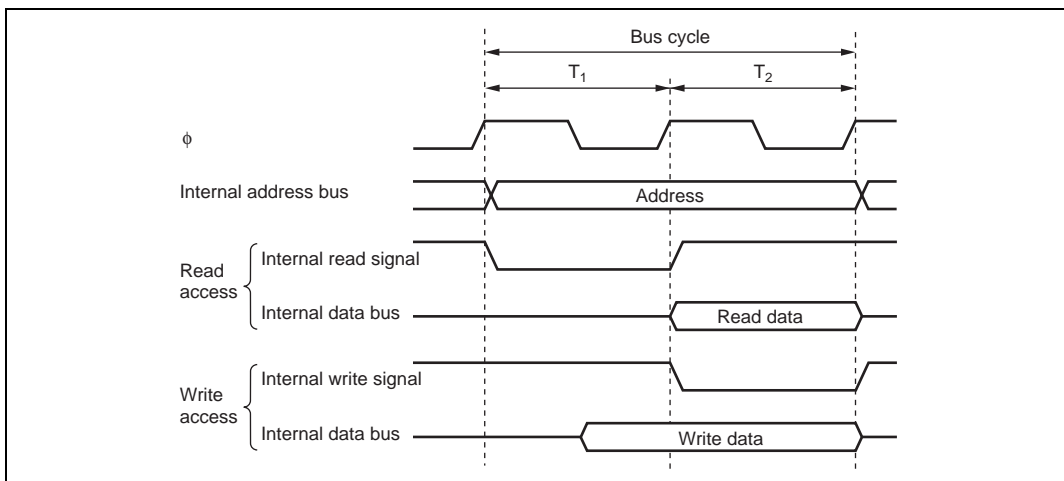


Figure 7.6 On-Chip Peripheral Module Access Cycle

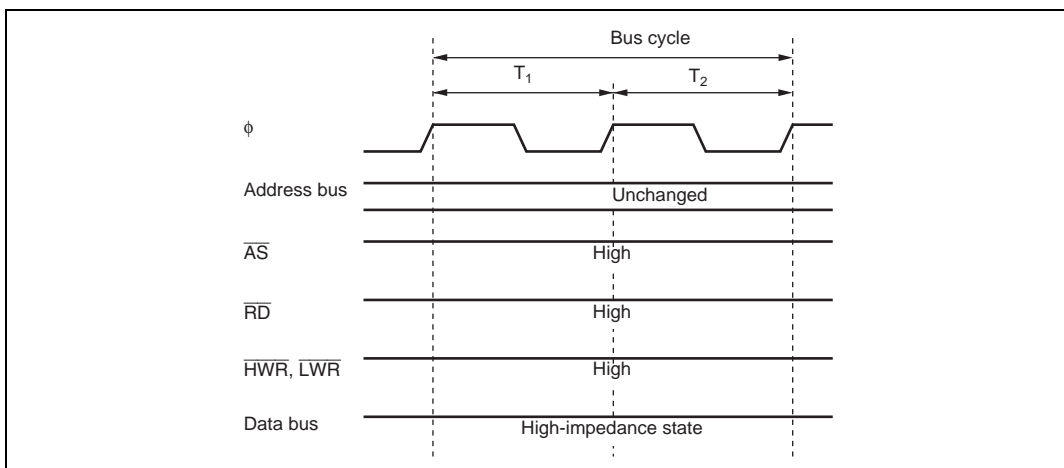


Figure 7.7 Pin States during On-Chip Peripheral Module Access

7.5.3 External Address Space Access Timing

The external address space is accessed with an 8-bit or 16-bit data bus width in a two-state or three-state bus cycle. In three-state access, wait states can be inserted. For further details, refer to section 7.6.3, Basic Timing.

7.6 Basic Bus Interface

The basic bus interface enables direct connection of ROM, SRAM, and so on.

7.6.1 Data Size and Data Alignment

Data sizes for the CPU and other internal bus masters are byte, word, and longword. The bus controller has a data alignment function, and when accessing external space, controls whether the upper data bus (D15 to D8) or lower data bus (D7 to D0) is used according to the bus specifications for the area being accessed (8-bit access space or 16-bit access space) and the data size.

8-Bit Access Space: Figure 7.8 illustrates data alignment control for the 8-bit access space. With the 8-bit access space, the upper data bus (D15 to D8) is always used for accesses. The amount of data that can be accessed at one time is one byte: a word transfer instruction is performed as two-byte accesses, and a longword transfer instruction, as four-byte accesses.

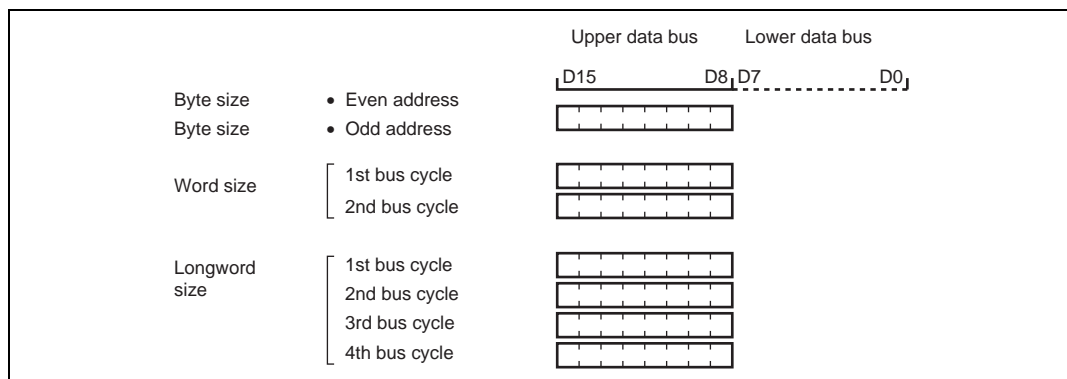


Figure 7.8 Access Sizes and Data Alignment Control (8-Bit Access Space)

16-Bit Access Space: Figure 7.9 illustrates data alignment control for the 16-bit access space. With the 16-bit access space, the upper data bus (D15 to D8) and lower data bus (D7 to D0) are used for accesses. The amount of data that can be accessed at one time is one byte or one word, and a longword transfer instruction is executed as two word transfer instructions.

In byte access, whether the upper or lower data bus is used is determined by whether the address is even or odd. The upper data bus is used for an even address, and the lower data bus for an odd address.

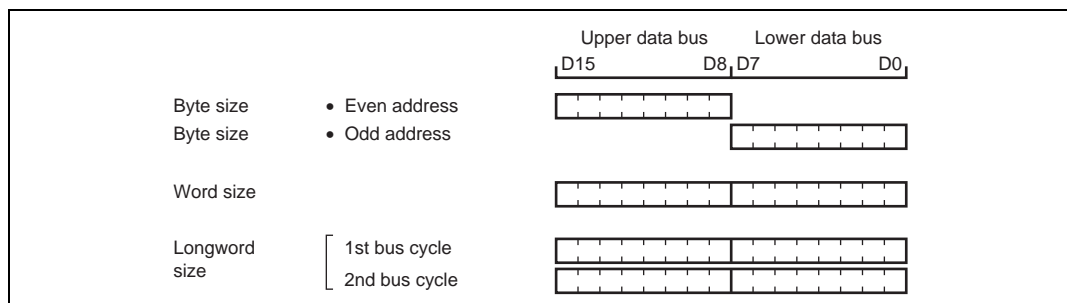


Figure 7.9 Access Sizes and Data Alignment Control (16-Bit Access Space)

7.6.2 Valid Strobes

Table 7.3 shows the data buses used and valid strobes for the access spaces.

In a read, the \overline{RD} signal is valid without discrimination between the upper and lower halves of the data bus.

In a write, the \overline{HWR} signal is valid for the upper half of the data bus, and the \overline{LWR} signal for the lower half.

Table 7.3 Data Buses Used and Valid Strobes

| Area | Access Size | Read/Write | Address | Valid Strobe | Upper Data Bus (D15 to D8) | Lower Data Bus (D7 to D0) |
|---------------------|-------------|------------|---------|----------------------------------|----------------------------|---------------------------|
| 8-bit access space | Byte | Read | — | \overline{RD} | Valid | Invalid |
| | | Write | — | \overline{HWR} | | Hi-Z |
| 16-bit access space | Byte | Read | Even | \overline{RD} | Valid | Invalid |
| | | | Odd | | Invalid | Valid |
| | | Write | Even | \overline{HWR} | Valid | Hi-Z |
| | | | Odd | \overline{LWR} | Hi-Z | Valid |
| | Word | Read | — | \overline{RD} | Valid | Valid |
| | | Write | — | $\overline{HWR}, \overline{LWR}$ | Valid | Valid |

Notes: Hi-Z: High impedance.

Invalid: Input state; input value is ignored.

7.6.3 Basic Timing

8-Bit 2-State Access Space: Figure 7.10 shows the bus timing for an 8-bit 2-state access space. When an 8-bit access space is accessed, the upper half (D15 to D8) of the data bus is used.

Wait states cannot be inserted.

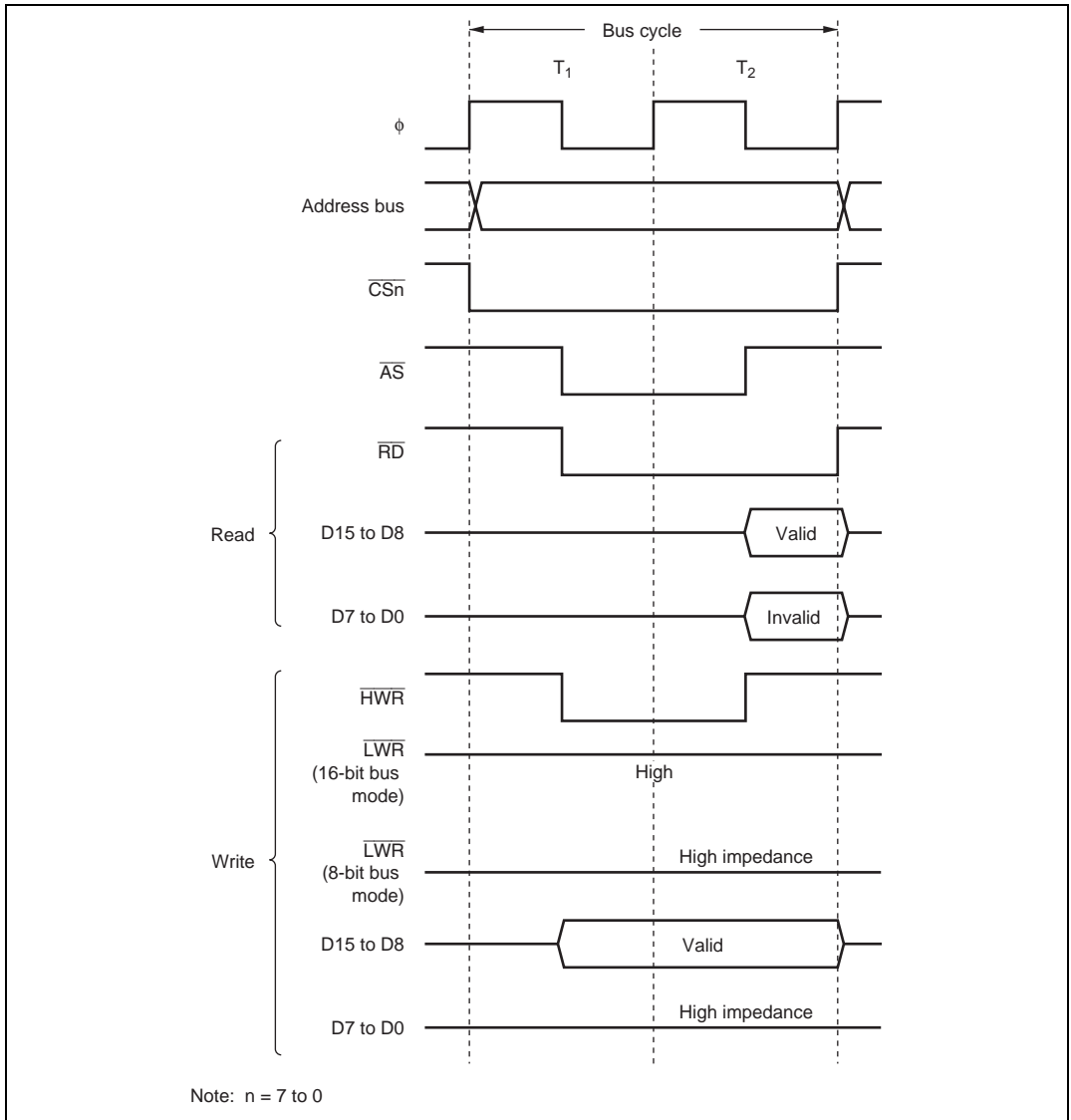


Figure 7.10 Bus Timing for 8-Bit 2-State Access Space

8-Bit 3-State Access Space: Figure 7.11 shows the bus timing for an 8-bit 3-state access space. When an 8-bit access space is accessed, the upper half (D15 to D8) of the data bus is used.

Wait states can be inserted.

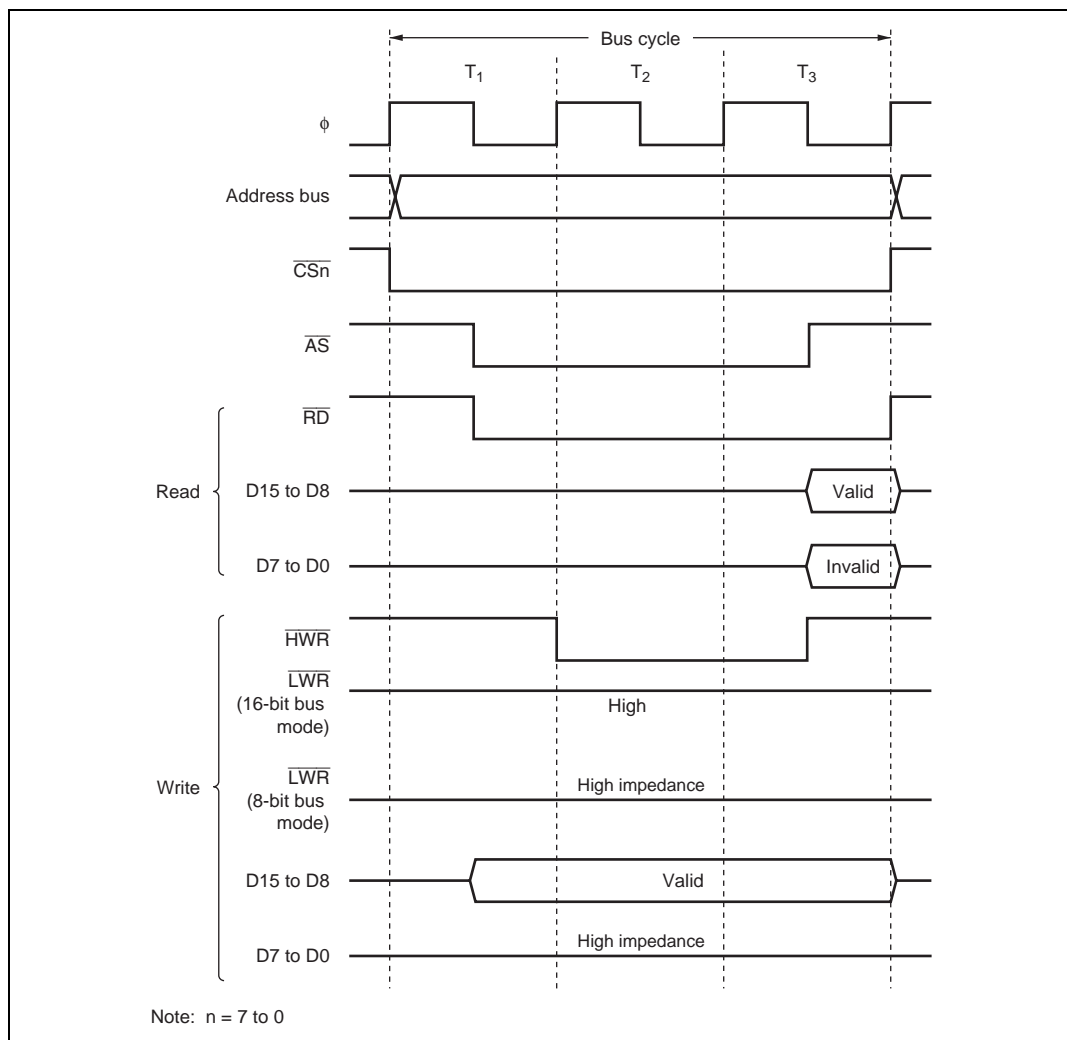


Figure 7.11 Bus Timing for 8-Bit 3-State Access Space

16-Bit 2-State Access Space: Figures 7.12 to 7.14 show bus timings for a 16-bit 2-state access space. When a 16-bit access space is accessed, the upper half (D15 to D8) of the data bus is used for the even address, and the lower half (D7 to D0) for the odd address.

Wait states cannot be inserted.

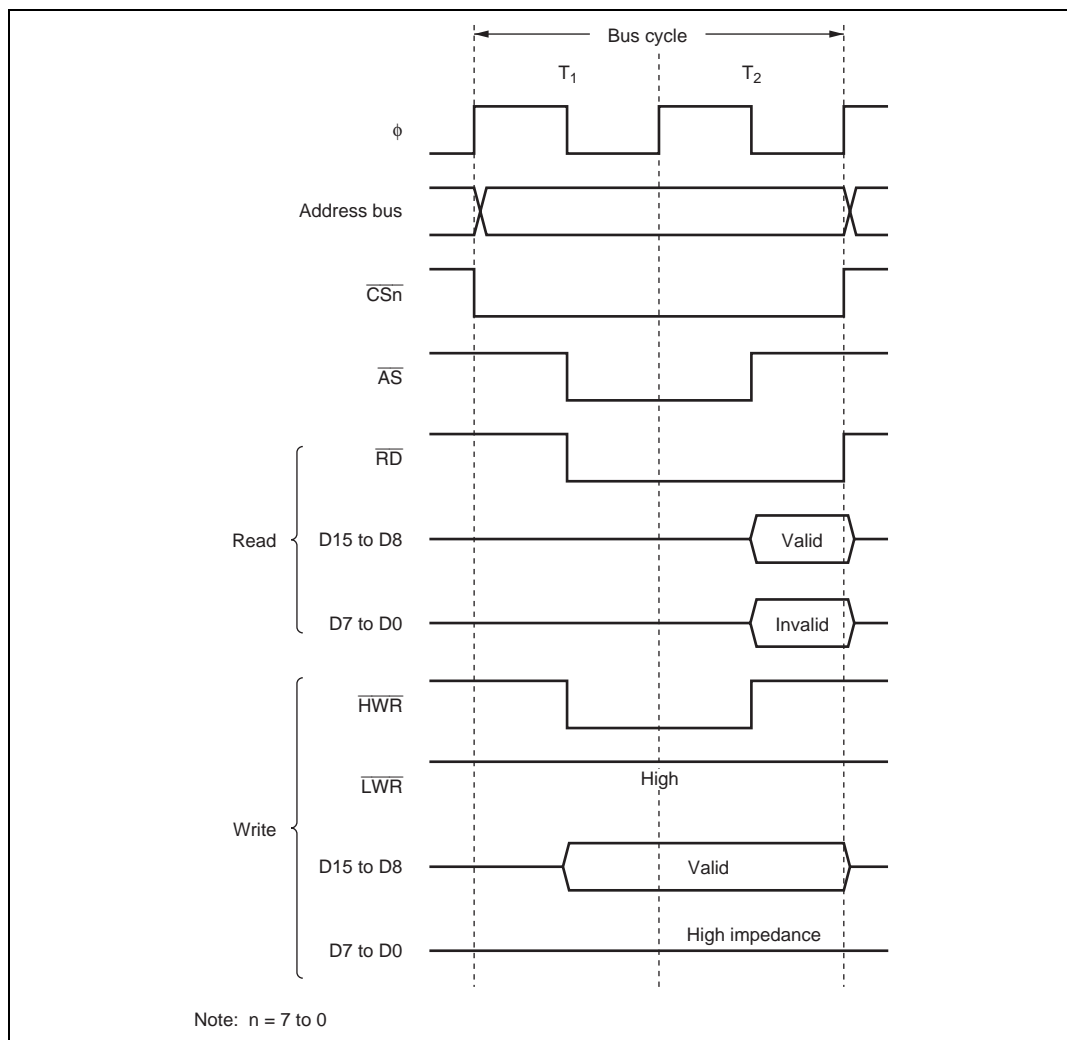


Figure 7.12 Bus Timing for 16-Bit 2-State Access Space (1) (Even Address Byte Access)

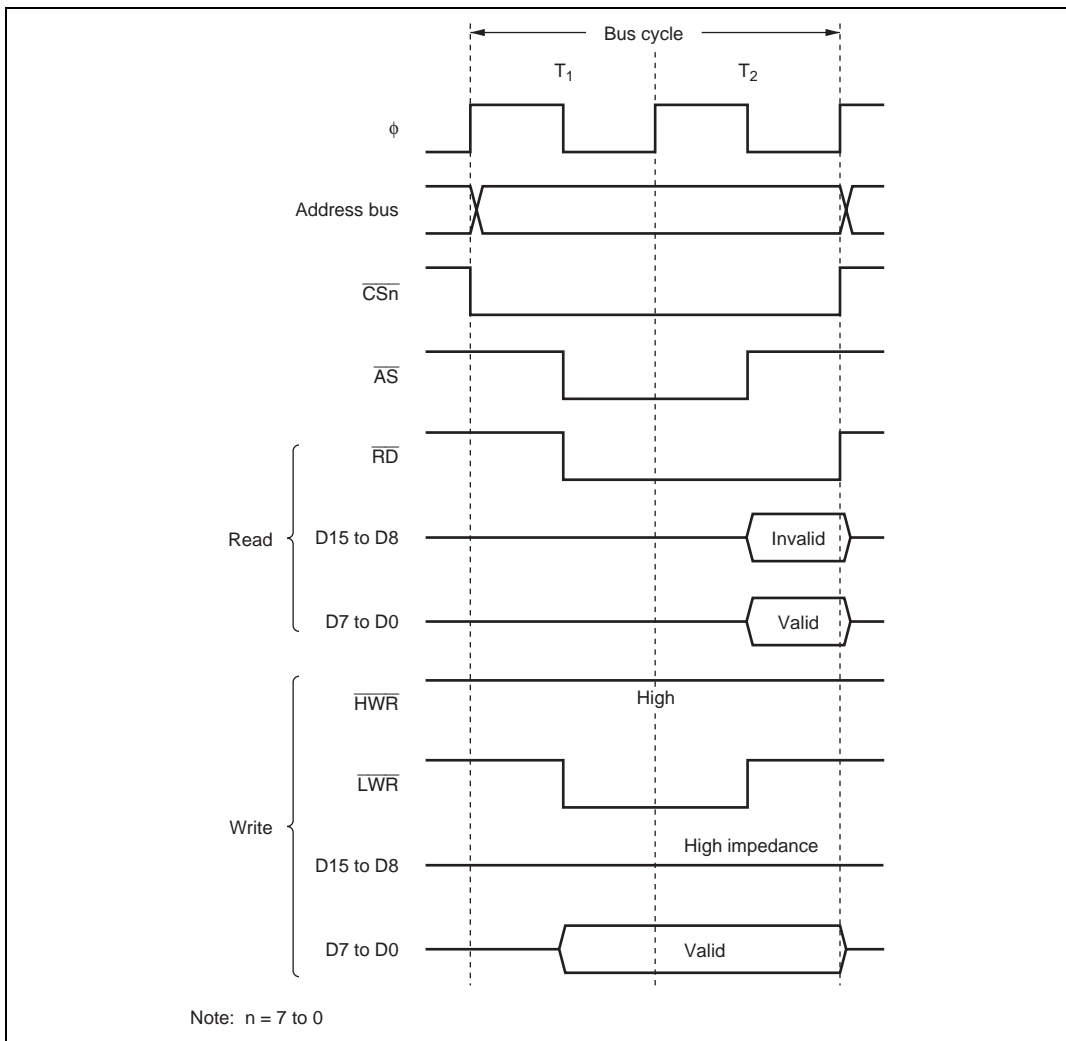


Figure 7.13 Bus Timing for 16-Bit 2-State Access Space (2) (Odd Address Byte Access)

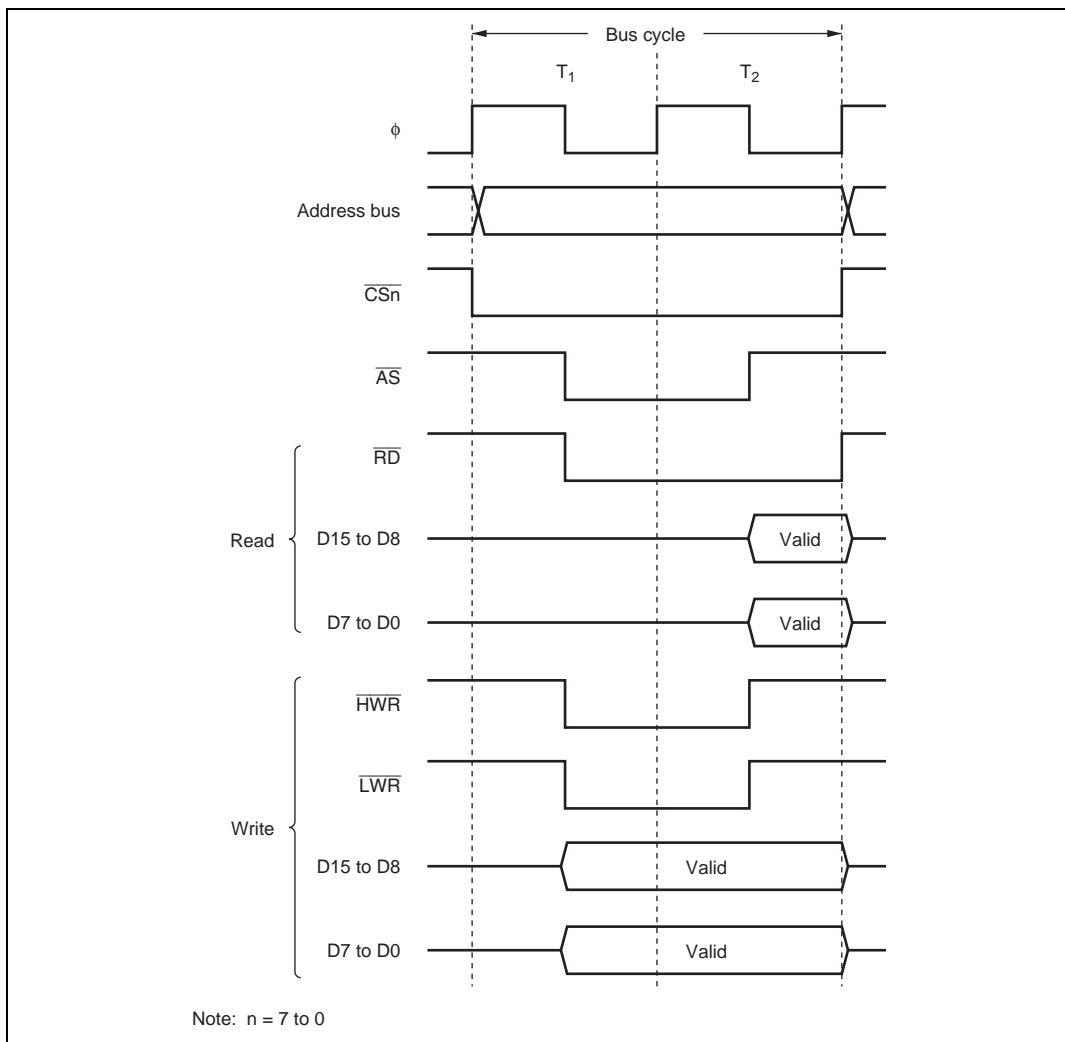


Figure 7.14 Bus Timing for 16-Bit 2-State Access Space (3) (Word Access)

16-Bit 3-State Access Space: Figures 7.15 to 7.17 show bus timings for a 16-bit 3-state access space. When a 16-bit access space is accessed, the upper half (D15 to D8) of the data bus is used for the even address, and the lower half (D7 to D0) for the odd address.

Wait states can be inserted.

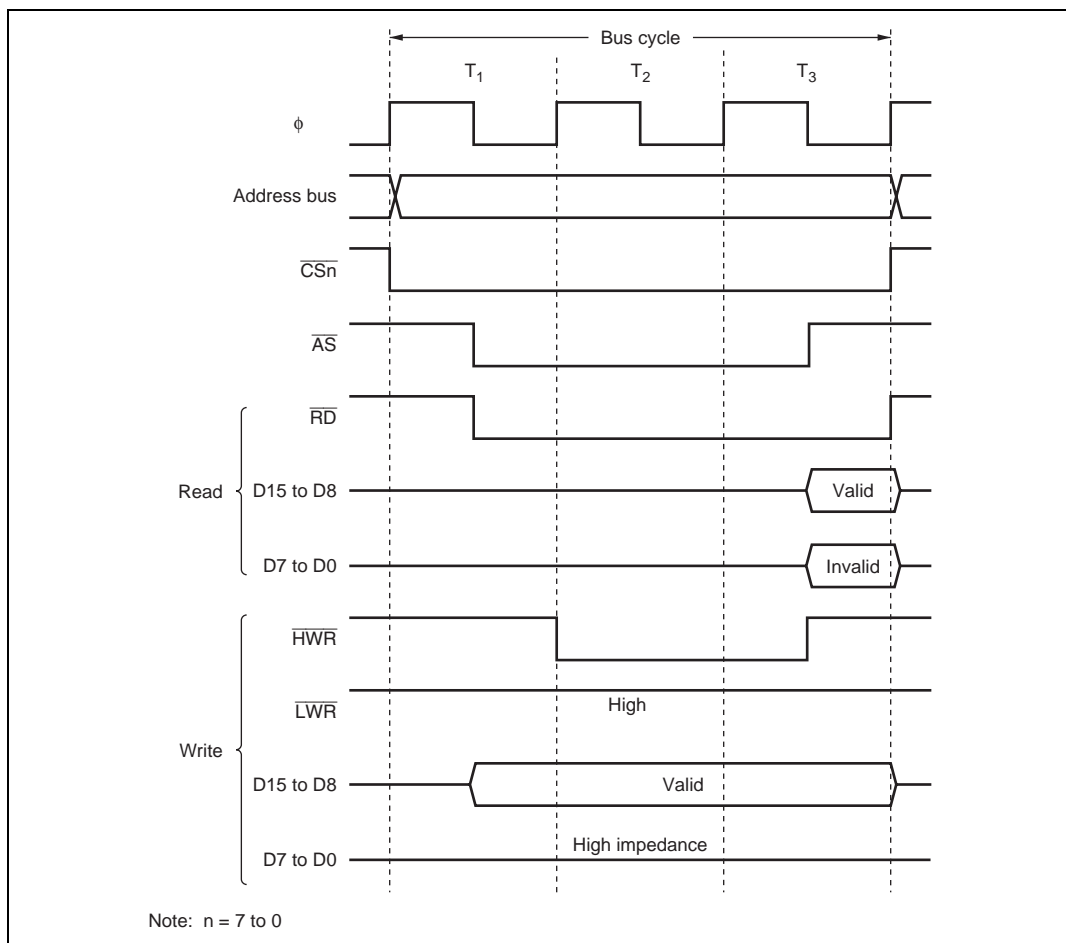


Figure 7.15 Bus Timing for 16-Bit 3-State Access Space (1) (Even Address Byte Access)

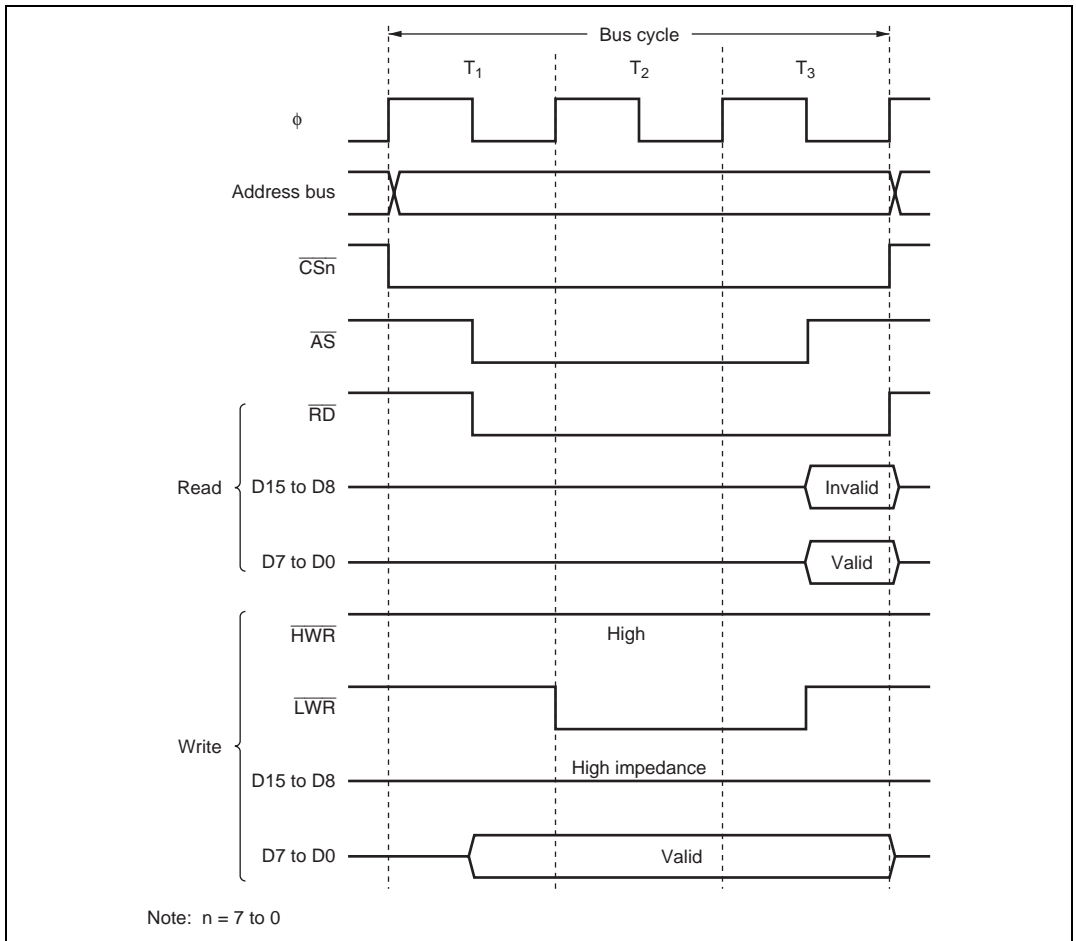


Figure 7.16 Bus Timing for 16-Bit 3-State Access Space (2) (Odd Address Byte Access)

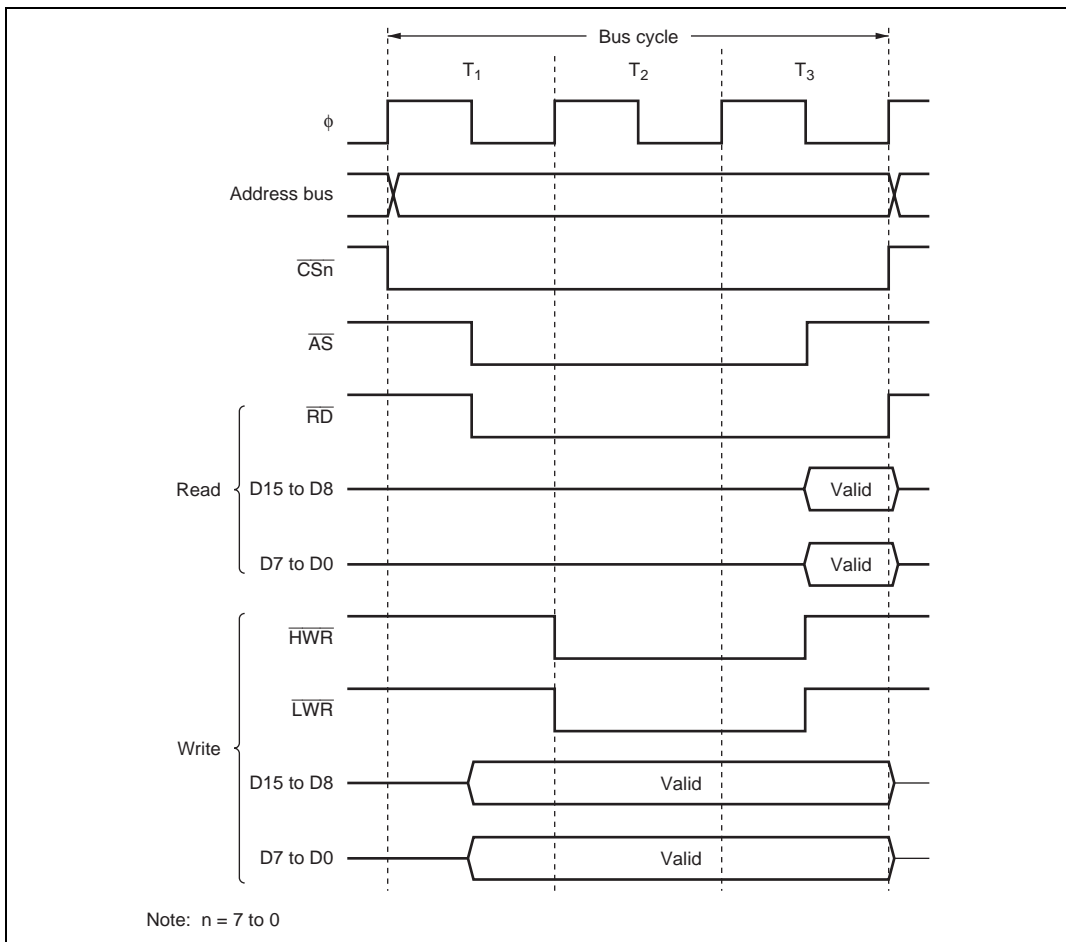


Figure 7.17 Bus Timing for 16-Bit 3-State Access Space (3) (Word Access)

7.6.4 Wait Control

When accessing external space, this LSI can extend the bus cycle by inserting one or more wait states (T_w). There are two ways of inserting wait states: program wait insertion and pin wait insertion using the $\overline{\text{WAIT}}$ pin.

(1) Program Wait Insertion

From 0 to 3 wait states can be inserted automatically between the T_2 state and T_3 state on an individual area basis in 3-state access space, according to the settings of WCRH and WCRL.

(2) Pin Wait Insertion

Setting the WAITE bit in BCRL to 1 enables wait insertion by means of the $\overline{\text{WAIT}}$ pin. When external space is accessed in this state, program wait insertion is first carried out according to the settings in WCRH and WCRL. Then, if the $\overline{\text{WAIT}}$ pin is low at the falling edge of ϕ in the last T_2 or T_w state, a T_w state is inserted. If the $\overline{\text{WAIT}}$ pin is held low, T_w states are inserted until it goes high.

Figure 7.18 shows an example of wait state insertion timing.

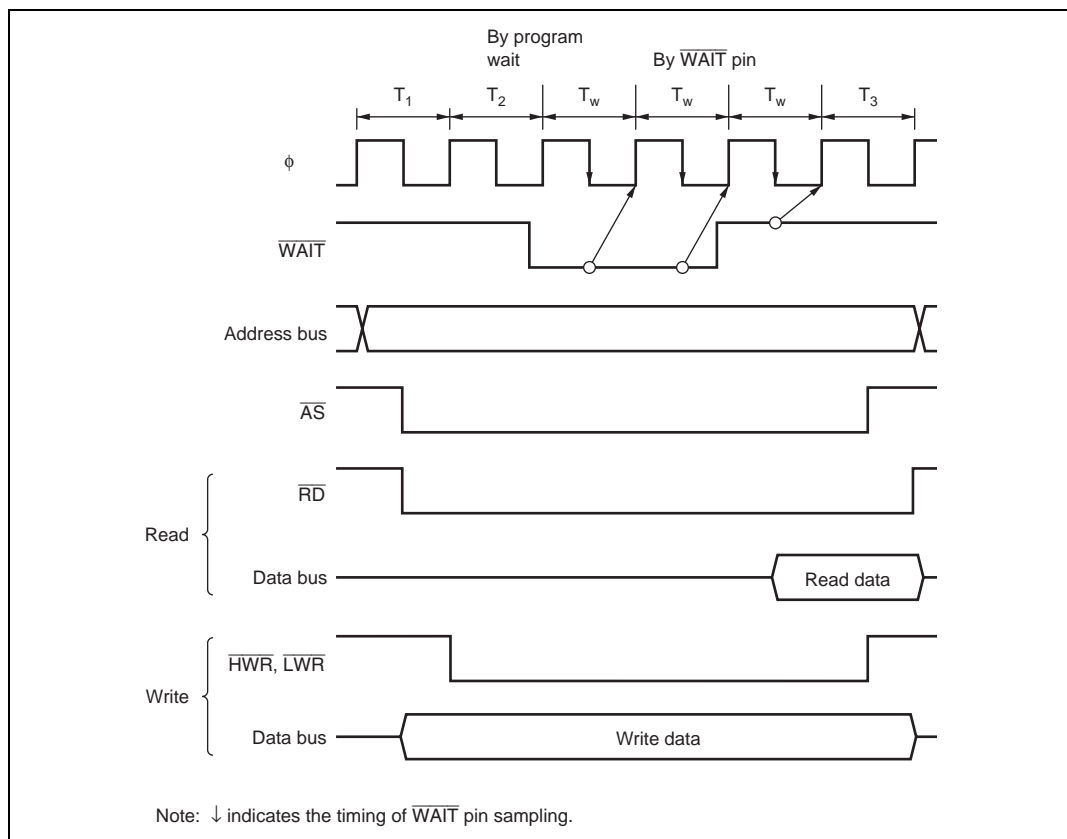


Figure 7.18 Example of Wait State Insertion Timing

7.7 Burst ROM Interface

With this LSI, external space area 0 can be designated as burst ROM space, and burst ROM interfacing can be performed. The burst ROM space interface enables 16-bit configuration ROM with burst access capability to be accessed at high speed.

Area 0 can be designated as burst ROM space by means of the BRSTRM bit in BCRH.

Consecutive burst accesses of a maximum of 4 words or 8 words can be performed for CPU instruction fetches only. One or two states can be selected for burst access.

Note: When the operating frequency ranges from 16 MHz to 20 MHz, the burst ROM interface is not available.

7.7.1 Basic Timing

The number of states in the initial cycle (full access) of the burst ROM interface is in accordance with the setting of the AST0 bit in ASTCR. Also, when the AST0 bit is set to 1, wait state insertion is possible. One or two states can be selected for the burst cycle, according to the setting of the BRSTS1 bit in BCRH. Wait states cannot be inserted. When area 0 is designated as burst ROM space, it becomes 16-bit access space regardless of the setting of the ABW0 bit in ABWCR.

When the BRSTS0 bit in BCRH is cleared to 0, burst access of up to 4 words is performed; when the BRSTS0 bit is set to 1, burst access of up to 8 words is performed.

The basic access timing for burst ROM space is shown in figures 7.19 and 7.20. The timing shown in figure 7.19 is for the case where the AST0 and BRSTS1 bits are both set to 1, and that in figure 7.20 is for the case where both these bits are cleared to 0.

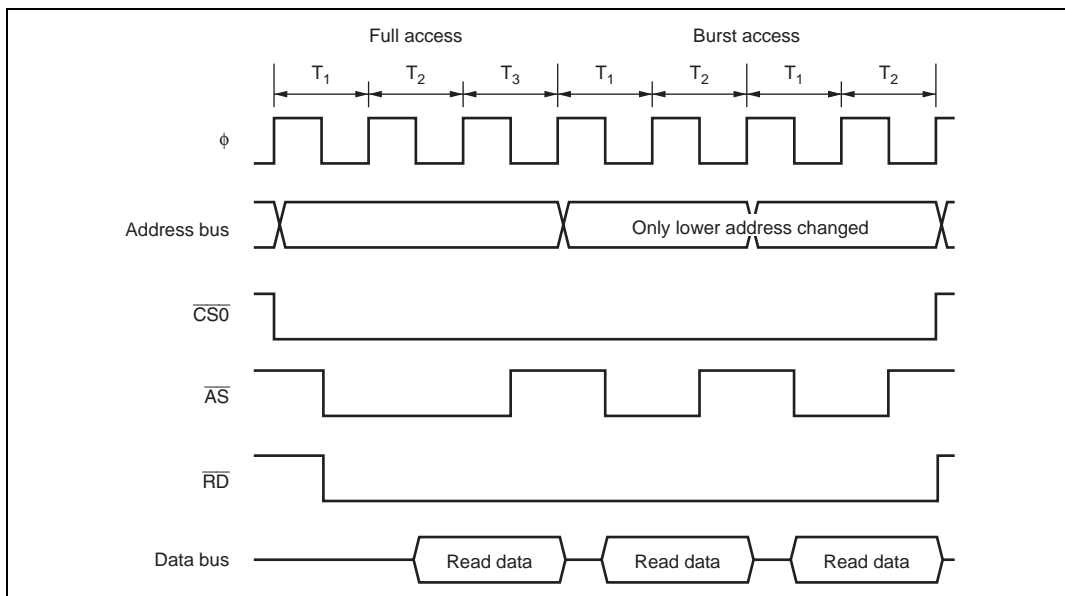


Figure 7.19 Example of Burst ROM Access Timing (When $AST0 = BRSTS1 = 1$)

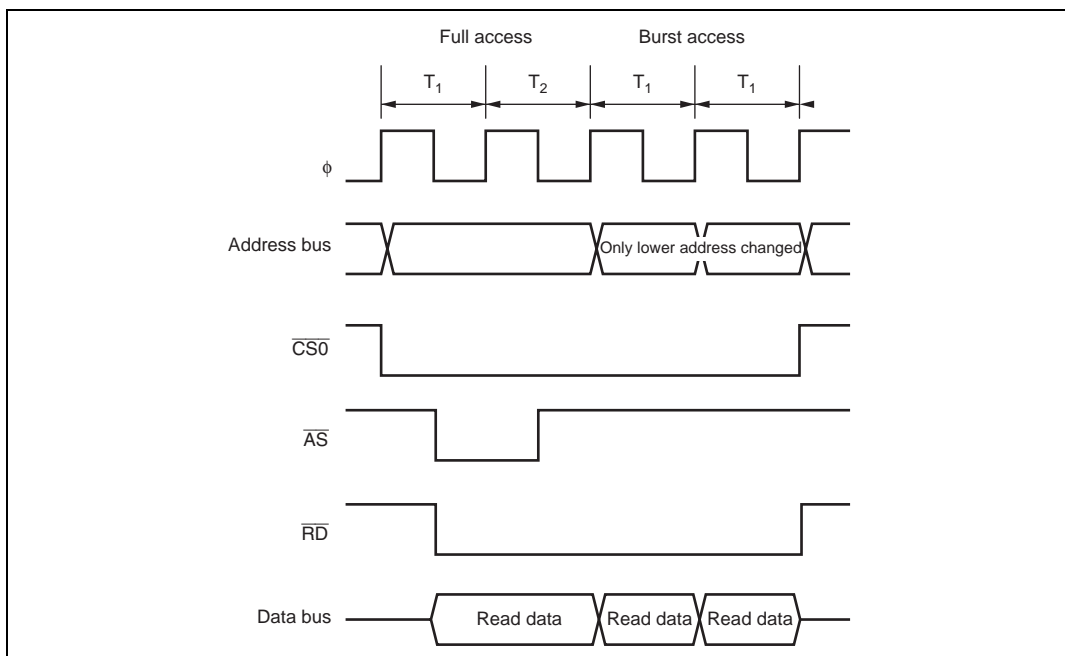


Figure 7.20 Example of Burst ROM Access Timing (When $AST0 = BRSTS1 = 0$)

7.7.2 Wait Control

As with the basic bus interface, either program wait insertion or pin wait insertion using the $\overline{\text{WAIT}}$ pin can be used in the initial cycle (full access) of the burst ROM interface. See section 7.6.4, Wait Control.

Wait states cannot be inserted in a burst cycle.

7.8 Idle Cycle

When this LSI accesses external space, it can insert a 1-state idle cycle (T_1) between bus cycles in the following two cases: (1) when read accesses between different areas occur consecutively, and (2) when a write cycle occurs immediately after a read cycle. By inserting an idle cycle it is possible, for example, to avoid data collisions between ROM, with a long output floating time, and high-speed memory, I/O interfaces, and so on.

(1) Consecutive Reads between Different Areas

If consecutive reads between different areas occur while the ICIS1 bit in BCRH is set to 1, an idle cycle is inserted at the start of the second read cycle.

Figure 7.21 shows an example of the operation in this case. In this example, bus cycle A is a read cycle from ROM with a long output floating time, and bus cycle B is a read cycle from SRAM, each being located in a different area. In (a), an idle cycle is not inserted, and a collision occurs in cycle B between the read data from ROM and that from SRAM. In (b), an idle cycle is inserted, and a data collision is prevented.

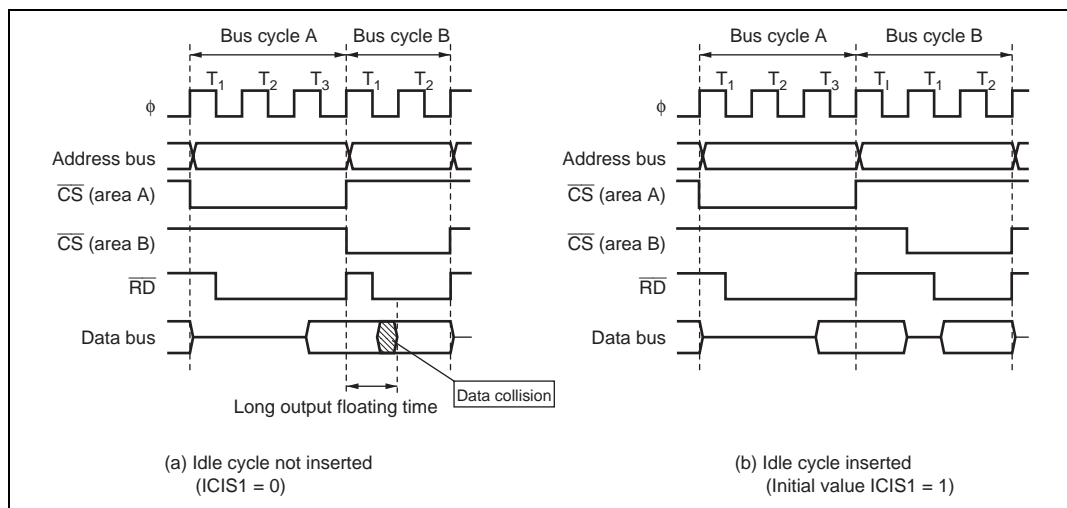


Figure 7.21 Example of Idle Cycle Operation (1)

(2) Write after Read

If an external write occurs after an external read while the ICIS0 bit in BCRH is set to 1, an idle cycle is inserted at the start of the write cycle.

Figure 7.22 shows an example of the operation in this case. In this example, bus cycle A is a read cycle from ROM with a long output floating time, and bus cycle B is a CPU write cycle. In (a), an idle cycle is not inserted, and a collision occurs in cycle B between the read data from ROM and the CPU write data. In (b), an idle cycle is inserted, and a data collision is prevented.

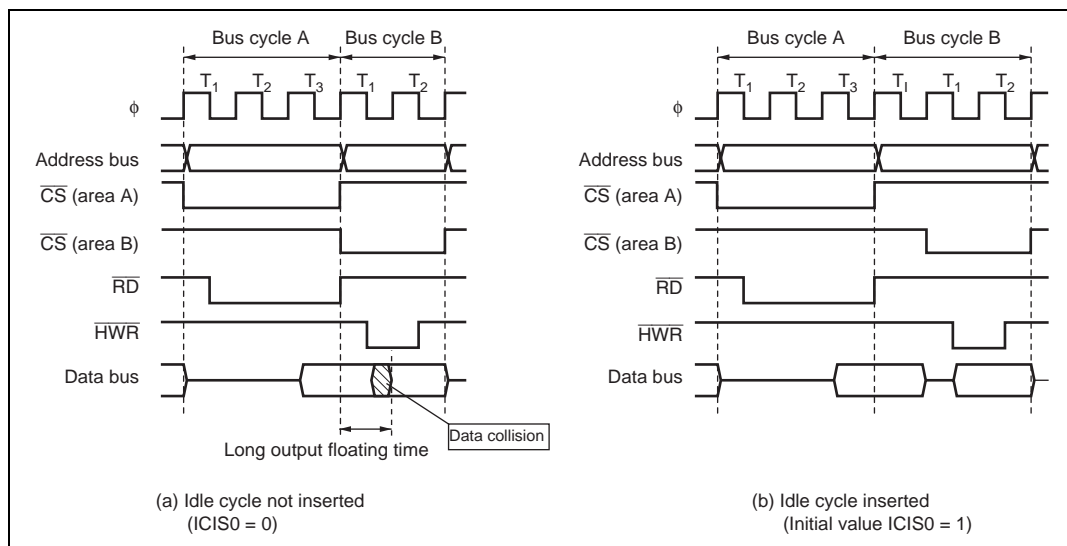


Figure 7.22 Example of Idle Cycle Operation (2)

(3) Relationship between Chip Select (\overline{CS}) Signal and Read (\overline{RD}) Signal

Depending on the system's load conditions, the \overline{RD} signal may lag behind the \overline{CS} signal. An example is shown in figure 7.23.

In this case, with the setting for no idle cycle insertion (a), there may be a period of overlap between the bus cycle A \overline{RD} signal and the bus cycle B \overline{CS} signal.

Setting idle cycle insertion, as in (b), however, will prevent any overlap between the \overline{RD} and \overline{CS} signals.

In the initial state after reset release, idle cycle insertion (b) is set.

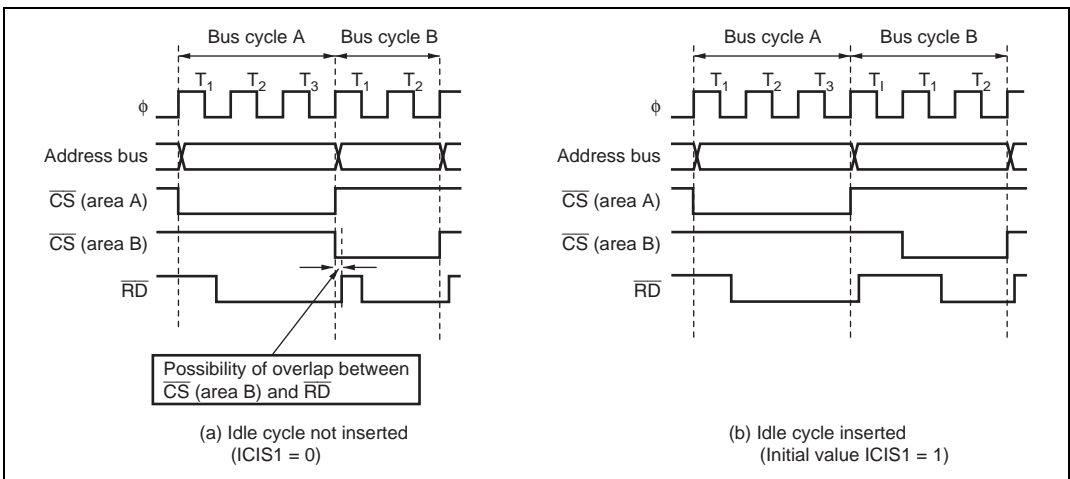


Figure 7.23 Relationship between Chip Select (\overline{CS}) and Read (\overline{RD})

Table 7.4 shows pin states in an idle cycle.

Table 7.4 Pin States in Idle Cycle

| Pins | Pin State |
|------------------|----------------------------|
| A23 to A0 | Contents of next bus cycle |
| D15 to D0 | High impedance |
| \overline{CSn} | High |
| \overline{AS} | High |
| \overline{RD} | High |
| \overline{HWR} | High |
| \overline{LWR} | High |

7.9 Bus Release

This LSI can release the external bus in response to a bus request from an external device. In the external bus released state, the internal bus master continues to operate as long as there is no external access.

In external extended mode, the bus can be released to an external device by setting the BRLE bit in BCRL to 1. Driving the $\overline{\text{BREQ}}$ pin low issues an external bus request to this LSI. When the $\overline{\text{BREQ}}$ pin is sampled, at the prescribed timing the $\overline{\text{BACK}}$ pin is driven low, and the address bus, data bus, and bus control signals are placed in the high-impedance state, establishing the external bus-released state.

In the external bus released state, an internal bus master can perform accesses using the internal bus. When an internal bus master wants to make an external access, it temporarily defers activation of the bus cycle, and waits for the bus request from the external bus master to be dropped.

When the $\overline{\text{BREQ}}$ pin is driven high, the $\overline{\text{BACK}}$ pin is driven high at the prescribed timing and the external bus released state is terminated.

In the event of simultaneous external bus release request and external access request generation, the order of priority is as follows:

(High) External bus release > Internal bus master external access (Low)

Table 7.5 shows pin states in the external bus released state.

Table 7.5 Pin States in Bus Released State

| Pins | Pin State |
|-------------------------|----------------|
| A23 to A0 | High impedance |
| D15 to D0 | High impedance |
| $\overline{\text{CSn}}$ | High impedance |
| $\overline{\text{AS}}$ | High impedance |
| $\overline{\text{RD}}$ | High impedance |
| $\overline{\text{HWR}}$ | High impedance |
| $\overline{\text{LWR}}$ | High impedance |

Figure 7.24 shows the timing for transition to the bus-released state.

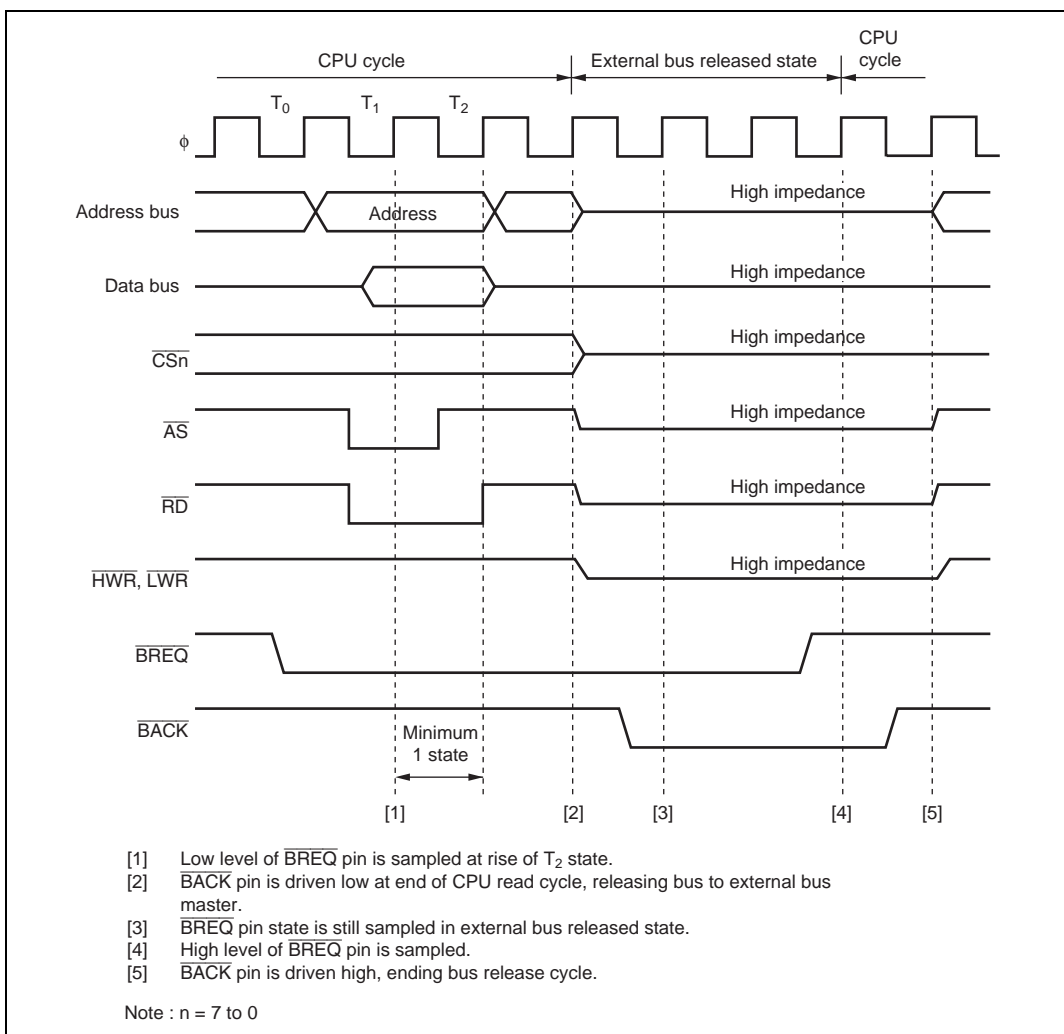


Figure 7.24 Bus-Released State Transition Timing

7.9.1 Bus Release Usage Note

When MSTPCR is set to H'FFFFFF and transmitted to sleep mode, the external bus release does not function. To activate the external bus release in sleep mode, do not set MSTPCR to H'FFFFFF.

7.10 Bus Arbitration

This LSI has a bus arbiter that arbitrates bus master operations.

There are three bus masters, the CPU, DMAC*, and DTC, which perform read/write operations when they have possession of the bus. Each bus master requests the bus by means of a bus request signal. The bus arbiter determines priorities at the prescribed timing, and permits use of the bus by means of a bus request acknowledge signal. The selected bus master then takes possession of the bus and begins its operation.

Note: * Supported only by the H8S/2239 Group.

7.10.1 Operation

The bus arbiter detects the bus masters' bus request signals, and if the bus is requested, sends a bus request acknowledge signal to the bus master making the request. If there are bus requests from more than one bus master, the bus request acknowledge signal is sent to the one with the highest priority. When a bus master receives the bus request acknowledge signal, it takes possession of the bus until that signal is canceled.

The order of priority of the bus masters is as follows:

(High) DMAC* > DTC > CPU (Low)

An internal bus access by an internal bus master, and external bus release, can be executed in parallel.

In the event of simultaneous external bus release request, and internal bus master external access request generation, the order of priority is as follows:

(High) External bus release > Internal bus master external access (Low)

Note: * Supported only by the H8S/2239 Group.

7.10.2 Bus Transfer Timing

Even if a bus request is received from a bus master with a higher priority than that of the bus master that has acquired the bus and is currently operating, the bus is not necessarily transferred immediately. There are specific times at which each bus master can relinquish the bus.

CPU: The CPU is the lowest-priority bus master, and if a bus request is received from the DMAC* and DTC, the bus arbiter transfers the bus to the bus master that issued the request. The timing for transfer of the bus is as follows:

- The bus is transferred at a break between bus cycles. However, if a bus cycle is executed in discrete operations, as in the case of a longword-size access, the bus is not transferred between the operations.
- If the CPU is in sleep mode, it transfers the bus immediately.

Note: * Supported only by the H8S/2239 Group.

DTC: The DTC sends the bus arbiter a request for the bus when an activation request is generated.

The DTC can release the bus after a vector read, a register information read (3 states), a single data transfer, or a register information write (3 states). It does not release the bus during a register information read (3 states), a single data transfer, or a register information write (3 states).

DMAC (Only by the H8S/2239 Group): The DMAC sends the bus arbiter a request for the bus when an activation request is generated.

In the case of an external request in short address mode or normal mode, and in cycle steal mode, the DMAC releases the bus after a single transfer.

In block transfer mode, it releases the bus after transfer of one block, and in burst mode, after completion of the transfer.

7.10.3 External Bus Release Usage Note

External bus release can be performed on completion of an external bus cycle. The $\overline{\text{CS}}$ signal remains low until the end of the external bus cycle. Therefore, when external bus release is performed, the $\overline{\text{CS}}$ signal may change from the low level to the high-impedance state.

7.11 Resets and the Bus Controller

In a power-on reset, this LSI, including the bus controller, enters the reset state at that point, and an executing bus cycle is discontinued.

In a manual reset, the bus controller's registers and internal state are maintained, and an executing external bus cycle is completed. In this case, $\overline{\text{WAIT}}$ input is ignored and write data is not guaranteed.

When the DMAC* is initialized at the manual reset, $\overline{\text{DACK}}$ and $\overline{\text{TEND}}$ output is disabled. The DMAC* operates as I/O port controlled by DDR and DR.

Note: * Supported only by the H8S/2239 Group.

Section 8 DMA Controller (DMAC)

The H8S/2239 Group has a built-in DMA controller (DMAC) which can carry out data transfer on up to 4 channels.

Note: The DMAC is supported only by the H8S/2239 Group. It is not available in the H8S/2258 Group, H8S/2238 Group, H8S/2237 Group, and H8S/2227 Group.

8.1 Features

- Selectable as short address mode or full address mode
 - Short Address Mode:
 - Maximum of 4 channels can be used
 - Dual address mode or single address mode can be selected
 - In dual address mode, one of the two addresses, transfer source and transfer destination, is specified as 24 bits and the other as 16 bits
 - In single address mode, transfer source or transfer destination address only is specified as 24 bits
 - In single address mode, transfer can be performed in one bus cycle
 - Choice of sequential mode, idle mode, or repeat mode for dual address mode and single address mode
 - Full Address Mode:
 - Maximum of 2 channels can be used
 - Transfer source and transfer destination addresses as specified as 24 bits
 - Choice of normal mode or block transfer mode
- 16-Mbyte address space can be specified directly
- Byte or word can be set as the transfer unit
- Activation sources: internal interrupt, external request, auto-request (depending on transfer mode)
 - Six 16-bit timer-pulse unit (TPU) compare match/input capture interrupts
 - Serial communication interface (SCI_0, SCI_1) transmit-data-empty interrupt, receive-data-full interrupt
 - A/D converter conversion end interrupt
 - External request
 - Auto-request
- Module stop mode can be set

A block diagram of the DMAC is shown in figure 8.1.

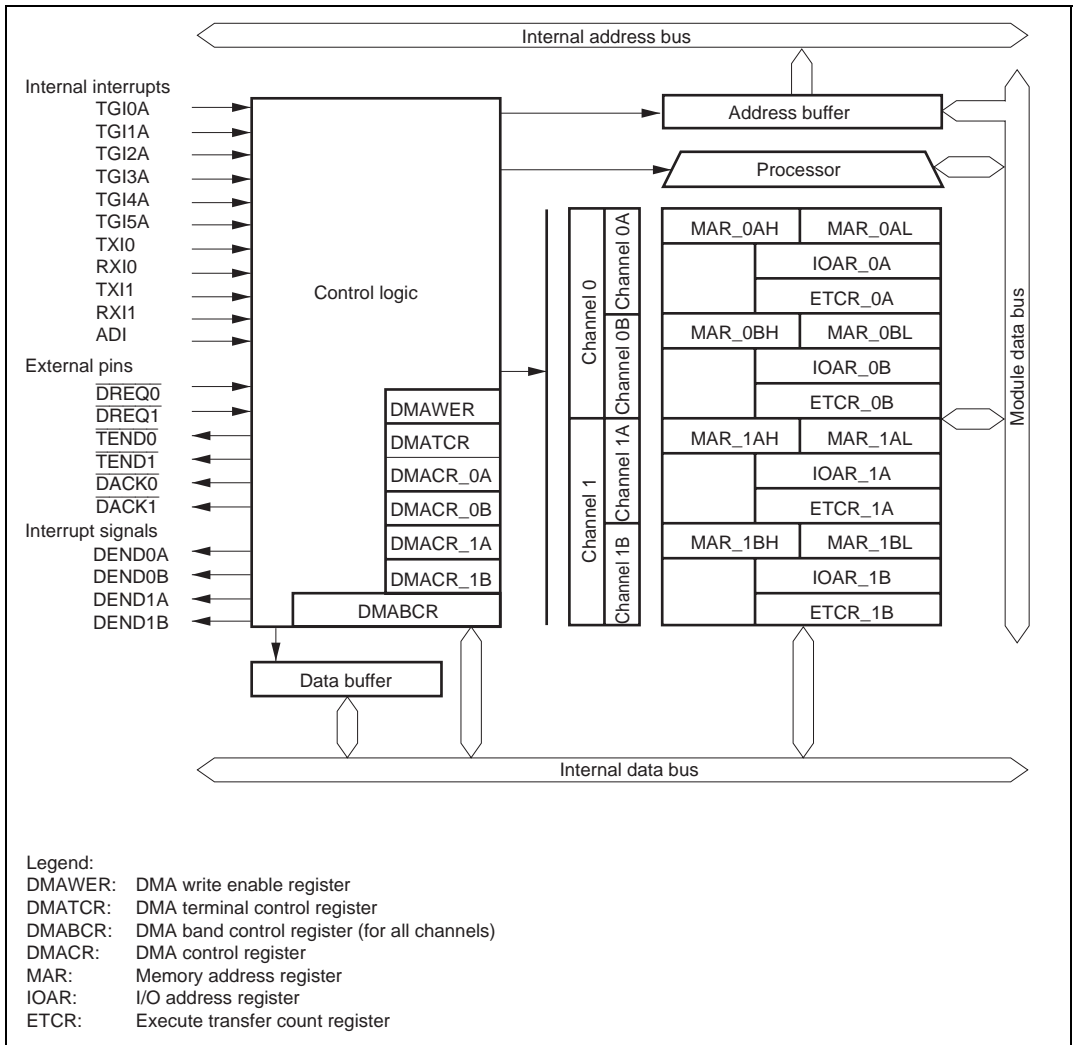


Figure 8.1 Block Diagram of DMAC

8.2 Input/Output Pins

Table 8.1 shows the pin configuration of the interrupt controller.

Table 8.1 Pin Configuration

| Channel | Pin Name | Symbol | I/O | Function |
|---------|----------------------------|---------------------------|--------|---|
| 0 | DMA request 0 | $\overline{\text{DREQ0}}$ | Input | Channel 0 external request |
| | DMA transfer acknowledge 0 | $\overline{\text{DACK0}}$ | Output | Channel 0 single address transfer acknowledge |
| | DMA transfer end 0 | $\overline{\text{TEND0}}$ | Output | Channel 0 transfer end |
| 1 | DMA request 1 | $\overline{\text{DREQ1}}$ | Input | Channel 1 external request |
| | DMA transfer acknowledge 1 | $\overline{\text{DACK1}}$ | Output | Channel 1 single address transfer acknowledge |
| | DMA transfer end 1 | $\overline{\text{TEND1}}$ | Output | Channel 1 transfer end |

8.3 Register Descriptions

- Memory address register_0AH (MAR_0AH)
- Memory address register_0AL (MAR_0AL)
- I/O address register_0A (IOAR_0A)
- Transfer count register_0A (ETCR_0A)
- Memory address register_0BH (MAR_0BH)
- Memory address register_0BL (MAR_0BL)
- I/O address register_0B (IOAR_0B)
- Transfer count register_0B (ETCR_0B)
- Memory address register_1AH (MAR_1AH)
- Memory address register_1AL (MAR_1AL)
- I/O address register_1A (IOAR_1A)
- Transfer count register_1A (ETCR_1B)
- Memory address register_1BH (MAR_1BH)
- Memory address register_1BL (MAR_1BL)
- I/O address register_1B (IOAR_1B)
- Transfer count register_1B (ETCR_1B)
- DMA control register_0A (DMACR_0A)
- DMA control register_0B (DMACR_0B)

- DMA control register_1A (DMACR_1A)
- DMA control register_1B (DMACR_1B)
- DMA band control register H (DMABCRH)
- DMA band control register L (DMABCRL)
- DMA write enable register (DMAWER)
- DMA terminal control register (DMATCR)

The functions of MAR, IOAR, ETCR, DMACR, and DMABCR differ according to the transfer mode (short address mode or full address mode). The transfer mode can be selected by means of the FAE1 and FAE0 bits in DMABCRH. The register configurations for short address mode and full address mode of channel 0 are shown in table 8.2.

Table 8.2 Short Address Mode and Full Address Mode (Channel 0)

FAE0 Description

0

Short address mode specified (channels 0A and 0B operate independently)

| | | | |
|------------|---------|----------|--|
| Channel 0A | MAR_0AH | MAR_0AL | ← Specifies transfer source/transfer destination address |
| | | IOAR_0A | ← Specifies transfer destination/transfer source address |
| | | ETCR_0A | ← Specifies number of transfers |
| | | DMACR_0A | ← Specifies transfer size, mode, activation source. |

| | | | |
|------------|---------|----------|--|
| Channel 0B | MAR_0BH | MAR_0BL | ← Specifies transfer source/transfer destination address |
| | | IOAR_0B | ← Specifies transfer destination/transfer source address |
| | | ETCR_0B | ← Specifies number of transfers |
| | | DMACR_0B | ← Specifies transfer size, mode, activation source. |

1

Full address mode specified (channels 0A and 0B operate in combination as channel 0)

| | | | |
|-----------|---------|----------|--|
| Channel 0 | MAR_0AH | MAR_0AL | ← Specifies transfer source address |
| | MAR_0BH | MAR_0BL | ← Specifies transfer destination address |
| | | IOAR_0A | ← Not used |
| | | IOAR_0B | ← Not used |
| | | ETCR_0A | ← Specifies number of transfers |
| | | ETCR_0B | ← Specifies number of transfers (used in block transfer mode only) |
| | | DMACR_0A | ← Specifies transfer size, mode, activation source, etc. |
| | | DMACR_0B | |

8.3.1 Memory Address Registers (MARA and MARB)

MAR is a 32-bit readable/writable register that specifies the source address (transfer source address) or destination address (transfer destination address). MAR consists of two 16-bit registers MARH and MARL. The upper 8 bits of MARH are reserved: they are always read as 0, and cannot be modified.

The DMA has four MAR registers: MAR_0A in channel 0 (channel 0A), MAR_0B in channel 0 (channel 0B), MAR_1A in channel 1 (channel 1A), and MAR_1B in channel 1 (channel 1B).

MAR is not initialized by a reset or in standby mode.

Short Address Mode: In short address mode, MARA and MARB operate independently. Whether MAR functions as the source address register or as the destination address register can be selected by means of the DTDIR bit in DMACR.

MAR is incremented or decremented each time a byte or word transfer is executed, so that the address specified by MAR is constantly updated.

Full Address Mode: In full address mode, MARA functions as the source address register, and MARB as the destination address register.

MAR is incremented or decremented each time a byte or word transfer is executed, so that the source or destination address is constantly updated.

8.3.2 I/O Address Registers (IOARA and IOARB)

IOAR is a 16-bit readable/writable register that specifies the lower 16 bits of the source address (transfer source address) or destination address (transfer destination address). The upper 8 bits of the transfer address are automatically set to H'FF.

The DMA has four IOAR registers: IOAR_0A in channel 0 (channel 0A), IOAR_0B in channel 0 (channel 0B), IOAR_1A in channel 1 (channel 1A), and IOAR_1B in channel 1 (channel 1B).

Whether IOAR functions as the source address register or as the destination address register can be selected by means of the DTDIR bit in DMACR.

IOAR is not incremented or decremented each time a data transfer is executed, so the address specified by IOAR is fixed.

IOAR is not initialized by a reset or in standby mode.

IOAR can be used in short address mode but not in full address mode.

8.3.3 Execute Transfer Count Registers (ETCRA and ETCRB)

ETCR is a 16-bit readable/writable register that specifies the number of transfers.

The DMA has four ETCR registers: ETCR_0A in channel 0 (channel 0A), ETCR_0B in channel 0 (channel 0B), ETCR_1A in channel 1 (channel 1A), and ETCR_1B in channel 1 (channel 1B).

ETCR is not initialized by a reset or in standby mode.

Short Address Mode: The function of ETCR in sequential mode and idle mode differs from that in repeat mode.

In sequential mode and idle mode, ETCR functions as a 16-bit transfer counter. ETCR is decremented by 1 each time a transfer is performed, and when the count reaches H'00, the DTE bit in DMABCRL is cleared, and transfer ends.

In repeat mode, ETCRL functions as an 8-bit transfer counter and ETCRH functions as a transfer count holding register. ETCRL is decremented by 1 each time a transfer is performed, and when the count reaches H'00, ETCRL is loaded with the value in ETCRH. At this point, MAR is automatically restored to the value it had when the count was started. The DTE bit in DMABCRL is not cleared, and so transfers can be performed repeatedly until the DTE bit is cleared by the user.

Full Address Mode: The function of ETCR in normal mode differs from that in block transfer mode.

In normal mode, ETCRA functions as a 16-bit transfer counter. ETCRA is decremented by 1 each time a data transfer is performed, and transfer ends when the count reaches H'0000. ETCRB is not used in normal mode.

In block transfer mode, ETCRAL functions as an 8-bit block size counter and ETCRAH functions as a block size holding register. ETCRAL is decremented by 1 each time a 1-byte or 1-word transfer is performed, and when the count reaches H'00, ETCRAL is loaded with the value in ETCRAH. So by setting the block size in ETCRAH and ETCRAL, it is possible to repeatedly transfer blocks consisting of any desired number of bytes or words.

In block transfer mode, ETCRB functions as a 16-bit block transfer counter. ETCRB is decremented by 1 each time a block is transferred, and transfer ends when the count reaches H'0000.

8.3.4 DMA Control Registers (DMACRA and DMACRB)

DMACR controls the operation of each DMAC channel.

The DMA has four DMACR registers: DMACR_0A in channel 0 (channel 0A), DMACR_0B in channel 0 (channel 0B), DMACR_1A in channel 1 (channel 1A), and DMACR_1B in channel 1 (channel 1B).

In short address mode, channels A and B operate independently, and in full address mode, channels A and B operate together. The bit functions in the DMACR registers differ according to the transfer mode.

(1) Short Address Mode

- DMACR_0A, DMACR_0B, DMACR_1A, and DMARC_1B

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | DTSZ | 0 | R/W | Data Transfer Size Selects the size of data to be transferred at one time. 0: Byte-size transfer 1: Word-size transfer |
| 6 | DTID | 0 | R/W | Data Transfer Increment/Decrement Selects incrementing or decrementing of MAR after every data transfer in sequential mode or repeat mode. In idle mode, MAR is neither incremented nor decremented. 0: MAR is incremented after a data transfer (Initial value) <ul style="list-style-type: none"> • When DTSZ = 0, MAR is incremented by 1 • When DTSZ = 1, MAR is incremented by 2 1: MAR is decremented after a data transfer <ul style="list-style-type: none"> • When DTSZ = 0, MAR is decremented by 1 • When DTSZ = 1, MAR is decremented by 2 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | RPE | 0 | R/W | <p>Repeat Enable</p> <p>Used in combination with the DTIE bit in DMABCR to select the mode (sequential, idle, or repeat) in which transfer is to be performed.</p> <p>When DTIE = 0 (no transfer end interrupt)</p> <p>0: Transfer in sequential mode</p> <p>1: Transfer in repeat mode</p> <p>When DTIE = 1 (with transfer end interrupt)</p> <p>0: Transfer in sequential mode</p> <p>1: Transfer in idle mode</p> |
| 4 | DTDIR | 0 | R/W | <p>Data Transfer Direction</p> <p>Used in combination with the SAE bit in DMABCR to specify the data transfer direction (source or destination). The function of this bit is therefore different in dual address mode and single address mode.</p> <p>When SAE = 0</p> <p>0: Transfer with MAR as source address and IOAR as destination address</p> <p>1: Transfer with IOAR as source address and MAR as destination address</p> <p>When SAE = 1</p> <p>0: Transfer with MAR as source address and DACK pin as write strobe</p> <p>1: Transfer with $\overline{\text{DACK}}$ pin as read strobe and MAR as destination address</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | DTF3 | 0 | R/W | Data Transfer Factor 3 to 0 |
| 2 | DTF2 | 0 | R/W | These bits select the data transfer factor (activation source). There are some differences in activation sources for channel A and channel B. Channel A: 0000: Setting prohibited 0001: Activated by A/D converter conversion end interrupt 0010: Setting prohibited 0011: Setting prohibited 0100: Activated by SCI channel 0 transmit-data-empty interrupt 0101: Activated by SCI channel 0 receive-data-full interrupt 0110: Activated by SCI channel 1 transmit-data-empty interrupt 0111: Activated by SCI channel 1 receive-data-full interrupt 1000: Activated by TPU channel 0 compare match/input capture A interrupt 1001: Activated by TPU channel 1 compare match/input capture A interrupt 1010: Activated by TPU channel 2 compare match/input capture A interrupt 1011: Activated by TPU channel 3 compare match/input capture A interrupt 1100: Activated by TPU channel 4 compare match/input capture A interrupt 1101: Activated by TPU channel 5 compare match/input capture A interrupt 1110: Setting prohibited 1111: Setting prohibited |
| 1 | DTF1 | 0 | R/W | |
| 0 | DTF0 | 0 | R/W | |
| | | | | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | DTF3 | 0 | R/W | Channel B: |
| 2 | DTF2 | 0 | R/W | 0000: Setting prohibited |
| 1 | DTF1 | 0 | R/W | 0001: Activated by A/D converter conversion end interrupt |
| 0 | DTF0 | 0 | R/W | 0010: Activated by $\overline{\text{DREQ}}$ pin falling edge input (detected as a low level in the first transfer after transfer is enabled) |
| | | | | 0011: Activated by $\overline{\text{DREQ}}$ pin low-level input |
| | | | | 0100: Activated by SCI channel 0 transmit-data-empty interrupt |
| | | | | 0101: Activated by SCI channel 0 receive-data-full interrupt |
| | | | | 0110: Activated by SCI channel 1 transmit-data-empty interrupt |
| | | | | 0111: Activated by SCI channel 1 receive-data-full interrupt |
| | | | | 1000: Activated by TPU channel 0 compare match/input capture A interrupt |
| | | | | 1001: Activated by TPU channel 1 compare match/input capture A interrupt |
| | | | | 1010: Activated by TPU channel 2 compare match/input capture A interrupt |
| | | | | 1011: Activated by TPU channel 3 compare match/input capture A interrupt |
| | | | | 1100: Activated by TPU channel 4 compare match/input capture A interrupt |
| | | | | 1101: Activated by TPU channel 5 compare match/input capture A interrupt |
| | | | | 1110: Setting prohibited |
| | | | | 1111: Setting prohibited |
| | | | | The same factor can be selected for more than one channel. In this case, activation starts with the highest-priority channel according to the relative channel priorities. For relative channel priorities, see section 8.5.11, Multi-Channel Operation. |

(2) Full Address Mode

- DMACR_0A and DMACR_1A

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | DTSZ | 0 | R/W | Data Transfer Size Selects the size of data to be transferred at one time. 0: Byte-size transfer 1: Word-size transfer |
| 14 | SAID | 0 | R/W | Source Address Increment/Decrement |
| 13 | SAIDE | 0 | R/W | Source Address Increment/Decrement Enable These bits specify whether source address register MARA is to be incremented, decremented, or left unchanged, when data transfer is performed. 00: MARA is fixed 01: MARA is incremented after a data transfer <ul style="list-style-type: none"> • When DTSZ = 0, MARA is incremented by 1 • When DTSZ = 1, MARA is incremented by 2 10: MARA is fixed 11: MARA is decremented after a data transfer <ul style="list-style-type: none"> • When DTSZ = 0, MARA is decremented by 1 • When DTSZ = 1, MARA is decremented by 2 |

| Bit | Bit Name | Initial Value | R/W | Description |
|---------|----------|---------------|-----|---|
| 12 | BLKDIR | 0 | R/W | Block Direction |
| 11 | BLKE | 0 | R/W | Block Enable |
| | | | | These bits specify whether normal mode or block transfer mode is to be used for data transfer. If block transfer mode is specified, the BLKDIR bit specifies whether the source side or the destination side is to be the block area. |
| | | | | ×0: Transfer in normal mode |
| | | | | 01: Transfer in block transfer mode (destination side is block area) |
| | | | | 11: Transfer in block transfer mode (source side is block area) |
| 10 to 8 | — | All 0 | R/W | Reserved |
| | | | | These bits can be read from or written to. However, the write value should always be 0. |

Legend:

×: Don't care

- DMACR_0B and DMACR_1B

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | — | 0 | R/W | Reserved This bit can be read from or written to. However, the write value should always be 0. |
| 6 | DAID | 0 | R/W | Destination Address Increment/Decrement |
| 5 | DAIDE | 0 | R/W | Destination Address Increment/Decrement Enable These bits specify whether destination address register MARB is to be incremented, decremented, or left unchanged, when data transfer is performed. 00: MARB is fixed 01: MARB is incremented after a data transfer <ul style="list-style-type: none"> When DTSZ = 0, MARB is incremented by 1 When DTSZ = 1, MARB is incremented by 2 10: MARB is fixed 11: MARB is decremented after a data transfer <ul style="list-style-type: none"> When DTSZ = 0, MARB is decremented by 1 When DTSZ = 1, MARB is decremented by 2 |
| 4 | — | 0 | R/W | Reserved This bit can be read from or written to. However, the write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| Bit | Bit Name | Initial Value | R/W | Description |
| 3 | DTF3 | 0 | R/W | Data Transfer Factor 3 to 0 |
| 2 | DTF2 | 0 | R/W | These bits select the data transfer factor (activation source). The factors that can be specified differ between normal mode and block transfer mode. |
| 1 | DTF1 | 0 | R/W | |
| 0 | DTF0 | 0 | R/W | |
| | | | | |
| | | | | Normal Mode |
| | | | | 0000: Setting prohibited |
| | | | | 0001: Setting prohibited |
| | | | | 0010: Activated by $\overline{\text{DREQ}}$ pin falling edge input (detected as a low level in the first transfer after transfer is enabled) |
| | | | | 0011: Activated by $\overline{\text{DREQ}}$ pin low-level input |
| | | | | 010x: Setting prohibited |
| | | | | 0110: Auto-request (cycle steal) |
| | | | | 0111: Auto-request (burst) |
| | | | | 1xxx: Setting prohibited |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | DTF3 | 0 | R/W | Block Transfer Mode |
| 2 | DTF2 | 0 | R/W | 0000: Setting prohibited |
| 1 | DTF1 | 0 | R/W | 0001: Activated by A/D converter conversion end interrupt |
| 0 | DTF0 | 0 | R/W | 0010: Activated by $\overline{\text{DREQ}}$ pin falling edge input (detected as a low level in the first transfer after transfer is enabled) |
| | | | | 0011: Activated by $\overline{\text{DREQ}}$ pin low-level input |
| | | | | 0100: Activated by SCI channel 0 transmit-data-empty interrupt |
| | | | | 0101: Activated by SCI channel 0 receive-data-full interrupt |
| | | | | 0110: Activated by SCI channel 1 transmit-data-empty interrupt |
| | | | | 0111: Activated by SCI channel 1 receive-data-full interrupt |
| | | | | 1000: Activated by TPU channel 0 compare match/input capture A interrupt |
| | | | | 1001: Activated by TPU channel 1 compare match/input capture A interrupt |
| | | | | 1010: Activated by TPU channel 2 compare match/input capture A interrupt |
| | | | | 1011: Activated by TPU channel 3 compare match/input capture A interrupt |
| | | | | 1100: Activated by TPU channel 4 compare match/input capture A interrupt |
| | | | | 1101: Activated by TPU channel 5 compare match/input capture A interrupt |
| | | | | 1110: Setting prohibited |
| | | | | 1111: Setting prohibited |
| | | | | The same factor can be selected for more than one channel. In this case, activation starts with the highest-priority channel according to the relative channel priorities. For relative channel priorities, see section 8.5.11, Multi-Channel Operation. |

Legend:

×: Don't care

8.3.5 DMA Band Control Registers H and L (DMABCRH and DMABCRL)

DMABCR controls the operation of each DMAC channel. The bit functions in the DMACR registers differ according to the transfer mode.

(1) Short Address Mode

- DMABCRH

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | FAE1 | 0 | R/W | Full Address Enable 1 Specifies whether channel 1 is to be used in short address mode or full address mode. In short address mode, channels 1A and 1B can be used as independent channels. 0: Short address mode 1: Full address mode |
| 14 | FAE0 | 0 | R/W | Full Address Enable 0 Specifies whether channel 0 is to be used in short address mode or full address mode. In short address mode, channels 0A and 0B can be used as independent channels. 0: Short address mode 1: Full address mode |
| 13 | SAE1 | 0 | R/W | Single Address Enable 1 Specifies whether channel 1B is to be used for transfer in dual address mode or single address mode. This bit is invalid in full address mode. 0: Dual address mode 1: Single address mode |
| 12 | SAE0 | 0 | R/W | Single Address Enable 0 Specifies whether channel 0B is to be used for transfer in dual address mode or single address mode. This bit is invalid in full address mode. 0: Dual address mode 1: Single address mode |

| Bit | Bit Name | Initial Value | R/W | Description |
|---|----------|---------------|-----|------------------------------|
| 11 | DTA1B | 0 | R/W | Data Transfer Acknowledge 1B |
| 10 | DTA1A | 0 | R/W | Data Transfer Acknowledge 1A |
| 9 | DTA0B | 0 | R/W | Data Transfer Acknowledge 0B |
| 8 | DTA0A | 0 | R/W | Data Transfer Acknowledge 0A |
| <p>These bits enable or disable clearing when DMA transfer is performed for the internal interrupt source selected by the DTF3 to DTF0 bits in DMACR.</p> <p>If the DTA bit is set to 1 when DTE = 1, the internal interrupt source is cleared automatically by DMA transfer. When DTE = 1 and DTA = 1, the internal interrupt source does not issue an interrupt request to the CPU or DTC.</p> <p>If the DTA bit is cleared to 0 when DTE = 1, the internal interrupt source is not cleared when a transfer is performed, and can issue an interrupt request to the CPU or DTC in parallel. In this case, the interrupt source should be cleared by the CPU or DTC transfer.</p> <p>When DTE = 0, the internal interrupt source issues an interrupt request to the CPU or DTC regardless of the DTA bit setting.</p> <p>0: Clearing is disabled when DMA transfer is performed for the selected internal interrupt source</p> <p>1: Clearing is enabled when DMA transfer is performed for the selected internal interrupt source</p> | | | | |

- DMABCRL

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|-------------------------|
| 7 | DTE1B | 0 | R/W | Data Transfer Enable 1B |
| 6 | DTE1A | 0 | R/W | Data Transfer Enable 1A |
| 5 | DTE0B | 0 | R/W | Data Transfer Enable 0B |
| 4 | DTE0A | 0 | R/W | Data Transfer Enable 0A |

If the DTE bit is cleared to 0 when DTIE = 1, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU or DTC.

When DTE = 0, data transfer is disabled and the DMAC ignores the activation source selected by the DTF3 to DTF0 bits in DMACR.

When DTE = 1, data transfer is enabled and the DMAC waits for a request by the activation source selected by the DTF3 to DTF0 bits in DMACR. When a request is issued by the activation source, DMA transfer is executed.

0: Data transfer is disabled
1: Data transfer is enabled

[Clearing conditions]

- When initialization is performed
- When the specified number of transfers have been completed in a transfer mode other than repeat mode
- When 0 is written to the DTE bit to forcibly suspend the transfer, or for a similar reason

[Setting condition]

When 1 is written to the DTE bit after reading DTE = 0

| Bit | Bit Name | Initial Value | R/W | Description |
|--|----------|---------------|-----|---------------------------------------|
| 3 | DTIE1B | 0 | R/W | Data Transfer End Interrupt Enable 1B |
| 2 | DTIE1A | 0 | R/W | Data Transfer End Interrupt Enable 1A |
| 1 | DTIE0B | 0 | R/W | Data Transfer End Interrupt Enable 0B |
| 0 | DTIE0A | 0 | R/W | Data Transfer End Interrupt Enable 0A |
| <p>These bits enable or disable an interrupt to the CPU or DTC when transfer ends. If the DTIE bit is set to 1 when DTE = 0, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU or DTC.</p> <p>A transfer end interrupt can be canceled either by clearing the DTIE bit to 0 in the interrupt handling routine, or by performing processing to continue transfer by setting the transfer counter and address register again, and then setting the DTE bit to 1.</p> <p>0: Transfer end interrupt is disabled 1: Transfer end interrupt is enabled</p> | | | | |

(2) Full Address Mode

- DMABCRH

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 15 | FAE1 | 0 | R/W | <p>Full Address Enable 1</p> <p>Specifies whether channel 1 is to be used in short address mode or full address mode. In full address mode, channels 1A and 1B are used together as channel 1.</p> <p>0: Short address mode 1: Full address mode</p> |
| 14 | FAE0 | 0 | R/W | <p>Full Address Enable 0</p> <p>Specifies whether channel 0 is to be used in short address mode or full address mode. In full address mode, channels 0A and 0B are used together as channel 0.</p> <p>0: Short address mode 1: Full address mode</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 13, 12 | — | All 0 | R/W | Reserved These bits can be read from or written to. However, the write value should always be 0. |
| 11 | DTA1 | 0 | R/W | <p>Data Transfer Acknowledge 1</p> <p>These bits enable or disable clearing when DMA transfer is performed for the internal interrupt source selected by the DTF3 to DTF0 bits in DMACR of channel 1.</p> <p>If the DTA1 bit is set to 1 when DTE1 = 1, the internal interrupt source is cleared automatically by DMA transfer. When DTE1 = 1 and DTA1 = 1, the internal interrupt source does not issue an interrupt request to the CPU or DTC.</p> <p>If the DTA1 bit is cleared to 0 when DTE1 = 1, the internal interrupt source is not cleared when a transfer is performed, and can issue an interrupt request to the CPU or DTC in parallel. In this case, the interrupt source should be cleared by the CPU or DTC transfer.</p> <p>When DTE1 = 0, the internal interrupt source issues an interrupt request to the CPU or DTC regardless of the DTA1 bit setting.</p> <p>The state of the DTME1 bit does not affect the above operations.</p> <p>0: Clearing is disabled when DMA transfer is performed for the selected internal interrupt source</p> <p>1: Clearing is enabled when DMA transfer is performed for the selected internal interrupt source</p> |
| 10 | — | 0 | R/W | Reserved This bit can be read from or written to. However, the write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 9 | DTA0 | 0 | R/W | <p>Data Transfer Acknowledge 0</p> <p>These bits enable or disable clearing when DMA transfer is performed for the internal interrupt source selected by the DTF3 to DTF0 bits in DMACR of channel 0.</p> <p>If the DTA0 bit is set to 1 when DTE0 = 1, the internal interrupt source is cleared automatically by DMA transfer. When DTE0 = 1 and DTA0 = 1, the internal interrupt source does not issue an interrupt request to the CPU or DTC.</p> <p>If the DTA0 bit is cleared to 0 when DTE0 = 1, the internal interrupt source is not cleared when a transfer is performed, and can issue an interrupt request to the CPU or DTC in parallel. In this case, the interrupt source should be cleared by the CPU or DTC transfer.</p> <p>When DTE0 = 0, the internal interrupt source issues an interrupt request to the CPU or DTC regardless of the DTA0 bit setting.</p> <p>The state of the DTME0 bit does not affect the above operations.</p> <p>0: Clearing is disabled when DMA transfer is performed for the selected internal interrupt source</p> <p>1: Clearing is enabled when DMA transfer is performed for the selected internal interrupt source</p> |
| 8 | — | 0 | R/W | <p>Reserved</p> <p>This bit can be read from or written to. However, the write value should always be 0.</p> |

• DMABCRL

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | DTME1 | 0 | R/W | <p>Data Transfer Master Enable 1</p> <p>Together with the DTE1 bit, this bit controls enabling or disabling of data transfer on channel 1. When both the DTME1 bit and DTE1 bit are set to 1, transfer is enabled for channel 1.</p> <p>If channel 1 is in the middle of a burst mode transfer when an NMI interrupt is generated, the DTME1 bit is cleared, the transfer is interrupted, and bus mastership passes to the CPU. When the DTME1 bit is subsequently set to 1 again, the interrupted transfer is resumed. In block transfer mode, however, the DTME1 bit is not cleared by an NMI interrupt, and transfer is not interrupted.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none">• When initialization is performed• When NMI is input in burst mode• When 0 is written to the DTME1 bit <p>[Setting condition]</p> <p>When 1 is written to DTME1 after reading DTME1 = 0</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 6 | DTE1 | 0 | R/W | <p>Data Transfer Enable 1</p> <p>Enables or disables DMA transfer for the activation source selected by the DTF3 to DTF0 bits in DMACR of channel 1.</p> <p>When DTE1 = 0, data transfer is disabled and the activation source is ignored. If the activation source is an internal interrupt, an interrupt request is issued to the CPU or DTC. If the DTE1 bit is cleared to 0 when DTIE1 = 1, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU.</p> <p>When DTE1 = 1 and DTME1 = 1, data transfer is enabled and the DMAC waits for a request by the activation source. When a request is issued by the activation source, DMA transfer is executed.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When initialization is performed • When the specified number of transfers have been completed • When 0 is written to the DTE1 bit to forcibly suspend the transfer, or for a similar reason <p>[Setting condition]</p> <p>When 1 is written to the DTE1 bit after reading DTE1 = 0</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | DTME0 | 0 | R/W | <p>Data Transfer Master Enable 0</p> <p>Together with the DTE0 bit, this bit controls enabling or disabling of data transfer on channel 0. When both the DTME0 bit and DTE0 bit are set to 1, transfer is enabled for channel 0.</p> <p>If channel 0 is in the middle of a burst mode transfer when an NMI interrupt is generated, the DTME0 bit is cleared, the transfer is interrupted, and bus mastership passes to the CPU. When the DTME0 bit is subsequently set to 1 again, the interrupted transfer is resumed. In block transfer mode, however, the DTME0 bit is not cleared by an NMI interrupt, and transfer is not interrupted.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none">• When initialization is performed• When NMI is input in burst mode• When 0 is written to the DTME0 bit <p>[Setting condition]</p> <p>When 1 is written to DTME0 after reading DTME0 = 0</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 4 | DTE0 | 0 | R/W | <p>Data Transfer Enable 0</p> <p>Enables or disables DMA transfer for the activation source selected by the DTF3 to DTF0 bits in DMACR of channel 0.</p> <p>When DTE0 = 0, data transfer is disabled and the activation source is ignored. If the activation source is an internal interrupt, an interrupt request is issued to the CPU or DTC. If the DTE0 bit is cleared to 0 when DTIE0 = 1, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU.</p> <p>When DTE0 = 1 and DTME0 = 1, data transfer is enabled and the DMAC waits for a request by the activation source. When a request is issued by the activation source, DMA transfer is executed.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When initialization is performed • When the specified number of transfers have been completed • When 0 is written to the DTE0 bit to forcibly suspend the transfer, or for a similar reason <p>[Setting condition]</p> <p>When 1 is written to the DTE0 bit after reading DTE0 = 0</p> |
| 3 | DTIE1B | 0 | R/W | <p>Data Transfer Interrupt Enable 1B</p> <p>Enables or disables an interrupt to the CPU or DTC when transfer on channel 1 is interrupted. If the DTME1 bit is cleared to 0 when DTIE1B = 1, the DMAC regards this as indicating a break in the transfer, and issues a transfer break interrupt request to the CPU or DTC.</p> <p>A transfer break interrupt can be canceled either by clearing the DTIE1B bit to 0 in the interrupt handling routine, or by performing processing to continue transfer by setting the DTME1 bit to 1.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | DTIE1A | 0 | R/W | <p>Data Transfer End Interrupt Enable 1A</p> <p>Enables or disables an interrupt to the CPU or DTC when transfer ends. If the DTE1 bit is cleared to 1 when DTIE1A = 1, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU or DTC.</p> <p>A transfer end interrupt can be canceled either by clearing the DTIE1A bit to 0 in the interrupt handling routine, or by performing processing to continue transfer by setting the transfer counter and address register again, and then setting the DTE1 bit to 1.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> |
| 1 | DTIE0B | 0 | R/W | <p>Data Transfer Interrupt Enable 0B</p> <p>Enables or disables an interrupt to the CPU or DTC when transfer on channel 1 is interrupted. If the DTME0 bit is cleared to 0 when DTIE0B = 1, the DMAC regards this as indicating a break in the transfer, and issues a transfer break interrupt request to the CPU or DTC.</p> <p>A transfer break interrupt can be canceled either by clearing the DTIE0B bit to 0 in the interrupt handling routine, or by performing processing to continue transfer by setting the DTME0 bit to 1.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> |
| 0 | DTIE0A | 0 | R/W | <p>Data Transfer End Interrupt Enable 0A</p> <p>Enables or disables an interrupt to the CPU or DTC when transfer ends. If the DTE0 bit is cleared to 0 when DTIE0A = 1, the DMAC regards this as indicating the end of a transfer, and issues a transfer end interrupt request to the CPU or DTC.</p> <p>A transfer end interrupt can be canceled either by clearing the DTIE0A bit to 0 in the interrupt handling routine, or by performing processing to continue transfer by setting the transfer counter and address register again, and then setting the DTE0 bit to 1.</p> <p>0: Data transfer is disabled 1: Data transfer is enabled</p> |

8.3.6 DMA Write Enable Register (DMAWER)

The DMAC can activate the DTC with a transfer end interrupt, rewrite the channel on which the transfer ended using a DTC chain transfer, and then reactivate the DTC. DMAWER applies restrictions for changing all bits of DMACR, and specific bits for DMATCR and DMABCR for the specific channel, to prevent inadvertent rewriting of registers other than those for the channel concerned. The restrictions applied by DMAWER are valid for the DTC.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 3 | WE1B | 0 | R/W | Write Enable 1B Enables or disables writes to all bits in DMACR1B, bits 11, 7, and 3 in DMABCR, and bit 5 in DMATCR. 0: Writes are disabled 1: Writes are enabled |
| 2 | WE1A | 0 | R/W | Write Enable 1A Enables or disables writes to all bits in DMACR1A, and bits 10, 6, and 2 in DMABCR. 0: Writes are disabled 1: Writes are enabled |
| 1 | WE0B | 0 | R/W | Write Enable 0B Enables or disables writes to all bits in DMACR0B, bits 9, 5, and 1 in DMABCR, and bit 4 in DMATCR. 0: Writes are disabled 1: Writes are enabled |
| 0 | WE0A | 0 | R/W | Write Enable 0A Enables or disables writes to all bits in DMACR0A, and bits 8, 4, and 0 in DMABCR. 0: Writes are disabled 1: Writes are enabled |

Figure 8.2 shows the transfer areas for activating the DTC with a channel 0A transfer end interrupt request, and reactivating channel 0A. The address register and count register areas are set again during the first DTC transfer, then the control register area is set again during the second DTC

chain transfer. When re-setting the control register area, perform masking by setting bits in DMAWER to prevent modification of the contents of other channels.

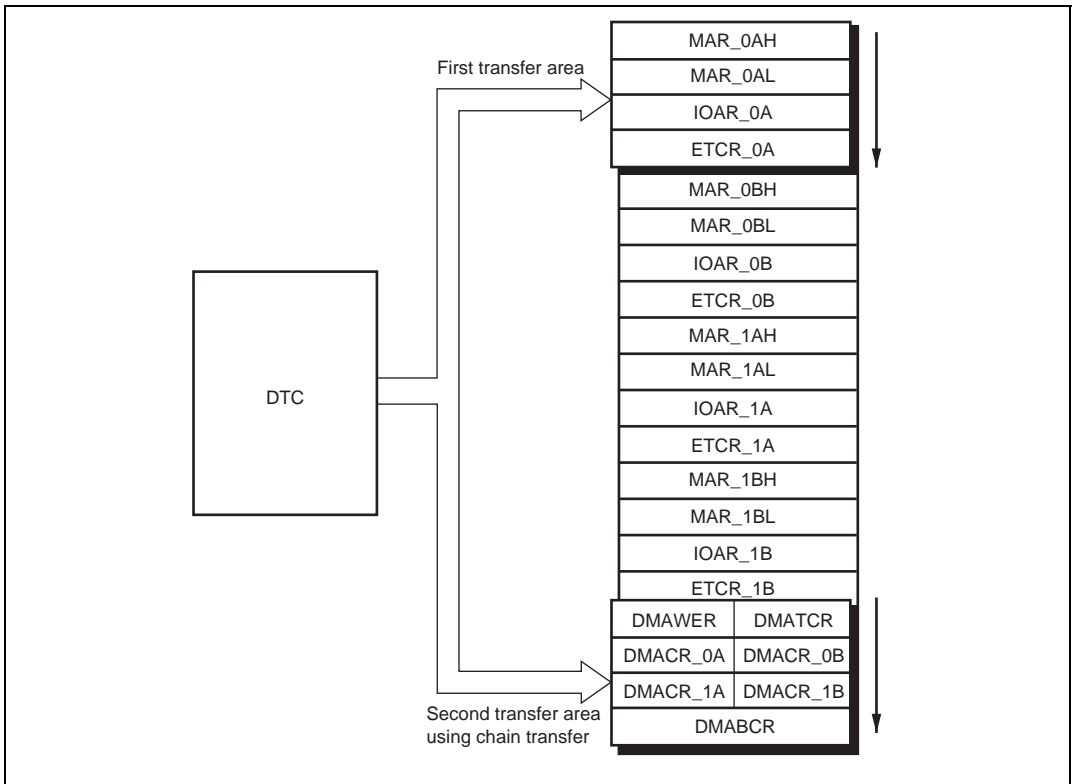


Figure 8.2 Areas for Register Re-Setting by DTC (Channel 0A)

Writes by the DTC to bits 15 to 12 (FAE and SAE) in DMABCR are invalid regardless of the DMAWER settings. These bits should be changed, if necessary, by CPU processing.

In writes by the DTC to bits 7 to 4 (DTE) in DMABCR, 1 can be written without first reading 0. To reactivate a channel set to full address mode, write 1 to both Write Enable A and Write Enable B for the channel to be reactivated.

MAR, IOAR, and ETCR can always be written to regardless of the DMAWER settings. When modifying these registers, the channel to be modified should be halted.

8.3.7 DMA Terminal Control Register (DMATCR)

DMATCR controls enabling or disabling of output from the DMAC transfer end pin. A port can be set for output automatically, and a transfer end signal output, by setting the appropriate bit. The TEND pin is available only for channel B in short address mode. Except for the block transfer mode, a transfer end signal asserts in the transfer cycle in which the transfer counter contents reaches 0 regardless of the activation source. In the block transfer mode, a transfer end signal asserts in the transfer cycle in which the block counter contents reaches 0.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7, 6 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 5 | TEE1 | 0 | R/W | Transfer End Enable 1 Enables or disables transfer end pin 1 ($\overline{TEND1}$) output. 0: $\overline{TEND1}$ pin output disabled 1: $\overline{TEND1}$ pin output enabled |
| 4 | TEE0 | 0 | R/W | Transfer End Enable 0 Enables or disables transfer end pin 0 ($\overline{TEND0}$) output. 0: $\overline{TEND0}$ pin output disabled 1: $\overline{TEND0}$ pin output enabled |
| 3 to 0 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |

8.4 Activation Sources

DMAC activation sources consist of internal interrupt requests, external requests, and auto-requests. The DMAC activation sources that can be specified depend on the transfer mode and channel, as shown in table 8.3.

Table 8.3 DMAC Activation Sources

| Activation Source | | Short Address Mode | | Full Address Mode | |
|---------------------|-----------------------------|--------------------|--------------------|-------------------|---------------------|
| | | Channels 0A and 1A | Channels 0B and 1B | Normal Mode | Block Transfer Mode |
| Internal interrupts | ADI | O | O | × | O |
| | TXI0 | O | O | × | O |
| | RXI0 | O | O | × | O |
| | TXI1 | O | O | × | O |
| | RXI1 | O | O | × | O |
| | TGI0A | O | O | × | O |
| | TGI1A | O | O | × | O |
| | TGI2A | O | O | × | O |
| | TGI3A | O | O | × | O |
| | TGI4A | O | O | × | O |
| | TGI5A | O | O | × | O |
| External requests | DREQ pin falling edge input | × | O | O | O |
| | DREQ pin low-level input | × | O | O | O |
| Auto-request | | × | × | O | × |

Legend:

O: Can be specified

×: Cannot be specified

8.4.1 Activation by Internal Interrupt Request

An interrupt request selected as a DMAC activation source can also simultaneously generate an interrupt request for the CPU or DTC. For details, see section 5, Interrupt Controller.

With activation by an internal interrupt request, the DMAC accepts the interrupt request independently of the interrupt controller. Consequently, interrupt controller priority settings are irrelevant.

If the DMAC is activated by a CPU interrupt source or an interrupt request that is not used as a DTC activation source ($DTA = 1$), the interrupt request flag is cleared automatically by the DMA transfer. With ADI, TXI, and RXI interrupts, however, the interrupt source flag is not cleared unless the relevant register is accessed in a DMA transfer. If the same interrupt is used as an activation source for more than one channel, the interrupt request flag is cleared when the highest-

priority channel is activated. Transfer requests for other channels are held pending in the DMAC, and activation is carried out in order of priority.

When $DTE = 0$ after completion of a transfer, an interrupt request from the selected activation source is not sent to the DMAC, regardless of the DTA bit setting. In this case, the relevant interrupt request is sent to the CPU or DTC.

When an interrupt request signal for DMAC activation is also used for an interrupt request to the CPU or DTC activation ($DTA = 0$), the interrupt request flag is not cleared by the DMAC.

8.4.2 Activation by External Request

If an external request (\overline{DREQ} pin) is specified as a DMAC activation source, the relevant port should be set to input mode in advance. Level sensing or edge sensing can be used for external requests.

External request operation in normal mode of short address mode or full address mode is described below.

When edge sensing is selected, a byte or word is transferred each time a high-to-low transition is detected on the \overline{DREQ} pin. The next data transfer may not be performed if the next edge is input before data transfer is completed.

When level sensing is selected, the DMAC stands by for a transfer request while the \overline{DREQ} pin is held high. While the \overline{DREQ} pin is held low, transfers continue in succession, with the bus being released each time a byte or word is transferred. If the \overline{DREQ} pin goes high in the middle of a transfer, the transfer is interrupted and the DMAC stands by for a transfer request.

8.4.3 Activation by Auto-Request

Auto-request is activated by register setting only, and transfer continues to the end. With auto-request activation, cycle steal mode or burst mode can be selected.

In cycle steal mode, the DMAC releases the bus to another bus master each time a byte or word is transferred. DMA and CPU cycles are usually repeated alternately. In burst mode, the DMAC keeps possession of the bus until the end of the transfer so that transfer is performed continuously.

8.5 Operation

8.5.1 Transfer Modes

Table 8.4 lists the DMAC transfer modes.

Table 8.4 DMAC Transfer Modes

| Transfer Mode | Transfer Source | Remarks |
|---|--|---|
| Short address mode | <ul style="list-style-type: none"> TPU channel 0 to 5 compare match/input capture A interrupt SCI transmit-data-empty interrupt SCI receive-data-full interrupt A/D converter conversion end interrupt External request | <ul style="list-style-type: none"> Up to 4 channels can operate independently External request applies to channel B only Single address mode applies to channel B only |
| Dual address mode <ul style="list-style-type: none"> 1-byte or 1-word transfer for a single transfer request Specify source and destination addresses to transfer data in two bus cycles. (1) Sequential mode <ul style="list-style-type: none"> Memory address incremented or decremented by 1 or 2 Number of transfers: 1 to 65,536 (2) Idle mode <ul style="list-style-type: none"> Memory address fixed Number of transfers: 1 to 65,536 (3) Repeat mode <ul style="list-style-type: none"> Memory address incremented or decremented by 1 or 2 Continues transfer after sending number of transfers (1 to 256) and restoring the initial value | | |
| Single address mode <ul style="list-style-type: none"> 1-byte or 1-word transfer for a single transfer request 1-bus cycle transfer by means of $\overline{\text{DACK}}$ pin instead of using address for specifying I/O Sequential mode, idle mode, or repeat mode can be specified | <ul style="list-style-type: none"> External request | |

| Transfer Mode | | Transfer Source | Remarks |
|-------------------|---|--|--|
| Full address mode | Normal mode | • Auto-request | • Max. 2-channel operation, combining channels A and B |
| | (1) Auto-request | | |
| | <ul style="list-style-type: none"> • Transfer request is internally held • Number of transfers (1 to 65,536) is continuously sent • Burst/cycle steal transfer can be selected | | |
| | (2) External request | • External request | |
| | <ul style="list-style-type: none"> • 1-byte or 1-word transfer for a single transfer request • Number of transfers: 1 to 65,536 | | |
| | Block transfer mode | <ul style="list-style-type: none"> • TPU channel 0 to 5 compare match/input capture A interrupt • SCI transmit-data-empty interrupt • SCI receive-data-full interrupt • A/D converter conversion end interrupt • External request | |

8.5.2 Sequential Mode

Sequential mode can be specified by clearing the RPE bit in DMACR to 0. In sequential mode, MAR is updated after each byte or word transfer in response to a single transfer request, and this is executed the number of times specified in ETCR. One address is specified by MAR, and the other by IOAR. The transfer direction can be specified by the DTDIR bit in DMACR.

Table 8.5 summarizes register functions in sequential mode.

Table 8.5 Register Functions in Sequential Mode

| Register | Function | | Initial Setting | Operation |
|--|------------------------------|------------------------------|--|---|
| | DTDIR = 0 | DTDIR = 1 | | |
| <div> <div>23</div> <div>0</div> <div> <div></div> <div>MAR</div> <div></div> </div> </div> | Source address register | Destination address register | Start address of transfer destination or transfer source | Incremented/decremented every transfer |
| <div> <div>23</div> <div>15</div> <div>0</div> <div> <div>H'FF</div> <div>IOAR</div> </div> </div> | Destination address register | Source address register | Start address of transfer source or transfer destination | Fixed |
| <div> <div>15</div> <div>0</div> <div>ETCR</div> </div> | Transfer counter | | Number of transfers | Decrement every transfer; transfer ends when count reaches H'0000 |

MAR specifies the start address of the transfer source or transfer destination as 24 bits. MAR is incremented or decremented by 1 or 2 each time a byte or word is transferred. IOAR specifies the lower 16 bits of the other address. The 8 bits above IOAR have a value of H'FF.

Figure 8.3 illustrates operation in sequential mode.

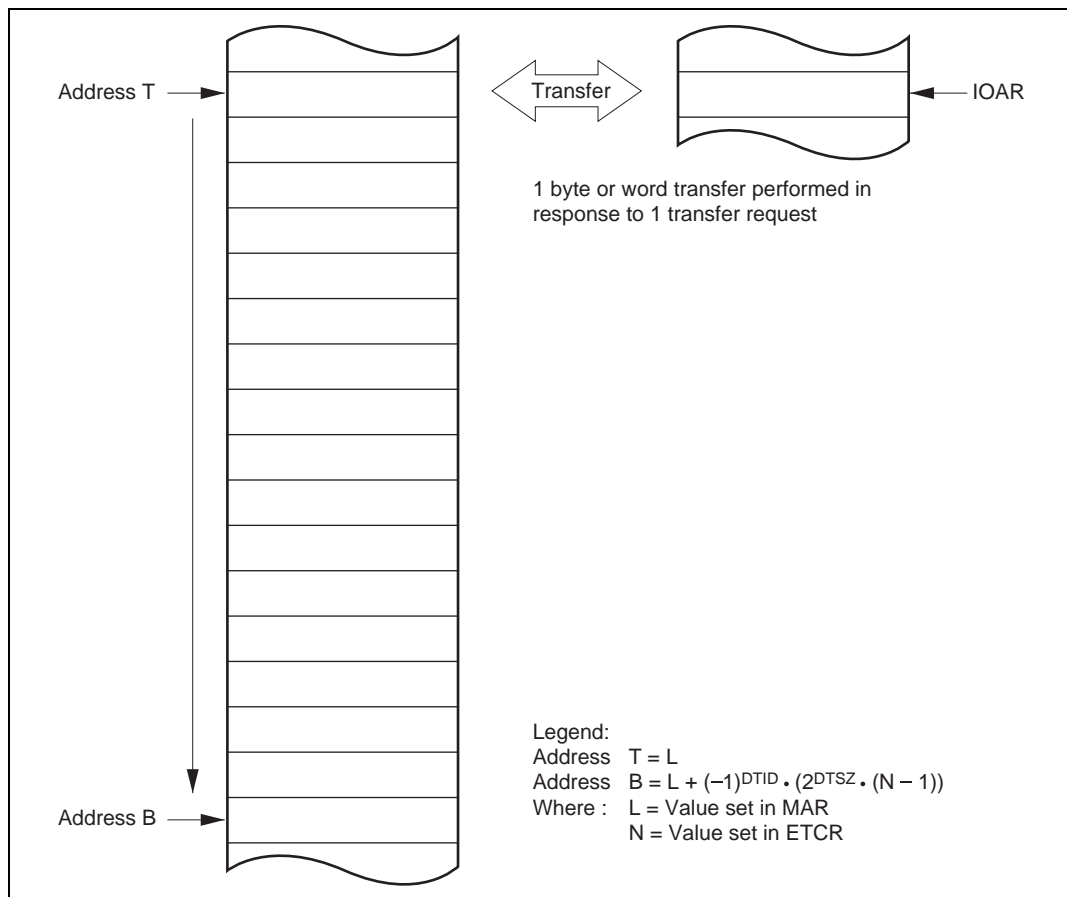
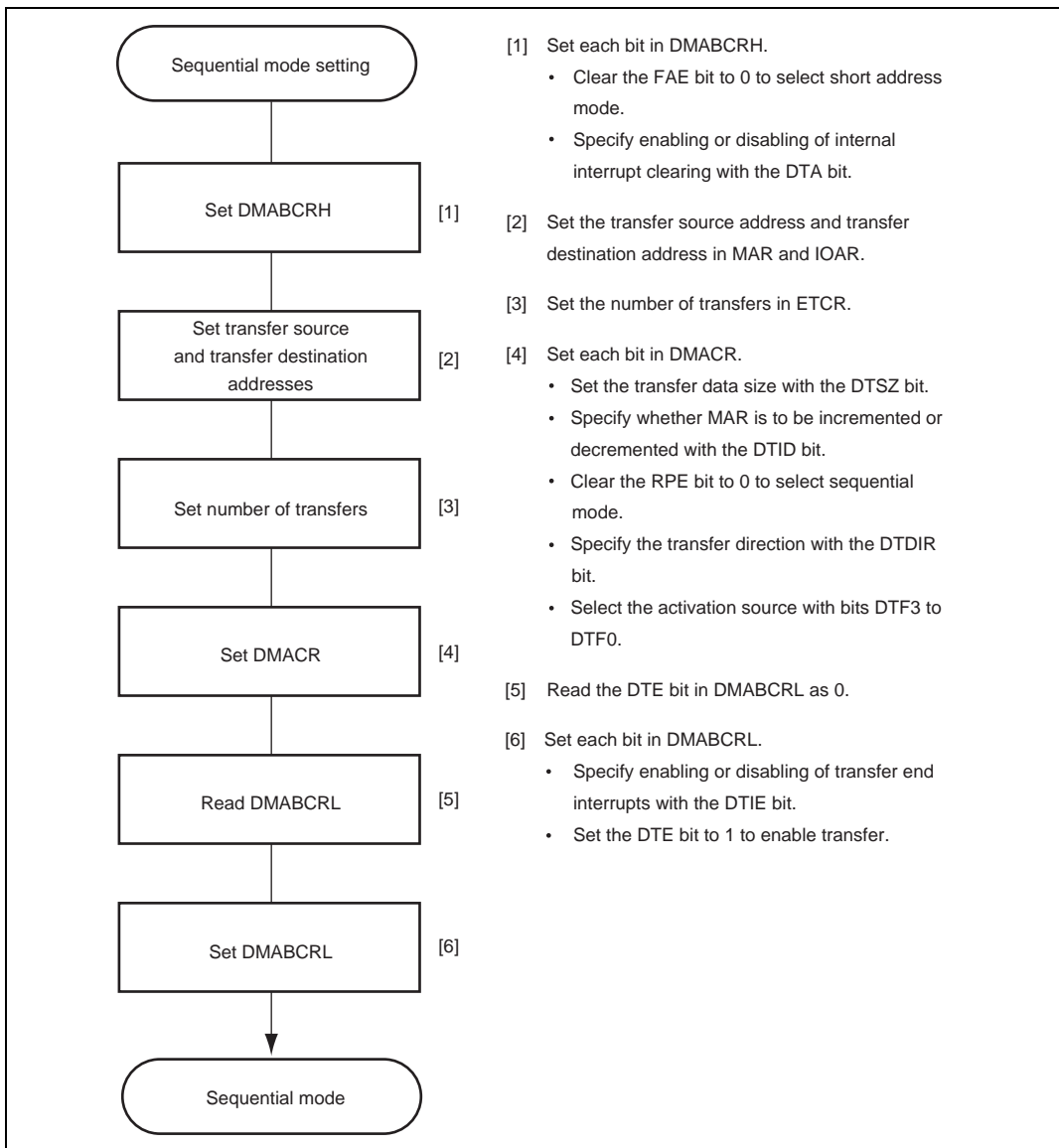


Figure 8.3 Operation in Sequential Mode

The number of transfers is specified as 16 bits in ETCR. ETCR is decremented by 1 each time a data transfer is executed, and when its value reaches H'0000, the DTE bit is cleared and data transfer ends. If the DTIE bit is set to 1 at this time, an interrupt request is sent to the CPU or DTC. The maximum number of transfers, when H'0000 is set in ETCR, is 65,536.

Transfer requests (activation sources) consist of A/D converter conversion end interrupts, external requests, SCI transmit-data-empty and receive-data-full interrupts, and TPU channel 0 to 5 compare match/input capture A interrupts. External requests can only be specified for channel B.

Figure 8.4 shows an example of the setting procedure for sequential mode.

**Figure 8.4 Example of Sequential Mode Setting Procedure**

8.5.3 Idle Mode

Idle mode can be specified by setting the RPE bit in DMACR and DTIE bit in DMABCRL to 1. In idle mode, one byte or word is transferred in response to a single transfer request, and this is executed the number of times specified in ETCR. One address is specified by MAR, and the other by IOAR. The transfer direction can be specified by the DTDIR bit in DMACR. Table 8.6 summarizes register functions in idle mode.

Table 8.6 Register Functions in Idle Mode

| Register | Function | | Initial Setting | Operation |
|---|-----------|-----------|--|---|
| | DTDIR = 0 | DTDIR = 1 | | |
| <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 10px;"> <div style="display: flex; justify-content: space-between; font-size: 0.8em;"> 230 </div> <div style="text-align: center; border: 1px solid black; width: 100px; height: 20px; margin: 2px auto;">MAR</div> </div> <div> <div style="display: flex; justify-content: space-between; font-size: 0.8em;"> Source address registerDestination address register </div> </div> </div> | | | Start address of transfer destination or transfer source | Fixed |
| <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 10px;"> <div style="display: flex; justify-content: space-between; font-size: 0.8em;"> 23150 </div> <div style="display: flex; justify-content: space-between; border: 1px solid black; width: 100px; height: 20px; margin: 2px auto;"> H'FFIOAR </div> </div> <div> <div style="display: flex; justify-content: space-between; font-size: 0.8em;"> Destination address registerSource address register </div> </div> </div> | | | Start address of transfer source or transfer destination | Fixed |
| <div style="display: flex; align-items: center;"> <div style="border: 1px solid black; padding: 2px; margin-right: 10px;"> <div style="display: flex; justify-content: space-between; font-size: 0.8em;"> 150 </div> <div style="text-align: center; border: 1px solid black; width: 60px; height: 20px; margin: 2px auto;">ETCR</div> </div> <div> <div style="display: flex; justify-content: space-between; font-size: 0.8em;"> Transfer counterNumber of transfers </div> </div> </div> | | | | Decrement every transfer; transfer ends when count reaches H'0000 |

MAR specifies the start address of the transfer source or transfer destination as 24 bits. MAR is neither incremented nor decremented by a data transfer. IOAR specifies the lower 16 bits of the other address. The upper 8 bits of IOAR have a value of H'FF.

Figure 8.5 illustrates operation in idle mode.

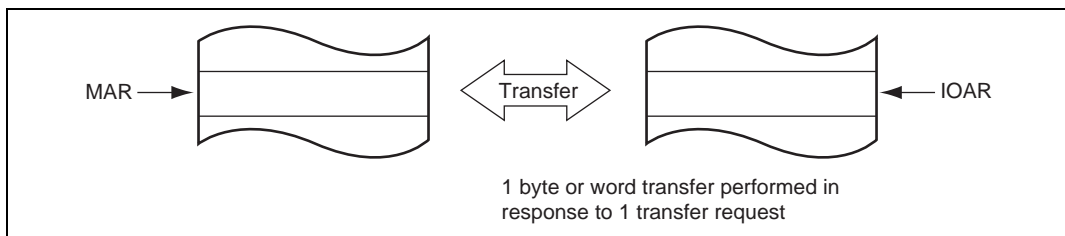


Figure 8.5 Operation in Idle Mode

The number of transfers is specified as 16 bits in ETCR. ETCR is decremented by 1 each time a transfer is executed, and when its value reaches H'0000, the DTE bit is cleared and data transfer ends. If the DTIE bit is set to 1 at this time, an interrupt request is sent to the CPU or DTC. The maximum number of transfers, when H'0000 is set in ETCR, is 65,536.

Transfer requests (activation sources) consist of A/D converter conversion end interrupts, external requests, SCI transmit-data-empty and receive-data-full interrupts, and TPU channel 0 to 5 compare match/input capture A interrupts. External requests can only be specified for channel B.

Figure 8.6 shows an example of the setting procedure for idle mode.

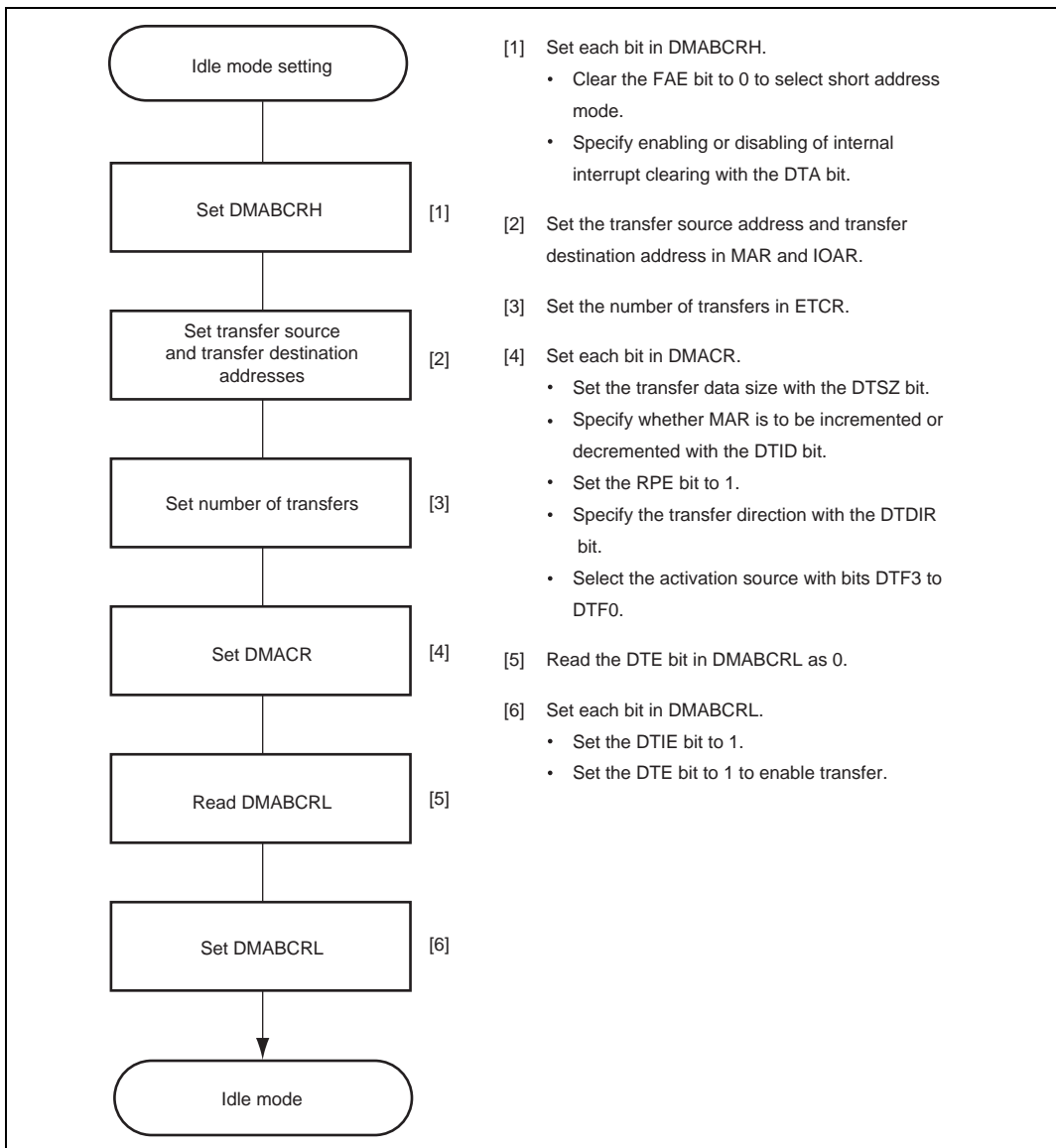






Figure 8.6 Example of Idle Mode Setting Procedure

8.5.4 Repeat Mode

Repeat mode can be specified by setting the RPE bit in DMACR to 1, and clearing the DTIE bit in DMABCRL to 0. In repeat mode, MAR is updated after each byte or word transfer in response to a single transfer request, and this is executed the number of times specified in ETCRL. On completion of the specified number of transfers, MAR and ETCRL are automatically restored to their original settings and operation continues. One address is specified by MAR, and the other by IOAR. The transfer direction can be specified by the DTDIR bit in DMACR. Table 8.7 summarizes register functions in repeat mode.

Table 8.7 Register Functions in Repeat Mode

| Register | Function | | Initial Setting | Operation |
|---|------------------------------|------------------------------|--|---|
| | DTDIR = 0 | DTDIR = 1 | | |
| <div style="text-align: center;">  </div> | Source address register | Destination address register | Start address of transfer destination or transfer source | Incremented/decremented every transfer Initial setting is restored when value reaches H'0000 |
| <div style="text-align: center;">  </div> | Destination address register | Source address register | Start address of transfer source or transfer destination | Fixed |
| <div style="text-align: center;">  </div> | Holds number of transfers | | Number of transfers | Fixed |
| <hr style="border-top: 1px dashed black;"/> | | | | |
| <div style="text-align: center;">  </div> | Transfer counter | | Number of transfers | Decrement every transfer Loaded with ETCRH value when count reaches H'00 |

MAR specifies the start address of the transfer source or transfer destination as 24 bits. MAR is incremented or decremented by 1 or 2 each time a byte or word is transferred. IOAR specifies the lower 16 bits of the other address. The upper 8 bits of IOAR have a value of H'FF. The number of transfers is specified as 8 bits by ETCRH and ETCRL. The maximum number of transfers, when H'00 is set in both ETCRH and ETCRL, is 256.

In repeat mode, ETCRL functions as the transfer counter, and ETCRH is used to hold the number of transfers. ETCRL is decremented by 1 each time a data transfer is executed, and when its value reaches H'00, it is loaded with the value in ETCRH. At the same time, the value set in MAR is

restored in accordance with the values of the DTSZ and DTID bits in DMACR. The MAR restoration operation is as shown below.

$$\text{MAR} = \text{MAR} - (-1)^{\text{DTID}} \cdot 2^{\text{DTSZ}} \cdot \text{ETCRH}$$

The same value should be set in ETCRH and ETCRL.

In repeat mode, operation continues until the DTE bit in DMABCRL is cleared. To end the transfer operation, therefore, the DTE bit should be cleared to 0. A transfer end interrupt request is not sent to the CPU or DTC. By setting the DTE bit to 1 again after it has been cleared, the operation can be restarted from the transfer after that terminated when the DTE bit was cleared.

Figure 8.7 illustrates operation in repeat mode.

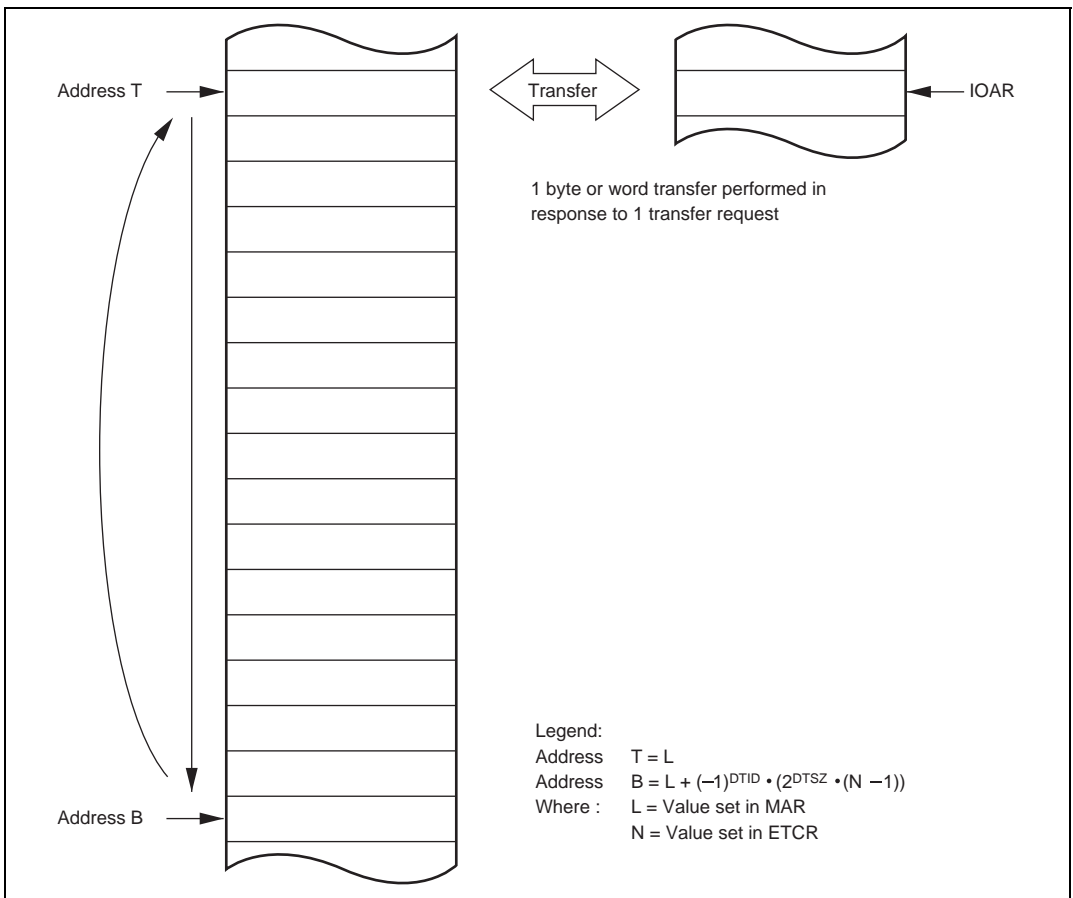


Figure 8.7 Operation in Repeat mode

Transfer requests (activation sources) consist of A/D converter conversion end interrupts, external requests, SCI transmit-data-empty and receive-data-full interrupts, and TPU channel 0 to 5 compare match/input capture A interrupts. External requests can only be specified for channel B.

Figure 8.8 shows an example of the setting procedure for repeat mode.

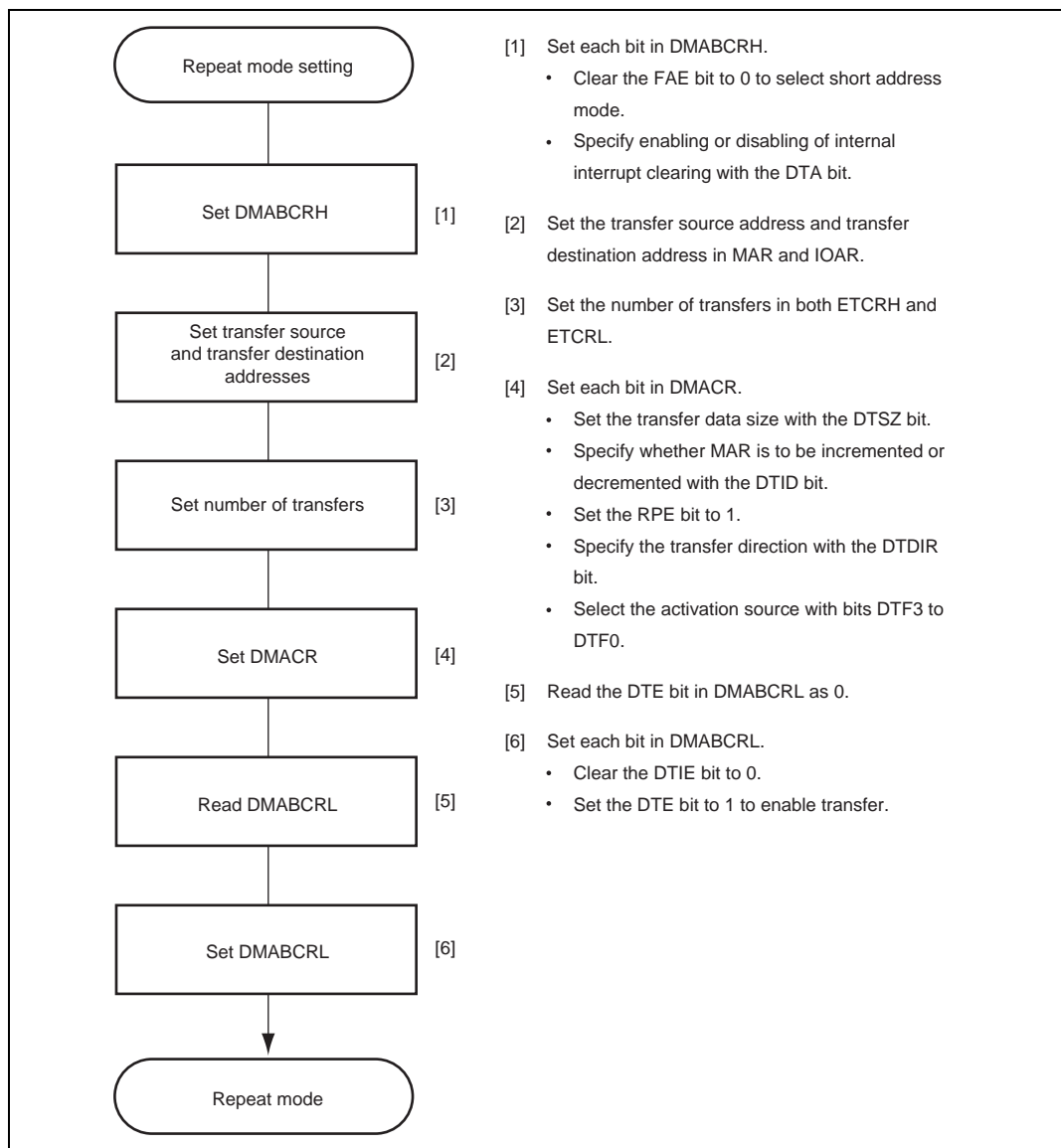


Figure 8.8 Example of Repeat Mode Setting Procedure

8.5.5 Single Address Mode

DMAC supports the dual address mode, in which two different cycles are used for reading and writing, and the single address mode, in which a single cycle is used for both reading and writing.

In dual address mode, the source address and the destination address are specified respectively for transferring data.

In single address mode, data is transferred between the external space, in which the transfer source or transfer destination is specified by the address, and the external device that is selected by $\overline{\text{DACK}}$ strobe regardless of the address. Figure 8.9 shows the data bus in single address mode.

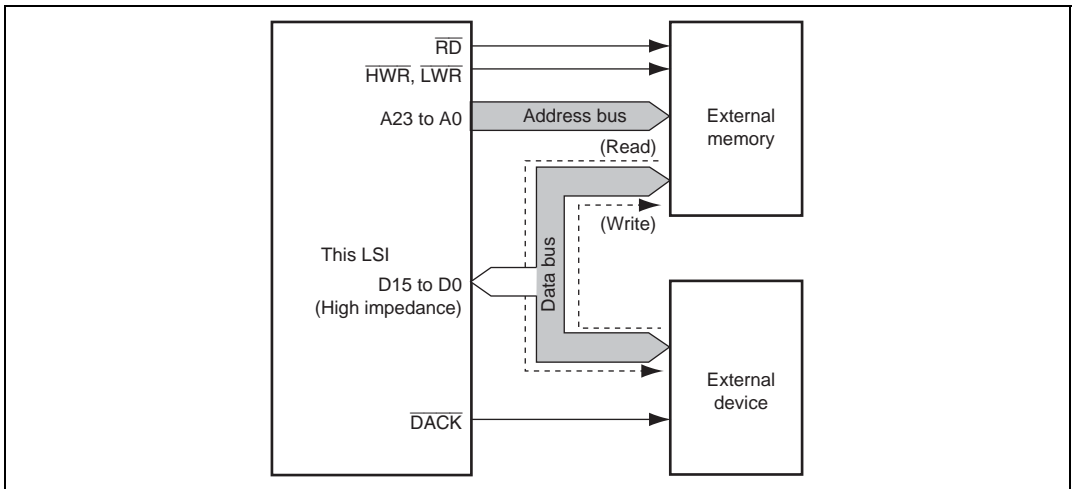


Figure 8.9 Data Bus in Single Address Mode

When the data bus is used for reading in single address mode, data is transferred from the external memory to the external device and the $\overline{\text{DACK}}$ pin functions as the write strobe for the external device. When the data bus is used for writing in single address mode, data is transferred from the external device to the external memory and the $\overline{\text{DACK}}$ pin functions as the read strobe for the external device. Since the direction for the external device cannot be controlled, choose one of the directions described above.

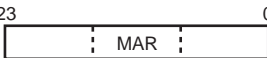

The setting of the bus controller for the external memory area controls the bus cycle in single address mode. To the external device, $\overline{\text{DACK}}$ is output in synchronization with the address strobe. For details on the bus cycle, see section 8.5.10, DMA Transfer (Single Address Mode) Bus Cycles.

In single address mode, do not specify the internal area for the transfer address.

Single address mode can only be specified for channel B. This mode can be specified by setting the SAE bit in DMABCRH to 1 in short address mode.

One address is specified by MAR, and the other is set automatically to the data transfer acknowledge pin (\overline{DACK}). The transfer direction can be specified by the DTDIR bit in DMACR. Table 8.8 summarizes register functions in single address mode.

Table 8.8 Register Functions in Single Address Mode

| Register | Function | | Initial Setting | Operation |
|---|-------------------------|------------------------------|--|--|
| | DTDIR = 0 | DTDIR = 1 | | |
|  | Source address register | Destination address register | Start address of transfer destination or transfer source | See sections 8.5.2, Sequential Mode, 8.5.3, Idle Mode, and 8.5.4, Repeat Mode. |
| \overline{DACK} pin | Write strobe | Read strobe | (Set automatically by SAE bit; IOAR is invalid) | Strobe for external device |
|  | Transfer counter | | Number of transfers | See sections 8.5.2, Sequential Mode, 8.5.3, Idle Mode, and 8.5.4, Repeat Mode. |

MAR specifies the start address of the transfer source or transfer destination as 24 bits. IOAR is invalid; in its place the strobe for external devices (\overline{DACK}) is output.

Figure 8.10 illustrates operation in single address mode (when sequential mode is specified).

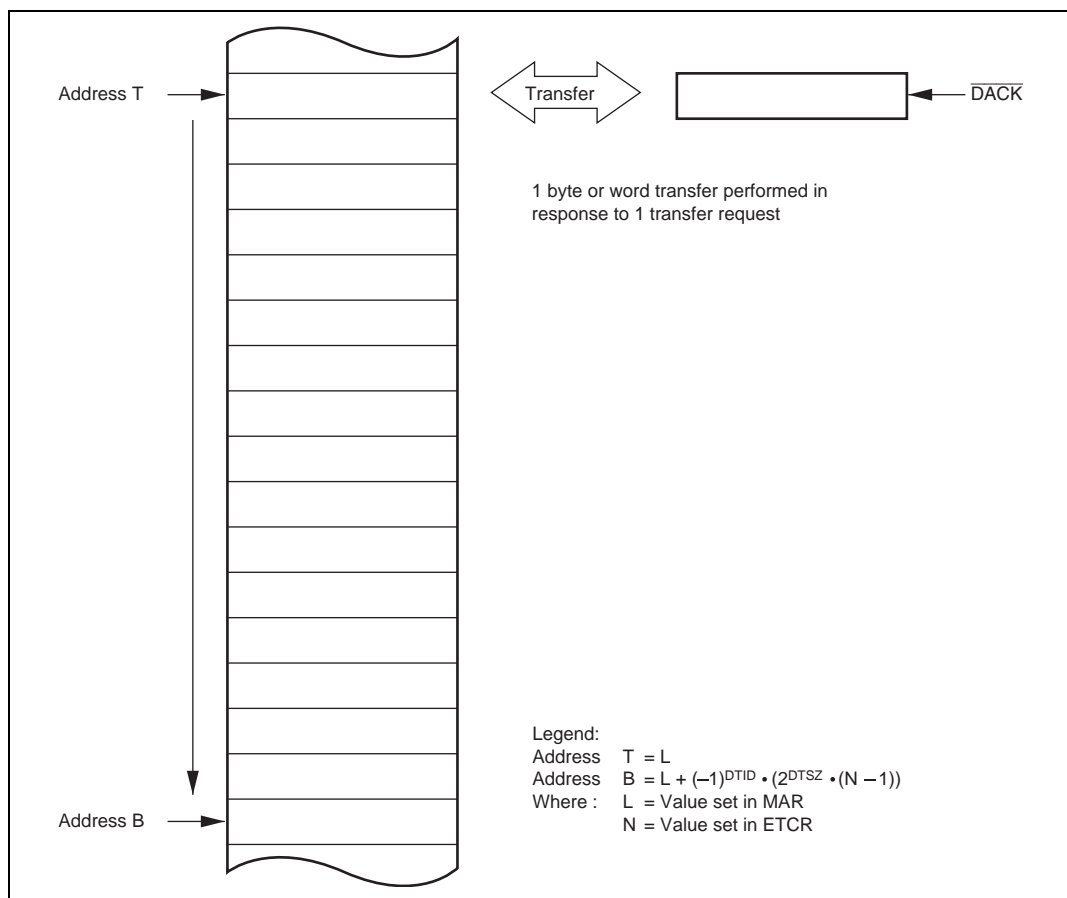
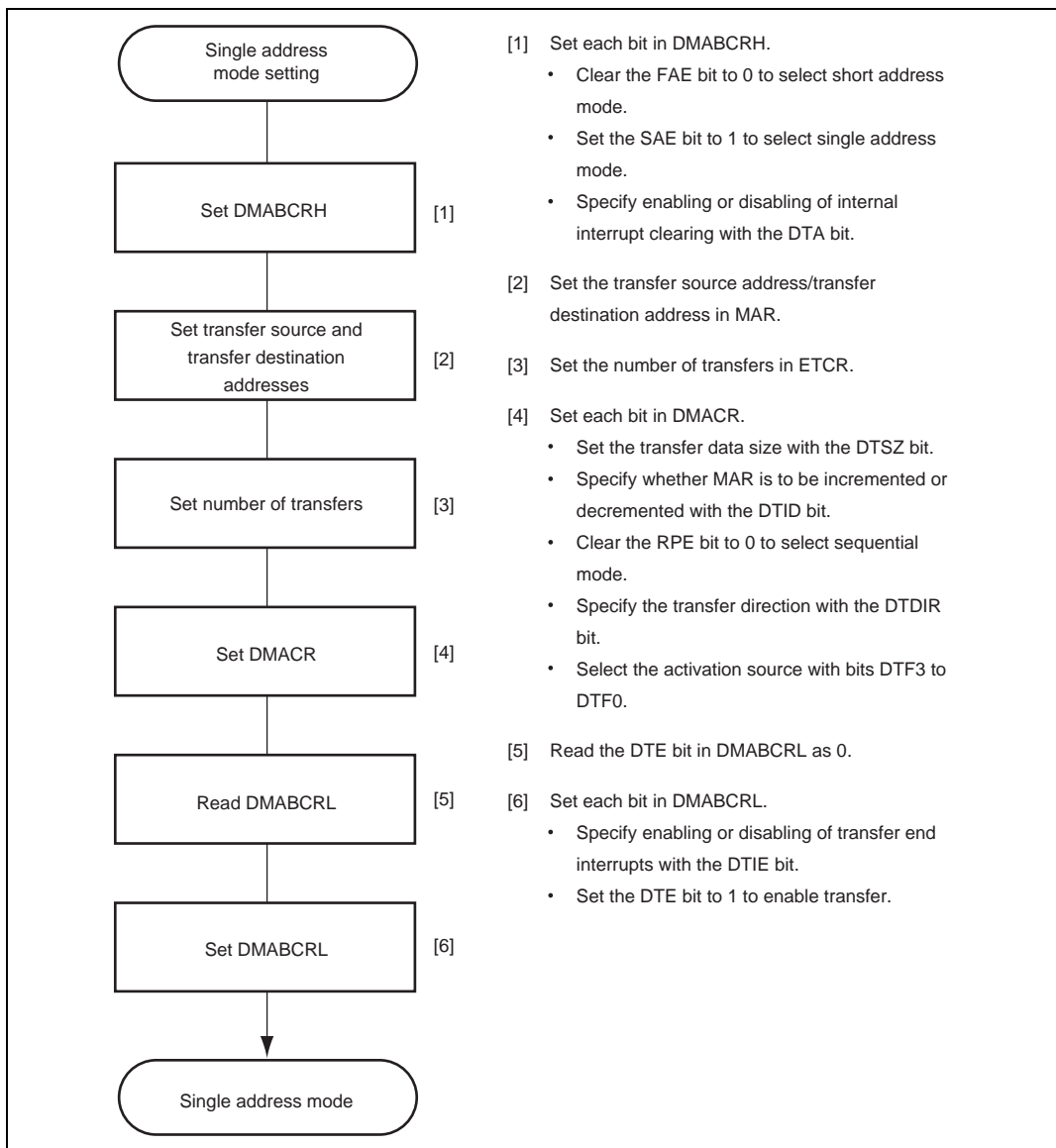


Figure 8.10 Operation in Single Address Mode (when Sequential Mode Is Specified)

Figure 8.11 shows an example of the setting procedure for single address mode (when sequential mode is specified).



**Figure 8.11 Example of Single Address Mode Setting Procedure
(when Sequential Mode Is Specified)**

8.5.6 Normal Mode

In normal mode, transfer is performed with channels A and B used in combination. Normal mode can be specified by setting the FAE bit in DMABCRH to 1 and clearing the BLKE bit in DMACRA to 0. In normal mode, MAR is updated after data transfer of a byte or word in response to a single transfer request, and this is executed the number of times specified in ETCRA. The transfer source is specified by MARA, and the transfer destination by MARB. Table 8.9 summarizes register functions in normal mode.

Table 8.9 Register Functions in Normal Mode

| Register | Function | Initial Setting | Operation |
|--|------------------------------|---------------------------------------|---|
| <div> <div>23</div> <div>0</div> <div>MARA</div> </div> | Source address register | Start address of transfer source | Incremented/decremented every transfer, or fixed |
| <div> <div>23</div> <div>0</div> <div>MARB</div> </div> | Destination address register | Start address of transfer destination | Incremented/decremented every transfer, or fixed |
| <div> <div>15</div> <div>0</div> <div>ETCRA</div> </div> | Transfer counter | Number of transfers | Decrement every transfer; transfer ends when count reaches H'0000 |

MARA and MARB specify the start addresses of the transfer source and transfer destination, respectively, as 24 bits. MAR can be incremented or decremented by 1 or 2 each time a byte or word is transferred, or can be fixed. Incrementing, decrementing, or holding a fixed value can be set separately for MARA and MARB.

The number of transfers is specified by ETCRA as 16 bits. ETCRA is decremented by 1 each time a transfer is performed, and when its value reaches H'0000 the DTE bit in DMABCRL is cleared and transfer ends. If the DTIE bit in DMABCRL is set to 1 at this time, an interrupt request is sent to the CPU or DTC. The maximum number of transfers, when H'0000 is set in ETCRA, is 65,536.

Figure 8.12 illustrates operation in normal mode.

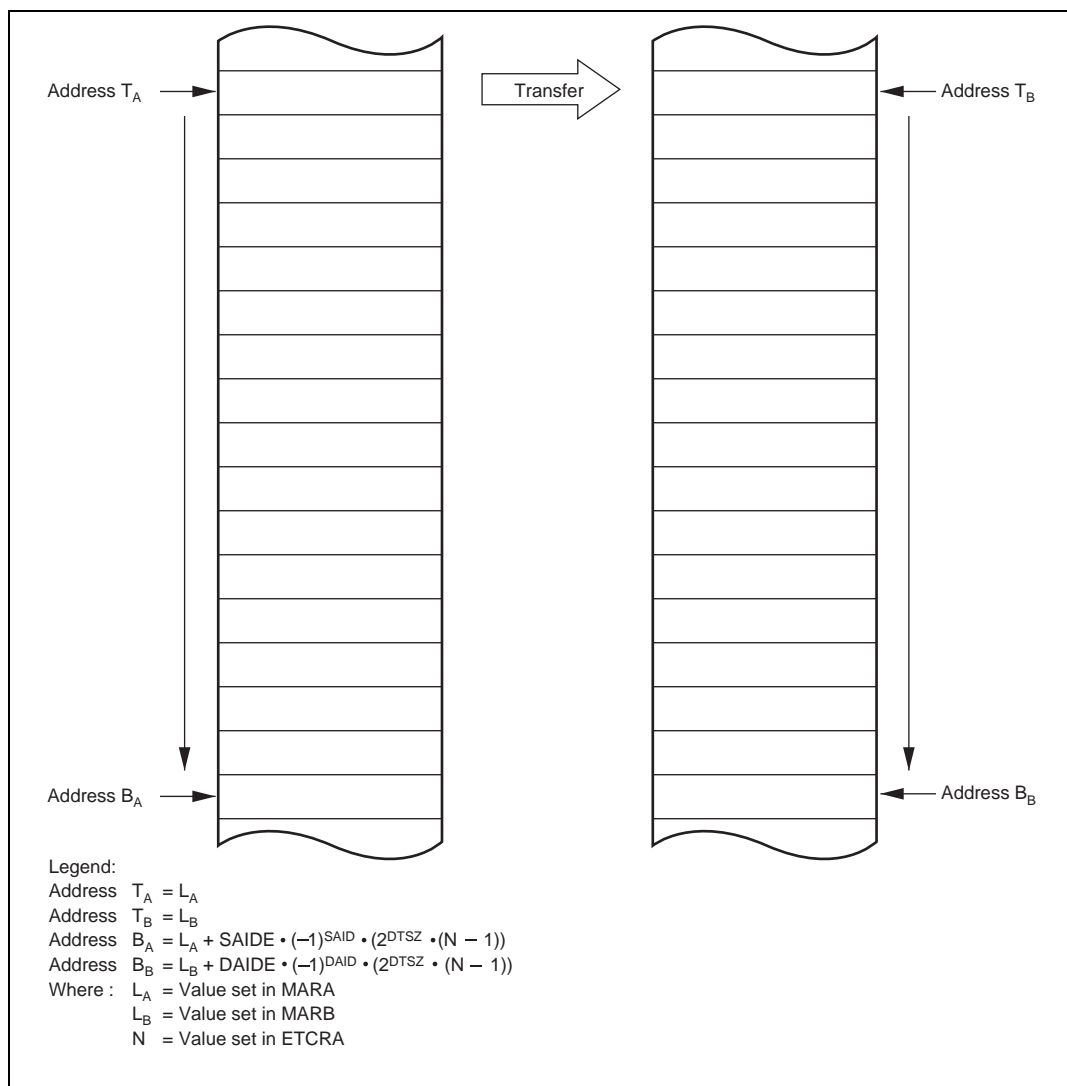


Figure 8.12 Operation in Normal Mode

Transfer requests (activation sources) are external requests and auto-requests. With auto-request, the DMAC is only activated by register setting, and the specified number of transfers are performed automatically. With auto-request, cycle steal mode or burst mode can be selected. In cycle steal mode, the bus is released to another bus master each time a transfer is performed. In burst mode, the bus is held continuously until transfer ends.

Figure 8.13 shows an example of the setting procedure for normal mode.

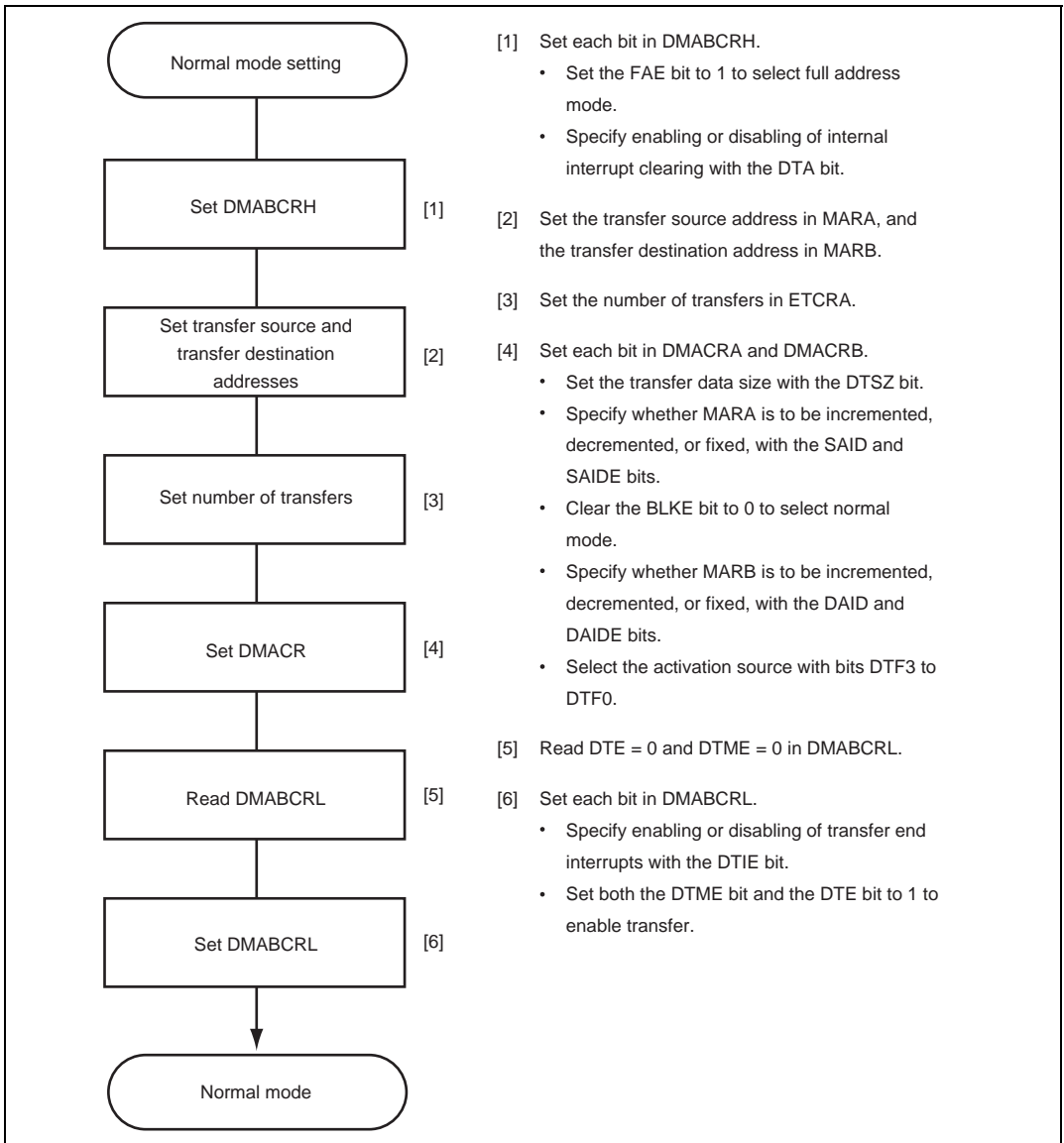







Figure 8.13 Example of Normal Mode Setting Procedure

8.5.7 Block Transfer Mode

In block transfer mode, data transfer is performed with channels A and B used in combination. Block transfer mode can be specified by setting the FAE bit in DMABCRH and the BLKE bit in DMACRA to 1. In block transfer mode, a data transfer of the specified block size is carried out in response to a single transfer request, and this is executed for the number of times specified in ETCRB. The transfer source is specified by MARA, and the transfer destination by MARB. Either the transfer source or the transfer destination can be selected as a block area (an area composed of a number of bytes or words). Table 8.10 summarizes register functions in block transfer mode.

Table 8.10 Register Functions in Block Transfer Mode

| Register | Function | Initial Setting | Operation |
|---|------------------------------|---------------------------------------|---|
|  | Source address register | Start address of transfer source | Incremented/decremented every transfer, or fixed |
|  | Destination address register | Start address of transfer destination | Incremented/decremented every transfer, or fixed |
|  | Holds block size | Block size | Fixed |
|  | Block size counter | Block size | Decrement every transfer; ETCRH value copied when count reaches H'00 |
|  | Block transfer counter | Number of block transfers | Decrement every block transfer; transfer ends when count reaches H'0000 |

MARA and MARB specify the start addresses of the transfer source and transfer destination, respectively, as 24 bits. MAR can be incremented or decremented by 1 or 2 each time a byte or word is transferred, or can be fixed. Incrementing, decrementing, or holding a fixed value can be set separately for MARA and MARB. Whether a block is to be designated for MARA or for MARB is specified by the BLKDIR bit in DMACRA.

To specify the number of transfers, if M is the size of one block (where M = 1 to 256) and N transfers are to be performed (where N = 1 to 65,536), M is set in both ETCRAH and ETCRAL, and N in ETCRB.

Figure 8.14 illustrates operation in block transfer mode when MARB is designated as a block area.

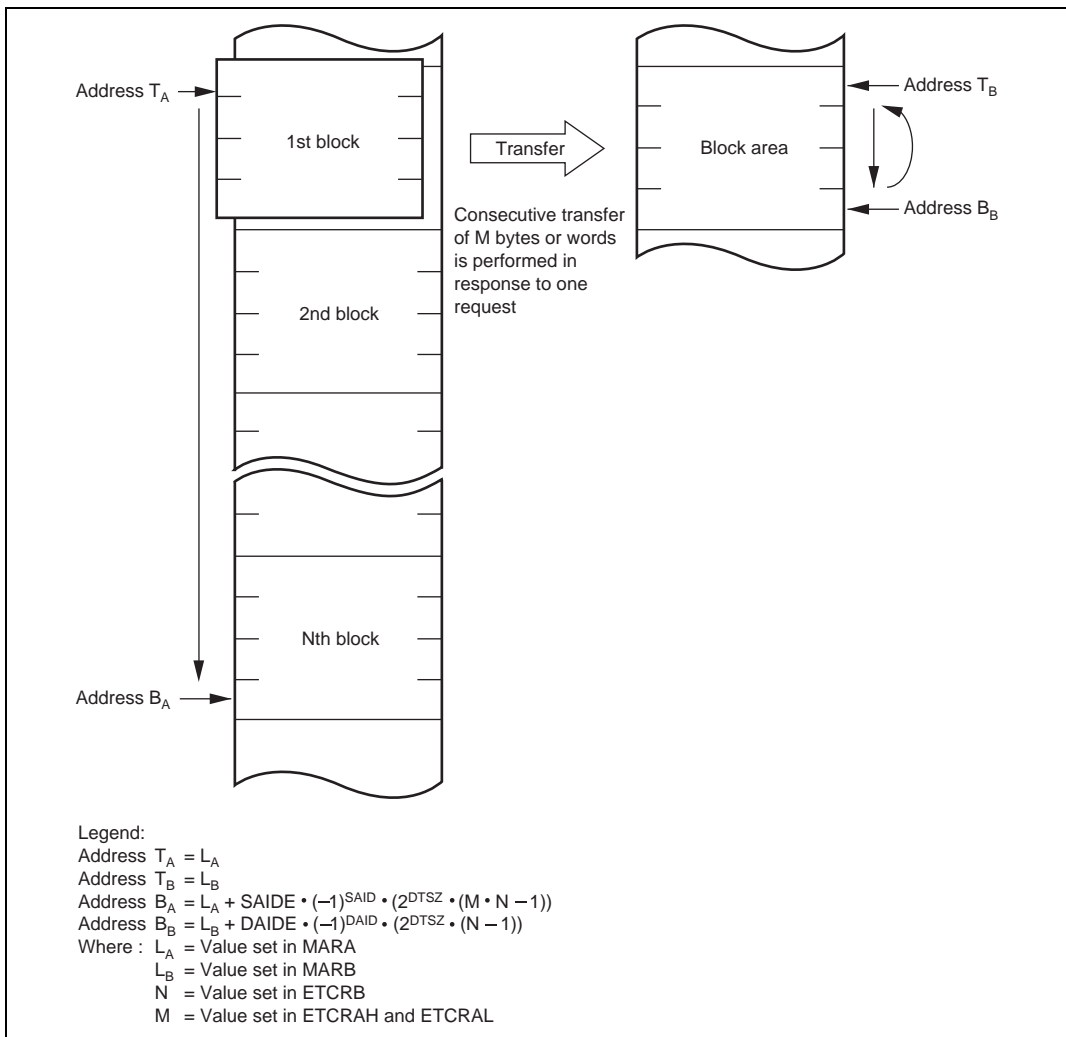


Figure 8.14 Operation in Block Transfer Mode (BLKDIR = 0)

Figure 8.15 illustrates operation in block transfer mode when MARA is designated as a block area.

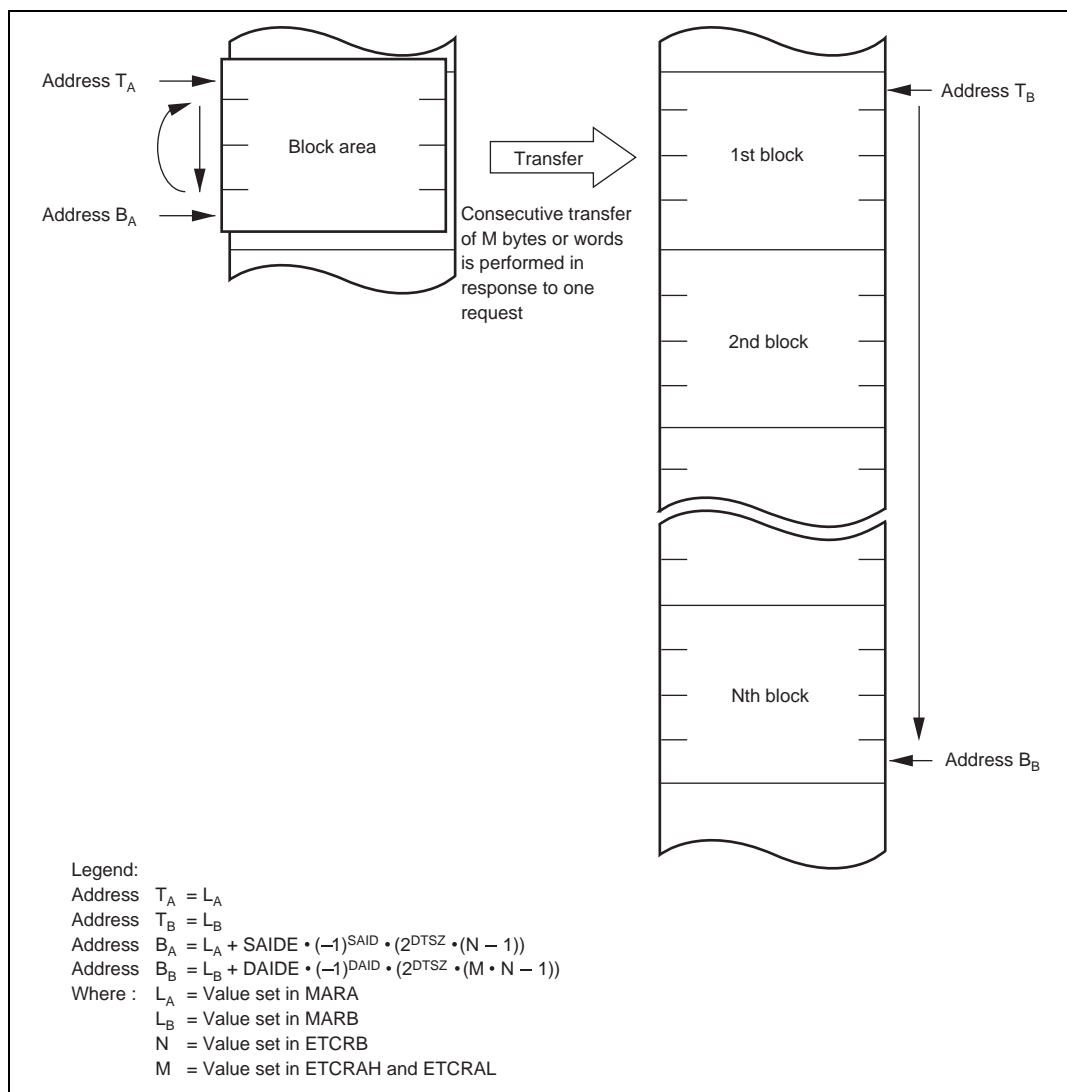


Figure 8.15 Operation in Block Transfer Mode (BLKDIR = 1)

ETCRAL is decremented by 1 each time a byte or word transfer is performed. In response to a single transfer request, burst transfer is performed until the value in ETCRAL reaches H'00. ETCRAL is then loaded with the value in ETCRAH. At this time, the value in the MAR register for which a block designation has been given by the BLKDIR bit in DMACRA is restored in accordance with the DTSZ, SAID/DAID, and SAIDE/DAIDE bits in DMACR.

ETCRB is decremented by 1 after every block transfer, and when the count reaches H'0000 the DTE bit in DMABCRL is cleared and transfer ends. If the DTIE bit in DMABCRL is set to 1 at this point, an interrupt request is sent to the CPU or DTC.

Figure 8.16 shows the operation flow in block transfer mode.

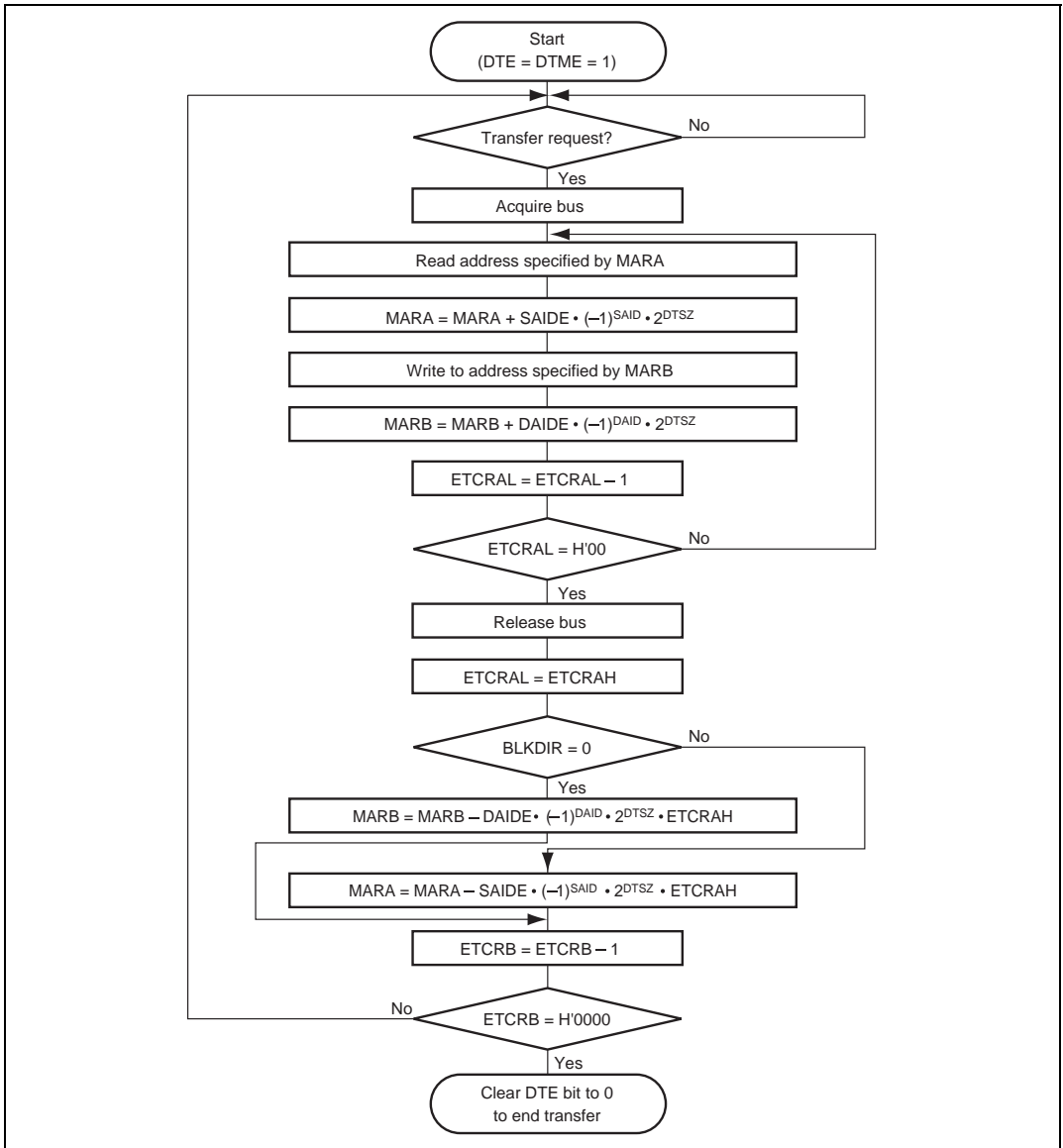


Figure 8.16 Operation Flow in Block Transfer Mode

Transfer requests (activation sources) consist of A/D converter conversion end interrupts, external requests, SCI transmit-data-empty and receive-data-full interrupts, and TPU channel 0 to 5 compare match/input capture A interrupts.

Figure 8.17 shows an example of the setting procedure for block transfer mode.

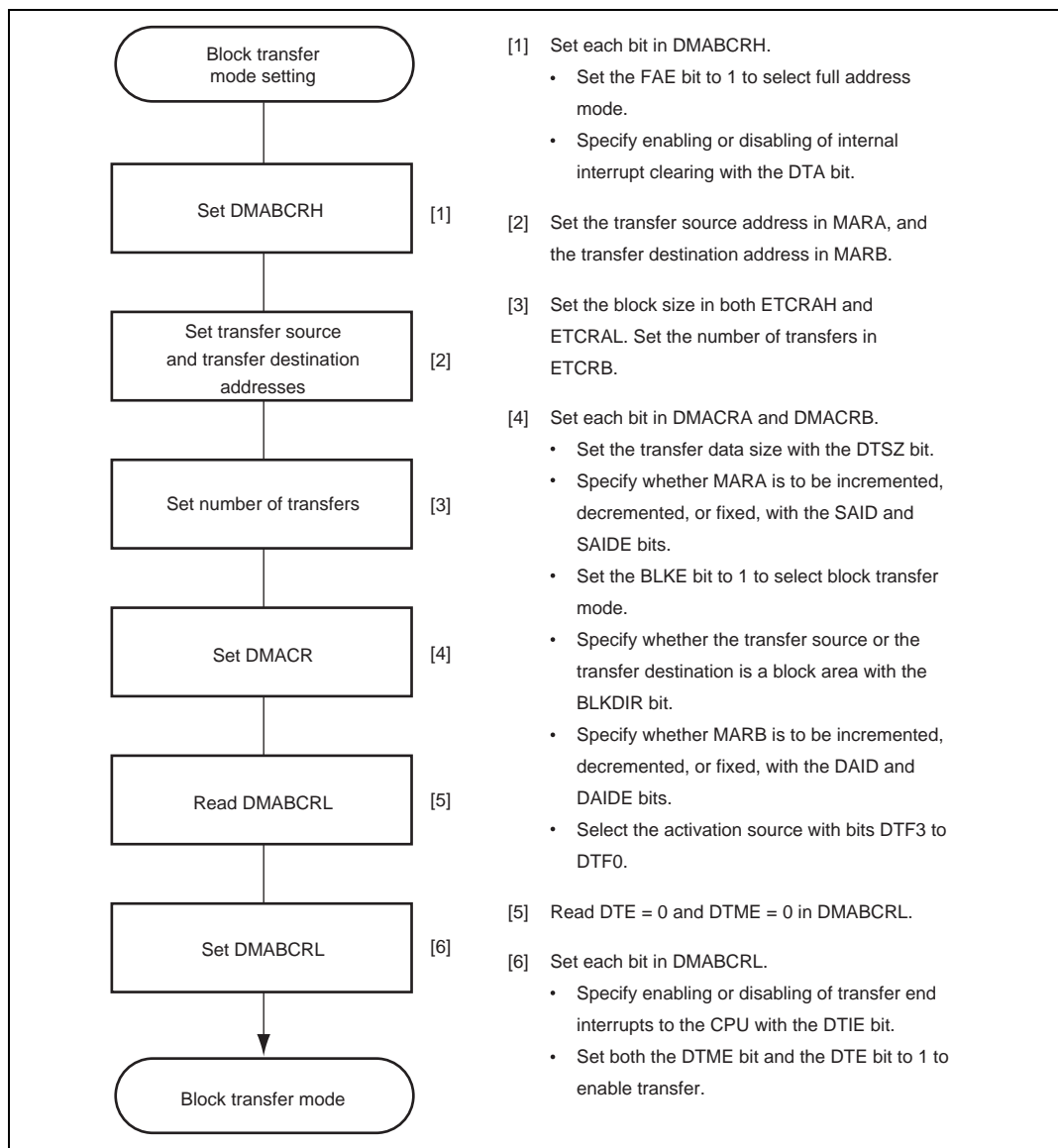


Figure 8.17 Example of Block Transfer Mode Setting Procedure

8.5.8 Basic Bus Cycles

An example of the basic DMAC bus cycle timing is shown in figure 8.18. In this example, word-size transfer is performed from 16-bit, 2-state access space to 8-bit, 3-state access space. When the bus is transferred from the CPU to the DMAC, a source address read and destination address write are performed. The bus is not released in response to another bus request, etc., between these read and write operations. As like CPU cycles, DMA cycles conform to the bus controller settings.

The address is not output to the external address bus in an access to on-chip memory or an internal I/O register.

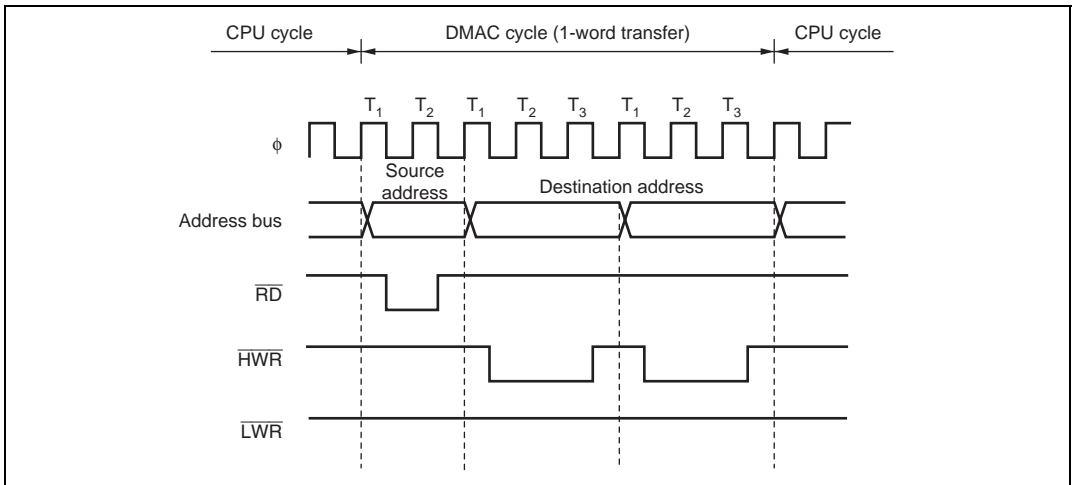


Figure 8.18 Example of DMA Transfer Bus Timing

8.5.9 DMA Transfer (Dual Address Mode) Bus Cycles

Short Address Mode: Figure 8.19 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and byte-size short address mode transfer (sequential/idle/repeat mode) is performed from external 8-bit, 2-state access space to internal I/O space.

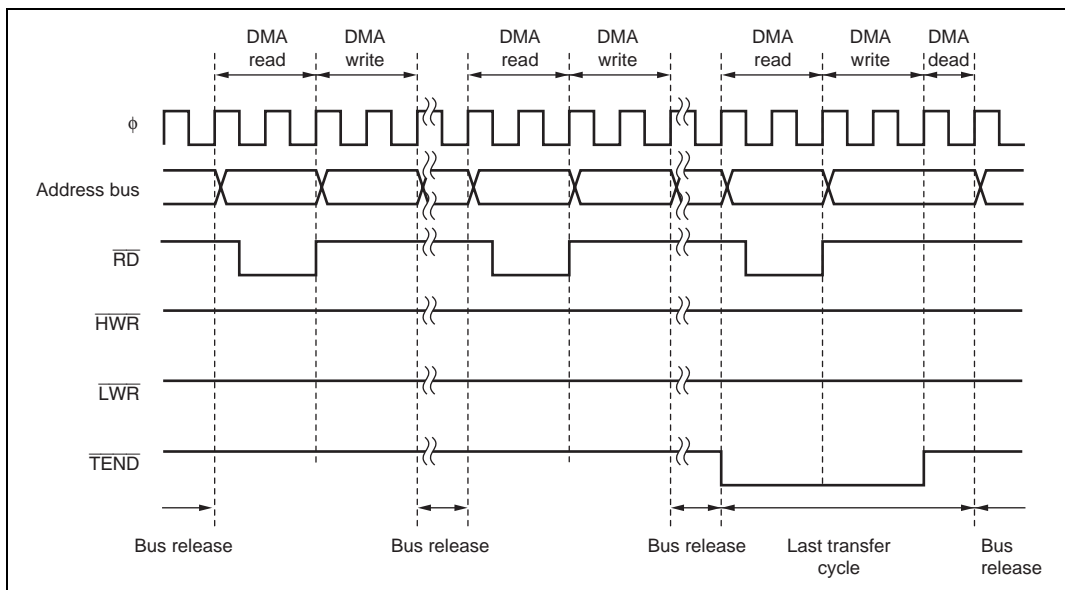


Figure 8.19 Example of Short Address Mode Transfer

A byte or word transfer is performed for a single transfer request, and after the transfer, the bus is released. While the bus is released, one or more bus cycles are executed by the CPU or DTC.

In the transfer end cycle (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle.

In repeat mode, when $\overline{\text{TEND}}$ output is enabled, $\overline{\text{TEND}}$ output goes low in the transfer end cycle.

Full Address Mode (Cycle Steal Mode): Figure 8.20 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and word-size full address mode transfer (cycle steal mode) is performed from external 16-bit, 2-state access space to external 16-bit, 2-state access space.

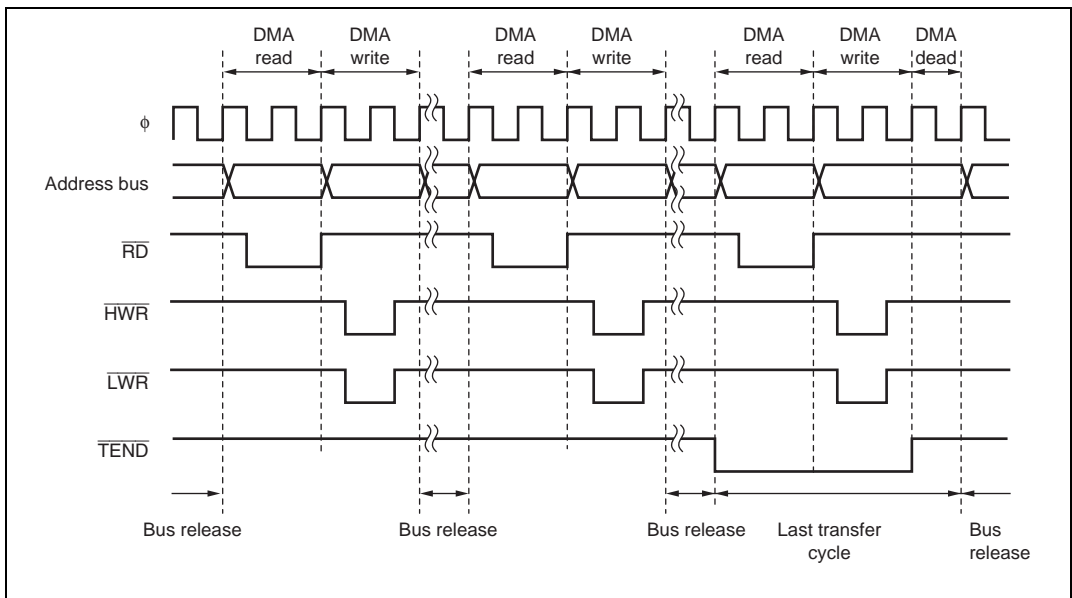


Figure 8.20 Example of Full Address Mode Transfer (Cycle Steal)

A byte or word transfer is performed for a single transfer request, and after the transfer, the bus is released. While the bus is released, one bus cycle is executed by the CPU or DTC.

In the transfer end cycle (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle.

Full Address Mode (Burst Mode): Figure 8.21 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and word-size full address mode transfer (burst mode) is performed from external 16-bit, 2-state access space to external 16-bit, 2-state access space.

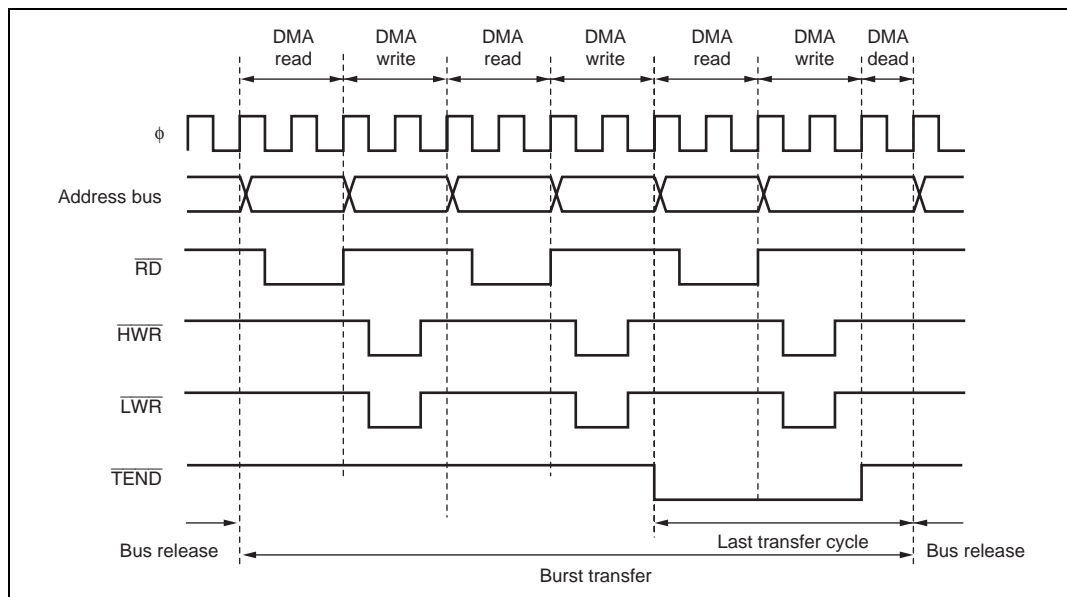


Figure 8.21 Example of Full Address Mode Transfer (Burst Mode)

In burst mode, one-byte or one-word transfers are executed consecutively until transfer ends.

In the transfer end cycle (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle.

If a request from another higher-priority channel is generated after burst transfer starts, that channel has to wait until the burst transfer ends.

If an NMI interrupt is generated while a channel designated for burst transfer is in the transfer enabled state, the DTME bit in DMABCRL is cleared and the channel is placed in the transfer disabled state. If burst transfer has already been activated inside the DMAC, the bus is released on completion of a one-byte or one-word transfer within the burst transfer, and burst transfer is suspended. If the last transfer cycle of the burst transfer has already been activated inside the DMAC, execution continues to the end of the transfer even if the DTME bit is cleared.

Full Address Mode (Block Transfer Mode): Figure 8.22 shows a transfer example in which TEND output is enabled and word-size full address mode transfer (block transfer mode) is performed from internal 16-bit, 1-state access space to external 16-bit, 2-state access space.

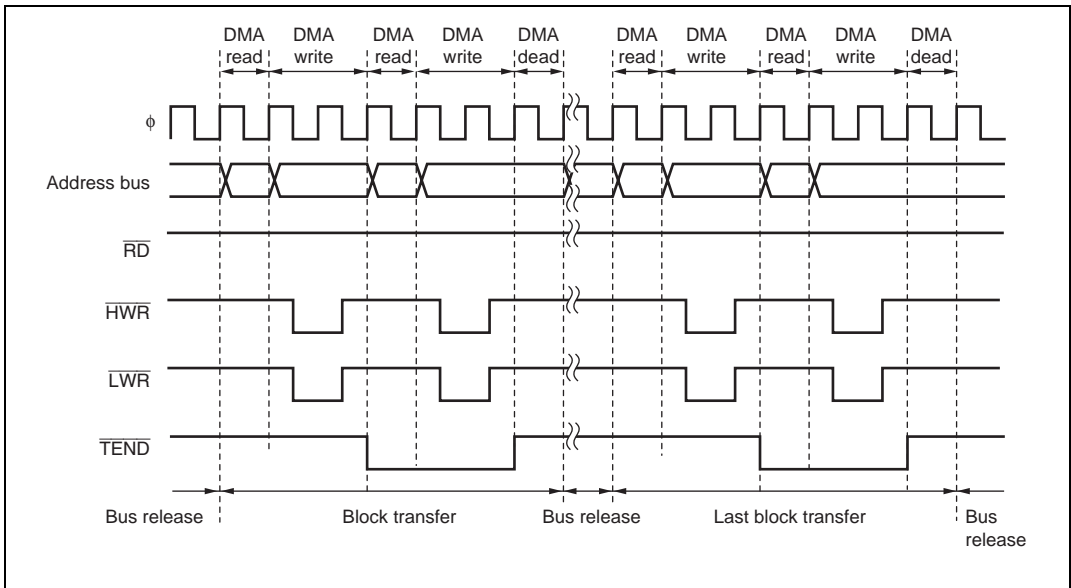


Figure 8.22 Example of Full Address Mode Transfer (Block Transfer Mode)

A one-block transfer is performed for a single transfer request, and after the transfer the bus is released. While the bus is released, one or more bus cycles are executed by the CPU or DTC.

In the transfer end cycle of each block (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle. Even if an NMI interrupt is generated during data transfer, block transfer operation is not affected until data transfer for one block has ended.

DREQ Pin Falling Edge Activation Timing: Set the DTA bit in DMABCRH to 1 for the channel for which the DREQ pin is selected.

Figure 8.23 shows an example of normal mode transfer activated by the $\overline{\text{DREQ}}$ pin falling edge.

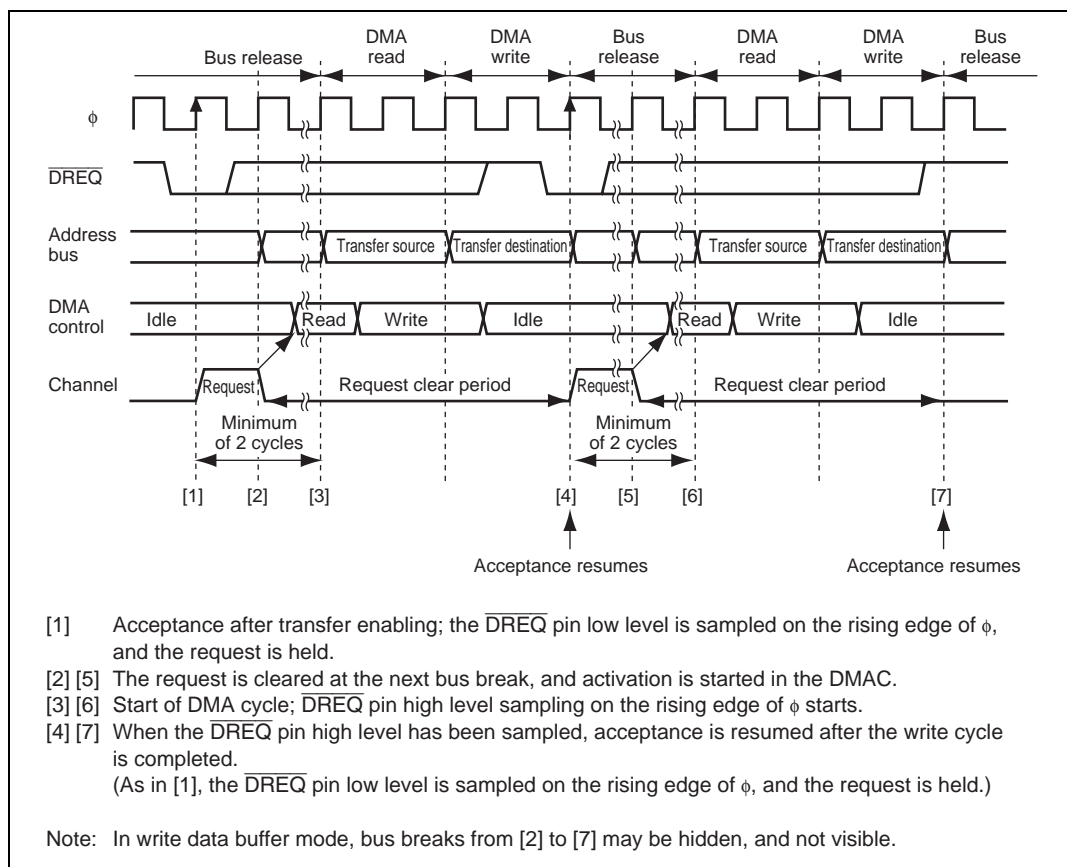


Figure 8.23 Example of $\overline{\text{DREQ}}$ Pin Falling Edge Activated Normal Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next ϕ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared, and $\overline{\text{DREQ}}$ pin high level sampling for edge detection is started. If $\overline{\text{DREQ}}$ pin high level sampling has been completed by the time the DMA write cycle ends, acceptance resumes after the end of the write cycle, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

Figure 8.24 shows an example of block transfer mode transfer activated by the $\overline{\text{DREQ}}$ pin falling edge.

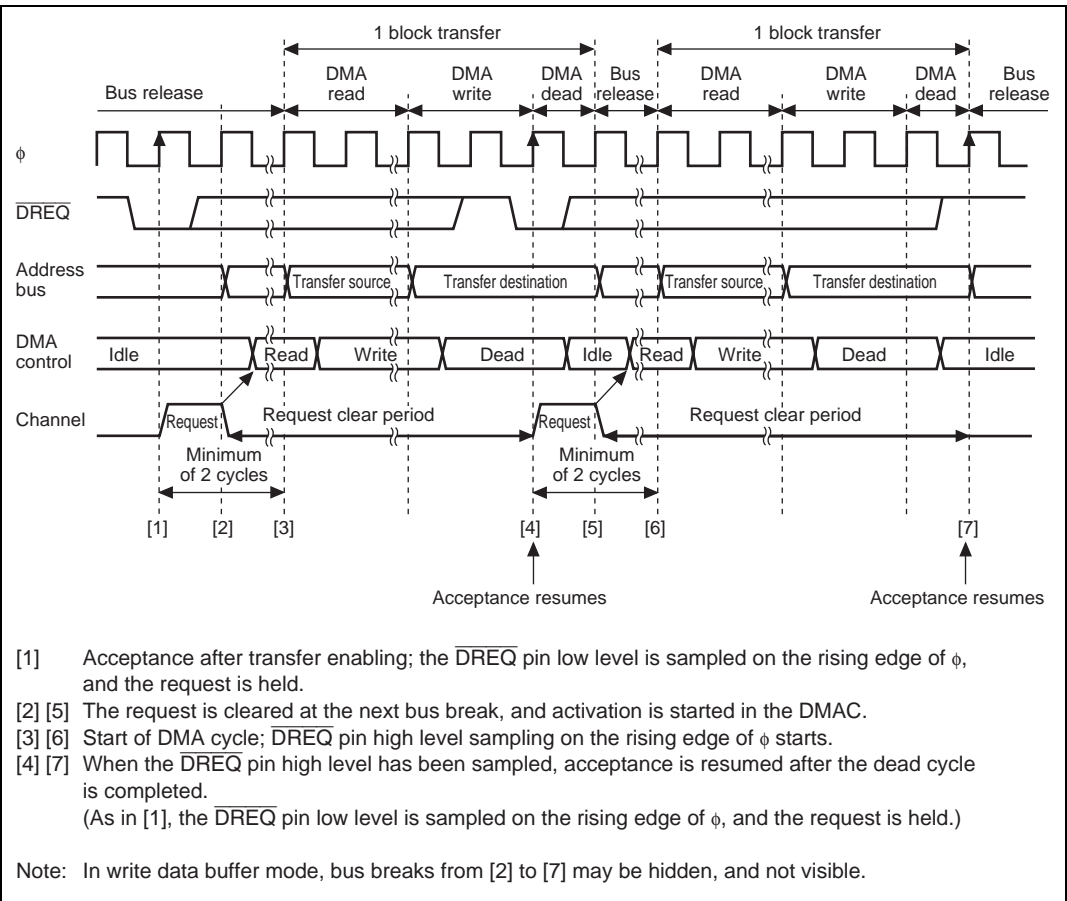


Figure 8.24 Example of $\overline{\text{DREQ}}$ Pin Falling Edge Activated Block Transfer Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next ϕ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared, and $\overline{\text{DREQ}}$ pin high level sampling for edge detection is started. If $\overline{\text{DREQ}}$ pin high level sampling has been completed by the time the DMA dead cycle ends, acceptance resumes after the end of the dead cycle, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

$\overline{\text{DREQ}}$ Pin Low Level Activation Timing (Normal Mode): Set the DTA bit in DMABCRH to 1 for the channel for which the $\overline{\text{DREQ}}$ pin is selected.

Figure 8.25 shows an example of normal mode transfer activated by the $\overline{\text{DREQ}}$ pin low level.

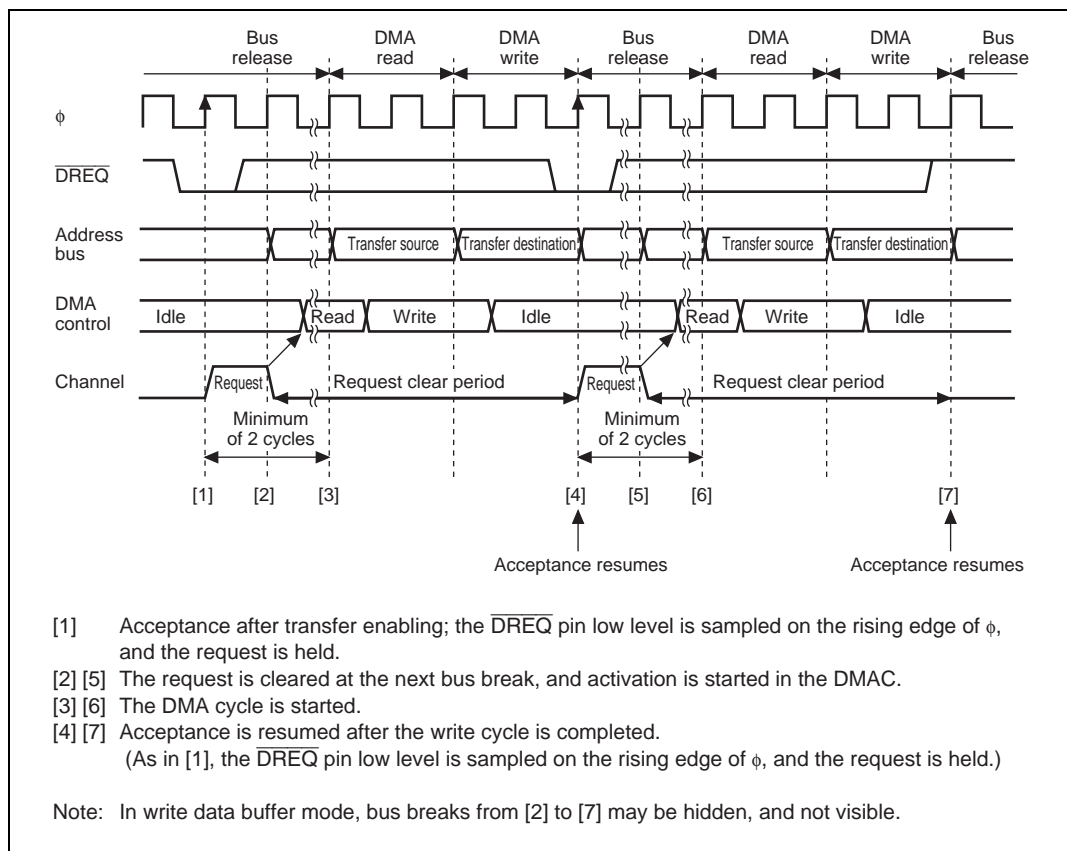


Figure 8.25 Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Normal Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next ϕ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared. After the end of the write cycle, acceptance resumes, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

Figure 8.26 shows an example of block transfer mode transfer activated by $\overline{\text{DREQ}}$ pin low level.

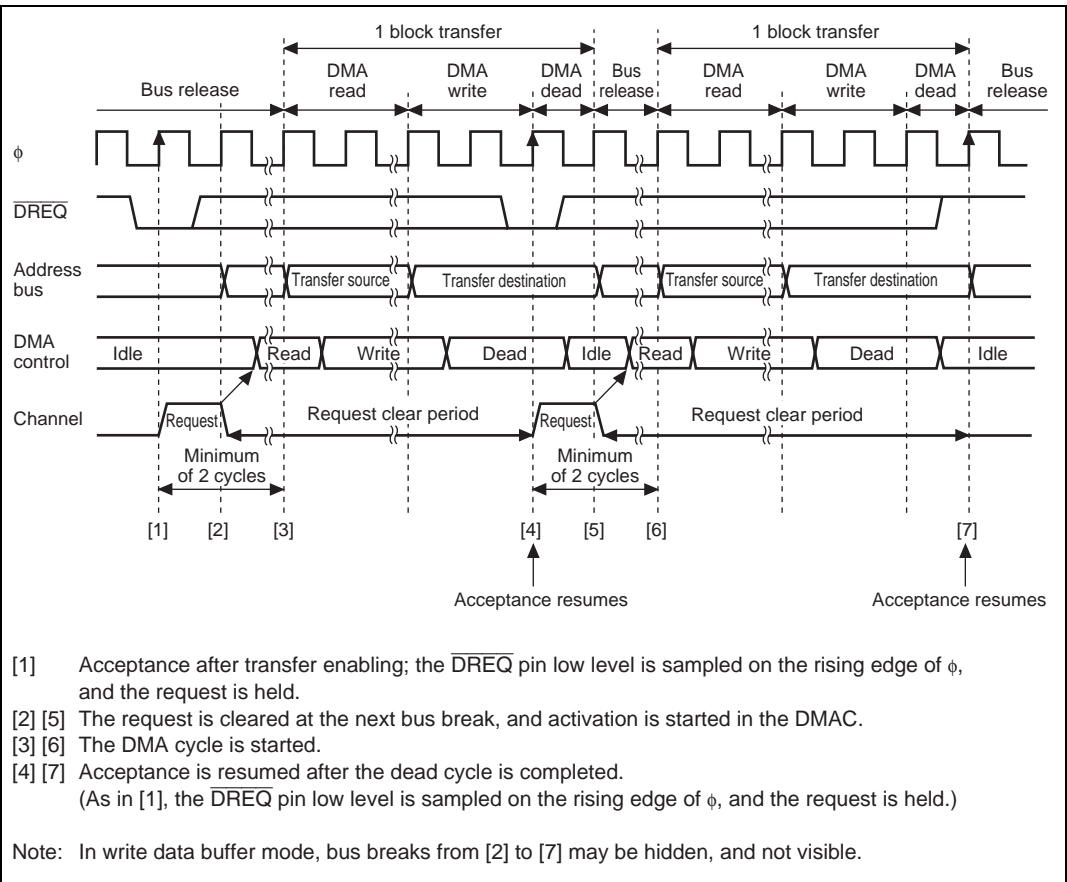


Figure 8.26 Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Block Transfer Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next ϕ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared. After the end of the dead cycle, acceptance resumes, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

8.5.10 DMA Transfer (Single Address Mode) Bus Cycles

Single Address Mode (Read): Figure 8.27 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and byte-size single address mode transfer (read) is performed from external 8-bit, 2-state access space to an external device.

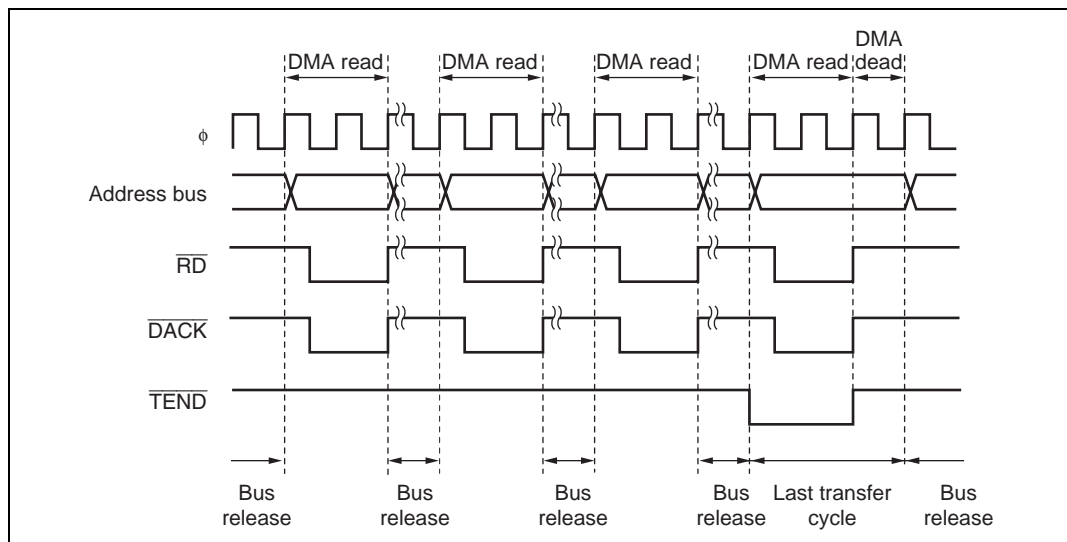


Figure 8.27 Example of Single Address Mode Transfer (Byte Read)

Figure 8.28 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and word-size single address mode transfer (read) is performed from external 8-bit, 2-state access space to an external device.

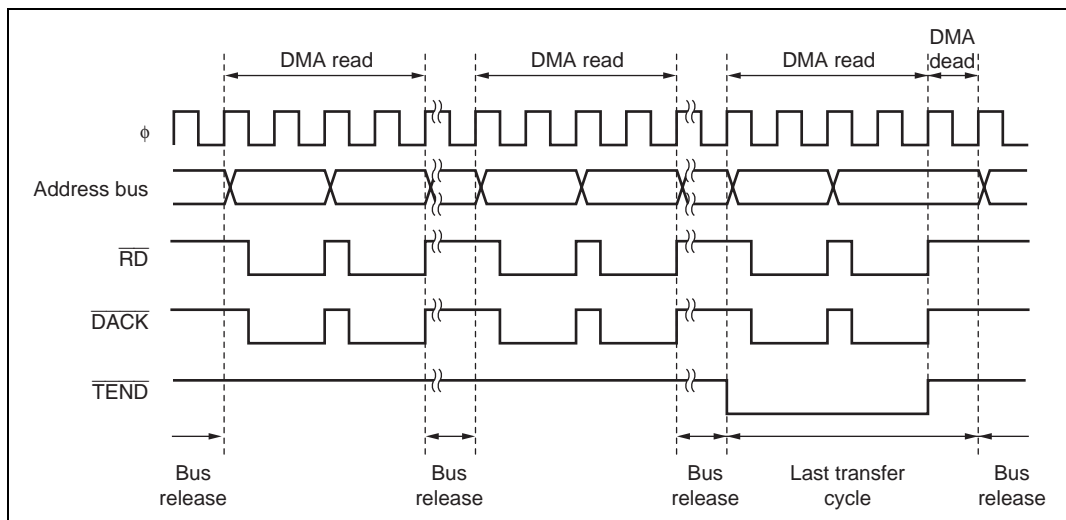


Figure 8.28 Example of Single Address Mode (Word Read) Transfer

A byte or word transfer is performed for a single transfer request, and after the transfer, the bus is released. While the bus is released, one or more bus cycles are executed by the CPU or DTC.

In the transfer end cycle (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle.

Single Address Mode (Write): Figure 8.29 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and byte-size single address mode transfer (write) is performed from an external device to external 8-bit, 2-state access space.

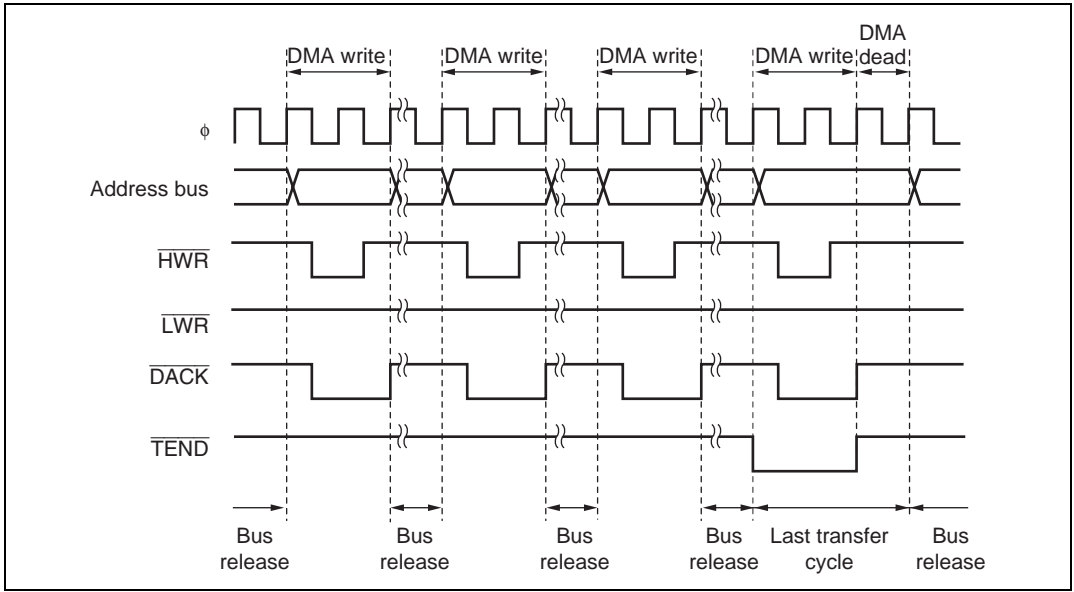


Figure 8.29 Example of Single Address Mode Transfer (Byte Write)

Figure 8.30 shows a transfer example in which $\overline{\text{TEND}}$ output is enabled and word-size single address mode transfer (write) is performed from an external device to external 8-bit, 2-state access space.

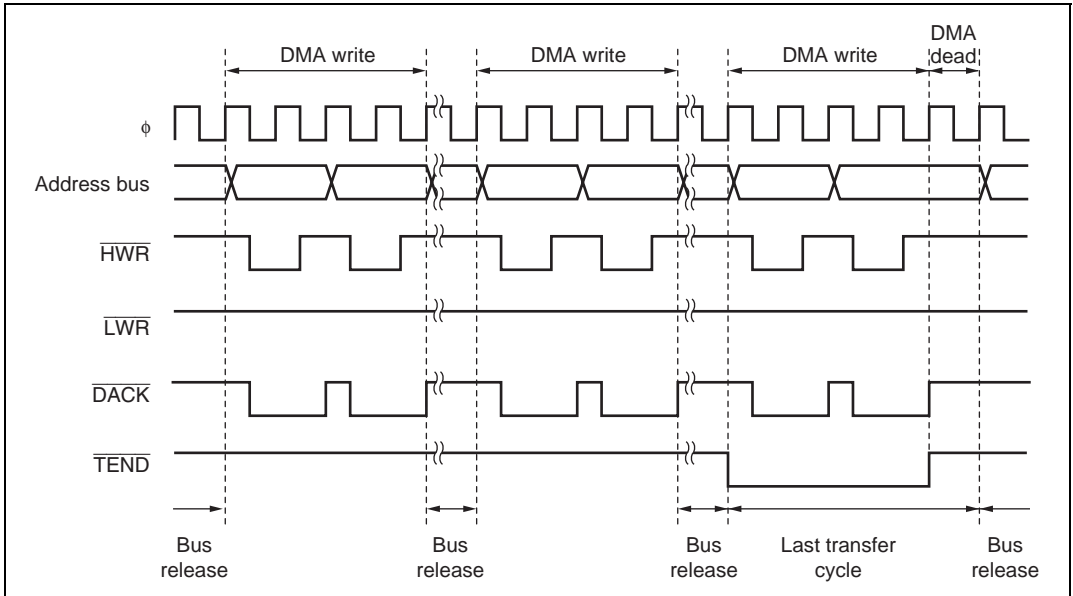


Figure 8.30 Example of Single Address Mode Transfer (Word Write)

A byte or word transfer is performed for a single transfer request, and after the transfer, the bus is released. While the bus is released, one or more bus cycles are executed by the CPU or DTC.

In the transfer end cycle (the cycle in which the transfer counter reaches 0), a one-state DMA dead cycle is inserted after the DMA write cycle.

$\overline{\text{DREQ}}$ Pin Falling Edge Activation Timing: Set the DTA bit in DMABCRH to 1 for the channel for which the $\overline{\text{DREQ}}$ pin is selected.

Figure 8.31 shows an example of single address mode transfer activated by the $\overline{\text{DREQ}}$ pin falling edge.

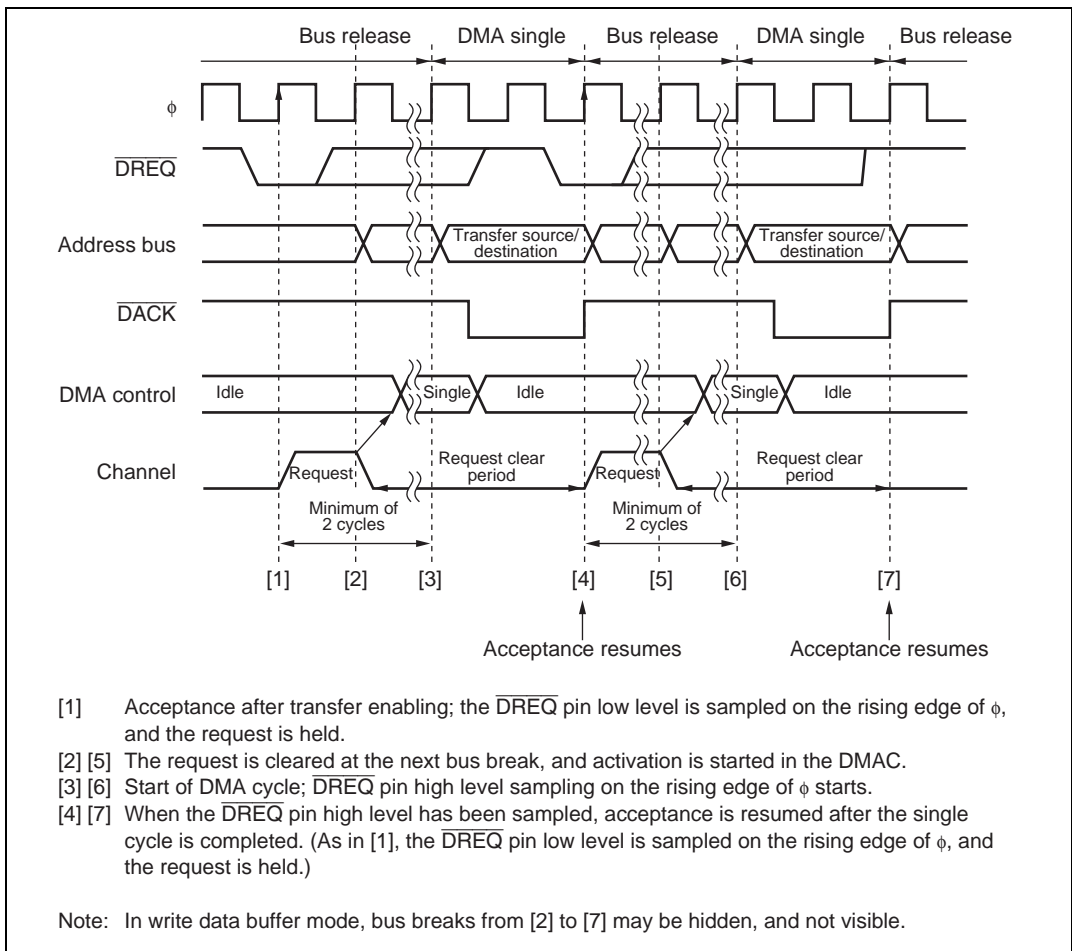


Figure 8.31 Example of $\overline{\text{DREQ}}$ Pin Falling Edge Activated Single Address Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next ϕ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared, and $\overline{\text{DREQ}}$ pin high level sampling for edge detection is started. If $\overline{\text{DREQ}}$ pin high level sampling has been completed by the time the DMA single cycle ends, acceptance

resumes after the end of the single cycle, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

$\overline{\text{DREQ}}$ Pin Low Level Activation Timing: Set the DTA bit in DMABCRH to 1 for the channel for which the $\overline{\text{DREQ}}$ pin is selected.

Figure 8.32 shows an example of single address mode transfer activated by the $\overline{\text{DREQ}}$ pin low level.

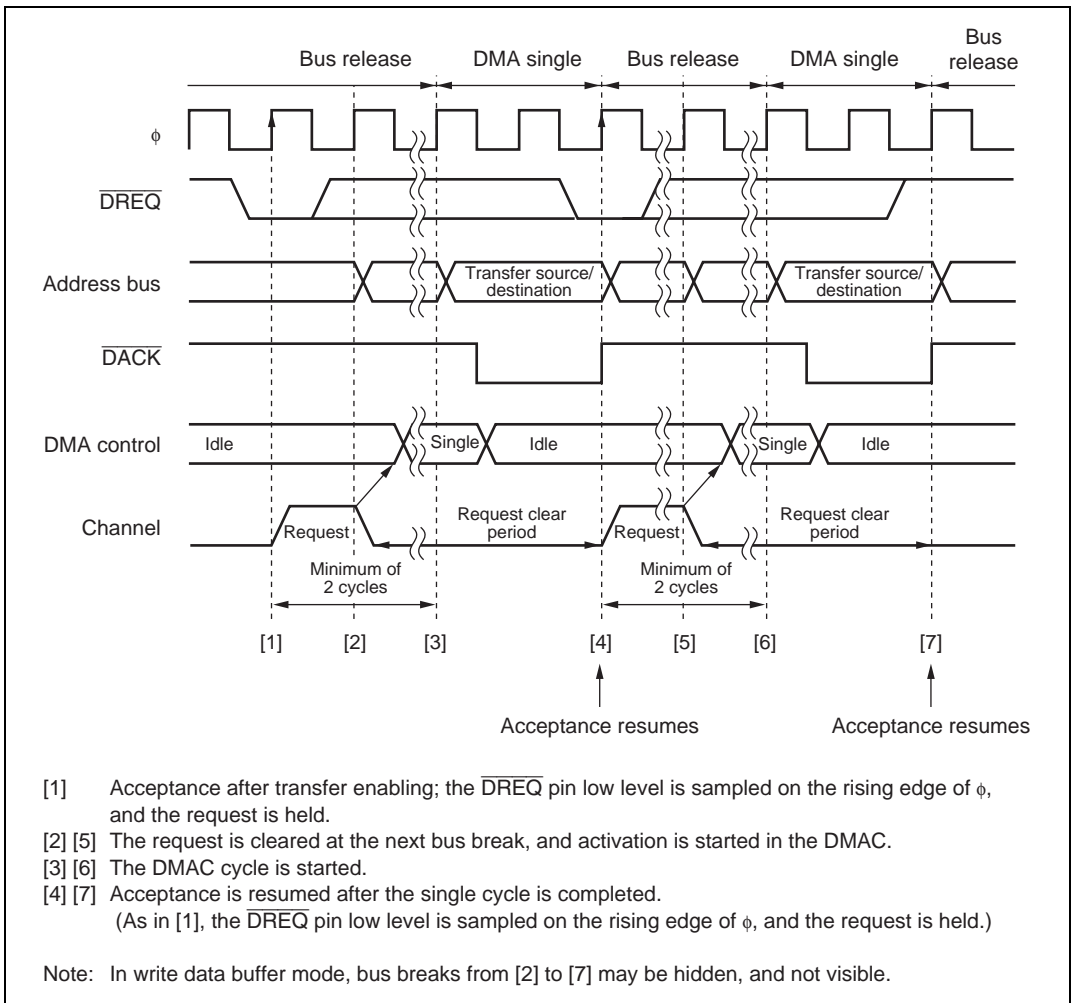


Figure 8.32 Example of $\overline{\text{DREQ}}$ Pin Low Level Activated Single Address Mode Transfer

$\overline{\text{DREQ}}$ pin sampling is performed every cycle, with the rising edge of the next ϕ cycle after the end of the DMABCR write cycle for setting the transfer enabled state as the starting point.

When the $\overline{\text{DREQ}}$ pin low level is sampled while acceptance by means of the $\overline{\text{DREQ}}$ pin is possible, the request is held in the DMAC. Then, when activation is initiated in the DMAC, the request is cleared. After the end of the single cycle, acceptance resumes, $\overline{\text{DREQ}}$ pin low level sampling is performed again, and this operation is repeated until the transfer ends.

8.5.11 Multi-Channel Operation

The DMAC channel priority order is: channel 0 > channel 1, and channel A > channel B. Table 8.11 summarizes the priority order for DMAC channels.

Table 8.11 DMAC Channel Priority Order

| Short Address Mode | Full Address Mode | Priority |
|--------------------|-------------------|----------|
| Channel 0A | Channel 0 | High |
| Channel 0B | | |
| Channel 1A | Channel 1 | Low |
| Channel 1B | | |

If transfer requests are issued simultaneously for more than one channel, or if a transfer request for another channel is issued during a transfer, when the bus is released, the DMAC selects the highest-priority channel from among those issuing a request according to the priority order shown in table 8.11. During burst transfer, or when one block is being transferred in block transfer, the channel will not be changed until the end of the transfer. Figure 8.33 shows a transfer example in which transfer requests are issued simultaneously for channels 0A, 0B, and 1.

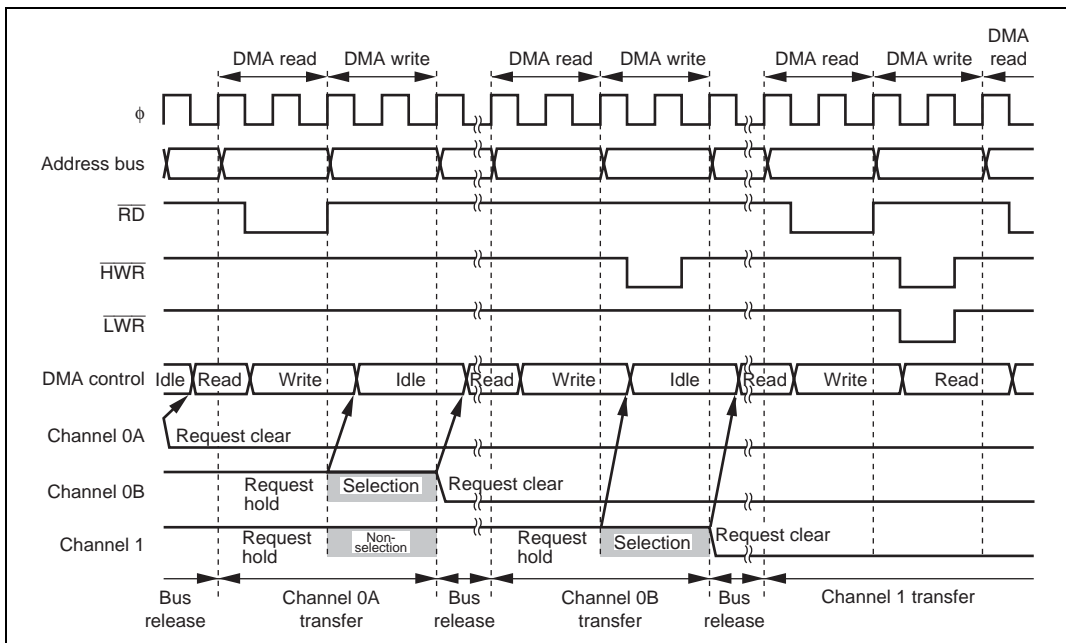


Figure 8.33 Example of Multi-Channel Transfer

8.5.12 Relation between DMAC and External Bus Requests, and DTC

The DMA read cycle and write cycle are inseparable, and so the external bus release cycle and DTC cycle do not arise between the DMA external read cycle and internal write cycle.

When the read cycle and write cycle are set in series as in a burst transfer or block transfer, the external bus release may be inserted after the write cycle. As the DTC has a lower priority than the DMAC, it is not executed until the DMAC releases the bus.

When the DMA read cycle or write cycle accesses the on-chip memory or an internal I/O register, the DMAC cycle or external bus release may be executed at the same time.

8.5.13 DMAC and NMI Interrupts

When an NMI interrupt is requested, burst mode transfer in full address mode is interrupted. An NMI interrupt does not affect the operation of the DMAC in other modes.

In full address mode, transfer is enabled for a channel when both the DTE bit and DTME bit are set to 1. With burst mode setting, the DTME bit is cleared when an NMI interrupt is requested.

If the DTME bit is cleared during burst mode transfer, the DMAC discontinues transfer on completion of the 1-byte or 1-word transfer in progress, then releases the bus, which passes to the CPU.

The channel on which transfer was interrupted can be restarted by setting the DTME bit to 1 again. Figure 8.34 shows the procedure for continuing transfer when it has been interrupted by an NMI interrupt on a channel designated for burst mode transfer.

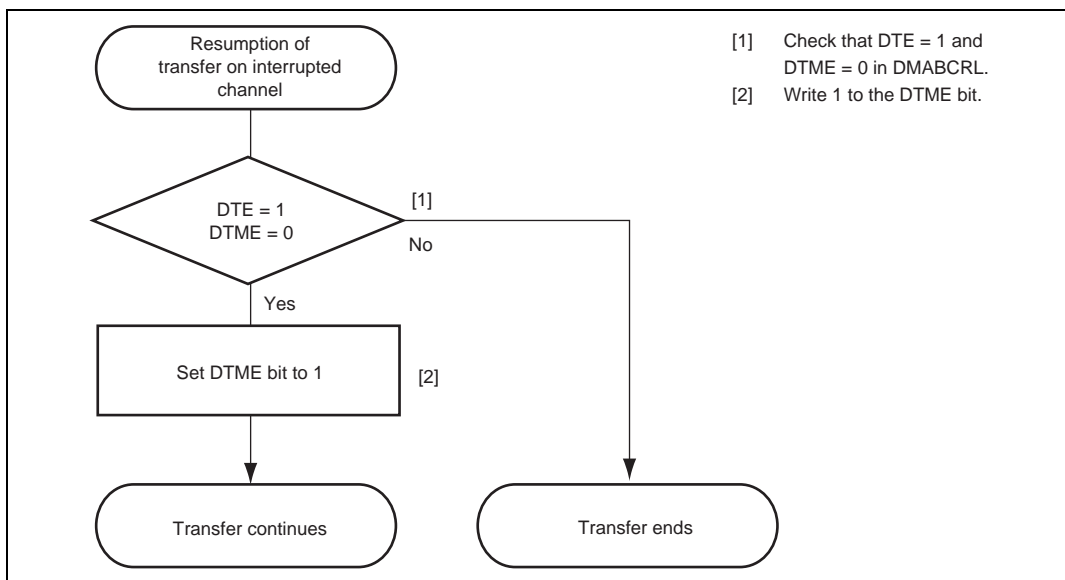


Figure 8.34 Example of Procedure for Continuing Transfer on Channel Interrupted by NMI Interrupt

8.5.14 Forced Termination of DMAC Operation

If the DTE bit in DMABCRL is cleared to 0 for the channel currently operating, the DMAC stops on completion of the 1-byte or 1-word transfer in progress. DMAC operation resumes when the DTE bit is set to 1 again. In full address mode, the same applies to the DTME bit in DMABCRL. Figure 8.35 shows the procedure for forcibly terminating DMAC operation by software.

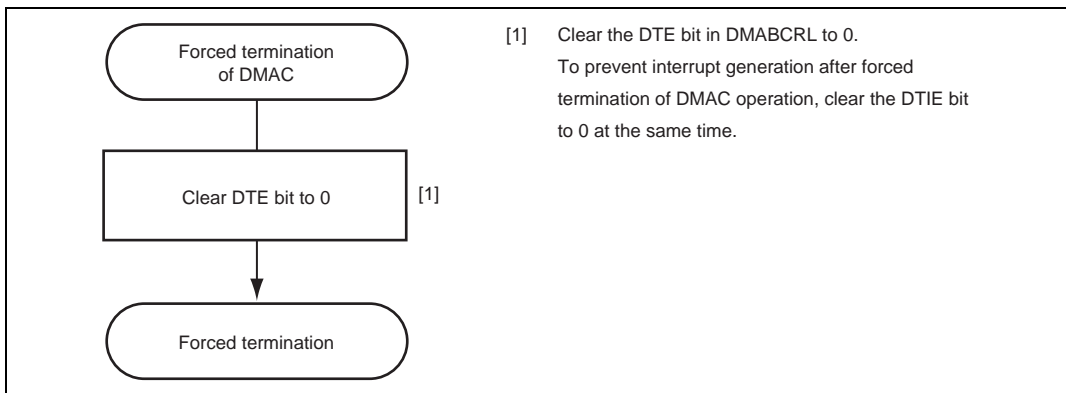


Figure 8.35 Example of Procedure for Forcibly Terminating DMAC Operation

8.5.15 Clearing Full Address Mode

Figure 8.36 shows the procedure for releasing and initializing a channel designated for full address mode. After full address mode has been cleared, the channel can be set to another transfer mode using the appropriate setting procedure.

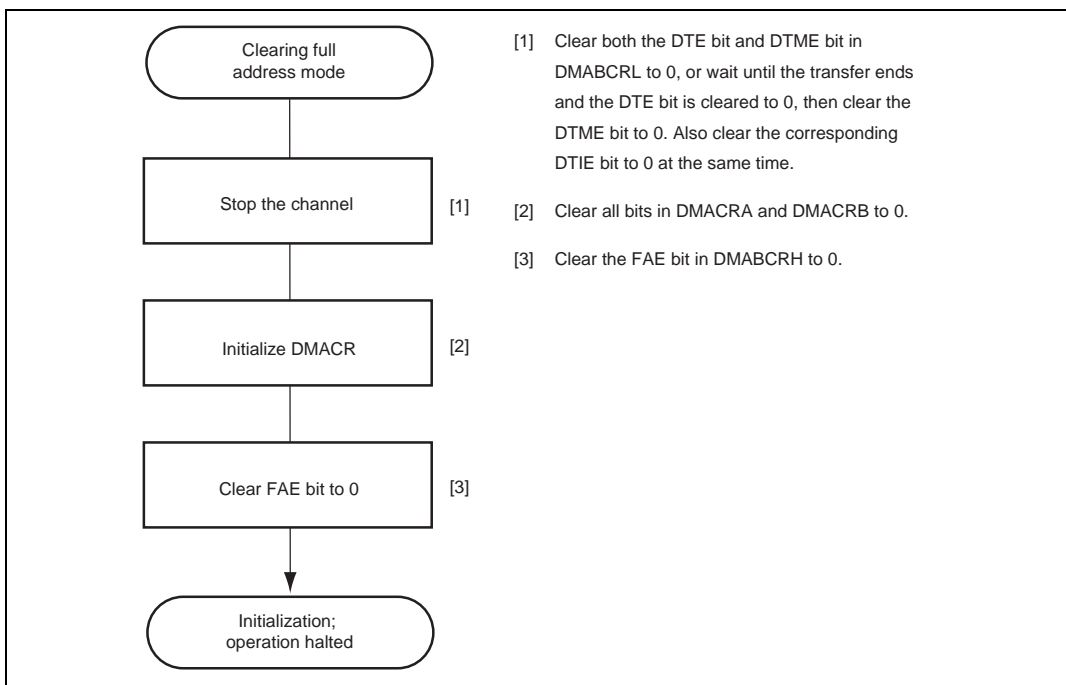


Figure 8.36 Example of Procedure for Clearing Full Address Mode

8.6 Interrupt Sources

The sources of interrupts generated by the DMAC are transfer end and transfer break. Table 8.12 shows the interrupt sources and their priority order.

Table 8.12 Interrupt Sources and Priority Order

| Interrupt Name | Interrupt Source | | Interrupt Priority Order |
|----------------|--|---|---|
| | Short Address Mode | Full Address Mode | |
| DEND0A | Interrupt due to end of transfer on channel 0A | Interrupt due to end of transfer on channel 0 | <div style="display: flex; align-items: center;"> <div style="flex: 1; border-left: 1px solid black; position: relative;"> <div style="position: absolute; top: 0; right: 0; width: 10px; height: 10px; background: white; border: 1px solid black;"></div> <div style="position: absolute; bottom: 0; right: 0; width: 10px; height: 10px; background: white; border: 1px solid black;"></div> </div> </div> |
| DEND0B | Interrupt due to end of transfer on channel 0B | Interrupt due to break in transfer on channel 0 | |
| DEND1A | Interrupt due to end of transfer on channel 1A | Interrupt due to end of transfer on channel 1 | |
| DEND1B | Interrupt due to end of transfer on channel 1B | Interrupt due to break in transfer on channel 1 | |

Enabling or disabling of each interrupt source is set by means of the DTIE bit in DMABCRL for the corresponding channel in DMABCRL, and interrupts from each source are sent to the interrupt controller independently. The priority of transfer end interrupts on each channel is decided by the interrupt controller, as shown in table 8.12.

Figure 8.37 shows a block diagram of a transfer end/transfer break interrupt. An interrupt is always generated when the DTIE bit is set to 1 while the DTE bit in DMABCRL is cleared to 0.

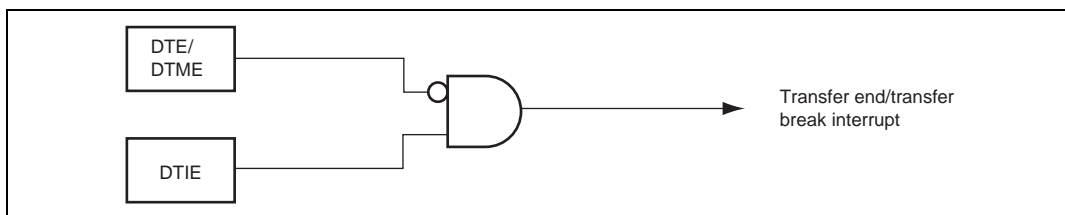


Figure 8.37 Block Diagram of Transfer End/Transfer Break Interrupt

In full address mode, a transfer break interrupt is generated when the DTME bit is cleared to 0 while the DTIE bit is set to 1. In both short address mode and full address mode, DMABCR should be set so as to prevent the occurrence of a combination that constitutes a condition for interrupt generation during setting.

8.7 Usage Notes

8.7.1 DMAC Register Access during Operation

Except for forced termination of the DMAC, the operating (including transfer waiting state) channel setting should not be changed. The operating channel setting should only be changed when transfer is disabled. Also, DMAC registers should not be written to in a DMA transfer.

DMAC register reads during operation (including the transfer waiting state) are described below.

- DMAC control starts one cycle before the bus cycle, with output of the internal address. Consequently, MAR is updated in the bus cycle before DMA transfer. Figure 8.38 shows an example of the update timing for DMAC registers in dual address transfer mode.

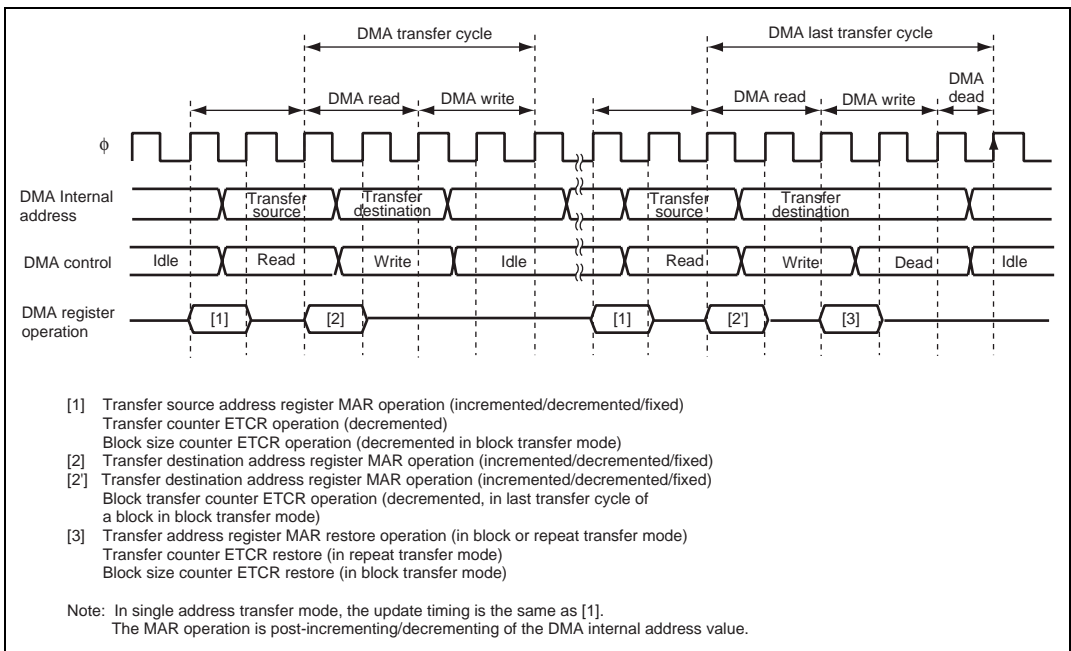


Figure 8.38 DMAC Register Update Timing

- If a DMAC transfer cycle occurs immediately after a DMAC register read cycle, the DMAC register is read as shown in figure 8.39.

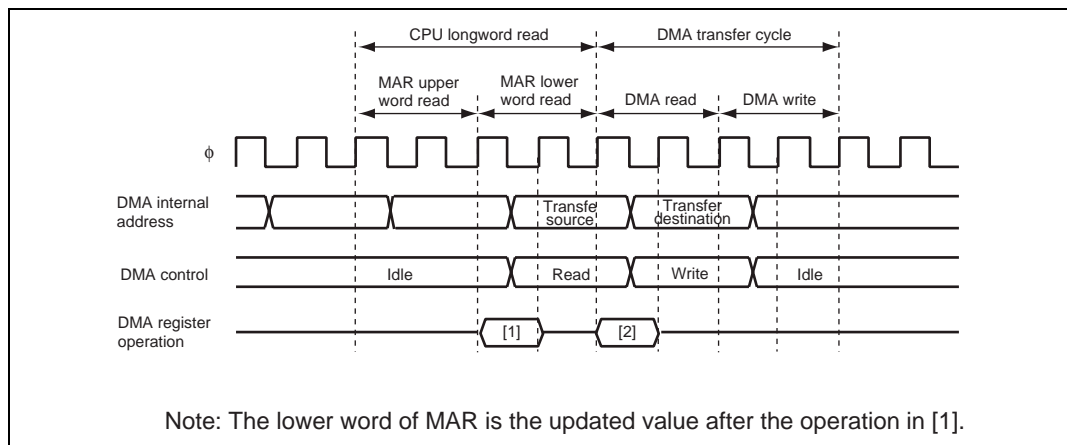


Figure 8.39 Contention between DMAC Register Update and CPU Read

8.7.2 Module Stop

When the MSTPA7 bit in MSTPCRA is set to 1, the DMAC clock stops, and the module stop state is entered. However, 1 cannot be written to the MSTPA7 bit if any of the DMAC channels is enabled. This setting should therefore be made when DMAC operation is stopped.

When the DMAC clock stops, DMAC register accesses can no longer be made. Since the following DMAC register settings are valid even in the module stop state, they should be invalidated, if necessary, before a module stop.

- Transfer end/break interrupt (DTE = 0 and DTIE = 1)
- $\overline{\text{TEND}}$ pin enable (TEE = 1)
- $\overline{\text{DACK}}$ pin enable (FAE = 0 and SAE = 1)

8.7.3 Medium-Speed Mode

When the DTA bit is cleared to 0, the internal interrupt signal that is specified for the DMAC transfer source is detected at the edge. In medium-speed mode, the DMAC operates by the medium-speed clock and the internal peripheral module operates by the high-speed clock. Therefore, when the corresponding interruption source is cleared by the CPU, DTC, or other channels of the DMAC and the period until the next interruption is executed is less than one state regarding to the DMAC clock (bus master clock), the signal is not detected at the edge and ignored.

In medium-speed mode, the $\overline{\text{DREQ}}$ pin is sampled at the rising edge of the medium clock.

8.7.4 Activation by Falling Edge on $\overline{\text{DREQ}}$ Pin

$\overline{\text{DREQ}}$ pin falling edge detection is performed in synchronization with DMAC internal operations. The operation is as follows:

- [1] Activation request wait state: Waits for detection of a low level on the $\overline{\text{DREQ}}$ pin, and switches to [2].
- [2] Transfer wait state: Waits for DMAC data transfer to become possible, and switches to [3].
- [3] Activation request disabled state: Waits for detection of a high level on the $\overline{\text{DREQ}}$ pin, and switches to [1].

After DMAC transfer is enabled, a transition is made to [1]. Thus, initial activation after transfer is enabled is performed on detection of a low level.

8.7.5 Activation Source Acceptance

At the start of activation source acceptance, a low level is detected in both $\overline{\text{DREQ}}$ pin falling edge sensing and low level sensing. Similarly, in the case of an internal interrupt, the interrupt request is detected. Therefore, a request is accepted from an internal interrupt or $\overline{\text{DREQ}}$ pin low level that occurs before write to DMABCRL to enable transfer.

When the DMAC is activated, take any necessary steps to prevent an internal interrupt or $\overline{\text{DREQ}}$ pin low level remaining from the end of the previous transfer, etc.

8.7.6 Internal Interrupt after End of Transfer

When the DTE bit in DMABCRL is cleared to 0 at the end of a transfer or by a forcible termination, the selected internal interrupt request will be sent to the CPU or DTC even if the DTA bit in DMABCRH is set to 1.

Also, if internal DMAC activation has already been initiated when operation is forcibly terminated, the transfer is executed but flag clearing is not performed for the selected internal interrupt even if the DTA bit is set to 1.

An internal interrupt request following the end of transfer or a forcible termination should be handled by the CPU as necessary.

8.7.7 Channel Re-Setting

To reactivate a number of channels when multiple channels are enabled, use exclusive handling of transfer end interrupts, and perform DMABCR control bit operations exclusively.

Note, in particular, that in cases where multiple interrupts are generated between reading and writing of DMABCR, and a DMABCR operation is performed during new interrupt handling, the DMABCR write data in the original interrupt handling routine will be incorrect, and the write may invalidate the results of the operations by the multiple interrupts. Ensure that overlapping DMABCR operations are not performed by multiple interrupts, and that there is no separation between read and write operations by the use of a bit-manipulation instruction.

Also, when the DTE and DTME bits are cleared by the DMAC or are written with 0, they must first be read while cleared to 0 before the CPU can write 1 to them.

Section 9 Data Transfer Controller (DTC)

This LSI includes a data transfer controller (DTC). The DTC can be activated by an interrupt or software, to transfer data.

Figure 9.1 shows a block diagram of the DTC.

The DTC's register information is stored in the on-chip RAM. When the DTC is used, the RAME bit in SYSCR must be set to 1. A 32-bit bus connects the DTC to the on-chip RAM (1 kbyte), enabling 32-bit/1-state reading and writing of the DTC register information.

9.1 Features

- Transfer is possible over any number of channels
- Three transfer modes
Normal, repeat, and block transfer modes are available
- One activation source can trigger a number of data transfers (chain transfer)
- The direct specification of 16-Mbyte address space is possible
- Activation by software is possible
- Transfer can be set in byte or word units
- A CPU interrupt can be requested for the interrupt that activated the DTC
- Module stop mode can be set

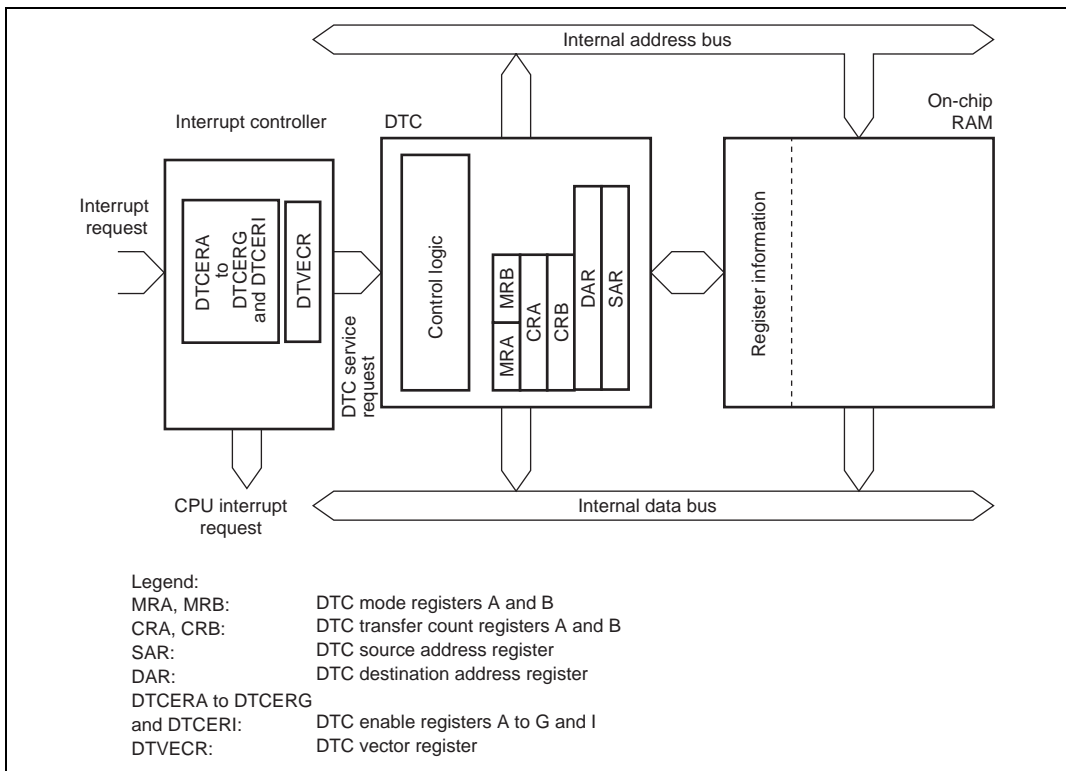


Figure 9.1 Block Diagram of DTC

9.2 Register Descriptions

The DTC has the following registers.

- DTC mode register A (MRA)
- DTC mode register B (MRB)
- DTC source address register (SAR)
- DTC destination address register (DAR)
- DTC transfer count register A (CRA)
- DTC transfer count register B (CRB)

These six registers cannot be directly accessed from the CPU.

When activated, the DTC reads a set of register information that is stored in on-chip RAM to the corresponding DTC registers and transfers data. After the data transfer, it writes a set of updated register information back to the RAM.

- DTC enable registers A to G, and I (DTCERA to DTCERG, and DTCERI)
- DTC vector register (DTVECR)

9.2.1 DTC Mode Register A (MRA)

MRA selects the DTC operating mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | SM1 | Undefined | — | Source Address Mode 1 and 0 |
| 6 | SM0 | Undefined | — | These bits specify an SAR operation after a data transfer. 0x: SAR is fixed 10: SAR is incremented after a transfer (by +1 when Sz = 0; by +2 when Sz = 1) 11: SAR is decremented after a transfer (by -1 when Sz = 0; by -2 when Sz = 1) |
| 5 | DM1 | Undefined | — | Destination Address Mode 1 and 0 |
| 4 | DM0 | Undefined | — | These bits specify a DAR operation after a data transfer. 0x: DAR is fixed 10: DAR is incremented after a transfer (by +1 when Sz = 0; by +2 when Sz = 1) 11: DAR is decremented after a transfer (by -1 when Sz = 0; by -2 when Sz = 1) |
| 3 | MD1 | Undefined | — | DTC Mode 1 and 0 |
| 2 | MD0 | Undefined | — | These bits specify the DTC transfer mode. 00: Normal mode 01: Repeat mode 10: Block transfer mode 11: Setting prohibited |
| 1 | DTS | Undefined | — | DTC Transfer Mode Select Specifies whether the source side or the destination side is set to be a repeat area or block area, in repeat mode or block transfer mode. 0: Destination side is repeat area or block area 1: Source side is repeat area or block area |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | Sz | Undefined | — | DTC Data Transfer Size Specifies the size of data to be transferred. 0: Byte-size transfer 1: Word-size transfer |

Legend: x: Don't care

9.2.2 DTC Mode Register B (MRB)

MRB is an 8-bit register that selects the DTC operating mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | CHNE | Undefined | — | DTC Chain Transfer Enable This bit specifies a chain transfer. For details, refer to section 9.5.4, Chain Transfer. In data transfer with CHNE set to 1, determination of the end of the specified number of transfers, clearing of the interrupt source flag, and clearing of DTCER, are not performed. 0: DTC data transfer completed (waiting for start) 1: DTC chain transfer (reads new register information and transfers data) |
| 6 | DISEL | Undefined | — | DTC Interrupt Select This bit specifies whether CPU interrupt is disabled or enabled after a data transfer. 0: Interrupt request is issued to the CPU when the specified data transfer is completed 1: DTC issues interrupt request to the CPU in every data transfer (DTC does not clear the interrupt request flag that is a cause of the activation) |
| 5 to 0 | — | Undefined | — | Reserved These bits have no effect on DTC operation. The write value should always be 0. |

9.2.3 DTC Source Address Register (SAR)

SAR is a 24-bit register that designates the source address of data to be transferred by the DTC. For word-size transfer, specify an even source address.

9.2.4 DTC Destination Address Register (DAR)

DAR is a 24-bit register that designates the destination address of data to be transferred by the DTC. For word-size transfer, specify an even destination address.

9.2.5 DTC Transfer Count Register A (CRA)

CRA is a 16-bit register that designates the number of times data is to be transferred by the DTC.

In normal mode, the entire CRA functions as a 16-bit transfer counter (1 to 65,536). It is decremented by 1 every time data is transferred, and transfer ends when the count reaches H'0000.

In repeat mode or block transfer mode, the CRA is divided into two parts; the upper 8 bits (CRAH) and the lower 8 bits (CRAL). In repeat mode, CRAH holds the number of transfers while CRAL functions as an 8-bit transfer counter (1 to 256). In block transfer mode, CRAH holds the block size while CRAL functions as an 8-bit block size counter (1 to 256). CRAL is decremented by 1 every time data is transferred, and the contents of CRAH are sent when the count reaches H'00. This operation is repeated.

9.2.6 DTC Transfer Count Register B (CRB)

CRB is a 16-bit register that designates the number of times data is to be transferred by the DTC in block transfer mode. It functions as a 16-bit transfer counter (1 to 65,536) that is decremented by 1 every time data is transferred, and transfer ends when the count reaches H'0000.

9.2.7 DTC Enable Registers A to G, and I (DTCERA to DTCERG, and DTCERI)

DTCER is a set of registers to specify the DTC activation interrupt source, and comprised of eight registers; DTCERA to DTCERG, and DTCERI. The correspondence between interrupt sources and DTCE bits, and vector numbers generated by the interrupt controller are shown in table 9.2. For DTCE bit setting, use bit manipulation instructions such as BSET and BCLR for reading and writing. When multiple activation sources are to be set at one time, only at the initial setting, writing data is enabled after executing a dummy read on the relevant register with all the interrupts being masked.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | DTCEn7 | 0 | R/W | DTC Activation Enable |
| 6 | DTCEn6 | 0 | R/W | 0: Disables an interrupt for DTC activation. |
| 5 | DTCEn5 | 0 | R/W | 1: Specifies a relevant interrupt source as a DTC activation source. |
| 4 | DTCEn4 | 0 | R/W | [Clearing conditions] |
| 3 | DTCEn3 | 0 | R/W | |
| 2 | DTCEn2 | 0 | R/W | • When the DISEL bit in MRB is 1 and the data transfer has ended |
| 1 | DTCEn1 | 0 | R/W | |
| 0 | DTCEn0 | 0 | R/W | • When the specified number of transfers have ended |
| | | | | [Retaining condition] |
| | | | | When the DISEL bit is 0 and the specified number of transfers have not been completed |

Note: n = A to G, and I

9.2.8 DTC Vector Register (DTVECR)

DTVECR is an 8-bit readable/writable register that enables or disables DTC activation by software, and sets a vector number for the software activation interrupt.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | SWDTE | 0 | R/W | <p>DTC Software Activation Enable</p> <p>Enables or disables the DTC software activation.</p> <p>0: Disables the DTC software activation.</p> <p>1: Enables the DTC software activation.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When the DISEL bit is 0 and the specified number of transfers have not ended When 0 is written to the DISEL bit after a software-activated data transfer end interrupt (SWDTEND) request has been sent to the CPU. <p>[Retaining conditions]</p> <ul style="list-style-type: none"> When the DISEL bit is 1 and data transfer has ended When the specified number of transfers have ended When the software-activated data transfer is in process |
| 6 | DTVEC6 | 0 | R/W | DTC Software Activation Vectors 0 to 6 |
| 5 | DTVEC5 | 0 | R/W | These bits specify a vector number for DTC software activation. |
| 4 | DTVEC4 | 0 | R/W | |
| 3 | DTVEC3 | 0 | R/W | The vector address is expressed as $H'0400 + (\text{vector number} \times 2)$. For example, when DTVEC6 to DTVEC0 = H'10, the vector address is H'0420. |
| 2 | DTVEC2 | 0 | R/W | |
| 1 | DTVEC1 | 0 | R/W | These bits are writable when SWDTE = 0. |
| 0 | DTVEC0 | 0 | R/W | |

9.3 Activation Sources

The DTC operates when activated by an interrupt or by a write to DTVECR by software. An interrupt request can be directed to the CPU or DTC, as designated by the corresponding DTCER bit. At the end of a data transfer (or the last consecutive transfer in the case of chain transfer), the activation source or corresponding DTCER bit is cleared. The activation source flag, in the case of RXI0, for example, is the RDRF flag of SCI_0. As there are a number of activation sources, the activation source flag is not cleared with the last byte (or word) transfer. Take appropriate measures at each interrupt as shown in table 9.1, Activation source and DTCER clearance.

Table 9.1 Activation Source and DTCER Clearance

| Activation Source | When the DIESEL Bit is 0 and the Specified Number of Transfers Have Not Ended | When the DIESEL Bit is 1, or when the Specified Number of Transfers Have Ended |
|--------------------------|--|--|
| Software activation | <ul style="list-style-type: none"> The SWDTE bit is cleared to 0 | <ul style="list-style-type: none"> The SWDTE bit remains set to 1 An interrupt is issued to the CPU |
| Interrupt activation | <ul style="list-style-type: none"> The corresponding DTCER bit remains set to 1 The activation source flag is cleared to 0 | <ul style="list-style-type: none"> The corresponding DTCER bit is cleared to 0 The activation source flag remains set to 1 A request is issued to the CPU for the activation source interrupt |

When an interrupt has been designated a DTC activation source, the existing CPU mask level and interrupt controller priorities have no effect. If there is more than one activation source at the same time, the DTC operates in accordance with the default priorities.

Figure 9.2 shows a block diagram of activation source control. For details, see section 5, Interrupt Controller.

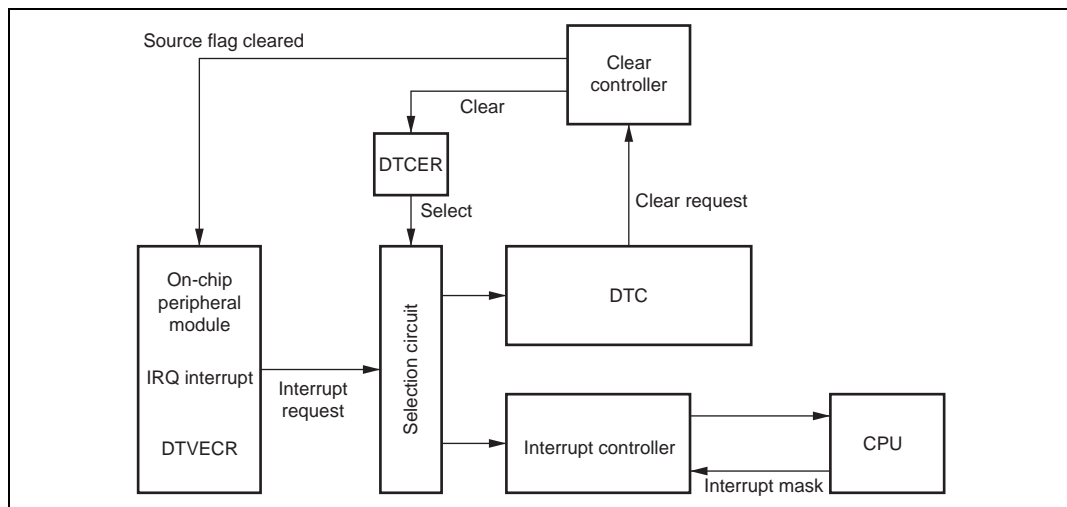


Figure 9.2 Block Diagram of DTC Activation Source Control

9.4 Location of Register Information and DTC Vector Table

Locate the register information in the on-chip RAM (addresses: H'FFEBC0 to H'FFEFBF). Register information should be located at an address that is a multiple of four within the range. Locating the register information in address space is shown in figure 9.3. Locate the MRA, SAR, MRB, DAR, CRA, and CRB registers, in that order, from the start address of the register information.

In the case of chain transfer, register information should be located in consecutive areas as shown in figure 9.3, and the register information start address should be located at the vector address corresponding to the interrupt source. Figure 9.4 shows the correspondence between DTC vector address and register information. The DTC reads the start address of the register information from the vector address set for each activation source, and then reads the register information from that start address.

When the DTC is activated by software, the vector address is obtained from: $H'0400 + (DTVECR[6:0] \times 2)$. For example, if DTVECR is H'10, the vector address is H'0420.

The configuration of the vector address is the same in both normal* and advanced modes, a 2-byte unit being used in both cases. These two bytes specify the lower bits of the register information start address.

Note: * Normal mode cannot be used in this LSI.

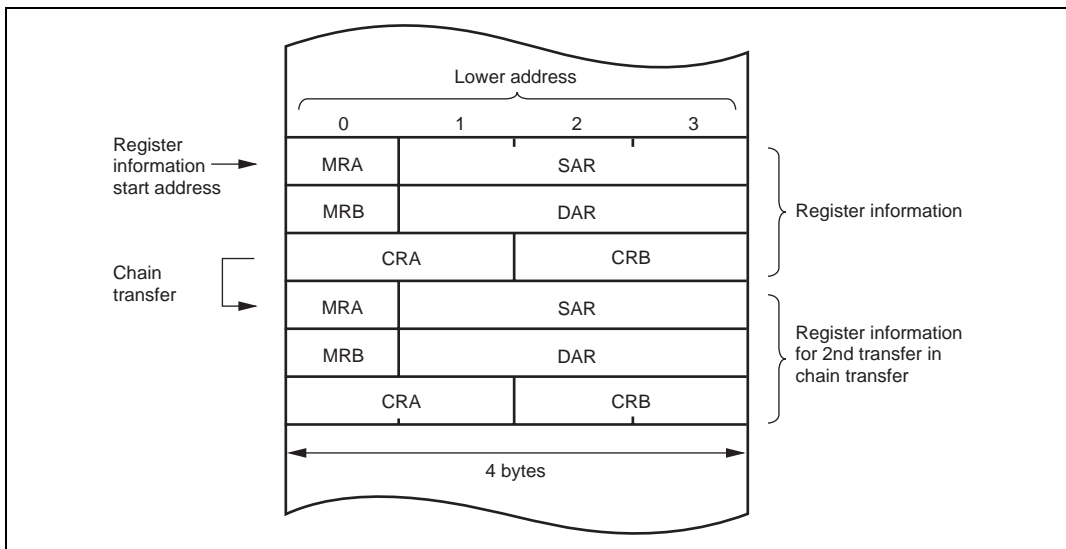


Figure 9.3 The Location of the DTC Register Information in the Address Space

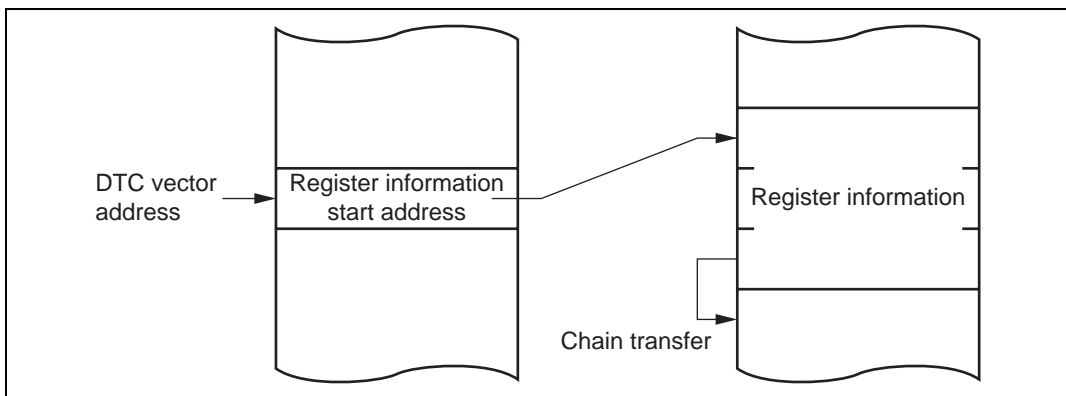


Figure 9.4 Correspondence between DTC Vector Address and Register Information

Table 9.2 Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs

| Interrupt Source | Origin of Interrupt Source | Vector Number | DTC Vector Address | DTCE ^{*1} | Priority |
|-----------------------------|----------------------------|---------------|-------------------------------|--------------------|----------|
| Software | Write to DTVECR | DTVECR | H'0400 + vector number × 2 | — | High |
| External pin | IRQ0 | 16 | H'0420 | DTCEA7 | ↑ |
| | IRQ1 | 17 | H'0422 | DTCEA6 | |
| | IRQ2 | 18 | H'0424 | DTCEA5 | |
| | IRQ3 | 19 | H'0426 | DTCEA4 | |
| | IRQ4 | 20 | H'0428 | DTCEA3 | |
| | IRQ5 | 21 | H'042A | DTCEA2 | |
| | IRQ6 | 22 | H'042C | DTCEA1 | |
| | IRQ7 | 23 | H'042E | DTCEA0 | |
| A/D converter | ADI (A/D conversion end) | 28 | H'0438 | DTCEB6 | |
| TPU Channel 0 | TGI0A | 32 | H'0440 | DTCEB5 | |
| | TGI0B | 33 | H'0442 | DTCEB4 | |
| | TGI0C | 34 | H'0444 | DTCEB3 | |
| | TGI0D | 35 | H'0446 | DTCEB2 | |
| TPU Channel 1 | TGI1A | 40 | H'0450 | DTCEB1 | |
| | TGI1B | 41 | H'0452 | DTCEB0 | |
| TPU Channel 2 | TGI2A | 44 | H'0458 | DTCEC7 | |
| | TGI2B | 45 | H'045A | DTCEC6 | |
| TPU Channel 3 ^{*4} | TGI3A | 48 | H'0460 | DTCEC5 | |
| | TGI3B | 49 | H'0462 | DTCEC4 | |
| | TGI3C | 50 | H'0464 | DTCEC3 | |
| | TGI3D | 51 | H'0466 | DTCEC2 | |
| TPU Channel 4 ^{*4} | TGI4A | 56 | H'0470 | DTCEC1 | |
| | TGI4B | 57 | H'0472 | DTCEC0 | |
| TPU Channel 5 ^{*4} | TGI5A | 60 | H'0478 | DTCED5 | |
| | TGI5B | 61 | H'047A | DTCED4 | |
| 8-bit timer channel 0 | CMIA0 | 64 | H'0480 | DTCED3 | |
| | CMIB0 | 65 | H'0482 | DTCED2 | Low |

| Interrupt Source | Origin of Interrupt Source | Vector Number | DTC Vector Address | DTCE ^{*1} | Priority |
|--|----------------------------|---------------|--------------------|--------------------|-----------|
| 8-bit timer channel 1 | CMIA1 | 68 | H'0488 | DTCED1 | High ↑ |
| | CMIB1 | 69 | H'048A | DTCED0 | |
| DMAC ^{*2} | DEND0A | 72 | H'0490 | DTCEE7 | |
| | DEND0A | 73 | H'0492 | DTCEE6 | |
| | DEND1A | 74 | H'0494 | DTCEE5 | |
| | DEND1A | 75 | H'0496 | DTCEE4 | |
| SCI channel 0 | RXI0 | 81 | H'04A2 | DTCEE3 | |
| | TXI0 | 82 | H'04A4 | DTCEE2 | |
| SCI channel 1 | RXI1 | 85 | H'04AA | DTCEE1 | |
| | TXI1 | 86 | H'04AC | DTCEE0 | |
| SCI channel 2 ^{*4} | RXI2 | 89 | H'04B2 | DTCEF7 | |
| | TXI2 | 90 | H'04B4 | DTCEF6 | |
| 8-bit timer channel 2 ^{*3} | CMIA2 | 92 | H'04B8 | DTCEF5 | Low ↓ |
| | CMIB2 | 93 | H'04BA | DTCEF4 | |
| 8-bit timer channel 3 ^{*3} | CMIA3 | 96 | H'04C0 | DTCEF3 | |
| | CMIB3 | 97 | H'04C2 | DTCEF2 | |
| IIC channel 0 (optional) ^{*3} | IICI0 | 100 | H'04C8 | DTCEF1 | |
| IIC channel 1 (optional) ^{*3} | IICI1 | 102 | H'04CC | DTCEF0 | |
| IEB ^{*5} | IERxI (RxRDY) | 105 | H'04D2 | DTCEG6 | |
| | IETxI (TxRDY) | 106 | H'04D4 | DTCEG5 | |
| SCI channel 3 | RXI3 | 121 | H'04F2 | DTCEI7 | |
| | TXI3 | 122 | H'04F4 | DTCEI6 | |

- Notes: 1. DTCE bits with no corresponding interrupt are reserved, and should be written with 0.
 2. Supported only by the H8S/2239 Group.
 3. These channels are not available in the H8S/2237 Group or H8S/2227 Group.
 4. These channels are not available in the H8S/2227 Group.
 5. Supported only by the H8S/2258 Group.

9.5 Operation

Register information is stored in on-chip RAM. When activated, the DTC reads register information in on-chip RAM and transfers data. After the data transfer, the DTC writes updated register information back to the memory.

The pre-storage of register information in memory makes it possible to transfer data over any required number of channels. The transfer mode can be specified as normal, repeat, and block transfer mode. Setting the CHNE bit in MRB to 1 makes it possible to perform a number of transfers with a single activation source (chain transfer).

The 24-bit SAR designates the DTC transfer source address, and the 24-bit DAR designates the transfer destination address. After each transfer, SAR and DAR are independently incremented, decremented, or left fixed depending on its register information.

Figure 9.5 shows the flowchart of DTC operation.

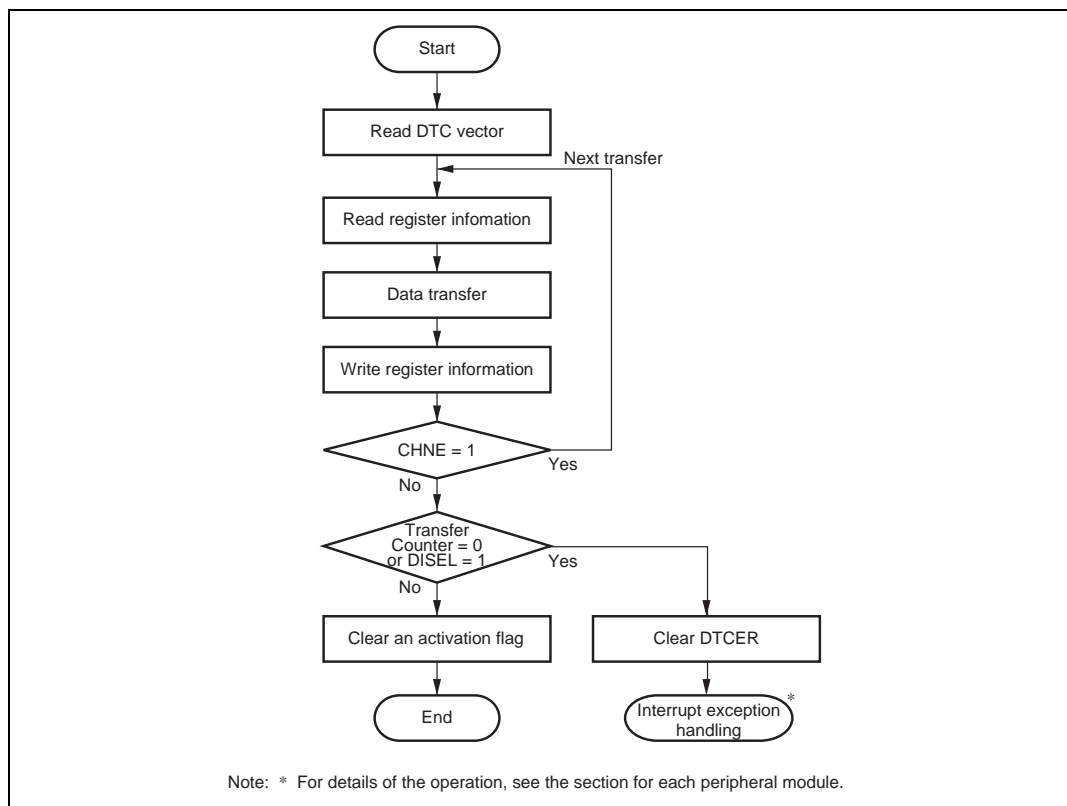


Figure 9.5 Flowchart of DTC Operation

9.5.1 Normal Mode

In normal mode, one operation transfers one byte or one word of data.

From 1 to 65,536 transfers can be specified. Once the specified number of transfers have been completed, a CPU interrupt can be requested.

Table 9.3 lists the register information in normal mode. Figure 9.6 shows the memory mapping in normal mode.

Table 9.3 Register Information in Normal Mode

| Name | Abbreviation | Function |
|----------------------------------|--------------|--------------------------------|
| DTC source address register | SAR | Designates source address |
| DTC destination address register | DAR | Designates destination address |
| DTC transfer count register A | CRA | Designates transfer count |
| DTC transfer count register B | CRB | Not used |

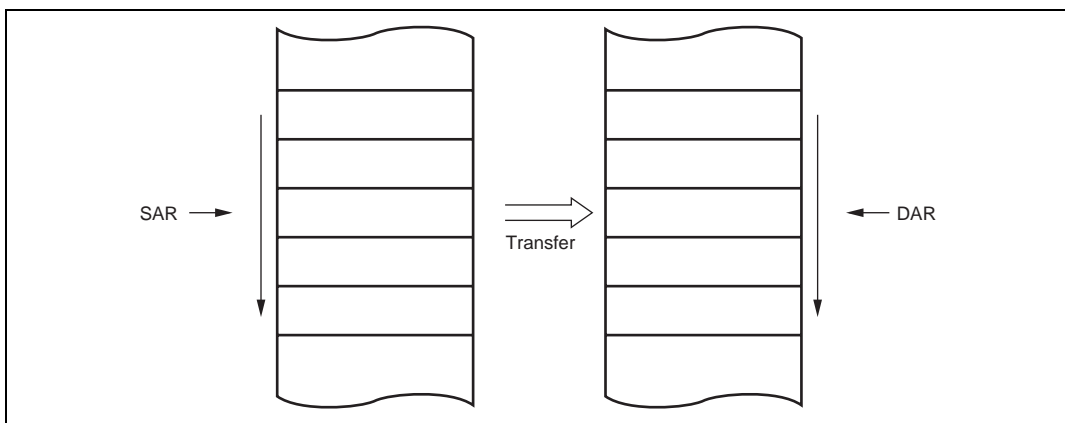


Figure 9.6 Memory Mapping in Normal Mode

9.5.2 Repeat Mode

In repeat mode, one operation transfers one byte or one word of data.

From 1 to 256 transfers can be specified. Once the specified number of transfers have ended, the initial state of the transfer counter and the address register specified as the repeat area is restored, and transfer is repeated. In repeat mode the transfer counter value does not reach H'00, and therefore CPU interrupts cannot be requested when DISEL = 0.

Table 9.4 lists the register information in repeat mode. Figure 9.7 shows the memory mapping in repeat mode.

Table 9.4 Register Information in Repeat Mode

| Name | Abbreviation | Function |
|----------------------------------|--------------|--------------------------------|
| DTC source address register | SAR | Designates source address |
| DTC destination address register | DAR | Designates destination address |
| DTC transfer count register AH | CRAH | Holds number of transfers |
| DTC transfer count register AL | CRAL | Designates transfer count |
| DTC transfer count register B | CRB | Not used |

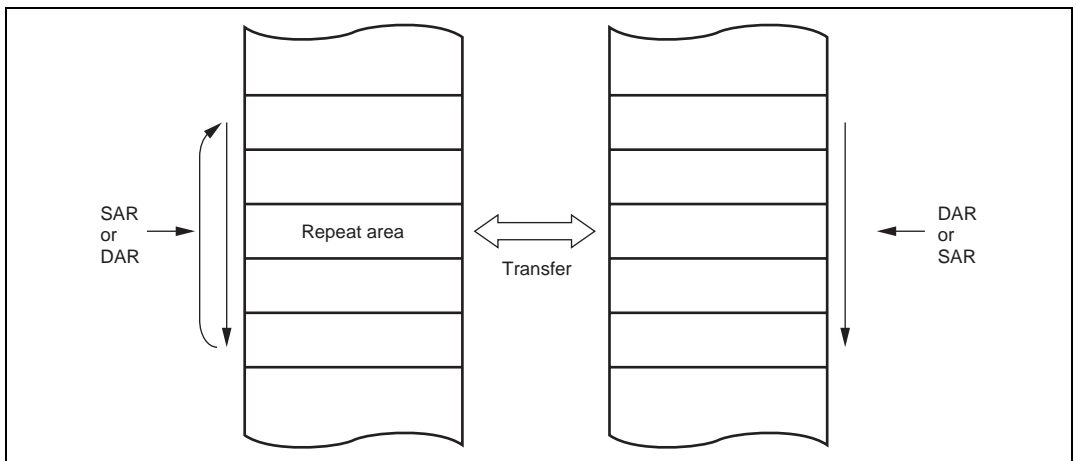


Figure 9.7 Memory Mapping in Repeat Mode

9.5.3 Block Transfer Mode

In block transfer mode, one operation transfers one block of data. Either the transfer source or the transfer destination is designated as a block area.

The block size can be between 1 to 256. When the transfer of one block ends, the initial state of the block size counter and the address register specified as the block area is restored. The other address register is then incremented, decremented, or left fixed.

From 1 to 65,536 transfers can be specified. Once the specified number of transfers have been completed, a CPU interrupt is requested.

Table 9.5 lists the register information in block transfer mode. Figure 9.8 shows the memory mapping in block transfer mode.

Table 9.5 Register Information in Block Transfer Mode

| Name | Abbreviation | Function |
|----------------------------------|--------------|--------------------------------|
| DTC source address register | SAR | Designates source address |
| DTC destination address register | DAR | Designates destination address |
| DTC transfer count register AH | CRAH | Holds block size |
| DTC transfer count register AL | CRAL | Designates block size count |
| DTC transfer count register B | CRB | Transfer count |

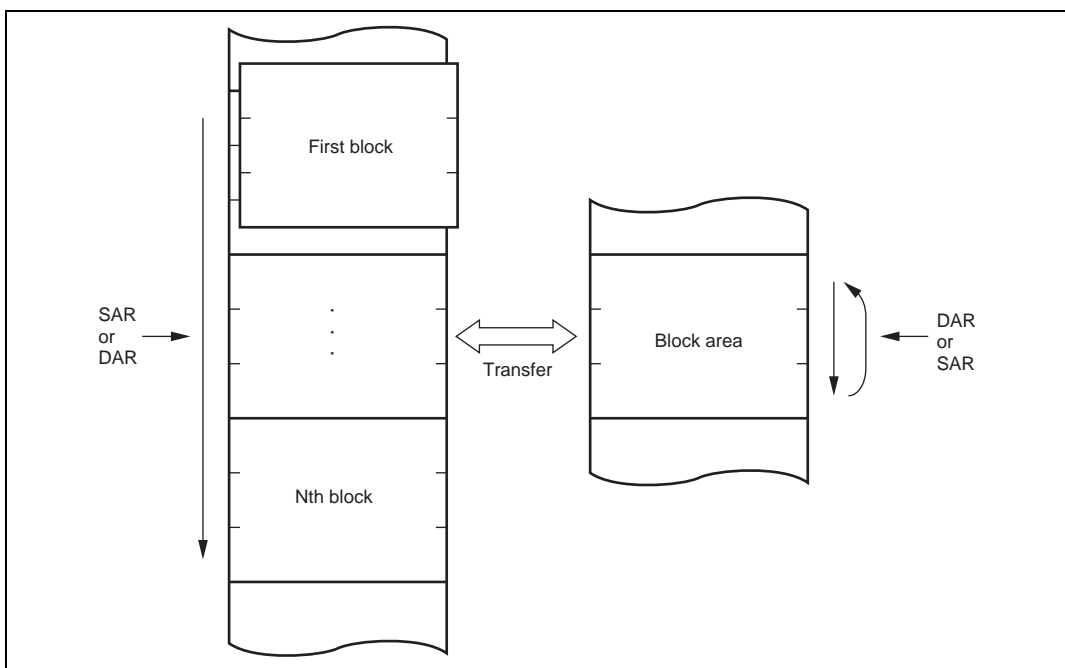


Figure 9.8 Memory Mapping in Block Transfer Mode

9.5.4 Chain Transfer

Setting the CHNE bit in MRB to 1 enables a number of data transfers to be performed consecutively in response to a single transfer request. SAR, DAR, CRA, CRB, MRA, and MRB, which define data transfers, can be set independently.

Figure 9.9 shows the memory map for chain transfer.

When activated, the DTC reads the register information start address stored at the vector address, and then reads the first register information at that start address. After the data transfer, the CHNE bit will be tested. When it has been set to 1, DTC reads the next register information located in a consecutive area and performs the data transfer. These sequences are repeated until the CHNE bit is cleared to 0.

In the case of transfer with CHNE set to 1, an interrupt request to the CPU is not generated at the end of the specified number of transfers or by setting of the DISEL bit to 1, and the interrupt source flag for the activation source is not affected.

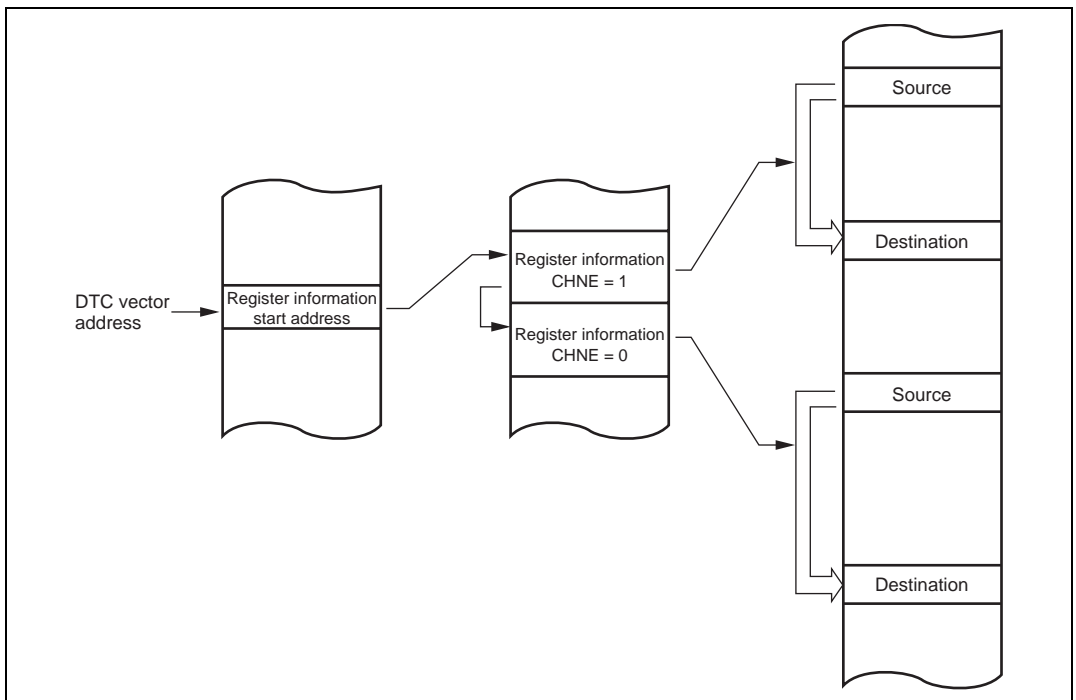


Figure 9.9 Chain Transfer Operation

9.5.5 Interrupts

An interrupt request is issued to the CPU when the DTC has completed the specified number of data transfers, or a data transfer for which the DISEL bit was set to 1. In the case of interrupt activation, the interrupt set as the activation source is generated. These interrupts to the CPU are subject to CPU mask level and interrupt controller priority level control.

In the case of software activation, a software-activated data transfer end interrupt (SWDTEND) is generated.

When the DISEL bit is 1 and one data transfer has been completed, or the specified number of transfers have been completed, after data transfer ends the SWDTE bit is held at 1 and an SWDTEND interrupt is generated. The interrupt handling routine will then clear the SWDTE bit to 0.

When the DTC is activated by software, an SWDTEND interrupt is not generated during a data transfer wait or during data transfer even if the SWDTE bit is set to 1.

9.5.6 Operation Timing

Figures 9.10 to 9.12 show the DTC operation timings.

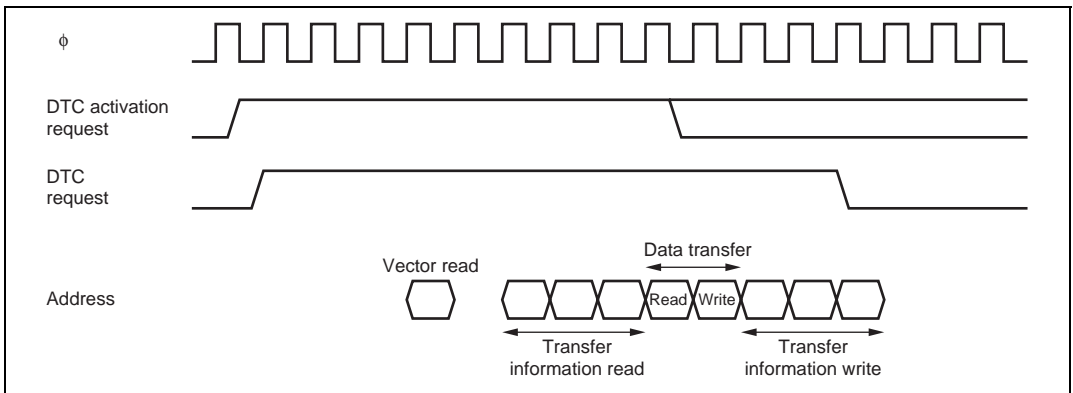


Figure 9.10 DTC Operation Timing (Example in Normal Mode or Repeat Mode)

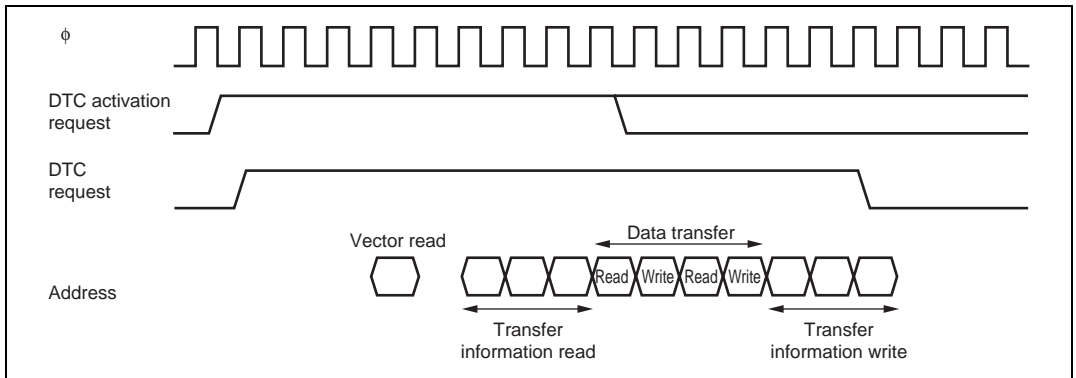


Figure 9.11 DTC Operation Timing
(Example of Block Transfer Mode, with Block Size of 2)

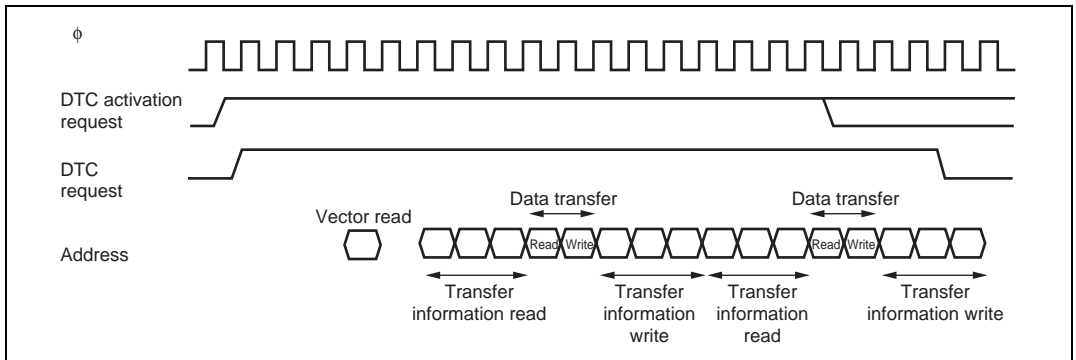


Figure 9.12 DTC Operation Timing (Example of Chain Transfer)

9.5.7 Number of DTC Execution States

Table 9.6 lists execution status for a single DTC data transfer, and table 9.7 shows the number of states required for each execution status.

Table 9.6 DTC Execution Status

| Mode | Vector Read I | Register Information | | Data Read K | Data Write L | Internal Operations M |
|----------------|------------------|----------------------|--|----------------|-----------------|-----------------------------|
| | | Read/Write J | | | | |
| Normal | 1 | 6 | | 1 | 1 | 3 |
| Repeat | 1 | 6 | | 1 | 1 | 3 |
| Block transfer | 1 | 6 | | N | N | 3 |

Legend:

N: Block size (initial setting of CRAH and CRAL)

Table 9.7 Number of States Required for Each Execution Status

| Object to be Accessed | | On- Chip RAM | On- Chip ROM | Internal I/O Registers | | External Devices | | | |
|--------------------------|--|--------------------|--------------------|---------------------------|----|------------------|----------|----|---------|
| | | | | | | | | | |
| Bus width | | 32 | 16 | 8 | 16 | 8 | 8 | 16 | 16 |
| Access states | | 1 | 1 | 2 | 2 | 2 | 3 | 2 | 3 |
| Execution Status | Vector read S_I | — | 1 | — | — | 4 | $6 + 2m$ | 2 | $3 + m$ |
| | Register information read/write S_J | 1 | — | — | — | — | — | — | — |
| | Byte data read S_K | 1 | 1 | 2 | 2 | 2 | $3 + m$ | 2 | $3 + m$ |
| | Word data read S_K | 1 | 1 | 4 | 2 | 4 | $6 + 2m$ | 2 | $3 + m$ |
| | Byte data write S_L | 1 | 1 | 2 | 2 | 2 | $3 + m$ | 2 | $3 + m$ |
| | Word data write S_L | 1 | 1 | 4 | 2 | 4 | $6 + 2m$ | 2 | $3 + m$ |
| Internal operation S_M | | 1 | | | | | | | |

Legend:

m: The number of wait states for accessing external devices.

The number of execution states is calculated from using the formula below. Note that Σ is the sum of all transfers activated by one activation event (the number in which the CHNE bit is set to 1, plus 1).

$$\text{Number of execution states} = I \cdot S_I + \Sigma (J \cdot S_J + K \cdot S_K + L \cdot S_L) + M \cdot S_M$$

For example, when the DTC vector address table is located in the on-chip ROM, normal mode is set, and data is transferred from on-chip ROM to an internal I/O register, then the time required for the DTC operation is 13 states. The time from activation to the end of the data write is 10 states.

9.6 Procedures for Using DTC

9.6.1 Activation by Interrupt

The procedure for using the DTC with interrupt activation is as follows:

1. Set the MRA, MRB, SAR, DAR, CRA, and CRB register information in on-chip RAM.
2. Set the start address of the register information in the DTC vector address.
3. Set the corresponding bit in DTCER to 1.
4. Set the enable bits for the interrupt sources to be used as the activation sources to 1. The DTC is activated when an interrupt used as an activation source is generated.
5. After one data transfer has been completed, or after the specified number of data transfers have been completed, the DTCE bit is cleared to 0 and a CPU interrupt is requested. If the DTC is to continue transferring data, set the DTCE bit to 1.

9.6.2 Activation by Software

The procedure for using the DTC with software activation is as follows:

1. Set the MRA, MRB, SAR, DAR, CRA, and CRB register information in on-chip RAM.
2. Set the start address of the register information in the DTC vector address.
3. Check that the SWDTE bit is 0.
4. Write 1 to SWDTE bit and the vector number to DTVECR.
5. Check the vector number written to DTVECR.
6. After one data transfer has been completed, if the DISEL bit is 0 and a CPU interrupt is not requested, the SWDTE bit is cleared to 0. If the DTC is to continue transferring data, set the SWDTE bit to 1. When the DISEL bit is 1, or after the specified number of data transfers have been completed, the SWDTE bit is held at 1 and a CPU interrupt is requested.

9.7 Examples of Use of the DTC

9.7.1 Normal Mode

An example is shown in which the DTC is used to receive 128 bytes of data via the SCI.

1. Set MRA to a fixed source address ($SM1 = SM0 = 0$), incrementing destination address ($DM1 = 1, DM0 = 0$), normal mode ($MD1 = MD0 = 0$), and byte size ($Sz = 0$). The DTS bit can have any value. Set MRB for one data transfer by one interrupt ($CHNE = 0, DISEL = 0$). Set the SCI RDR address in SAR, the start address of the RAM area where the data will be received in DAR, and 128 (H'0080) in CRA. CRB can be set to any value.
2. Set the start address of the register information at the DTC vector address.
3. Set the corresponding bit in DTCER to 1.
4. Set the SCI to the appropriate receive mode. Set the RIE bit in SCR to 1 to enable the reception complete (RXI) interrupt. Since the generation of a receive error during the SCI reception operation will disable subsequent reception, the CPU should be enabled to accept receive error interrupts.
5. Each time the reception of one byte of data has been completed on the SCI, the RDRF flag in SSR is set to 1, an RXI interrupt is generated, and the DTC is activated. The receive data is transferred from RDR to RAM by the DTC. DAR is incremented and CRA is decremented. The RDRF flag is automatically cleared to 0.
6. When CRA becomes 0 after the 128 data transfers have been completed, the RDRF flag is held at 1, the DTCE bit is cleared to 0, and an RXI interrupt request is sent to the CPU. The interrupt handling routine will perform wrap-up processing.

9.7.2 Software Activation

An example is shown in which the DTC is used to transfer a block of 128 bytes of data by means of software activation. The transfer source address is H'1000 and the destination address is H'2000. The vector number is H'60, so the vector address is H'04C0.

1. Set MRA to incrementing source address ($SM1 = 1, SM0 = 0$), incrementing destination address ($DM1 = 1, DM0 = 0$), block transfer mode ($MD1 = 1, MD0 = 0$), and byte size ($Sz = 0$). The DTS bit can have any value. Set MRB for one block transfer by one interrupt ($CHNE = 0$). Set the transfer source address (H'1000) in SAR, the destination address (H'2000) in DAR, and 128 (H'8080) in CRA. Set 1 (H'0001) in CRB.
2. Set the start address of the register information at the DTC vector address (H'04C0).
3. Check that the SWDTE bit in DTVECR is 0. Check that there is currently no transfer activated by software.

4. Write 1 to the SWDTE bit and the vector number (H'60) to DTVECR. The write data is H'E0.
5. Read DTVECR again and check that it is set to the vector number (H'60). If it is not, this indicates that the write failed. This is presumably because an interrupt occurred between steps 3 and 4 and led to a different software activation. To activate this transfer, go back to step 3.
6. If the write was successful, the DTC is activated and a block of 128 bytes of data is transferred.
7. After the transfer, an SWDTEND interrupt occurs. The interrupt handling routine should clear the SWDTE bit to 0 and perform other wrap-up processing.

9.8 Usage Notes

9.8.1 Module Stop Mode Setting

DTC operation can be disabled or enabled using the module stop control register. The initial setting is for DTC operation to be enabled. Register access is disabled by setting module stop mode. Module stop mode cannot be set during DTC operation. For details, refer to section 24, Power-Down Modes.

9.8.2 On-Chip RAM

The MRA, MRB, SAR, DAR, CRA, and CRB registers are all located in on-chip RAM. When the DTC is used, the RAME bit in SYSCR should not be cleared to 0.

9.8.3 DTCE Bit Setting

For DTCE bit setting, use bit manipulation instructions such as BSET and BCLR. If all interrupts are masked, multiple activation sources can be set at one time (only at the initial setting) by writing data after executing a dummy read on the relevant register.

Section 10 I/O Ports

Table 10.1 summarizes the port functions. The pins of each port also have other functions such as input/output or interrupt input pins of on-chip peripheral modules.

Each I/O port includes a data direction register (DDR) that controls input/output, a data register (DR) that stores output data, and a port register (PORT) used to read the pin states. The input-only ports do not have DR and DDR registers.

Ports A to E have a built-in input pull-up MOS function and an input pull-up MOS control register (PCR) to control the on/off state of input pull-up MOS respectively.

Ports 3 and A include an open-drain control register (ODR) that controls the on/off state of the output buffer PMOS respectively.

All the I/O ports can drive a single TTL load and a 30-pF capacitive load.

The P35 and P34 pins on port 3 are NMOS push pull outputs.*

The IRQ pin is Schmitt-trigger input.

Note: * Supported only by the H8S/2258 Group, H8S/2239 Group, and H8S/2238 Group.

Table 10.1 Port Functions

| Port | Description | Mode4 | Mode5 | Mode 6 | Mode 7 | Input/Output and Output Type |
|--------|--|---|-------|--------|--|---|
| Port 1 | General I/O port also functioning as TPU_2, TPU_1, and TPU_0 I/O pins, interrupt input pins, address output pins, and DMAC output pins | P17/TIOCB2/TCLKD P16/TIOCA2/ $\overline{\text{IRQ1}}$ P15/TIOCB1/TCLKC P14/TIOCA1/ $\overline{\text{IRQ0}}$ P13/TIOCD0/TCLKB/A23 P12/TIOCC0/TCLKA/A22 P11/TIOCB0/ $\overline{\text{DACK1}}$ ^{*3} /A21 P10/TIOCA0/ $\overline{\text{DACK0}}$ ^{*3} /A20 | | | P17/TIOCB2/TCLKD P16/TIOCA2/ $\overline{\text{IRQ1}}$ P15/TIOCB1/TCLKC P14/TIOCA1/ $\overline{\text{IRQ0}}$ P13/TIOCD0/TCLKB P12/TIOCC0/TCLKA P11/TIOCB0/ $\overline{\text{DACK1}}$ ^{*3} P10/TIOCA0/ $\overline{\text{DACK0}}$ ^{*3} | Schmitt-trigger input ($\overline{\text{IRQ0}}$, $\overline{\text{IRQ1}}$) |
| Port 3 | General I/O port also functioning as I ² C bus interface ^{*1} I/O pins, SCI_1 and SCI_0 I/O pins, and interrupt input pins | P36 P35/SCK1/SCL0 ^{*1} / $\overline{\text{IRQ5}}$ P34/RxD1/SDA0 ^{*1} P33/TxD1/SDA0 ^{*1} P32/SCK0/SDA1 ^{*1} / $\overline{\text{IRQ4}}$ P31/RxD0 P30/TxD0 | | | | Specifiable of open drain output Schmitt-trigger input ($\overline{\text{IRQ4}}$, $\overline{\text{IRQ5}}$) NMOS push-pull output ^{*1} (P35, P34, SCK1) |
| Port 4 | General input port also functioning as A/D converter analog input pins | P47/AN7 P46/AN6 P45/AN5 P44/AN4 P43/AN3 P42/AN2 P41/AN1 P40/AN0 | | | | |
| Port 7 | General I/O port also functioning as SCI_3 I/O pins, TMR_3 ^{*1} , TMR_2 ^{*1} , TMR_1, TMR_0 I/O pins, and DMAC I/O pins | P77/TxD3 P76/RxD3 P75/TMO3 ^{*1} /SCK3 P74/TMO2 ^{*1} / $\overline{\text{MRES}}$ P73/TMO1/ $\overline{\text{TEND1}}$ ^{*3} /CS7 P72/TMO0/ $\overline{\text{TEND0}}$ ^{*3} /CS6 P71/TMRI23 ^{*1} /TMCI23 ^{*1} / $\overline{\text{DREQ1}}$ ^{*3} /CS5 P70/TMRI01/TMCI01/ $\overline{\text{DREQ0}}$ ^{*3} /CS4 | | | P73/TMO1/ $\overline{\text{TEND1}}$ ^{*3} P72/TMO0/ $\overline{\text{TEND0}}$ ^{*3} P71/TMRI23 ^{*1} /TMCI23 ^{*1} / $\overline{\text{DREQ1}}$ ^{*3} P70/TMRI01/TMCI01/ $\overline{\text{DREQ0}}$ ^{*3} | |

| Port | Description | Mode 4 | Mode5 | Mode 6 | Mode 7 | Input/Output and Output Type |
|--------|---|--|--|--|--|---|
| Port 9 | General I/O port also functioning as D/A converter ^{*2} analog output pins | P97/DA1 ^{*2} P96/ DA0 ^{*2} | | | | |
| Port A | General I/O port also functioning as SCI_2 ^{*2} I/O pins and address output pins | PA3/A19/SCK2 ^{*2} PA2/A18/RxD2 ^{*2} PA1/A17/TxD2 ^{*2} PA0/A16 | | | PA3/SCK2 ^{*2} PA2/RxD2 ^{*2} PA1/TxD2 ^{*2} PA0 | Specifiable of built-in input pull-up MOS open drain output |
| Port B | General I/O port also functioning as TPU_5 ^{*2} , TPU_4 ^{*2} , TPU_3 ^{*2} I/O pins, and address output pins | PB7/A15/TIOCB5 ^{*2} PB6/A14/TIOCA5 ^{*2} PB5/A13/TIOCB4 ^{*2} PB4/A12/TIOCA4 ^{*2} PB3/A11/TIOCD3 ^{*2} PB2/A10/TIOCC3 ^{*2} PB1/A9/TIOCB3 ^{*2} PB0/A8/TIOCA3 ^{*2} | | | PB7/TIOCB5 ^{*2} PB6/TIOCA5 ^{*2} PB5/TIOCB4 ^{*2} PB4/TIOCA4 ^{*2} PB3/TIOCD3 ^{*2} PB2/TIOCC3 ^{*2} PB1/TIOCB3 ^{*2} PB0/TIOCA3 ^{*2} | Built-in input pull-up MOS |
| Port C | General I/O port also functioning as address output pins | A7 A6 A5 A4 A3 A2 A1 A0 | PC7/A7 PC6/A6 PC5/A5 PC4/A4 PC3/A3 PC2/A2 PC1/A1 PC0/A0 | PC7 PC6 PC5 PC4 PC3 PC2 PC1 PC0 | | Built-in input pull-up MOS |
| Port D | General I/O port also functioning as data I/O pins | D15 D14 D13 D12 D11 D10 D9 D8 | | | PD7 PD6 PD5 PD4 PD3 PD2 PD1 PD0 | Built-in input pull-up MOS |

| Port | Description | Mode 4 | Mode5 | Mode 6 | Mode 7 | Input/Output and Output Type |
|--------|--|--|-------|--------|--|--|
| Port E | General I/O port also functioning as data I/O pins | PE7/D7 PE6/D6 PE5/D5 PE4/D4 PE3/D3 PE2/D2 PE1/D1 PE0/D0 | | | PE7 PE6 PE5 PE4 PE3 PE2 PE1 PE0 | Built-in input pull-up MOS |
| Port F | General I/O port also functioning as interrupt input pins, bus control I/O pins, an A/D converter input pins and WDT output pins | PF7/ ϕ \overline{AS} \overline{RD} \overline{HWR} PF3/ $\overline{LWR}/\overline{ADTRG}/\overline{IRQ3}$ PF2/ \overline{WAIT} PF1/ $\overline{BACK}/BUZZ$ PF0/ $\overline{BREQ}/\overline{IRQ2}$ | | | PF7/ ϕ PF6 PF5 PF4 PF3/ $\overline{ADTRG}/\overline{IRQ3}$ PF2 PF1/ \overline{BUZZ} PF0/ $\overline{IRQ2}$ | Schmit-trigger input ($\overline{IRQ2}$, $\overline{IRQ3}$) |
| Port G | General I/O port also functioning as interrupt input pins | PG4/ $\overline{CS0}$ PG3/ $\overline{Rx}/\overline{CS1}^{*4}$ PG2/ $\overline{Tx}/\overline{CS2}^{*4}$ PG1/ $\overline{CS3}/\overline{IRQ7}$ PG0/ $\overline{IRQ6}$ | | | PG4 PG3/ \overline{Rx} PG2/ \overline{Tx} PG1/ $\overline{IRQ7}$ PG0/ $\overline{IRQ6}$ | Schmit-trigger input ($\overline{IRQ6}$, $\overline{IRQ7}$) |

- Notes:
1. Not available in the H8S/2237 Group and H8S/2227 Group.
 2. Not available in the H8S/2227 Group.
 3. Supported only by the H8S/2239 Group.
 4. Supported only by the H8S/2258 Group.

10.1 Port 1

Port 1 is an 8-bit I/O port and has the following registers.

- Port 1 data direction register (P1DDR)
- Port 1 data register (P1DR)
- Port 1 register (PORT1)

10.1.1 Port 1 Data Direction Register (P1DDR)

P1DDR specifies input or output of the port 1 pins using the individual bits. P1DDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | P17DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port 1 pin an output pin. Clearing this bit to 0 makes the pin an input pin. |
| 6 | P16DDR | 0 | W | |
| 5 | P15DDR | 0 | W | |
| 4 | P14DDR | 0 | W | |
| 3 | P13DDR | 0 | W | |
| 2 | P12DDR | 0 | W | |
| 1 | P11DDR | 0 | W | |
| 0 | P10DDR | 0 | W | |

10.1.2 Port 1 Data Register (P1DR)

P1DR stores output data for port 1 pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | P17DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | P16DR | 0 | R/W | |
| 5 | P15DR | 0 | R/W | |
| 4 | P14DR | 0 | R/W | |
| 3 | P13DR | 0 | R/W | |
| 2 | P12DR | 0 | R/W | |
| 1 | P11DR | 0 | R/W | |
| 0 | P10DR | 0 | R/W | |

10.1.3 Port 1 Register (PORT1)

PORT1 shows the pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | P17 | —* | R | If a port 1 read is performed while P1DDR bits are set to 1, the P1DR values are read. If a port 1 read is performed while P1DDR bits are cleared to 0, the pin states are read. |
| 6 | P16 | —* | R | |
| 5 | P15 | —* | R | |
| 4 | P14 | —* | R | |
| 3 | P13 | —* | R | |
| 2 | P12 | —* | R | |
| 1 | P11 | —* | R | |
| 0 | P10 | —* | R | |

Note: * Determined by the states of pins P17 to P10.

10.1.4 Pin Functions

Port 1 pins also function as TPU I/O pins (TPU_0, TPU_1, and TPU_2), DMAC* output pins, interrupt input pins and address output pins. Values of the register and pin functions are shown below.

Note: * Supported only by the H8S/2239 Group.

- P17/TIOCB2/TCLKD

The pin functions are switched as shown below according to the combination of the TPU channel 2 setting, TPSC2 to TPS0 bits in TCR_0 and TCR_5, and the P17DDR bit.

| TPU Channel 2 Setting ^{*1} | Output | Input or Initial Value | |
|-------------------------------------|-------------------------------|--------------------------------|----------------|
| P17DDR | — | 0 | 1 |
| Pin functions | TIOCB2 output pin | P17 input pin | P17 output pin |
| | | TIOCB2 input pin ^{*2} | |
| | TCLKD input pin ^{*3} | | |

- Notes:
1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCB2 input when TPU channel 2 timer operating mode is set to normal operating or phase counting mode and IOB3 in TIOR_2 is set to 1.
 3. This pin functions as TCLKD input when TPSC2 to TPSC0 in TCR_0 or TCR_5 are set to 111 or when channels 2 and 4 are set to phase counting mode.

- P16/TIOCA2/IRQ1

The pin functions are switched as shown below according to the combination of the TPU channel 2 setting and the P16DDR bit.

| TPU Channel 2 Setting ^{*1} | Output | Input or Initial Value | |
|-------------------------------------|------------------------------|--------------------------------|----------------|
| P16DDR | — | 0 | 1 |
| Pin functions | TIOCA2 output pin | P16 input pin | P16 output pin |
| | | TIOCA2 input pin ^{*2} | |
| | IRQ1 input pin ^{*3} | | |

- Notes:
1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCA2 input when TPU channel 2 timer operating mode is set to normal operating or phase counting mode and IOA3 in TIOR_2 is 1.
 3. When this pin is used as an external interrupt pin, do not specify other functions.

- P15/TIOCB1/TCLKC

The pin functions are switched as shown below according to the combination of the TPU channel 1 setting, TPSC2 to TPS0 bits in TCR_0, TCR_2, TCR_4, and TCR_5 and the P15DDR bit.

| TPU Channel 1 Setting ^{*1} | Output | Input or Initial Value | |
|-------------------------------------|-------------------------------|--------------------------------|----------------|
| P15DDR | — | 0 | 1 |
| Pin functions | TIOCB1 output pin | P15 input pin | P15 output pin |
| | | TIOCB1 input pin ^{*2} | |
| | TCLKC input pin ^{*3} | | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCB1 input when TPU channel 1 timer operating mode is set to normal operating or phase counting mode and IOB3 to IOB0 in TIOR_1 are set to 10xx.
 3. This pin functions as TCLKC input when TPSC2 to TPSC0 in TCR_0 or TCR_2 are set to 110 or TPSC2 to TPSC0 in TCR_4 or TCR_0 are 101 or when channels 2 and 4 are set to phase counting mode.

- P14/TIOCA1/ $\overline{\text{IRQ0}}$

The pin functions are switched as shown below according to the combination of the TPU channel 1 setting and the P14DDR bit.

| TPU Channel 1 Setting ^{*1} | Output | Input or Initial Value | |
|-------------------------------------|------------------------------|--------------------------------|----------------|
| P14DDR | — | 0 | 1 |
| Pin functions | TIOCA1 output pin | P14 input pin | P14 output pin |
| | | TIOCA1 input pin ^{*2} | |
| | IRQ0 input pin ^{*3} | | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCA1 input when TPU channel 1 timer operating mode is set to normal operating or phase counting mode and IOA3 to IOA0 in TIOR_1 are set to 10xx.
 3. When this pin is used as an external interrupt pin, do not specify other functions.

- P13/TIOCD0/TCLKB/A23

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 0 setting, TPSC2 to TPSC0 bits in TCR_0 to TCR_2, AE3 to AE0 bits in PFCR and the P13DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|-------------------------------------|----------------|-------------------------------|----------------------------|----------------|-------------------------------|--------------------------------|----------------|
| AE3 to AE0 | B'1111 | Other than B'1111 | | | — | | |
| TPU Channel 0 Setting ^{*1} | — | Output | Input or Initial Value | | Output | Input or Initial Value | |
| P13DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A23 output pin | TIOCD0 output pin | P13 input pin | P13 output pin | TIOCD0 output pin | P13 input pin | P13 output pin |
| | | | TIOCD0 input ^{*2} | | | TIOCD0 input pin ^{*2} | |
| | | TCLKB input pin ^{*3} | | | TCLKB input pin ^{*3} | | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCD0 input when TPU channel 0 timer operating mode is set to normal operating and IOD3 to IOD0 in TIORL_0 are set to 10xx.
 3. This pin functions as TCLKB input when TPSC2 to TPSC0 in any of TCR_0 to TCR_2 are set to 101 or when channels 1 and 5 are set to phase counting mode.

- P12/TIOCC0/TCLKA/A22

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 0 setting, TPSC2 to TPSC0 bits in TCR_0 to TCR_5, AE3 to AE0 bits in PFCR, and the P12DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|-------------------------------------|----------------|-------------------------------|--------------------------------|----------------|-------------------|--------------------------------|----------------|
| AE3 to AE0 | B'1111 | Other than B'1111 | | | — | | |
| TPU Channel 0 Setting ^{*1} | — | Output | Input or Initial Value | | Output | Input or Initial Value | |
| P12DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A22 output pin | TIOCC0 output pin | P12 input pin | P12 output pin | TIOCC0 output pin | P12 input pin | P12 output pin |
| | | | TIOCC0 input pin ^{*2} | | | TIOCC0 input pin ^{*2} | |
| | | TCLKA input pin ^{*3} | | | | TCLKA input pin ^{*3} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCC0 input when TPU channel 0 timer operating mode is set to normal operating and IOC3 to IOC0 in TIORL_0 are set to 10xx.
 3. This pin functions as TCLKB input when TPSC2 to TPSC0 in any of TCR_0 to TCR_5 are set to 100 or when channels 1 and 5 are set to phase counting mode.

- P11/TIOCB0/ $\overline{\text{DACK1}}$ /A21

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 0 setting, AE3 to AE0 bits in PFCR, the SAE1 bit^{*3} in DMABCRH, and the P11DDR bit.

| Operating mode | Modes 4 to 6 | | | | | Mode 7 | | |
|-------------------------------------|----------------|-------------------|--------------------------------|----------------|--|-------------------|--------------------------------|----------------|
| AE3 to AE0 | B'111x | Other than B'111x | | | | — | | |
| SAE1 ^{*3} | — | 0 | | | 1 | — | | |
| TPU Channel 0 Setting ^{*1} | — | Output | Input or Initial Value | | — | Output | Input or Initial Value | |
| P11DDR | — | — | 0 | 1 | — | — | 0 | 1 |
| Pin functions | A21 output pin | TIOCB0 output pin | P11 input pin | P11 output pin | $\overline{\text{DACK1}}$ ^{*3} output pin | TIOCB0 output pin | P11 input pin | P11 output pin |
| | | | TIOCB0 input pin ^{*2} | | | | TIOCB0 input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCB0 input when TPU channel 0 timer operating mode is set to normal operating and IOB3 to IOB0 in TIORH_0 are set to 10xx.
 3. Supported only by the H8S/2239 Group.

- P10/TIOCA0/ $\overline{\text{DACK0}}$ /A20

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 0 setting, AE3 to AE0 bits in PFCR, the SAE0 bit^{*3} in DMABCRH, and the P10DDR bit.

| Operating mode | Modes 4 to 6 | | | | | Mode 7 | | |
|-------------------------------------|------------------|-------------------------------|--------------------------------|----------------|--|-------------------|--------------------------------|----------------|
| AE3 to AE0 | B'1101 or B'111x | Other than (B'1101 or B'111x) | | | | — | | |
| SAE0 ^{*3} | — | 0 | | | 1 | — | | |
| TPU Channel 0 Setting ^{*1} | — | Output | Input or Initial Value | | — | Output | Input or Initial Value | |
| P10DDR | — | — | 0 | 1 | — | — | 0 | 1 |
| Pin functions | A20 output pin | TIOCA0 output pin | P10 input pin | P10 output pin | $\overline{\text{DACK0}}$ ^{*3} output pin | TIOCA0 output pin | P10 input pin | P10 output pin |
| | | | TIOCA0 input pin ^{*2} | | | | TIOCA0 input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCA0 input when TPU channel 0 timer operating mode is set to normal operating and IOA3 to IOA0 in TIORH_0 are set to 10xx.
 3. Supported only by the H8S/2239 Group.

10.2 Port 3

Port 3 is a general 7-bit I/O port and has the following registers.

The P34, P35, and SCK1 function as NMOS push/pull outputs.*

- Port 3 data direction register (P3DDR)
- Port 3 data register (P3DR)
- Port 3 register (PORT3)
- Port 3 open drain control register (P3ODR)

Note: * Function as CMOS outputs in the H8S/2237 Group and H8S/2227 Group.

10.2.1 Port 3 Data Direction Register (P3DDR)

P3DDR specifies input or output of the port 3 pins using the individual bits.

P3DDR cannot be read; if it is, an undefined value will be read.

This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|--|----------|---------------|-----|---|
| 7 | — | Undefined | — | Reserved |
| These bits are always read as undefined value. | | | | |
| 6 | P36DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port 3 pin an output port. Clearing this bit to 0 makes the pin an input port. |
| 5 | P35DDR | 0 | W | |
| 4 | P34DDR | 0 | W | |
| 3 | P33DDR | 0 | W | |
| 2 | P32DDR | 0 | W | |
| 1 | P31DDR | 0 | W | |
| 0 | P30DDR | 0 | W | |

10.2.2 Port 3 Data Register (P3DR)

P3DR stores output data for port 3 pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 6 | P36DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 5 | P35DR | 0 | R/W | |
| 4 | P34DR | 0 | R/W | |
| 3 | P33DR | 0 | R/W | |
| 2 | P32DR | 0 | R/W | |
| 1 | P31DR | 0 | R/W | |
| 0 | P30DR | 0 | R/W | |

10.2.3 Port 3 Register (PORT3)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 6 | P36 | —* | R | If a port 3 read is performed while P3DDR bits are set to 1, the P3DR values are read. If a port 3 read is performed while P3DDR bits are cleared to 0, the pin states are read. |
| 5 | P35 | —* | R | |
| 4 | P34 | —* | R | |
| 3 | P33 | —* | R | |
| 2 | P32 | —* | R | |
| 1 | P31 | —* | R | |
| 0 | P30 | —* | R | |

Note: * Determined by the states of pins P36 to P30.

10.2.4 Port 3 Open Drain Control Register (P3ODR)

P3ODR controls on/off state of the PMOS for port 3 pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 6 | P36ODR | 0 | R/W | When each of P36ODR and P33ODR to P30ODR bits is set to 1, the corresponding pins P36 and P33 to P30 function as NMOS open drain outputs. When cleared to 0, the corresponding pins function as CMOS outputs. When each of P35ODR and P34ODR bits is set to 1, the corresponding pins P35 and P34 function as open drain outputs. When they are cleared to 0, the corresponding pins function as NMOS push pull outputs.* |
| 5 | P35ODR | 0 | R/W | |
| 4 | P34ODR | 0 | R/W | |
| 3 | P33ODR | 0 | R/W | |
| 2 | P32ODR | 0 | R/W | |
| 1 | P31ODR | 0 | R/W | |
| 0 | P30ODR | 0 | R/W | |

Note: * When they are cleared to 0, the corresponding pins function as CMOS outputs in the H8S/2237 Group and H8S/2227 Group.

10.2.5 Pin Functions

The port 3 pins also function as SCI I/O input pins, I2C bus interface* I/O pins, and as external interrupt input pins.

As shown in figure 10.1, when the pins P35, P34, SCK1, SCL0, or SDA0 type open drain output is used, a bus line is not affected even if the power supply for this LSI fails. Use (a) type open drain output when using a bus line having a state in which the power is not supplied to this LSI.

Note: * The I²C bus interface is not available in the H8S/2237 Group and H8S/2227 Group.

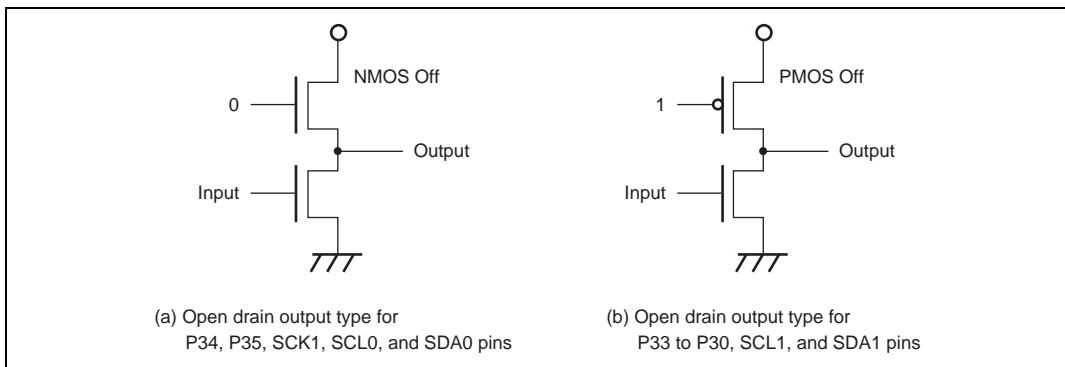


Figure 10.1 Types of Open Drain Outputs

The P34, P35, and SCK1 NMOS push-pull outputs will not output the V_{cc} level, regardless of the load, even if set to the high output state. External pull-up resistors are required to output the V_{cc} level.

- Notes: 1. Note that the signal rise and fall times become longer when external pull-up resistors are connected. If signals with long rise and fall times are input, use input circuits with noise absorbing functions, such as Schmitt trigger circuits.
2. Implement external circuit countermeasures such as inserting level shifters if the device is operated at high speeds.
3. See the output high-level voltage items in tables 27.2, 27.14, 27.27, and 27.39 on pages 34 to 35 for the output characteristics. Use values for the pull-up resistors such that the allowable output current conditions in tables 27.3, 27.15, 27.28, and 27.40 are met.

* This is not present in the H8S/2227 Group and the H8S/2237 Group products.

The H8S/2227 Group and the H8S/2237 Group products do not have an IIC bus, and the P34 and P35 pin outputs are CMOS outputs (when the P34ODR and P35ODR bits for the pins are 0).

When using an emulator that includes either an H8S/2633 evaluation chip or an H8S/2238 evaluation chip, these pins will be NMOS push-pull outputs. Therefore the pin output characteristics will differ from those in the H8S/2227 Group and the H8S/2237 Group products. If CMOS output characteristics are required in pins P34 and P35, pull up the emulator P34 and P35 pins with an appropriate resistor.

- P36

The pin functions are switched as shown below according to the P36DDR bit condition.

| P36DDR | 0 | 1 |
|---------------|---------------|-----------------|
| Pin functions | P36 input pin | P36 output pin* |

Note: * When P36ODR is set to 1, functions as NMOS open drain output.

- P35/SCK1/SCL0/ $\overline{\text{IRQ5}}$

The pin functions are switched as shown below according to the combination of the ICE bit^{*3} in ICCR_0 of IIC_0, the C/ $\overline{\text{A}}$ bit in SMR_1 of SCI_1, CKE0 and CKE1 bits in SCR_1, and the P35DDR bit. To use this port as SCL0 I/O pin, clear the C/ $\overline{\text{A}}$ bit, CKE1 bit, and CKE0 bit to 0.

The SCL0 functions as NMOS open drain output and the pin can drive bus directly. When this pin is specified as the P35 output pin or SCK1 output pin, it functions as NMOS push/pull output.^{*4}

| ICE ^{*3} | 0 | | | | | 1 |
|--------------------------|--|------------------------------|-------------------------------|-------------------------------|----------------|----------------------------|
| CKE1 | 0 | | | | 1 | 0 |
| C/ $\overline{\text{A}}$ | 0 | | | 1 | — | 0 |
| CKE0 | 0 | | 1 | — | — | 0 |
| P35DDR | 0 | 1 | — | — | — | — |
| Pin functions | P35 input pin | P35 output pin ^{*1} | SCK1 output pin ^{*1} | SCK1 output pin ^{*1} | SCK1 input pin | SCL0 I/O pin ^{*3} |
| | $\overline{\text{IRQ5}}$ Input pin ^{*2} | | | | | |

Notes: 1. When the P35ODR is set to 1, it functions as NMOS open drain output. When the P35ODR is cleared to 0, it functions as NMOS push/pull output.^{*4}

2. When this pin is used as an external interrupt pin, do not specify other functions.

3. Not available in the H8S/2237 Group and H8S/2227 Group.

4. It functions as CMOS output in the H8S/2237 Group and H8S/2227 Group.

- P34/RxD1/SDA0

The pin functions are switched as shown below according to the combination of the ICE bit^{*2} in ICCR_0 of IIC_0, the RE bit in SCR_1 of SCI_1, and the P34DDR bit. When this pin is specified as P34 output pin, it functions as NMOS push-pull output.^{*3} The SDA0 also functions as NMOS open drain outputs and can drive bus directly.

| ICE ^{*2} | 0 | | | 1 |
|-------------------|---------------|------------------------------|----------------|----------------------------|
| RE | 0 | | 1 | — |
| P34DDR | 0 | 1 | — | — |
| Pin functions | P34 input pin | P34 output pin ^{*1} | RxD1 input pin | SDA0 I/O pin ^{*2} |

Notes: 1. When P34ODR is set to 1, it functions as NMOS open drain output. When the P34ODR is cleared to 0, it functions as NMOS push/pull output.^{*3}

2. Not available in the H8S/2237 Group and H8S/2227 Group.

3. It functions as CMOS output in the H8S/2237 Group and H8S/2227 Group.

- P33/TxD1/SCL1

The pin functions are switched as shown below according to the combination of the ICE bit^{*2} in ICCR_1 of IIC_1, the TE bit in SCR_1 of SCI_1, and the P33DDR bit. SCL1 functions as NMOS open drain output and can drive bus directly.

| ICE ^{*2} | 0 | | | 1 |
|-------------------|---------------|------------------------------|-------------------------------|----------------------------|
| TE | 0 | | 1 | — |
| P33DDR | 0 | 1 | — | — |
| Pin functions | P33 input pin | P33 output pin ^{*1} | TxD1 output pin ^{*1} | SCL1 I/O pin ^{*2} |

Notes: 1. When P33ODR is set to 1, it functions as NMOS open drain output.

2. Not available in the H8S/2237 Group and H8S/2227 Group.

- P32/SCK0/SDA1/ $\overline{\text{IRQ4}}$

The pin functions are switched as shown below according to the combination of the ICE bit^{*3} in ICCR_1 of IIC_1, the C/ $\overline{\text{A}}$ bit in SMR_0 of SCI_0, CKE1 and CKE0 bits in SCR, and the P32DDR bit. To use this port as SDA1 input pin, clear the C/ $\overline{\text{A}}$ bit, CKE0 bit, and CKE1 bit to 0. The SDA1 functions as NMOS open drain output and can drive bus directly.

| ICE ^{*3} | 0 | | | | | 1 |
|--------------------------|--|------------------------------|-------------------------------|-------------------------------|----------------|----------------------------|
| CKE1 | 0 | | | 1 | | 0 |
| C/ $\overline{\text{A}}$ | 0 | | 1 | — | | 0 |
| CKE0 | 0 | 1 | — | — | | 0 |
| P32DDR | 0 | 1 | — | — | — | — |
| Pin functions | P32 input pin | P32 output pin ^{*1} | SCK0 output pin ^{*1} | SCK0 output pin ^{*1} | SCK0 input pin | SDA1 I/O pin ^{*3} |
| | $\overline{\text{IRQ4}}$ Input ^{*2} | | | | | |

Notes: 1. When P32ODR is set to 1, it functions as NMOS open drain output.

2. When this pin is used as an external interrupt pin, do not specify other functions.

3. Not available in the H8S/2237 Group and H8S/2227 Group.

- P31/RxD0

The pin functions are switched as shown below according to the combination of the RE bit in SCR_0 of SCI_0 and the P31DDR bit.

| RE | 0 | | 1 |
|---------------|---------------|-----------------------------|------------|
| P31DDR | 0 | 1 | — |
| Pin functions | P31 input pin | P31 output pin [*] | RxD0 input |

Note: * When P31ODR is set to 1, it functions as NMOS open drain output.

- P30/TxD0

The pin functions are switched as shown below according to the combination of the TE bit in SCR_0 of SCI_0 and the P30DDR bit.

| | | | |
|---------------|---------------|-----------------|--------------|
| TE | 0 | | 1 |
| P30DDR | 0 | 1 | — |
| Pin functions | P30 input pin | P30 output pin* | TxD0 output* |

Note: * When P30ODR is set to 1, it functions as NMOS open drain output.

10.3 Port 4

Port 4 is an 8-bit input port and has the following register.

- Port 4 register (PORT4)

10.3.1 Port 4 Register (PORT4)

PORT4 shows port 4 pin states.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | P47 | —* | R | The pin states are always read when a port 4 read is performed. |
| 6 | P46 | —* | R | |
| 5 | P45 | —* | R | |
| 4 | P44 | —* | R | |
| 3 | P43 | —* | R | |
| 2 | P42 | —* | R | |
| 1 | P41 | —* | R | |
| 0 | P40 | —* | R | |

Note: * Determined by the states of pins P47 to P40.

10.3.2 Pin Functions

Port 4 pins also function as A/D converter analog input pins (AN7 to AN0).

10.4 Port 7

Port 7 is an 8-bit I/O port and has the following registers.

- Port 7 data direction register (P7DDR)
- Port 7 data register (P7DR)
- Port 7 register (PORT7)

10.4.1 Port 7 Data Direction Register (P7DDR)

P7DDR specifies input or output of the port 7 pins using the individual bits. P7DDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | P77DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port 7 pin an output pin. Clearing this bit to 0 makes the pin an input pin. |
| 6 | P76DDR | 0 | W | |
| 5 | P75DDR | 0 | W | |
| 4 | P74DDR | 0 | W | |
| 3 | P73DDR | 0 | W | |
| 2 | P72DDR | 0 | W | |
| 1 | P71DDR | 0 | W | |
| 0 | P70DDR | 0 | W | |

10.4.2 Port 7 Data Register (P7DR)

P7DR stores output data for port 7 pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | P77DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | P76DR | 0 | R/W | |
| 5 | P75DR | 0 | R/W | |
| 4 | P74DR | 0 | R/W | |
| 3 | P73DR | 0 | R/W | |
| 2 | P72DR | 0 | R/W | |
| 1 | P71DR | 0 | R/W | |
| 0 | P70DR | 0 | R/W | |

10.4.3 Port 7 Register (PORT7)

PORT7 shows the pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | P77 | —* | R | If a port 1 read is performed while P7DDR bits are set to 1, the P7DR values are read. If a port 1 read is performed while P7DDR bits are cleared to 0, the pin states are read. |
| 6 | P76 | —* | R | |
| 5 | P75 | —* | R | |
| 4 | P74 | —* | R | |
| 3 | P73 | —* | R | |
| 2 | P72 | —* | R | |
| 1 | P71 | —* | R | |
| 0 | P70 | —* | R | |

Note: * Determined by the states of pins P77 to P70.

10.4.4 Pin Functions

Port 7 pins also function as TMR I/O pins (TMR_0, TMR_1, TMR_2*¹, and TMR_3*¹), bus control output pin, SCI I/O pins, and DMAC*² I/O pins. Values of the register and pin functions are shown below.

Notes: 1. Not available in the H8S/2237 Group and H8S/2227 Group.
2. Supported only by the H8S/2239 Group.

- P77/TxD3

The pin functions are switched as shown below according to the combination of the TE bit in SCR_3 of SCI_3 and the P77DDR bit.

| TE | 0 | | 1 |
|---------------|---------------|----------------|-------------|
| P77DDR | 0 | 1 | — |
| Pin functions | P77 input pin | P77 output pin | TxD3 output |

- P76/RxD3

The pin functions are switched as shown below according to the combination of the RE bit in SCR_3 of SCI_3 and the P76DDR bit.

| RE | 0 | | 1 |
|---------------|---------------|----------------|------------|
| P76DDR | 0 | 1 | — |
| Pin functions | P76 input pin | P76 output pin | RxD3 Input |

- P75/TMO3/SCK3

The pin functions are switched as shown below according to the combination of OS3 to OS0 bits in TCSR_3 of TMR_3*, CKE1 and CKE0 bits in SCR_3 of SCI_3, the C/A bit in SMR_3, and the P75DDR bit.

| OS3 to OS0* | All bits are 0 | | | | Any bit is 1 | |
|---------------|----------------|----------------|-----------------|-----------------|----------------|------------------|
| CKE1 | 0 | | | 1 | — | |
| C/A | 0 | | 1 | — | — | |
| CKE0 | 0 | | 1 | — | — | — |
| P75DDR | 0 | 1 | — | — | — | — |
| Pin functions | P75 input pin | P75 output pin | SCK3 output pin | SCK3 output pin | SCK3 input pin | TMO3* output pin |

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

- **P74/TMO2/ $\overline{\text{MRES}}$**

The pin functions are switched as shown below according to the combination of OS3 to OS0 bits in TCSR_2 of TMR_2*, the MRESE bit in SYSCR, and the P74DDR bit.

| MRESE | 0 | | | 1 |
|---------------|----------------|----------------|--------------|--------------------------------|
| OS3 to OS0* | All bits are 0 | | | Any bit is 1 |
| P74DDR | 0 | 1 | — | 0 |
| Pin functions | P74 input pin | P74 output pin | TMO2* output | $\overline{\text{MRES}}$ input |

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

- **P73/TMO1/ $\overline{\text{TEND1}}$ / $\overline{\text{CS7}}$**

The pin functions are switched as shown below according to the combination of operating mode, the TEE1 bit in DMATCR of DMAC*, OS3 to OS0 bits in TCSR_1 of TMR_1, and the P73DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | | |
|----------------|----------------|------------------------------------|-----------------|--|----------------|----------------|-----------------|--|
| TEE1* | 0 | | 1 | — | 0 | | 1 | — |
| OS3 to OS0 | All bits are 0 | | Any bit is 1 | — | All bits are 0 | | Any bit is 1 | — |
| P73DDR | 0 | 1 | — | — | 0 | 1 | — | — |
| Pin functions | P73 input pin | $\overline{\text{CS7}}$ output pin | TMO1 output pin | $\overline{\text{TEND1}}$ * output pin | P73 input pin | P73 output pin | TMO1 output pin | $\overline{\text{TEND1}}$ * output pin |

Note: * Supported only by the H8S/2239 Group.

- **P72/TMO0/ $\overline{\text{TEND0}}$ / $\overline{\text{CS6}}$**

The pin functions are switched as shown below according to the combination of operating mode the TEE0 bit in DMATCR of DMAC*, OS3 to OS0 bits in TCSR_0 of TMR_0, and the P72DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | | |
|----------------|----------------|------------------------------------|-----------------|--|----------------|----------------|-----------------|--|
| TEE0* | 0 | | 1 | — | 0 | | 1 | — |
| OS3 to OS0 | All bits are 0 | | Any bit is 1 | — | All bits are 0 | | Any bit is 1 | — |
| P72DDR | 0 | 1 | — | — | — | — | — | — |
| Pin functions | P72 input pin | $\overline{\text{CS6}}$ output pin | TMO0 output pin | $\overline{\text{TEND0}}$ * output pin | P72 input pin | P72 output pin | TMO0 output pin | $\overline{\text{TEND0}}$ * output pin |

Note: * Supported only by the H8S/2239 Group.

- P71/TMRI23/TMCI23/ $\overline{\text{DREQ1}}$ / $\overline{\text{CS5}}$

The pin functions are switched as shown below according to the combination of operating mode and the P71DDR bit.

| Operating mode | Modes 4 to 6 | | Mode 7 | |
|----------------|---|------------------------------------|---|----------------|
| P71DDR | 0 | 1 | 0 | 1 |
| Pin functions | P71 input pin | $\overline{\text{CS5}}$ output pin | P71 input pin | P71 output pin |
| | TMRI23 ^{*1} , TMCI23 ^{*1} , $\overline{\text{DREQ1}}$ ^{*2} input pin | — | TMRI23 ^{*1} , TMCI23 ^{*1} , $\overline{\text{DREQ1}}$ ^{*2} input pin | |

Notes: 1. Not available in the H8S/2237 Group and H8S/2227 Group.

2. Supported only by the H8S/2239 Group.

- P70/TMRI01/TMCI01/ $\overline{\text{DREQ0}}$ / $\overline{\text{CS4}}$

The pin functions are switched as shown below according to the combination of operating mode and the P70DDR bit.

| Operating mode | Modes 4 to 6 | | Mode 7 | |
|----------------|---|------------------------------------|--|----------------|
| P70DDR | 0 | 1 | 0 | 1 |
| Pin functions | P70 input pin | $\overline{\text{CS4}}$ output pin | P70 input pin | P70 output pin |
| | TMRI01, TMCI01, $\overline{\text{DREQ0}}$ [*] input pin | — | TMRI01, TMCI01, $\overline{\text{DREQ0}}$ [*] input pin | |

Note: * Supported only by the H8S/2239 Group.

10.5 Port 9

Port 9 is a 2-bit input-only port and has the following register.

- Port 9 register (PORT9)

10.5.1 Port 9 Register (PORT9)

PORT9 shows port 9 pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|--|----------|---------------|-----|--|
| 7 | P97 | —* | R | The pin states are always read when these bits are read. |
| 6 | P96 | —* | R | |
| 5 to 0 | — | — | R | Reserved |
| These bits are always read as undefined value. | | | | |

Note: * Determined by the states of pins P97 and P96.

10.5.2 Pin Functions

Port 9 pins also function as D/A converter analog output pins (DA1 and DA0)*.

Note: * Not available in the H8S/2227 Group.

10.6 Port A

Port A is a 4-bit I/O port and has the following register.

- Port A data direction register (PADDR)
- Port A data register (PADR)
- Port A register (PORTA)
- Port A pull-up MOS control register (PAPCR)
- Port A open drain control register (PAODR)

10.6.1 Port A Data Direction Register (PADDR)

PADDR specifies input or output the port A pins using the individual bits. PADDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 4 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 3 | PA3DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port A pin an output pin. Clearing this bit to 0 makes the pin an input pin. |
| 2 | PA2DDR | 0 | W | |
| 1 | PA1DDR | 0 | W | |
| 0 | PA0DDR | 0 | W | |

10.6.2 Port A Data Register (PADR)

PADR stores output data for port A pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 4 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 3 | PA3DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 2 | PA2DR | 0 | R/W | |
| 1 | PA1DR | 0 | R/W | |
| 0 | PA0DR | 0 | R/W | |

10.6.3 Port A Register (PORTA)

PORTA shows the pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 3 | PA3 | —* | R | If this bit is read while PADDR is set to 1, the PADDR value is read. If this bit is read while PADDR is cleared, the PA3 pin states are read. |
| 2 | PA2 | —* | R | |
| 1 | PA1 | —* | R | |
| 0 | PA0 | —* | R | |

Note: * Determined by the states of PA3 to PA0 pins.

10.6.4 Port A Pull-Up MOS Control Register (PAPCR)

PAPCR controls the on/off state of port A input pull-up MOS.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 4 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 3 | PA3PCR | 0 | R/W | When the pin is specified as an input port, setting the corresponding bit to 1 turns on the input pull-up MOS for that pin. |
| 2 | PA2PCR | 0 | R/W | |
| 1 | PA1PCR | 0 | R/W | |
| 0 | PA0PCR | 0 | R/W | |

10.6.5 Port A Open Drain Control Register (PAODR)

PAODR selects output state of port A.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 3 | PAODR | 0 | R/W | When this bit is set to 1, the corresponding port A pin functions as open drain output. When this bit is cleared to 0, the corresponding pin functions as CMOS output. |
| 2 | PAODR | 0 | R/W | |
| 1 | PAODR | 0 | R/W | |
| 0 | PAODR | 0 | R/W | |

10.6.6 Pin Functions

Port A pins also function as an address output pin and SCI_2* I/O pins. The relationship between the value of register and pin is shown as below.

Note: * Not available in the H8S/2227 Group.

- PA3/A19/SCK2

The pin functions are switched as shown below according to the combination of operating mode, AE3 to AE0 bits in PFCR, the C/A in SMR_2 of SCI_2*, CKE1 and CKE0 bits in SCR_2, and the PA3DDR bit.

| Operating mode | Modes 4 to 6 | | | | | |
|----------------|----------------|-------------------|------------------|---------------------|---------------------|------------------|
| AE3 to AE0 | B'11xx | Other than B'11xx | | | | |
| CKE1 | — | 0 | | | | 1 |
| C/A*2 | — | 0 | | | 1 | — |
| CKE0 | — | 0 | | 1 | — | — |
| PA3DDR | — | 0 | 1 | — | — | — |
| Pin functions | A19 output pin | PA3 input pin | PA3 output pin*1 | SCK2*2 output pin*1 | SCK2*2 output pin*1 | SCK2*2 input pin |

| Operating mode | Mode 7 | | | | |
|----------------|---------------|------------------|---------------------|---------------------|------------------|
| AE3 to AE0 | — | | | | |
| CKE1 | 0 | | | | 1 |
| C/A*2 | 0 | | | 1 | — |
| CKE0 | 0 | | 1 | — | — |
| PA3DDR | 0 | 1 | — | — | — |
| Pin functions | PA3 input pin | PA3 output pin*1 | SCK2*2 output pin*1 | SCK2*2 output pin*1 | SCK2*2 input pin |

Notes: 1. When PA3ODR in PAODR is set to 1, the corresponding pin functions as NMOS open drain output.

2. Not available in the H8S/2227 Group.

- PA2/A18/RxD2

The pin functions are switched as shown below according to the combination of operating mode, AE3 to AE0 bits in PFCR, the RE bit in SCR_2 of SCI_2^{*2}, and the PA2DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|------------------|------------------|-------------------------------|------------------------------|------------------------------|---------------|------------------------------|------------------------------|
| AE3 to AE0 | B'1011 or B'11xx | Other than (B'1011 or B'11xx) | | | — | | |
| RE ^{*2} | — | 0 | | 1 | 0 | | 1 |
| PA2DDR | — | 0 | 1 | — | 0 | 1 | — |
| Pin functions | A18 output pin | PA2 input pin | PA2 output pin ^{*1} | RxD2 ^{*2} input pin | PA2 input pin | PA2 output pin ^{*1} | RxD2 ^{*2} input pin |

- Notes: 1. When PA2ODR in PAODR is set to 1, the corresponding pin functions as NMOS open drain output.
 2. Not available in the H8S/2227 Group.

- PA1/A17/TxD2

The pin functions are switched as shown below according to the combination of operating mode, AE3 to AE0 bits in PFCR, the TE bit in SCR_2 of SCI_2^{*2}, and the PA1DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|------------------|------------------|-------------------------------|------------------------------|---|---------------|------------------------------|---|
| AE3 to AE0 | B'101x or B'11xx | Other than (B'101x or B'11xx) | | | — | | |
| TE ^{*2} | — | 0 | | 1 | 0 | | 1 |
| PA1DDR | — | 0 | 1 | — | 0 | 1 | — |
| Pin functions | A17 output pin | PA1 input pin | PA1 output pin ^{*1} | TxD2 ^{*2} output pin ^{*1} | PA1 input pin | PA1 output pin ^{*1} | TxD2 ^{*2} output pin ^{*1} |

- Notes: 1. When PA1ODR in PAODR is set to 1, the corresponding pin functions as NMOS open drain output.
 2. Not available in the H8S/2227 Group.

- PA0/A16

The pin functions are switched as shown below according to the combination of operating mode, AE3 to AE0 bits in PFCR and the PA0DDR bit.

| Operating mode | Modes 4 to 6 | | | Mode 7 | |
|----------------|-------------------------------|------------------|-----------------|---------------|-----------------|
| AE3 to AE0 | Other than (B'0xxx or B'1000) | B'0xxx or B'1000 | | — | |
| PA0DDR | — | 0 | 1 | 0 | 1 |
| Pin functions | A16 output pin | PA0 input pin | PA0 output pin* | PA0 input pin | PA0 output pin* |

Note: * When PA0DDR in PAODR is set to 1, the corresponding pin functions as NMOS open drain output.

10.6.7 Input Pull-Up MOS States in Port A

Port A has a built-in input pull-up MOS function that can be controlled by software. Input pull-up MOS can be specified as on or off on an individual bit basis.

Table 10.2 summarizes the input pull-up MOS states.

Table 10.2 Input Pull-Up MOS States in Port A

| Pin States | Power-on Reset | Hardware Standby Mode | Manual Reset | Software Standby Mode | In Other Operations |
|---|----------------|-----------------------|--------------|-----------------------|---------------------|
| Address output, Port output, SCI output | OFF | OFF | OFF | OFF | OFF |
| Port input, SCI input | | | ON/OFF | ON/OFF | ON/OFF |

Legend:

OFF: Input pull-up MOS is always off.

ON/OFF: On when PADDR = 0 and PAPCR = 1; otherwise off.

10.7 Port B

Port B is a 8-bit I/O port. Port B has the following registers.

- Port B data direction register (PBDDR)
- Port B data register (PBDR)
- Port B register (PORTB)

- Port B pull-up MOS control register (PBPCR)

10.7.1 Port B Data Direction Register (PBDDR)

PBDDR specifies input or output the port B pins using the individual bits. PBDDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PB7DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting the bit to 1 makes the corresponding port B pin an output pin. Clearing the bit to 0 makes the pin an input pin. |
| 6 | PB6DDR | 0 | W | |
| 5 | PB5DDR | 0 | W | |
| 4 | PB4DDR | 0 | W | |
| 3 | PB3DDR | 0 | W | |
| 2 | PB2DDR | 0 | W | |
| 1 | PB1DDR | 0 | W | |
| 0 | PB0DDR | 0 | W | |

10.7.2 Port B Data Register (PBDR)

PBDR stores output data for port B pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PB7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PB6DR | 0 | R/W | |
| 5 | PB5DR | 0 | R/W | |
| 4 | PB4DR | 0 | R/W | |
| 3 | PB3DR | 0 | R/W | |
| 2 | PB2DR | 0 | R/W | |
| 1 | PB1DR | 0 | R/W | |
| 0 | PB0DR | 0 | R/W | |

10.7.3 Port B Register (PORTB)

PORTB shows the pin states and cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PB7 | —* | R | If these bits are read while the corresponding PBDDR bits are set to 1, the PBDR value is read. If these bits are read while PBDDR bits are cleared to 0, the pin states are read. |
| 6 | PB6 | —* | R | |
| 5 | PB5 | —* | R | |
| 4 | PB4 | —* | R | |
| 3 | PB3 | —* | R | |
| 2 | PB2 | —* | R | |
| 1 | PB1 | —* | R | |
| 0 | PB0 | —* | R | |

Note: * Determined by the states of pins PB7 to PB0.

10.7.4 Port B Pull-Up MOS Control Register (PBPCR)

PBPCR controls the on/off state of port B input pull-up MOS.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PB7PCR | 0 | R/W | When a pin is specified as an input port, setting the corresponding bit to 1 turns on the input pull-up MOS for that pin. |
| 6 | PB6PCR | 0 | R/W | |
| 5 | PB5PCR | 0 | R/W | |
| 4 | PB4PCR | 0 | R/W | |
| 3 | PB3PCR | 0 | R/W | |
| 2 | PB2PCR | 0 | R/W | |
| 1 | PB1PCR | 0 | R/W | |
| 0 | PB0PCR | 0 | R/W | |

10.7.5 Pin Functions

Port B pins also function as TPU I/O pins (TPU_3*, TPU_4*, and TPU_5*) and address output pins. The values of register and pin functions are shown below.

Note: * Not available in the H8S/2227 Group.

- PB7/A15/TIOCB5

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 5^{*3} setting, AE3 to AE0 bits in PFCR, and the PB7DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|----------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | B'1xxx | Other than B'1xxx | | | — | | |
| TPU channel 5 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB7DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A15 output pin | TIOCB5 ^{*3} output pin | PB7 input pin | PB7 output pin | TIOCB5 ^{*3} output pin | PB7 input pin | PB7 output pin |
| | | | TIOCB5 ^{*3} input pin ^{*2} | | | TIOCB5 ^{*3} input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCB5 input when TPU channel 5 timer operating mode is set to normal operating or phase counting mode and IOB3 in TIOR_5 is set to 1.
 3. Not available in the H8S/2227 Group.

- PB6/A14/TIOCA5

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 5^{*3} setting, AE3 to AE0 bits in PFCR, and the PB6DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | B'0111 or B'1xxx | Other than (B'0111 or B'1xxx) | | | — | | |
| TPU channel 5 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB6DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A14 output pin | TIOCA5 ^{*3} output pin | PB6 input pin | PB6 output pin | TIOCA5 ^{*3} output pin | PB6 input pin | PB6 output pin |
| | | | TIOCA5 ^{*3} input pin ^{*2} | | | TIOCA5 ^{*3} input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCA5 input when TPU channel 5 timer operating mode is set to normal operating or phase counting mode and IOA3 in TIOR_5 is set to 1.
 3. Not available in the H8S/2227 Group.

- PB5/A13/TIOCB4

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 4^{*3} setting, AE3 to AE0 bits in PFCR, and the PB5DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | B'011x or B'1xxx | Other than (B'011x or B'1xxx) | | | — | | |
| TPU channel 4 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB5DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A13 output pin | TIOCB4 ^{*3} output pin | PB5 input pin | PB5 output pin | TIOCB4 ^{*3} output pin | PB5 input pin | PB5 output pin |
| | | | TIOCB4 ^{*3} input pin ^{*2} | | | TIOCB4 ^{*3} input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCB4 input when TPU channel 4 timer operating mode is set to normal operating or phase counting mode and IOB3 to IOB0 in TIOR_4 are set to 10xx.
 3. Not available in the H8S/2227 Group.

- PB4/A12/TIOCA4

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 4^{*3} setting, AE3 to AE0 bits in PFCR, and the PB4DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|-------------------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | Other than (B'0100 or B'00xx) | B'0100 or B'00xx | | | — | | |
| TPU channel 4 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB4DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A12 output pin | TIOCA4 ^{*3} output pin | PB4 input pin | PB4 output pin | TIOCA4 ^{*3} output pin | PB4 input pin | PB4 output pin |
| | | | TIOCA4 ^{*3} input pin ^{*2} | | | TIOCA4 ^{*3} input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCA4 input when TPU channel 4 timer operating mode is set to normal operating or phase counting mode and IOA3 to IOA0 in TIOR_4 are set to 10xx.
 3. Not available in the H8S/2227 Group.

• PB3/A11/TIOCD3

The pin function is switched as shown below according to combination of the operating mode, the TPU channel 3^{*1*3} setting, AE3 to AE0 bits in PFCR, and the PB3DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|-------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | Other than B'00xx | B'00xx | | | — | | |
| TPU channel 3 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB3DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A11 output pin | TIOCD3 ^{*3} output pin | PB3 input pin | PB3 output pin | TIOCD3 ^{*3} output pin | PB3 input pin | PB3 output pin |
| | | | TIOCD3 ^{*3} input pin ^{*2} | | | TIOCD3 ^{*3} input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCD3 input when TPU channel 3 timer operating mode is set to normal operating and IOD3 to IOD0 in TIORL_3 are set to 10xx.
 3. Not available in the H8S/2227 Group.

• PB2/A10/TIOCC3

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 3^{*3} setting, AE3 to AE0 bits in PFCR, and the PB2DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|-------------------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | Other than (B'0010 or B'000x) | B'0010 or B'000x | | | — | | |
| TPU channel 3 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB2DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A10 output pin | TIOCC3 ^{*3} output pin | PB2 input pin | PB2 output pin | TIOCC3 ^{*3} output pin | PB2 input pin | PB2 output pin |
| | | | TIOCC3 ^{*3} input pin ^{*2} | | | TIOCC3 ^{*3} input pin ^{*2} | |

- Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).
 2. This pin functions as TIOCC3 input when TPU channel 3 timer operating mode is set to normal operating mode and IOC3 to IOC0 in TIORL_3 are set to 10xx.
 3. Not available in the H8S/2227 Group.

- PB1/A9/TIOCB3

The pin functions are switched as shown below according to the combination of operating mode, the TPU channel 3^{*3} setting, AE3 to AE0 bits in PFCR, and the PB1DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|-------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | Other than B'000x | B'000x | | | — | | |
| TPU channel 3 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB1DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A9 output pin | TIOCB3 ^{*3} output pin | PB1 input pin | PB1 output pin | TIOCB3 ^{*3} output pin | PB1 input pin | PB1 output pin |
| | | | TIOCB3 ^{*3} input pin ^{*2} | | | TIOCB3 ^{*3} input pin ^{*2} | |

Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).

2. This pin functions as TIOCB3 input when TPU channel 3 timer operating mode is set to normal operating mode and IOB3 to IOB0 in TIORH_3 are set to 10xx.

3. Not available in the H8S/2227 Group.

- PB0/A8/TIOCA3

The pin functions are switched as shown below according to the combination of the operating mode, TPU channel 3^{*3} setting, the AE3 to AE0 bits in PFCR, and the PB0DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|---------------------------------------|-------------------|---------------------------------|--|----------------|---------------------------------|--|----------------|
| AE3 to AE0 | Other than B'0000 | B'0000 | | | — | | |
| TPU channel 3 setting ^{*1*3} | — | Output | Input or initial value | | Output | Input or initial value | |
| PB0DDR | — | — | 0 | 1 | — | 0 | 1 |
| Pin functions | A8 output pin | TIOCA3 ^{*3} output pin | PB0 input pin | PB0 output pin | TIOCA3 ^{*3} output pin | PB0 input pin | PB0 output pin |
| | | | TIOCA3 ^{*3} input pin ^{*2} | | | TIOCA3 ^{*3} input pin ^{*2} | |

Notes: 1. For the setting of the TPU channel, see section 11, 16-Bit Timer Pulse Unit (TPU).

2. This pin functions as TIOCA3 input when TPU channel 3 timer operating mode is set to normal operating mode and IOA3 to IOA0 in TIORH_3 are set to 10xx.

3. Not available in the H8S/2227 Group.

10.7.6 Input Pull-Up MOS States in Port B

Port B has a built-in input pull-up MOS function that can be controlled by software. Input pull-up MOS can be specified as on or off on an individual bit basis.

Table 10.3 summarizes the input pull-up MOS states.

Table 10.3 Input Pull-Up MOS States in Port B

| Pin States | Power-on Reset | Hardware Standby Mode | Manual Reset | Software Standby Mode | In Other Operations |
|---|-------------------|-----------------------------|-----------------|-----------------------------|------------------------|
| Address output, Port output, TPU output | OFF | OFF | OFF | OFF | OFF |
| Port input, TPU input | | | ON/OFF | ON/OFF | ON/OFF |

Legend:

OFF: Input pull-up MOS is always off.

ON/OFF: On when PBDDR = 0 and PBPCR = 1; otherwise off.

10.8 Port C

Port C is an 8-bit I/O port and has the following registers.

- Port C data direction register (PCDDR)
- Port C data register (PCDR)
- Port C register (PORTC)
- Port C pull-up MOS control register (PCPCR)

10.8.1 Port C Data Direction Register (PCDDR)

PCDDR specifies input or output the port C pins using the individual bits. PCDDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PC7DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port C pin an output pin. Clearing this bit to 0 makes the pin an input pin. |
| 6 | PC6DDR | 0 | W | |
| 5 | PC5DDR | 0 | W | |
| 4 | PC4DDR | 0 | W | |
| 3 | PC3DDR | 0 | W | |
| 2 | PC2DDR | 0 | W | |
| 1 | PC1DDR | 0 | W | |
| 0 | PC0DDR | 0 | W | |

10.8.2 Port C Data Register (PCDR)

PCDR stores output data for port C pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PC7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PC6DR | 0 | R/W | |
| 5 | PC5DR | 0 | R/W | |
| 4 | PC4DR | 0 | R/W | |
| 3 | PC3DR | 0 | R/W | |
| 2 | PC2DR | 0 | R/W | |
| 1 | PC1DR | 0 | R/W | |
| 0 | PC0DR | 0 | R/W | |

10.8.3 Port C Register (PORTC)

PORTC shows port C pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PC7 | —* | R | If a port C read is performed while PCDDR bits are set to 1, the PCDR values are read. If a port C read is performed while PCDDR bits are cleared to 0, the pin states are read. |
| 6 | PC6 | —* | R | |
| 5 | PC5 | —* | R | |
| 4 | PC4 | —* | R | |
| 3 | PC3 | —* | R | |
| 2 | PC2 | —* | R | |
| 1 | PC1 | —* | R | |
| 0 | PC0 | —* | R | |

Note: * Determined by the states of pins PC7 to PC0.

10.8.4 Port C Pull-Up MOS Control Register (PCPCR)

PCPCR controls the input pull-up MOS specification as on or off for port C.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PC7PCR | 0 | R/W | When a pin is specified as an input port, setting the corresponding bit to 1 turns on the input pull-up MOS for that pin. |
| 6 | PC6PCR | 0 | R/W | |
| 5 | PC5PCR | 0 | R/W | |
| 4 | PC4PCR | 0 | R/W | |
| 3 | PC3PCR | 0 | R/W | |
| 2 | PC2PCR | 0 | R/W | |
| 1 | PC1PCR | 0 | R/W | |
| 0 | PC0PCR | 0 | R/W | |

10.8.5 Pin Functions

Port C pins also function as address output pin. The values of register and pin functions are shown below.

- PC7/A7, PC6/A6, PC5/A5, PC4/A4, PC3/A3, PC2/A2, PC1/A1, PC0/A0

The pin functions are switched as shown below according to the combination of operating mode and the PCnDDR bit.

| Operating mode | Modes 4 and 5 | Mode 6 | | Mode 7 | |
|----------------|--------------------|---------------|--------------------|---------------|----------------|
| PCnDDR | — | 0 | 1 | 0 | 1 |
| Pin functions | Address output pin | PCn input pin | Address output pin | PCn input pin | PCn output pin |

Note: n = 7 to 0

10.8.6 Input Pull-Up MOS States in Port C

Port C has a built-in input pull-up MOS function that can be controlled by software. Input pull-up MOS can be used in modes 6 and 7 and specified as on or off on an individual bit basis.

Table 10.4 summarizes the input pull-up MOS states in port C.

Table 10.4 Input Pull-Up MOS States in Port C

| Pin States | Power-on Reset | Hardware Standby Mode | Software Standby Mode | In Other Operations |
|--|----------------|-----------------------|-----------------------|---------------------|
| Address output (modes 4 and 5) and port output (modes 6 and 7) | OFF | OFF | OFF | OFF |
| Port input (modes 6 and 7) | | | ON/OFF | ON/OFF |

Legend:

OFF: Input pull-up MOS is always off.

ON/OFF: On when PCDDR = 0 and PCPCR = 1; otherwise off.

10.9 Port D

Port D is an 8-bit I/O port and has the following registers.

- Port D data direction register (PDDDR)
- Port D data register (PDDR)
- Port D register (PORTD)
- Port D pull-up MOS control register (PDPCR)

10.9.1 Port D Data Direction Register (PDDDR)

PDDDR specifies input or output the port D pins using the individual bits. PDDDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PD7DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port D pin an output port. Clearing this bit to 0 makes the pin an input port. |
| 6 | PD6DDR | 0 | W | |
| 5 | PD5DDR | 0 | W | |
| 4 | PD4DDR | 0 | W | |
| 3 | PD3DDR | 0 | W | |
| 2 | PD2DDR | 0 | W | |
| 1 | PD1DDR | 0 | W | |
| 0 | PD0DDR | 0 | W | |

10.9.2 Port D Data Register (PDDR)

PDDR stores output data for port D pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PD7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PD6DR | 0 | R/W | |
| 5 | PD5DR | 0 | R/W | |
| 4 | PD4DR | 0 | R/W | |
| 3 | PD3DR | 0 | R/W | |
| 2 | PD2DR | 0 | R/W | |
| 1 | PD1DR | 0 | R/W | |
| 0 | PD0DR | 0 | R/W | |

10.9.3 Port D Register (PORTD)

PORTD shows port D pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PD7 | —* | R | If a port D read is performed while PDDDR bits are set to 1, the PDDR values are read. If a port D read is performed while PDDDR bits are cleared to 0, the pin states are read. |
| 6 | PD6 | —* | R | |
| 5 | PD5 | —* | R | |
| 4 | PD4 | —* | R | |
| 3 | PD3 | —* | R | |
| 2 | PD2 | —* | R | |
| 1 | PD1 | —* | R | |
| 0 | PD0 | —* | R | |

Note: * Determined by the states of pins PD7 to PD0.

10.9.4 Port D Pull-Up MOS Control Register (PDPCR)

PDPCR controls the on/off state of port D input pull-up MOS.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PD7PCR | 0 | R/W | When a pin is specified as an input port, setting the corresponding bit to 1 turns on the input pull-up MOS for that pin. |
| 6 | PD6PCR | 0 | R/W | |
| 5 | PD5PCR | 0 | R/W | |
| 4 | PD4PCR | 0 | R/W | |
| 3 | PD3PCR | 0 | R/W | |
| 2 | PD2PCR | 0 | R/W | |
| 1 | PD1PCR | 0 | R/W | |
| 0 | PD0PCR | 0 | R/W | |

10.9.5 Pin Functions

Port D pins also function as data I/O pins. The values of register and pin functions are shown below.

- PD7/D15, PD6/D14, PD5/D13, PD4/D12, PD3/D11, PD2/D10, PD1/D9, PD0/D8

The pin functions are switched as shown below according to the combination of the operating mode and the PDnDDR bit.

| Operating mode | Modes 4 to 6 | Mode 7 | |
|----------------|--------------|---------------|----------------|
| PDnDDR | — | 0 | 1 |
| Pin functions | Data I/O pin | PDn input pin | PDn output pin |

Note: n = 7 to 0

10.9.6 Input Pull-Up MOS States in Port D

Port D has a built-in input pull-up MOS function that can be controlled by software. Input pull-up MOS can be used in mode 7 and specified as on or off on an individual bit basis.

Table 10.5 summarizes the input pull-up MOS states in port D.

Table 10.5 Input Pull-Up MOS States in Port D

| Pin States | Power-on Reset | Hardware Standby Mode | Manual Reset | Software Standby Mode | In Other Operations |
|---|-------------------|-----------------------------|-----------------|-----------------------------|------------------------|
| Data I/O (modes 4 to 6) and OFF port output (mode 7) | | OFF | OFF | OFF | OFF |
| Port input (mode 7) | | | ON/OFF | ON/OFF | ON/OFF |

Legend:

OFF: Input pull-up MOS is always off.

ON/OFF: On when PDDDR = 0 and PDPCR = 1; otherwise off.

10.10 Port E

Port E is an 8-bit I/O port and has the following registers.

- Port E data direction register (PEDDDR)
- Port E data register (PEDR)
- Port E register (PORTE)
- Port E pull-up MOS control register (PEPCR)

10.10.1 Port E Data Direction Register (PEDDR)

PEDDR specifies input or output of the port E pins using the individual bits. PEDDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PE7DDR | 0 | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port E pin an output port. Clearing this bit to 0 makes the pin an input port. |
| 6 | PE6DDR | 0 | W | |
| 5 | PE5DDR | 0 | W | |
| 4 | PE4DDR | 0 | W | |
| 3 | PE3DDR | 0 | W | |
| 2 | PE2DDR | 0 | W | |
| 1 | PE1DDR | 0 | W | |
| 0 | PE0DDR | 0 | W | |

10.10.2 Port E Data Register (PEDR)

PEDR stores output data for port E pins.

PEDR stores output data for port E pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PE7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PE6DR | 0 | R/W | |
| 5 | PE5DR | 0 | R/W | |
| 4 | PE4DR | 0 | R/W | |
| 3 | PE3DR | 0 | R/W | |
| 2 | PE2DR | 0 | R/W | |
| 1 | PE1DR | 0 | R/W | |
| 0 | PE0DR | 0 | R/W | |

10.10.3 Port E Register (PORTE)

PORTE shows port E pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PE7 | —* | R | If a port E read is performed while PEDDR bits are set to 1, the PEDR values are read. If a port E read is performed while PEDDR bits are cleared to 0, the pin states are read. |
| 6 | PE6 | —* | R | |
| 5 | PE5 | —* | R | |
| 4 | PE4 | —* | R | |
| 3 | PE3 | —* | R | |
| 2 | PE2 | —* | R | |
| 1 | PE1 | —* | R | |
| 0 | PE0 | —* | R | |

Note: * Determined by the states of pins PE7 to PE0.

10.10.4 Port E Pull-Up MOS Control Register (PEPCR)

PEPCR controls the on/off state of port E input pull-up MOS.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PE7PCR | 0 | R/W | When a pin is specified as an input port, setting the corresponding bit to 1 turns on the input pull-up MOS for that pin. |
| 6 | PE6PCR | 0 | R/W | |
| 5 | PE5PCR | 0 | R/W | |
| 4 | PE4PCR | 0 | R/W | |
| 3 | PE3PCR | 0 | R/W | |
| 2 | PE2PCR | 0 | R/W | |
| 1 | PE1PCR | 0 | R/W | |
| 0 | PE0PCR | 0 | R/W | |

10.10.5 Pin Functions

Port E pins also function as data I/O pins. The values of register and pin functions are shown below.

- PE7/D7, PE6/D6, PE5/D5, PE4/D4, PE3/D3, PE2/D2, PE1/D1, PE0/D0

The pin functions are switched as shown below according to the combination of the operating mode, bus mode, and the PEnDDR bit.

| Operating mode | Modes 4 to 6 | | | Mode 7 | |
|----------------|----------------|----------------|-----------------|---------------|----------------|
| Bus mode | 8-bit bus mode | | 16-bit bus mode | — | |
| PEnDDR | 0 | 1 | — | 0 | 1 |
| Pin functions | PEn input pin | PEn output pin | Data I/O pin | PEn input pin | PEn output pin |

Note: n = 7 to 0

10.10.6 Input Pull-Up MOS States in Port E

Port E has a built-in input pull-up MOS function that can be controlled by software. Input pull-up MOS can be used in modes 4 to 6 and 8-bit bus mode or in mode 7 and specified as on or off on an individual bit basis.

Table 10.6 summarizes the input pull-up MOS states in port E.

Table 10.6 Input Pull-Up MOS States in Port E

| Pin States | Power-on Reset | Hardware Standby Mode | Manual Reset | Software Standby Mode | In Other Operations |
|---|----------------|-----------------------|--------------|-----------------------|---------------------|
| Data I/O (16-bit bus in modes 4 to 6) and port output (8-bit bus in modes 4 to 6, and mode 7) | OFF | OFF | OFF | OFF | OFF |
| Port input (8-bit bus in modes 4 to 6, and mode 7) | | | ON/OFF | ON/OFF | ON/OFF |

Legend:

OFF: Input pull-up MOS is always off.

ON/OFF: On when PEnDDR = 0 and PEnPCR = 1; otherwise off.

10.11 Port F

Port F is an 8-bit I/O port and has the following registers.

- Port F data direction register (PFDDR)
- Port F data register (PFDR)
- Port F register (PORTF)

10.11.1 Port F Data Direction Register (PFDDR)

PFDDR specifies input or output of the port F pins using the individual bits. PFDDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PF7DDR | 0/1* | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port F pin an output port. Clearing this bit to 0 makes the pin an input port. |
| 6 | PF6DDR | 0 | W | |
| 5 | PF5DDR | 0 | W | |
| 4 | PF4DDR | 0 | W | |
| 3 | PF3DDR | 0 | W | |
| 2 | PF2DDR | 0 | W | |
| 1 | PF1DDR | 0 | W | |
| 0 | PF0DDR | 0 | W | |

Note: * In modes 4 to 6, initial value is 1. In mode 7, initial value is 0.

10.11.2 Port F Data Register (PFDR)

PFDR stores output data for port F pins.

PFDR stores output data for port F pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | PF7DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 6 | PF6DR | 0 | R/W | |
| 5 | PF5DR | 0 | R/W | |
| 4 | PF4DR | 0 | R/W | |
| 3 | PF3DR | 0 | R/W | |
| 2 | PF2DR | 0 | R/W | |
| 1 | PF1DR | 0 | R/W | |
| 0 | PF0DR | 0 | R/W | |

10.11.3 Port F Register (PORTF)

PORTF shows port F pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | PF7 | —* | R | If a port F read is performed while PFDDR bits are set to 1, the PFDR values are read. If a port F read is performed while PFDDR bits are cleared to 0, the pin states are read. |
| 6 | PF6 | —* | R | |
| 5 | PF5 | —* | R | |
| 4 | PF4 | —* | R | |
| 3 | PF3 | —* | R | |
| 2 | PF2 | —* | R | |
| 1 | PF1 | —* | R | |
| 0 | PF0 | —* | R | |

Note: * Determined by the states of pins PF7 to PF0.

10.11.4 Pin Functions

Port F pins also function as bus control signal input/output pin, interrupt input pin, system clock output pin, A/D trigger input pin, and BUZZ output pin. The values of register and pin functions are shown below.

- PF7/ ϕ

The pin functions are switched as shown below according to the PF7DDR bit.

| | | |
|---------------|---------------|-------------------|
| PF7DDR | 0 | 1 |
| Pin functions | PF7 input pin | ϕ output pin |

- PF6/ \overline{AS}

The pin functions are switched as shown below according to the combination of operating mode and the PF6DDR bit.

| | | | |
|----------------|----------------------------|---------------|----------------|
| Operating mode | Modes 4 to 6 | Mode 7 | |
| PF6DDR | — | 0 | 1 |
| Pin functions | \overline{AS} output pin | PF6 input pin | PF6 output pin |

- PF5/ \overline{RD}

The pin functions are switched as shown below according to the combination of operating mode and the PF5DDR bit.

| | | | |
|----------------|----------------------------|---------------|----------------|
| Operating mode | Modes 4 to 6 | Mode 7 | |
| PF5DDR | — | 0 | 1 |
| Pin functions | \overline{RD} output pin | PF5 input pin | PF5 output pin |

- PF4/ \overline{HWR}

The pin functions are switched as shown below according to the combination of operating mode and the PF4DDR bit.

| | | | |
|----------------|-----------------------------|---------------|----------------|
| Operating mode | Modes 4 to 6 | Mode 7 | |
| PF4DDR | — | 0 | 1 |
| Pin functions | \overline{HWR} output pin | PF4 input pin | PF4 output pin |

- PF3/LWR/ADTRG/IRQ3

The pin functions are switched as shown below according to the combination of operating mode and the PF3DDR bit.

| Operating mode | Modes 4 to 6 | | | Mode 7 | |
|----------------|-----------------|-------------------------------|----------------|---------------|----------------|
| Bus mode | 16-bit bus mode | 8-bit bus mode | | — | |
| PF3DDR | — | 0 | 1 | 0 | 1 |
| Pin functions | LWR output pin | PF3 input pin | PF3 output pin | PF3 input pin | PF3 output pin |
| | | ADTRG input pin ^{*1} | | | |
| | | IRQ3 input pin ^{*2} | | | |

Notes: 1. When TRGS0 and TRGS1 are set to 1, this pin is ADTRG input.

2. When this pin is used as an external interrupt pin, do not specify other functions.

- PF2/WAIT

The pin functions are switched as shown below according to the combination of operating mode, the WAITE bit, and the PF2DDR bit.

| Operating mode | Modes 4 to 6 | | | Mode 7 | |
|----------------|---------------|----------------|----------------|---------------|----------------|
| WAITE | 0 | | 1 | — | |
| PF2DDR | 0 | 1 | — | 0 | 1 |
| Pin functions | PF2 input pin | PF2 output pin | WAIT input pin | PF2 input pin | PF2 output pin |

- PF1/BACK/BUZZ

The pin functions are switched as shown below according to the combination of operating mode, the BUZZ bit in PFCR, and the PF1DDR bit.

| Operating mode | Modes 4 to 6 | | | | Mode 7 | | |
|----------------|---------------|----------------|-----------------|-----------------|---------------|----------------|-----------------|
| BRLE | 0 | | | 1 | — | | |
| BUZZE | 0 | | 1 | — | 0 | | 1 |
| PF1DDR | 0 | 1 | — | — | 0 | 1 | — |
| Pin functions | PF1 input pin | PF1 output pin | BUZZ output pin | BACK output pin | PF1 input pin | PF1 output pin | BUZZ output pin |

- PF0/BREQ/ $\overline{\text{IRQ2}}$

The pin functions are switched as shown below according to the combination of operating mode, the BRLE bit, and the PF0DDR bit.

| Operating mode | Modes 4 to 6 | | | Mode 7 | |
|----------------|-----------------|----------------|------------------------------------|---------------|----------------|
| BRLE | 0 | | 1 | — | |
| PF0DDR | 0 | 1 | — | 0 | 1 |
| Pin functions | PF0 input pin | PF0 output pin | $\overline{\text{BREQ}}$ input pin | PF0 input pin | PF0 output pin |
| | IRQ2 input pin* | | | | |

Note: * When this pin is used as an external interrupt pin, do not specify other functions.

10.12 Port G

Port G is a 5-bit I/O port and has the following registers.

- Port G data direction register (PGDDR)
- Port G data register (PGDR)
- Port G register (PORTG)

10.12.1 Port G Data Direction Register (PGDDR)

PGDDR specifies input or output of the port G pins using the individual bits. PGDDR cannot be read; if it is, an undefined value will be read. This register is a write-only register, and cannot be written by bit manipulation instruction. For details, see section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 5 | — | Undefined | — | Reserved |
| | | | | These bits are always read as undefined value. |
| 4 | PG4DDR | 0/1* | W | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the corresponding port G pin an output port. Clearing this bit to 0 makes the pin an input port. |
| 3 | PG3DDR | 0 | W | |
| 2 | PG2DDR | 0 | W | |
| 1 | PG1DDR | 0 | W | |
| 0 | PG0DDR | 0 | W | |

Note: * In modes 4 and 5, initial value is 1. In modes 6 and 7, initial value is 0.

10.12.2 Port G Data Register (PGDR)

PGDR stores output data for port G pins.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 5 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 4 | PG4DR | 0 | R/W | Output data for a pin is stored when the pin is specified as a general purpose output port. |
| 3 | PG3DR | 0 | R/W | |
| 2 | PG2DR | 0 | R/W | |
| 1 | PG1DR | 0 | R/W | |
| 0 | PG0DR | 0 | R/W | |

10.12.3 Port G Register (PORTG)

PORTG shows port G pin states. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 5 | — | Undefined | — | Reserved These bits are always read as undefined value. |
| 4 | PG4 | —* | R | If a port G read is performed while PGDDR bits are set to 1, the PGDR values are read. If a port G read is performed while PGDDR bits are cleared to 0, the pin states are read. |
| 3 | PG3 | —* | R | |
| 2 | PG2 | —* | R | |
| 1 | PG1 | —* | R | |
| 0 | PG0 | —* | R | |

Note: * Determined by the states of pins PG4 to PG0.

10.12.4 Pin Functions

Port G pins also function as IEB* input/output pin, bus control signal input/output pin, and interrupt input pin. The values of registers and pin functions are shown below.

Note: * Supported only by the H8S/2258 Group.

- PG4/ $\overline{\text{CS0}}$

The pin functions are switched as shown below according to the combination of operating mode and the PG4DDR bit.

| Operating mode | Modes 4 to 6 | | Mode 7 | |
|----------------|---------------|------------------------------------|---------------|----------------|
| PG4DDR | 0 | 1 | 0 | 1 |
| Pin functions | PG4 input pin | $\overline{\text{CS0}}$ output pin | PG4 input pin | PG4 output pin |

- PG3/ $\overline{\text{Rx}}/\overline{\text{CS1}}$

The pin functions are switched as shown below according to the combination of the IEE bit in IECTR of IEB*, operating mode, and the PG3DDR bit.

| IEE* | 0 | | | | 1 |
|----------------|---------------|------------------------------------|---------------|----------------|-----------------------------------|
| Operating mode | Modes 4 to 6 | | Mode 7 | | — |
| PG3DDR | 0 | 1 | 0 | 1 | — |
| Pin functions | PG3 input pin | $\overline{\text{CS1}}$ output pin | PG3 input pin | PG3 output pin | $\overline{\text{Rx}}$ input pin* |

Note: * Supported only by the H8S/2258 Group.

- PG2/ $\overline{\text{Tx}}/\overline{\text{CS2}}$

The pin functions are switched as shown below according to the combination of the IEE bit in IECTR of IEB*, operating mode, and the PG2DDR bit.

| IEE* | 0 | | | | 1 |
|----------------|---------------|------------------------------------|---------------|----------------|-----------------------------------|
| Operating mode | Modes 4 to 6 | | Mode 7 | | — |
| PG2DDR | 0 | 1 | 0 | 1 | — |
| Pin functions | PG2 input pin | $\overline{\text{CS2}}$ output pin | PG2 input pin | PG2 output pin | $\overline{\text{Tx}}$ input pin* |

Note: * Supported only by the H8S/2258 Group.

- PG1/ $\overline{\text{CS3}}$ / $\overline{\text{IRQ7}}$

The pin functions are switched as shown below according to the combination of operating mode and the PG1DDR bit.

| Operating mode | Modes 4 to 6 | | Mode 7 | |
|----------------|-------------------------------------|------------------------------------|---------------|----------------|
| PG1DDR | 0 | 1 | 0 | 1 |
| Pin functions | PG1 input pin | $\overline{\text{CS3}}$ output pin | PG1 input pin | PG1 output pin |
| | $\overline{\text{IRQ7}}$ input pin* | | | |

Note: * When this pin is used as an external interrupt pin, do not specify other functions.

- PG0/ $\overline{\text{IRQ6}}$

The pin functions are switched as shown below according to the PG0DDR bit.

| PG0DDR | 0 | 1 |
|---------------|-------------------------------------|----------------|
| Pin functions | PG0 input pin | PG0 output pin |
| | $\overline{\text{IRQ6}}$ input pin* | |

Note: * When this pin is use as an external interrupt pin, do not specify other functions.

10.13 Handling of Unused Pins

Unused input pins should be fixed high or low. Generally, the input pins of CMOS products are high-impedance. Leaving unused pins open can cause the generation of intermediate levels due to peripheral noise induction. This can result in shoot-through current inside the device and cause it to malfunction. Table 10.7 lists examples of ways to handle unused pins. Pins marked NC should be left open.

Table 10.7 Examples of Ways to Handle Unused Input Pins

| Port Name | Pin Handling Example |
|-----------|---|
| Port 1 | Connect each pin to Vcc (pull-up) or to Vss (pull-down) via a resistor. |
| Port 3 | |
| Port 4 | Connect each pin to AVcc (pull-up) or to AVss (pull-down) via a resistor. |
| Port 7 | Connect each pin to Vcc (pull-up) or to Vss (pull-down) via a resistor. |
| Port 9 | Connect each pin to AVcc (pull-up) or to AVss (pull-down) via a resistor. |
| Port A | Connect each pin to Vcc (pull-up) or to Vss (pull-down) via a resistor. |
| Port B | |
| Port C | |
| Port D | |
| Port E | |
| Port F | |
| Port G | |

Section 11 16-Bit Timer Pulse Unit (TPU)

This LSI has an on-chip 16-bit timer pulse unit (TPU) that comprises three 16-bit timer channels or six 16-bit timer channels. The function list of the 16-bit timer unit and its block diagram are shown in table 11.1 and figure 11.1, respectively.

11.1 Features

- The number of channels
H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group: Six channels (channels 0, 1, 2, 3, 4, and 5)
H8S/2227 Group: three channels (channels 0, 1, and 2)
- Pulse input/output
H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group: Maximum of 16-pulse input/output
H8S/2227 Group: Maximum of eight-pulse input/output
- Selection of 8 counter input clocks for each channel
- The following operations can be set for each channel:
Waveform output at compare match
Input capture function
Counter clear operation
Synchronous operations:
Multiple timer counters (TCNT) can be written to simultaneously
Simultaneous clearing by compare match and input capture possible
Register simultaneous input/output possible by counter synchronous operation
Maximum of 15-phase PWM output possible by combination with synchronous operation
- Buffer operation settable for channels 0 and 3
- Phase counting mode settable independently for each of channels 1, 2, 4, and 5
- Cascaded operation*
- Fast access via internal 16-bit bus
- 26 interrupt sources
- Automatic transfer of register data
- A/D converter conversion start trigger can be generated
- Module stop mode can be set

Note: * Not available in the H8S/2227 Group.

Table 11.1 TPU Functions

| Item | Channel 0 | Channel 1 | Channel 2 | Channel 3 ^{*1} | Channel 4 ^{*1} | Channel 5 ^{*1} |
|--|--|--|--|--|--|--|
| Count clock | $\phi/1$ | $\phi/1$ | $\phi/1$ | $\phi/1$ | $\phi/1$ | $\phi/1$ |
| | $\phi/4$ | $\phi/4$ | $\phi/4$ | $\phi/4$ | $\phi/4$ | $\phi/4$ |
| | $\phi/16$ | $\phi/16$ | $\phi/16$ | $\phi/16$ | $\phi/16$ | $\phi/16$ |
| | $\phi/64$ | $\phi/64$ | $\phi/64$ | $\phi/64$ | $\phi/64$ | $\phi/64$ |
| | TCLKA | $\phi/256$ | $\phi/1024$ | $\phi/256$ | $\phi/1024$ | $\phi/256$ |
| | TCLKB | TCLKA | TCLKA | $\phi/1024$ | TCLKA | TCLKA |
| | TCLKC | TCLKB | TCLKB | $\phi/4096$ | TCLKC | TCLKC |
| | TCLKD | | TCLKC | TCLKA | | TCLKD |
| General registers (TGR) | TGRA_0 | TGRA_1 | TGRA_2 | TGRA_3 | TGRA_4 | TGRA_5 |
| | TGRB_0 | TGRB_1 | TGRB_2 | TGRB_3 | TGRB_4 | TGRB_5 |
| General registers/ buffer registers | TGRC_0 | — | — | TGRC_3 | — | — |
| | TGRD_0 | | | TGRD_3 | | |
| I/O pins | TIOCA0 | TIOCA1 | TIOCA2 | TIOCA3 | TIOCA4 | TIOCA5 |
| | TIOCB0 | TIOCB1 | TIOCB2 | TIOCB3 | TIOCB4 | TIOCB5 |
| | TIOCC0 | | | TIOCC3 | | |
| | TIOCD0 | | | TIOCD3 | | |
| Counter clear function | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture |
| Compare match output | 0 output | O | O | O | O | O |
| | 1 output | O | O | O | O | O |
| | Toggle output | O | O | O | O | O |
| Input capture function | O | O | O | O | O | O |
| Synchronous operation | O | O | O | O | O | O |
| PWM mode | O | O | O | O | O | O |
| Phase counting mode | — | O | O | — | O | O |
| Buffer operation | O | — | — | O | — | — |

| Item | Channel 0 | Channel 1 | Channel 2 | Channel 3 | Channel 4 | Channel 5 |
|-------------------------------|---|--|--|---|--|--|
| DTC activation | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture | TGR compare match or input capture |
| DMAC ^{*2} activation | TGRA_0 compare match or input capture | TGRA_1 compare match or input capture | TGRA_2 compare match or input capture | TGRA_3 compare match or input capture | TGRA_4 compare match or input capture | TGRA_5 compare match or input capture |
| A/D converter trigger | TGRA_0 compare match or input capture | TGRA_1 compare match or input capture | TGRA_2 compare match or input capture | TGRA_3 compare match or input capture | TGRA_4 compare match or input capture | TGRA_5 compare match or input capture |
| Interrupt sources | 5 sources • Compare match or input capture 0A • Compare match or input capture 0B • Compare match or input capture 0C • Compare match or input capture 0D • Overflow | 4 sources • Compare match or input capture 1A • Compare match or input capture 1B • Overflow • Underflow | 4 sources • Compare match or input capture 2A • Compare match or input capture 2B • Overflow • Underflow | 5 sources • Compare match or input capture 3A • Compare match or input capture 3B • Compare match or input capture 3D • Overflow | 4 sources • Compare match or input capture 4A • Compare match or input capture 4B • Overflow • Underflow | 4 sources • Compare match or input capture 5A • Compare match or input capture 5B • Overflow • Underflow |

Legend:

O: Possible

—: Not possible

Notes: 1. Not available in the H8S/2227 Group.
2. Supported only by the H8S/2239 Group.

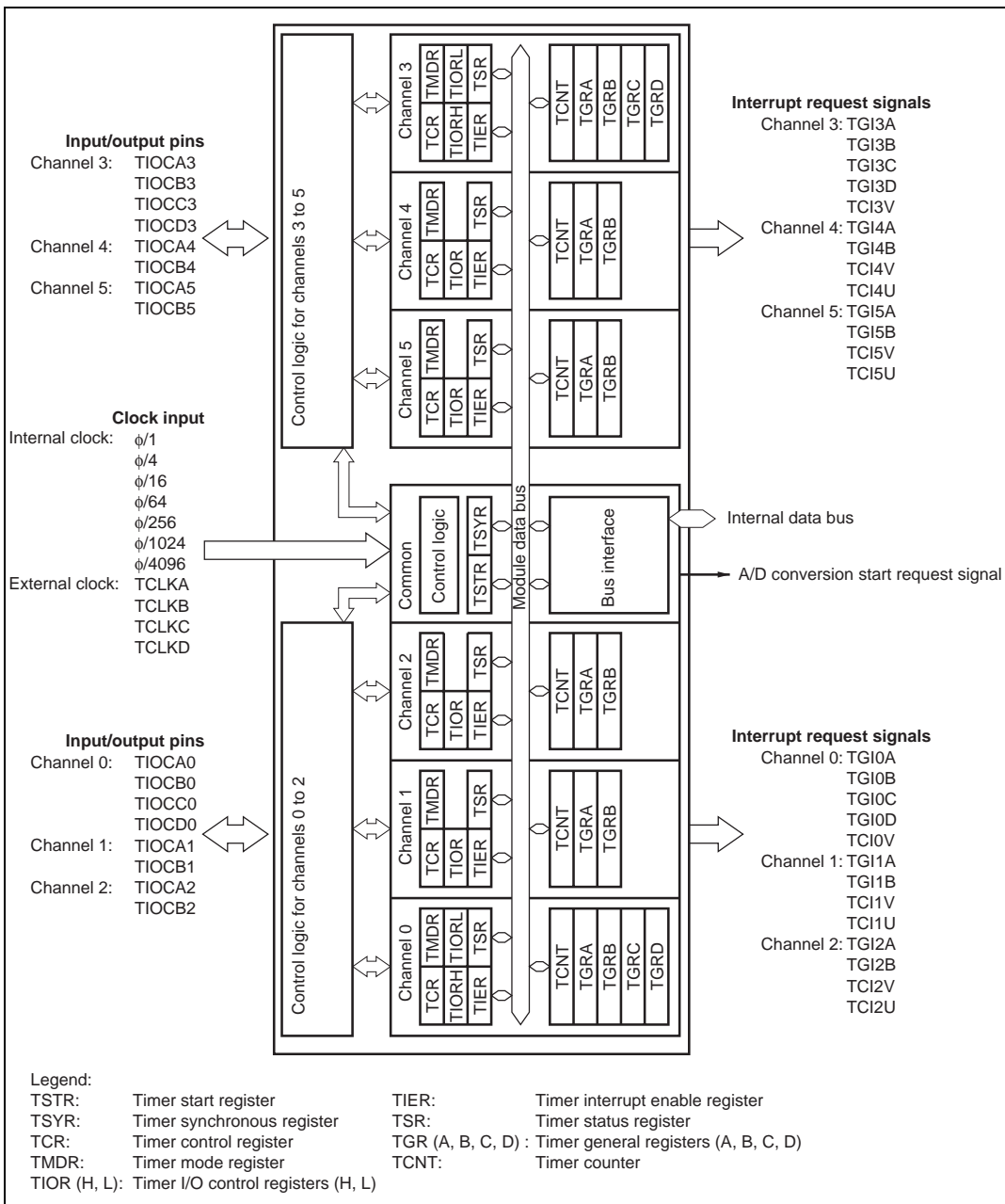


Figure 11.1 Block Diagram of TPU
 (H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group)

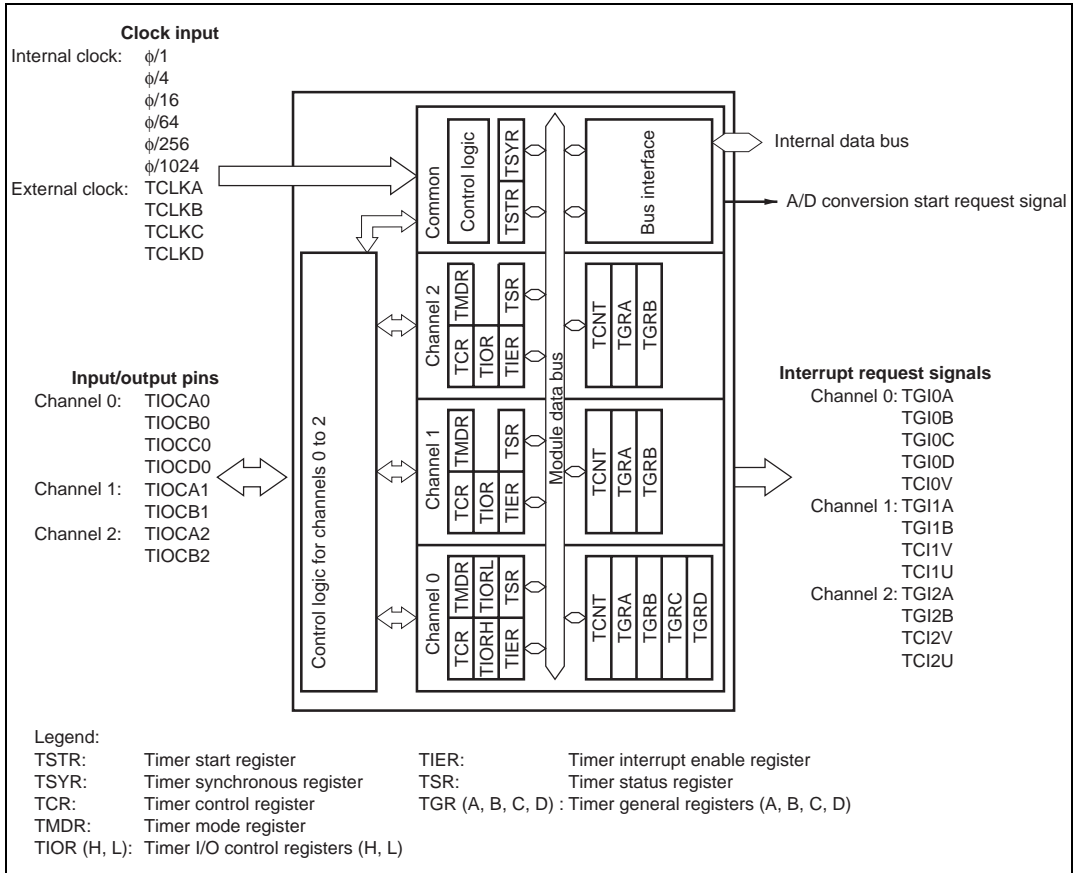


Figure 11.2 Block Diagram of TPU (H8S/2227 Group)

11.2 Input/Output Pins

Table 11.2 Pin Configuration

| Channel | Symbol | I/O | Function |
|---------|--------|-------|---|
| All | TCLKA | Input | External clock A input pin (Channels 1 and 5* phase counting mode A phase input) |
| | TCLKB | Input | External clock B input pin (Channels 1 and 5* phase counting mode B phase input) |
| | TCLKC | Input | External clock C input pin (Channels 2 and 4* phase counting mode A phase input) |
| | TCLKD | Input | External clock D input pin (Channels 2 and 4* phase counting mode B phase input) |
| 0 | TIOCA0 | I/O | TGRA_0 input capture input/output compare output/PWM output pin |
| | TIOCB0 | I/O | TGRB_0 input capture input/output compare output/PWM output pin |
| | TIOCC0 | I/O | TGRC_0 input capture input/output compare output/PWM output pin |
| | TIOCD0 | I/O | TGRD_0 input capture input/output compare output/PWM output pin |
| 1 | TIOCA1 | I/O | TGRA_1 input capture input/output compare output/PWM output pin |
| | TIOCB1 | I/O | TGRB_1 input capture input/output compare output/PWM output pin |
| 2 | TIOCA2 | I/O | TGRA_2 input capture input/output compare output/PWM output pin |
| | TIOCB2 | I/O | TGRB_2 input capture input/output compare output/PWM output pin |
| 3* | TIOCA3 | I/O | TGRA_3 input capture input/output compare output/PWM output pin |
| | TIOCB3 | I/O | TGRB_3 input capture input/output compare output/PWM output pin |
| | TIOCC3 | I/O | TGRC_3 input capture input/output compare output/PWM output pin |
| | TIOCD3 | I/O | TGRD_3 input capture input/output compare output/PWM output pin |
| 4* | TIOCA4 | I/O | TGRA_4 input capture input/output compare output/PWM output pin |
| | TIOCB4 | I/O | TGRB_4 input capture input/output compare output/PWM output pin |
| 5* | TIOCA5 | I/O | TGRA_5 input capture input/output compare output/PWM output pin |
| | TIOCB5 | I/O | TGRB_5 input capture input/output compare output/PWM output pin |

Note: * Not available in the H8S/2227 Group.

11.3 Register Descriptions

The TPU has the following registers in each channel.

- Timer control register_0 (TCR_0)
- Timer mode register_0 (TMDR_0)
- Timer I/O control register H_0 (TIORH_0)
- Timer I/O control register L_0 (TIORL_0)
- Timer interrupt enable register_0 (TIER_0)
- Timer status register_0 (TSR_0)
- Timer counter_0 (TCNT_0)
- Timer general register A_0 (TGRA_0)
- Timer general register B_0 (TGRB_0)
- Timer general register C_0 (TGRC_0)
- Timer general register D_0 (TGRD_0)
- Timer control register_1 (TCR_1)
- Timer mode register_1 (TMDR_1)
- Timer I/O control register _1 (TIOR_1)
- Timer interrupt enable register_1 (TIER_1)
- Timer status register_1 (TSR_1)
- Timer counter_1 (TCNT_1)
- Timer general register A_1 (TGRA_1)
- Timer general register B_1 (TGRB_1)
- Timer control register_2 (TCR_2)
- Timer mode register_2 (TMDR_2)
- Timer I/O control register_2 (TIOR_2)
- Timer interrupt enable register_2 (TIER_2)
- Timer status register_2 (TSR_2)
- Timer counter_2 (TCNT_2)
- Timer general register A_2 (TGRA_2)
- Timer general register B_2 (TGRB_2)
- Timer control register_3 (TCR_3)*
- Timer mode register_3 (TMDR_3)*
- Timer I/O control register H_3 (TIORH_3)*
- Timer I/O control register L_3 (TIORL_3)*

- Timer interrupt enable register_3 (TIER_3)*
- Timer status register_3 (TSR_3)*
- Timer counter_3 (TCNT_3)*
- Timer general register A_3 (TGRA_3)*
- Timer general register B_3 (TGRB_3)*
- Timer general register C_3 (TGRC_3)*
- Timer general register D_3 (TGRD_3)*
- Timer control register_4 (TCR_4)*
- Timer mode register_4 (TMDR_4)*
- Timer I/O control register_4 (TIOR_4)*
- Timer interrupt enable register_4 (TIER_4)*
- Timer status register_4 (TSR_4)*
- Timer counter_4 (TCNT_4)*
- Timer general register A_4 (TGRA_4)*
- Timer general register B_4 (TGRB_4)*
- Timer control register_5 (TCR_5)*
- Timer mode register_5 (TMDR_5)*
- Timer I/O control register_5 (TIOR_5)*
- Timer interrupt enable register_5 (TIER_5)*
- Timer status register_5 (TSR_5)*
- Timer counter_5 (TCNT_5)*
- Timer general register A_5 (TGRA_5)*
- Timer general register B_5 (TGRB_5)*

Common Registers

- Timer start register (TSTR)
- Timer synchronous register (TSYR)

Note: * Not available in the H8S/2227 Group.

11.3.1 Timer Control Register (TCR)

The TCR registers control the TCNT operation for each channel. The TPU of the H8S/2227 Group has a total of three TCR registers, one each for channels 0 to 2. In other groups, the TPU has a total of six TCR registers, one each for channels 0 to 5. TCR register settings should be made only when TCNT operation is stopped.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | CCLR2 | 0 | R/W | Counter Clear 2 to 0 |
| 6 | CCLR1 | 0 | R/W | These bits select the TCNT counter clearing source. See tables 11.3 and 11.4 for details. |
| 5 | CCLR0 | 0 | R/W | |
| 4 | CKEG1 | 0 | R/W | Clock Edge 1 and 0 |
| 3 | CKEG0 | 0 | R/W | <p>These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. $\phi/4$ both edges = $\phi/2$ rising edge). If phase counting mode is used on channels 1, 2, 4*, and 5*, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is $\phi/4$ or slower. When the input clock is $\phi/1$ or when overflow/underflow of another channel is selected, this setting is ignored and the input clock is counted at the falling edge of ϕ.</p> <p>00: Count at rising edge 01: Count at falling edge 1×: Count at both edges</p> <p>Legend: ×: Don't care</p> |
| 2 | TPSC2 | 0 | R/W | Time Prescaler 2 to 0 |
| 1 | TPSC1 | 0 | R/W | These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See tables 11.5 to 11.10 for details. |
| 0 | TPSC0 | 0 | R/W | |

Note: * Not available in the H8S/2227 Group.

Table 11.3 CCLR2 to CCLR0 (Channels 0 and 3)

| Channel | Bit 7 CCLR2 | Bit 6 CCLR1 | Bit 5 CCLR0 | Description |
|--------------------|----------------|----------------|----------------|--|
| 0, 3 ^{*3} | 0 | 0 | 0 | TCNT clearing disabled |
| | | | 1 | TCNT cleared by TGRA compare match/input capture |
| | | 1 | 0 | TCNT cleared by TGRB compare match/input capture |
| | | | 1 | TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation ^{*1} |
| | 1 | 0 | 0 | TCNT clearing disabled |
| | | | 1 | TCNT cleared by TGRC compare match/input capture ^{*2} |
| | | 1 | 0 | TCNT cleared by TGRD compare match/input capture ^{*2} |
| | | | 1 | TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation ^{*1} |

Notes: 1. Synchronous operation setting is performed by setting the SYNC bit in TSYR to 1.
 2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.
 3. Not available in the H8S/2227 Group.

Table 11.4 CCLR2 to CCLR0 (Channels 1, 2, 4, and 5)

| Channel | Bit 7 Reserved ^{*2} | Bit 6 CCLR1 | Bit 5 CCLR0 | Description |
|---|---------------------------------|----------------|----------------|--|
| 1, 2, 4 ^{*3} , 5 ^{*3} | 0 | 0 | 0 | TCNT clearing disabled |
| | | | 1 | TCNT cleared by TGRA compare match/input capture |
| | | 1 | 0 | TCNT cleared by TGRB compare match/input capture |
| | | | 1 | TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation ^{*1} |

Notes: 1. Synchronous operation setting is performed by setting the SYNC bit in TSYR to 1.
 2. Bit 7 is reserved in channels 1, 2, 4, and 5. It is always read as 0 and cannot be modified.
 3. Not available in the H8S/2227 Group.

Table 11.5 TPSC2 to TPSC0 (Channel 0)

| Channel | Bit 2 TPSC2 | Bit 1 TPSC1 | Bit 0 TPSC0 | Description |
|---------|----------------|----------------|----------------|---|
| 0 | 0 | 0 | 0 | Internal clock: counts on $\phi/1$ |
| | | | 1 | Internal clock: counts on $\phi/4$ |
| | | 1 | 0 | Internal clock: counts on $\phi/16$ |
| | | | 1 | Internal clock: counts on $\phi/64$ |
| | 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | | 1 | External clock: counts on TCLKB pin input |
| | | 1 | 0 | External clock: counts on TCLKC pin input |
| | | | 1 | External clock: counts on TCLKD pin input |

Table 11.6 TPSC2 to TPSC0 (Channel 1)

| Channel | Bit 2 TPSC2 | Bit 1 TPSC1 | Bit 0 TPSC0 | Description |
|---------|----------------|----------------|----------------|--|
| 1 | 0 | 0 | 0 | Internal clock: counts on $\phi/1$ |
| | | | 1 | Internal clock: counts on $\phi/4$ |
| | | 1 | 0 | Internal clock: counts on $\phi/16$ |
| | | | 1 | Internal clock: counts on $\phi/64$ |
| | 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | | 1 | External clock: counts on TCLKB pin input |
| | | 1 | 0 | Internal clock: counts on $\phi/256$ |
| | | | 1 | Counts on TCNT2 overflow/underflow Setting is prohibited in the H8S/2227 Group. |

Note: This setting is ignored when channel 1 is in phase counting mode.

Table 11.7 TPSC2 to TPSC0 (Channel 2)

| Channel | Bit 2 TPSC2 | Bit 1 TPSC1 | Bit 0 TPSC0 | Description |
|---------|----------------|----------------|----------------|---|
| 2 | 0 | 0 | 0 | Internal clock: counts on $\phi/1$ |
| | | | 1 | Internal clock: counts on $\phi/4$ |
| | | 1 | 0 | Internal clock: counts on $\phi/16$ |
| | | | 1 | Internal clock: counts on $\phi/64$ |
| | 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | | 1 | External clock: counts on TCLKB pin input |
| | | 1 | 0 | External clock: counts on TCLKC pin input |
| | | | 1 | Internal clock: counts on $\phi/1024$ |

Note: This setting is ignored when channel 2 is in phase counting mode.

Table 11.8 TPSC2 to TPSC0 (Channel 3)

| Channel | Bit 2 TPSC2 | Bit 1 TPSC1 | Bit 0 TPSC0 | Description |
|---------|----------------|----------------|----------------|---|
| 3* | 0 | 0 | 0 | Internal clock: counts on $\phi/1$ |
| | | | 1 | Internal clock: counts on $\phi/4$ |
| | | 1 | 0 | Internal clock: counts on $\phi/16$ |
| | | | 1 | Internal clock: counts on $\phi/64$ |
| | 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | | 1 | Internal clock: counts on $\phi/1024$ |
| | | 1 | 0 | Internal clock: counts on $\phi/256$ |
| | | | 1 | Internal clock: counts on $\phi/4096$ |

Note: * Not available in the H8S/2227 Group.

Table 11.9 TPSC2 to TPSC0 (Channel 4)

| Channel | Bit 2 TPSC2 | Bit 1 TPSC1 | Bit 0 TPSC0 | Description |
|---------|----------------|----------------|----------------|---|
| 4* | 0 | 0 | 0 | Internal clock: counts on $\phi/1$ |
| | | | 1 | Internal clock: counts on $\phi/4$ |
| | | 1 | 0 | Internal clock: counts on $\phi/16$ |
| | | | 1 | Internal clock: counts on $\phi/64$ |
| | 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | | 1 | External clock: counts on TCLKC pin input |
| | | 1 | 0 | Internal clock: counts on $\phi/1024$ |
| | | | 1 | Counts on TCNT5 overflow/underflow |

Notes: This setting is ignored when channel 4 is in phase counting mode.

* Not available in the H8S/2227 Group.

Table 11.10 TPSC2 to TPSC0 (Channel 5)

| Channel | Bit 2 TPSC2 | Bit 1 TPSC1 | Bit 0 TPSC0 | Description |
|---------|----------------|----------------|----------------|---|
| 5* | 0 | 0 | 0 | Internal clock: counts on $\phi/1$ |
| | | | 1 | Internal clock: counts on $\phi/4$ |
| | | 1 | 0 | Internal clock: counts on $\phi/16$ |
| | | | 1 | Internal clock: counts on $\phi/64$ |
| | 1 | 0 | 0 | External clock: counts on TCLKA pin input |
| | | | 1 | External clock: counts on TCLKC pin input |
| | | 1 | 0 | Internal clock: counts on $\phi/256$ |
| | | | 1 | External clock: counts on TCLKD pin input |

Notes: This setting is ignored when channel 5 is in phase counting mode.

* Not available in the H8S/2227 Group.

11.3.2 Timer Mode Register (TMDR)

The TMDR registers are used to set the operating mode for each channel. The TPU of the H8S/2227 Group has a total of three TMDR registers, one each for channels 0 to 2. In other groups, the TPU has a total of six TMDR registers, one each for channels 0 to 5. TMDR register settings should be made only when TCNT operation is stopped.

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 7, 6 | — | All 1 | — | Reserved These bits are always read as 1 and cannot be modified. |
| 5 | BFB | 0 | R/W | Buffer Operation B Specifies whether TGRB is to operate in the normal way, or TGRB and TGRD are to be used together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare is not generated. In channels 1, 2, 4*, and 5*, which have no TGRD, bit 5 is reserved. It is always read as 0 and cannot be modified. 0: TGRB operates normally 1: TGRB and TGRD used together for buffer operation |
| 4 | BFA | 0 | R/W | Buffer Operation A Specifies whether TGRA is to operate in the normal way, or TGRA and TGRC are to be used together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare is not generated. In channels 1, 2, 4*, and 5*, which have no TGRC, bit 4 is reserved. It is always read as 0 and cannot be modified. 0: TGRA operates normally 1: TGRA and TGRC used together for buffer operation |
| 3 | MD3 | 0 | R/W | Modes 3 to 0 |
| 2 | MD2 | 0 | R/W | These bits are used to set the timer operating mode. |
| 1 | MD1 | 0 | R/W | |
| 0 | MD0 | 0 | R/W | MD3 is a reserved bit. The write value should always be 0. See table 11.11 for details. |

Note: * Not available in the H8S/2227 Group.

Table 11.11 MD3 to MD0

| Bit 3 MD3*1 | Bit 2 MD2*2 | Bit 1 MD1 | Bit 0 MD0 | Description |
|----------------|----------------|--------------|--------------|-----------------------|
| 0 | 0 | 0 | 0 | Normal operation |
| | | | 1 | Reserved |
| | | 1 | 0 | PWM mode 1 |
| | | | 1 | PWM mode 2 |
| | 1 | 0 | 0 | Phase counting mode 1 |
| | | | 1 | Phase counting mode 2 |
| | | 1 | 0 | Phase counting mode 3 |
| | | | 1 | Phase counting mode 4 |
| 1 | × | × | × | — |

Legend: ×: Don't care

Notes: 1. MD3 is a reserved bit. In a write, it should always be written with 0.

2. Phase counting mode cannot be set for channels 0 and 3. In this case, 0 should always be written to MD2.

11.3.3 Timer I/O Control Register (TIOR)

The TIOR registers control the TGR registers. The TPU of the H8S/2227 Group has a total of four TIOR registers, two for channel 0 and one each for channels 1 and 2. In other groups, the TPU has a total of eight TIOR registers, two each for channels 0 and 3, and one each for channels 1, 2, 4, and 5. Care is required since TIOR is affected by the TMDR setting.

The initial output specified by TIOR is valid when the counter is stopped (the CST bit in TSTR is cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified.

When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

TIORH_0, TIOR_1, TIOR_2, TIORH_3*, TIOR_4*, TIOR_5*

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | IOB3 | 0 | R/W | I/O Control B3 to B0 |
| 6 | IOB2 | 0 | R/W | Specify the function of TGRB. |
| 5 | IOB1 | 0 | R/W | For details, see tables 11.12, 11.14, 11.15, 11.16, 11.18, and 11.19. |
| 4 | IOB0 | 0 | R/W | |
| 3 | IOA3 | 0 | R/W | I/O Control A3 to A0 |
| 2 | IOA2 | 0 | R/W | Specify the function of TGRA. |
| 1 | IOA1 | 0 | R/W | For details, see tables 11.20, 11.22, 11.23, 11.24, 11.26, and 11.27. |
| 0 | IOA0 | 0 | R/W | |

Note: * Not available in the H8S/2227 Group.

TIORL_0, TIORL_3*

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | IOD3 | 0 | R/W | I/O Control D3 to D0 |
| 6 | IOD2 | 0 | R/W | Specify the function of TGRD. |
| 5 | IOD1 | 0 | R/W | For details, see tables 11.13 and 11.17. |
| 4 | IOD0 | 0 | R/W | |
| 3 | IOC3 | 0 | R/W | I/O Control C3 to C0 |
| 2 | IOC2 | 0 | R/W | Specify the function of TGRC. |
| 1 | IOC1 | 0 | R/W | For details, see tables 11.21 and 11.25 |
| 0 | IOC0 | 0 | R/W | |

Note: * Not available in the H8S/2227 Group.

Table 11.12 TIORH_0

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| Bit 7 IOB3 | Bit 6 IOB2 | Bit 5 IOB1 | Bit 4 IOB0 | TGRB_0 Function | TIOCB0 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 1 | | Initial output is 0 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 0 output Toggle output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | | 1 | | Initial output is 1 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 1 output Toggle output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| 1 | 0 | 0 | 0 | Input capture register | Capture input source is TIOCB0 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCB0 pin Input capture at falling edge |
| | | | 1 | | Capture input source is TIOCB0 pin Input capture at both edges |
| | | 1 | × | | Capture input source is TIOCB0 pin Input capture at both edges |
| | | | × | | Capture input source is channel 1/count clock Input capture at TCNT_1 count- up/count- down*1*2 |
| | | | × | | Capture input source is channel 1/count clock Input capture at TCNT_1 count- up/count- down*1*2 |

Legend: ×: Don't care

- Notes: 1. When bits TPSC2 to TPSC0 in TCR_1 are set to B'000 and $\phi/1$ is used as the TCNT_1 count clock, this setting is invalid and input capture is not generated.
2. Not available in the H8S/2227 Group.

Table 11.13 TIORL_0

| | | | | Description | |
|---------------|---------------|---------------|---------------|---|--|
| Bit 7 IOD3 | Bit 6 IOD2 | Bit 5 IOD1 | Bit 4 IOD0 | TGRD_0 Function | TIOCD0 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register ^{*2} | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | | 1 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | | 1 | 0 | | Capture input source is TIOCD0 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCD0 pin Input capture at falling edge |
| | | | × | | Capture input source is TIOCD0 pin Input capture at both edges |
| | | | × | | Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down ^{*1*3} |

Legend: ×: Don't care

- Notes: 1. When bits TPSC2 to TPSC0 in TCR_1 are set to B'000 and $\phi/1$ is used as the TCNT_1 count clock, this setting is invalid and input capture is not generated.
2. When the BFB bit in TMDR_0 is set to 1 and TGRD_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.
3. Not available in the H8S/2227 Group.

Table 11.14 TIOR_1

| Bit 7 IOB3 | Bit 6 IOB2 | Bit 5 IOB1 | Bit 4 IOB0 | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| | | | | TGRB_1 Function | TIOCB1 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | 1 | 0 | 0 | Input capture register | Capture input source is TIOCB1 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCB1 pin Input capture at falling edge |
| | | 1 | × | | Capture input source is TIOCB1 pin Input capture at both edges |
| | | | × | | TGRC_0 compare match/input capture Input capture at generation of TGRC_0 compare match/input capture* |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.15 TIOR_2

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| Bit 7 IOB3 | Bit 6 IOB2 | Bit 5 IOB1 | Bit 4 IOB0 | TGRB_2 Function | TIOCB2 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 0 | | Initial output is 0 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 0 output Toggle output at compare match |
| | | 1 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | 1 | 0 | 0 | Input capture register | Capture input source is TIOCB2 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCB2 pin Input capture at falling edge |
| | | | 0 | | Capture input source is TIOCB2 pin Input capture at both edges |
| | | 1 | 0 | | |
| | | | 1 | | |
| | | | 0 | | |

Legend: ×: Don't care

Table 11.16 TIORH_3

| Bit 7 IOB3 | Bit 6 IOB2 | Bit 5 IOB1 | Bit 4 IOB0 | Description | |
|---------------|---------------|---------------|---------------|----------------------------------|--|
| | | | | TGRB_3 Function ^{*2} | TIOCB3 Pin Function ^{*2} |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 0 | | Initial output is 0 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 0 output Toggle output at compare match |
| | | | 1 | | Output disabled |
| | | | 0 | | Initial output is 1 output 0 output at compare match |
| | 1 | 0 | 0 | Input capture register | Initial output is 1 output 0 output at compare match |
| | | | 1 | | Initial output is 1 output 1 output at compare match |
| | | | 0 | | Initial output is 1 output Toggle output at compare match |
| | | 1 | 0 | | Capture input source is TIOCB3 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCB3 pin Input capture at falling edge |
| | | | × | | Capture input source is TIOCB3 pin Input capture at both edges |
| 1 | 0 | 0 | × | | Capture input source is channel 4/count clock Input capture at TCNT_4 count-up/count-down ^{*1} |
| | | | × | | |

Legend: ×: Don't care

Notes: 1. When bits TPSC2 to TPSC0 in TCR_4 are set to B'000 and $\phi/1$ is used as the TCNT_4 count clock, this setting is invalid and input capture is not generated.

2. Not available in the H8S/2227 Group.

Table 11.17 TIORL_3

| | | | | Description | |
|---------------|---------------|---------------|---------------|---|--|
| Bit 7 IOD3 | Bit 6 IOD2 | Bit 5 IOD1 | Bit 4 IOD0 | TGRD_3 Function ^{*3} | TIOCD3 Pin Function ^{*3} |
| 0 | 0 | 0 | 0 | Output compare register ^{*2} | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 1 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | | 1 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | 1 | 0 | 0 | Input capture register ^{*2} | Capture input source is TIOCD3 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCD3 pin Input capture at falling edge |
| | | | 1 | | Capture input source is TIOCD3 pin Input capture at both edges |
| | | | 1 | | Capture input source is channel 4/count clock Input capture at TCNT_4 count-up/count-down ^{*1} |

Legend: ×: Don't care

- Notes: 1. When bits TPSC2 to TPSC0 in TCR_4 are set to B'000 and $\phi/1$ is used as the TCNT_4 count clock, this setting is invalid and input capture is not generated.
2. When the BFB bit in TMDR_3 is set to 1 and TGRD_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.
3. Not available in the H8S/2227 Group.

Table 11.18 TIOR_4

| Bit 7 IOB3 | Bit 6 IOB2 | Bit 5 IOB1 | Bit 4 IOB0 | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| | | | | TGRB_4 Function* | TIOCB4 Pin Function* |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | 1 | 0 | 0 | Input capture register | Capture input source is TIOCB4 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCB4 pin Input capture at falling edge |
| | | 1 | × | | Capture input source is TIOCB4 pin Input capture at both edges |
| | | | × | | Capture input source is TGRC_3 compare match/input capture Input capture at generation of TGRC_3 compare match/input capture |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.19 TIOR_5

| Bit 7 IOB3 | Bit 6 IOB2 | Bit 5 IOB1 | Bit 4 IOB0 | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| | | | | TGRB_5 Function* | TIOCB5 Pin Function* |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | | 1 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | 1 | 0 | 0 | | Capture input source is TIOCB5 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCB5 pin Input capture at falling edge |
| | | | 0 | | Capture input source is TIOCB5 pin Input capture at both edges |
| | | | 1 | | |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.20 TIORH_0

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| Bit 3 IOA3 | Bit 2 IOA2 | Bit 1 IOA1 | Bit 0 IOA0 | TGRA_0 Function | TIOCA0 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 1 | | Initial output is 0 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 0 output Toggle output at compare match |
| | | | 1 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | 1 | 0 | 0 | Input capture register | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | | | 1 | | Capture input source is TIOCA0 pin Input capture at rising edge |
| | | 1 | 0 | | Capture input source is TIOCA0 pin Input capture at falling edge |
| | | | 1 | | Capture input source is TIOCA0 pin Input capture at both edges |
| | | | × | | Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down* |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.21 TIORL_0

| | | | | Description | |
|---------------|---------------|---------------|---------------|---|--|
| Bit 3 IOC3 | Bit 2 IOC2 | Bit 1 IOC1 | Bit 0 IOC0 | TGRC_0 Function | TIOCC0 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register ^{*1} | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | 1 | 0 | 0 | Input capture register ^{*1} | Capture input source is TIOCC0 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCC0 pin Input capture at falling edge |
| | | 1 | × | | Capture input source is TIOCC0 pin Input capture at both edges |
| | | | × | | Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down ^{*2} |

Legend: ×: Don't care

Notes: 1. When the BFA bit in TMDR_0 is set to 1 and TGRC_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

2. Not available in the H8S/2227 Group.

Table 11.22 TIOR_1

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| Bit 3 IOA3 | Bit 2 IOA2 | Bit 1 IOA1 | Bit 0 IOA0 | TGRA_1 Function | TIOCA1 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| 1 | 0 | 0 | 0 | Input capture register | Capture input source is TIOCA1 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCA1 pin Input capture at falling edge |
| | | 1 | × | | Capture input source is TIOCA1 pin Input capture at both edges |
| | 1 | × | × | | Capture input source is TGRA_0 compare match/input capture |
| | | | | | Input capture at generation of channel 0/TGRA_0 compare match/input capture* |
| | | | | | |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.23 TIOR_2

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| Bit 3 IOA3 | Bit 2 IOA2 | Bit 1 IOA1 | Bit 0 IOA0 | TGRA_2 Function | TIOCA2 Pin Function |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| 1 | × | 0 | 0 | Input capture register | Capture input source is TIOCA2 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCA2 pin Input capture at falling edge |
| | | 1 | × | | Capture input source is TIOCA2 pin Input capture at both edges |
| | | | | | |

Legend: ×: Don't care

Table 11.24 TIORH_3

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|--|
| Bit 3 IOA3 | Bit 2 IOA2 | Bit 1 IOA1 | Bit 0 IOA0 | TGRA_3 Function* | TIOCA3 Pin Function* |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 1 | | Initial output is 0 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 0 output Toggle output at compare match |
| | | | 1 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | 1 | 0 | 0 | Input capture register | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | | | 1 | | Capture input source is TIOCA3 pin Input capture at rising edge |
| | | 1 | 0 | | Capture input source is TIOCA3 pin Input capture at falling edge |
| | | | 1 | | Capture input source is TIOCA3 pin Input capture at both edges |
| | | | × | | Capture input source is channel 4/count clock Input capture at TCNT_4 count-up/count-down |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.25 TIORL_3

| | | | | Description | |
|---------------|---------------|---------------|---------------|---------------------------------|--|
| Bit 3 IOC3 | Bit 2 IOC2 | Bit 1 IOC1 | Bit 0 IOC0 | TGRC_3 Function*2 | TIOCC3 Pin Function*2 |
| 0 | 0 | 0 | 0 | Output compare register*1 | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | 1 | | Initial output is 0 output 1 output at compare match |
| | | 1 | 0 | | Initial output is 0 output Toggle output at compare match |
| | | | 1 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | 1 | 0 | 0 | Input capture register*1 | Initial output is 1 output 0 output at compare match |
| | | | 1 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | | 1 | 0 | | Capture input source is TIOCC3 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCC3 pin Input capture at falling edge |
| | | | × | | Capture input source is TIOCC3 pin Input capture at both edges |
| 1 | 0 | 0 | 0 | Input capture register*1 | Capture input source is channel 4/count clock Input capture at TCNT_4 count-up/count-down |
| | | | 1 | | |
| 1 | 1 | × | × | | |
| | | | × | | |

Legend: ×: Don't care

Notes: 1. When the BFA bit in TMDR_3 is set to 1 and TGRC_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

2. Not available in the H8S/2227 Group.

Table 11.26 TIOR_4

| Bit 3 IOA3 | Bit 2 IOA2 | Bit 1 IOA1 | Bit 0 IOA0 | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| | | | | TGRA_4 Function* | TIOCA4 Pin Function* |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | | — | | — |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | — | 1 | | Initial output is 0 output Toggle output at compare match |
| | | — | — | | — |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | | — | | — |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | — | 1 | | Initial output is 1 output Toggle output at compare match |
| | | — | — | | — |
| 1 | 0 | 0 | 0 | Input capture register | Capture input source is TIOCA4 pin Input capture at rising edge |
| | | | 1 | | Capture input source is TIOCA4 pin Input capture at falling edge |
| | | — | — | | — |
| | 1 | 1 | × | | Capture input source is TIOCA4 pin Input capture at both edges |
| | | — | — | | — |
| | | — | × | | Capture input source is TGRA_3 compare match/input capture Input capture at generation of TGRA_3 compare match/input capture |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

Table 11.27 TIOR_5

| | | | | Description | |
|---------------|---------------|---------------|---------------|-------------------------------|---|
| Bit 3 IOA3 | Bit 2 IOA2 | Bit 1 IOA1 | Bit 0 IOA0 | TGRA_5 Function* | TIOCA5 Pin Function* |
| 0 | 0 | 0 | 0 | Output compare register | Output disabled |
| | | | 1 | | Initial output is 0 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 0 output 1 output at compare match |
| | | | 1 | | Initial output is 0 output Toggle output at compare match |
| | 1 | 0 | 0 | | Output disabled |
| | | | 1 | | Initial output is 1 output 0 output at compare match |
| | | 1 | 0 | | Initial output is 1 output 1 output at compare match |
| | | | 1 | | Initial output is 1 output Toggle output at compare match |
| | × | 0 | 0 | Input capture register | Input capture source is TIOCA5 pin Input capture at rising edge |
| | | | 1 | | Input capture source is TIOCA5 pin Input capture at falling edge |
| | | 1 | × | | Input capture source is TIOCA5 pin Input capture at both edges |
| | | | | | |

Legend: ×: Don't care

Note: * Not available in the H8S/2227 Group.

11.3.4 Timer Interrupt Enable Register (TIER)

The TIER registers control enabling or disabling of interrupt requests for each channel. The TPU of the H8S/2227 Group has a total of three TIER registers, one each for channels 0 to 2. In other groups, the TPU has a total of six TIER registers, one each for channels 0 to 5.

| Bit | Bit Name | Initial value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | TTGE | 0 | R/W | <p>A/D Conversion Start Request Enable</p> <p>Enables or disables generation of A/D conversion start requests by TGRA input capture/compare match.</p> <p>0: A/D conversion start request generation disabled 1: A/D conversion start request generation enabled</p> |
| 6 | — | 1 | — | <p>Reserved</p> <p>This bit is always read as 1 and cannot be modified.</p> |
| 5 | TCIEU | 0 | R/W | <p>Underflow Interrupt Enable</p> <p>Enables or disables interrupt requests (TCIU) by the TCFU flag when the TCFU flag in TSR is set to 1 in channels 1, 2, 4*, and 5*.</p> <p>In channels 0 and 3*, bit 5 is reserved. It is always read as 0 and cannot be modified.</p> <p>0: Interrupt requests (TCIU) by TCFU disabled 1: Interrupt requests (TCIU) by TCFU enabled</p> |
| 4 | TCIEV | 0 | R/W | <p>Overflow Interrupt Enable</p> <p>Enables or disables interrupt requests (TCIV) by the TCFV flag when the TCFV flag in TSR is set to 1.</p> <p>0: Interrupt requests (TCIV) by TCFV disabled 1: Interrupt requests (TCIV) by TCFV enabled</p> |
| 3 | TGIED | 0 | R/W | <p>TGR Interrupt Enable D</p> <p>Enables or disables interrupt requests (TGID) by the TGFD bit when the TGFD bit in TSR is set to 1 in channels 0 and 3*.</p> <p>In channels 1, 2, 4*, and 5*, bit 3 is reserved. It is always read as 0 and cannot be modified.</p> <p>0: Interrupt requests (TGID) by TGFD bit disabled 1: Interrupt requests (TGID) by TGFD bit enabled</p> |

| Bit | Bit Name | Initial value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | TGIEC | 0 | R/W | <p>TGR Interrupt Enable C</p> <p>Enables or disables interrupt requests (TGIC) by the TGFC bit when the TGFC bit in TSR is set to 1 in channels 0 and 3*.</p> <p>In channels 1, 2, 4*, and 5*, bit 2 is reserved. It is always read as 0 and cannot be modified.</p> <p>0: Interrupt requests (TGIC) by TGFC bit disabled</p> <p>1: Interrupt requests (TGIC) by TGFC bit enabled</p> |
| 1 | TGIEB | 0 | R/W | <p>TGR Interrupt Enable B</p> <p>Enables or disables interrupt requests (TGIB) by the TGFB bit when the TGFB bit in TSR is set to 1.</p> <p>0: Interrupt requests (TGIB) by TGFB bit disabled</p> <p>1: Interrupt requests (TGIB) by TGFB bit enabled</p> |
| 0 | TGIEA | 0 | R/W | <p>TGR Interrupt Enable A</p> <p>Enables or disables interrupt requests (TGIA) by the TGFA bit when the TGFA bit in TSR is set to 1.</p> <p>0: Interrupt requests (TGIA) by TGFA bit disabled</p> <p>1: Interrupt requests (TGIA) by TGFA bit enabled</p> |

Note: * Not available in the H8S/2227 Group.

11.3.5 Timer Status Register (TSR)

The TSR registers indicate the status of each channel. The TPU of the H8S/2227 Group has a total of three TSR registers, one each for channels 0 to 2. In other groups, the TPU has a total of six TSR registers, one each for channels 0 to 5.

| Bit | Bit Name | Initial value | R/W | Description |
|-----|----------|---------------|---------------------|---|
| 7 | TCFD | 1 | R | Count Direction Flag Status flag that shows the direction in which TCNT counts in channels 1, 2, 4 ^{*3} , and 5 ^{*3} . In channels 0 and 3 ^{*3} , bit 7 is reserved. It is always read as 1 and cannot be modified. 0: TCNT counts down 1: TCNT counts up |
| 6 | — | 1 | — | Reserved This bit is always read as 1 and cannot be modified. |
| 5 | TCFU | 0 | R/(W) ^{*1} | Underflow Flag Status flag that indicates that TCNT underflow has occurred when channels 1, 2, 4 ^{*3} , and 5 ^{*3} are set to phase counting mode. In channels 0 and 3 ^{*3} , bit 5 is reserved. It is always read as 0 and cannot be modified. [Setting condition] When the TCNT value underflows (changes from H'0000 to H'FFFF) [Clearing condition] When 0 is written to TCFU after reading TCFU = 1 |
| 4 | TCFV | 0 | R/(W) ^{*1} | Overflow Flag Status flag that indicates that TCNT overflow has occurred. [Setting condition] When the TCNT value overflows (changes from H'FFFF to H'0000) [Clearing condition] When 0 is written to TCFV after reading TCFV = 1 |

| Bit | Bit Name | Initial value | R/W | Description |
|-----|----------|---------------|---------|--|
| 3 | TGFD | 0 | R/(W)*1 | <p>Input Capture/Output Compare Flag D</p> <p>Status flag that indicates the occurrence of TGRD input capture or compare match in channels 0 and 3*3.</p> <p>In channels 1, 2, 4*3, and 5*3, bit 3 is reserved. It is always read as 0 and cannot be modified.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> When TCNT = TGRD while TGRD is functioning as output compare register When TCNT value is transferred to TGRD by input capture signal while TGRD is functioning as input capture register <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When DTC is activated by TGID interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 When 0 is written to TGFD after reading TGFD = 1 |
| 2 | TGFC | 0 | R/(W)*1 | <p>Input Capture/Output Compare Flag C</p> <p>Status flag that indicates the occurrence of TGRC input capture or compare match in channels 0 and 3*3.</p> <p>In channels 1, 2, 4*3, and 5*3, bit 2 is reserved. It is always read as 0 and cannot be modified.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> When TCNT = TGRC while TGRC is functioning as output compare register When TCNT value is transferred to TGRC by input capture signal while TGRC is functioning as input capture register <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When DTC is activated by TGIC interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 When 0 is written to TGFC after reading TGFC = 1 |

| Bit | Bit Name | Initial value | R/W | Description |
|-----|----------|---------------|---------------------|---|
| 1 | TGFB | 0 | R/(W)* ¹ | <p>Input Capture/Output Compare Flag B</p> <p>Status flag that indicates the occurrence of TGRB input capture or compare match.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> When TCNT = TGRB while TGRB is functioning as output compare register When TCNT value is transferred to TGRB by input capture signal while TGRB is functioning as input capture register <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When DTC is activated by TGIB interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 When 0 is written to TGFB after reading TGFB = 1 |
| 0 | TGFA | 0 | R/(W)* ¹ | <p>Input Capture/Output Compare Flag A</p> <p>Status flag that indicates the occurrence of TGRA input capture or compare match.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> When TCNT = TGRA while TGRA is functioning as output compare register When TCNT value is transferred to TGRA by input capture signal while TGRA is functioning as input capture register <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When DTC is activated by TGIA interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 When DMAC is activated by TGIA interrupt while DTE bit of DMABCR in DMAC is 1*² When 0 is written to TGFA after reading TGFA = 1 |

Notes: 1. Only 0 can be written, for flag clearing.
2. Supported only by the H8S/2239 Group.
3. Not available in the H8S/2227 Group.

11.3.6 Timer Counter (TCNT)

The TCNT registers are 16-bit readable/writable counters. The TPU of the H8S/2227 Group has a total of three TCNT registers, one each for channels 0 to 2. In other groups, the TPU has a total of six TCNT registers, one each for channels 0 to 5.

The TCNT counters are initialized to H'0000 by a reset, or in hardware standby mode.

The TCNT counters cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit.

11.3.7 Timer General Register (TGR)

The TGR registers are 16-bit readable/writable registers with a dual function as output compare and input capture registers. The TPU of the H8S/2227 Group has a total of four TGR registers, two for channel 0 and one each for channels 1 and 2. In other groups, the TPU has a total of eight TGR registers, two each for channels 0 and 3, and one each for channels 1, 2, 4, and 5. TGRC and TGRD for channels 0 and 3 can also be designated for operation as buffer registers. The TGR registers cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit. TGR buffer register combinations are TGRA-TGRC and TGRB-TGRD.

11.3.8 Timer Start Register (TSTR)

In the H8S/2227 Group, TSTR selects operate/stop for channels 0 to 2. In other groups, TSTR selects operate/stop for channels 0 to 5. When setting the operating mode in TMDR or setting the count clock in TCR, first stop the TCNT counter.

| Bit | Bit Name | Initial value | R/W | Description |
|------|----------|---------------|-----|---|
| 7, 6 | — | All 0 | — | Reserved The write value should always be 0. |
| 5 | CST5* | 0 | R/W | Counter Start 5 to 0 |
| 4 | CST4* | 0 | R/W | These bits select operation or stoppage for TCNT. |
| 3 | CST3* | 0 | R/W | If 0 is written to the CST bit during operation with the |
| 2 | CST2 | 0 | R/W | TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. |
| 1 | CST1 | 0 | R/W | If TIOR is written to when the CST bit is cleared to 0, |
| 0 | CST0 | 0 | R/W | the pin output level will be changed to the set initial output value. 0: TCNT_5 to TCNT_0 count operation is stopped 1: TCNT_5 to TCNT_0 performs count operation |

Note: * In the H8S/2227 Group, bits 5 to 3 are reserved. The write value should always be 0.

11.3.9 Timer Synchronous Register (TSYR)

In the H8S/2227 Group, TSYR selects independent or synchronous TCNT operation for channels 0 to 2. In other groups, TSYR selects independent or synchronous TCNT operation for channels 0 to 5. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

| Bit | Bit Name | Initial value | R/W | Description |
|------|----------|---------------|-----|---|
| 7, 6 | — | All 0 | R/W | Reserved The write value should always be 0. |
| 5 | SYNC5* | 0 | R/W | Timer Synchronization 5 to 0 |
| 4 | SYNC4* | 0 | R/W | These bits select whether operation is independent of or synchronized with other channels. |
| 3 | SYNC3* | 0 | R/W | |
| 2 | SYNC2 | 0 | R/W | When synchronous operation is selected, synchronous presetting of multiple channels, and synchronous clearing through counter clearing on another channel are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR2 to CCLR0 in TCR. 0: TCNT_5 to TCNT_0 operates independently (TCNT presetting /clearing is unrelated to other channels) 1: TCNT_5 to TCNT_0 performs synchronous operation (TCNT synchronous presetting/ synchronous clearing is possible) |
| 1 | SYNC1 | 0 | R/W | |
| 0 | SYNC0 | 0 | R/W | |

Note: * In the H8S/2227 Group, bits 5 to 3 are reserved. The write value should always be 0.

11.4 Operation

11.4.1 Basic Functions

Each channel has a TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, periodic counting, and external event counting.

Each TGR can be used as an input capture register or output compare register.

Counter Operation: When one of bits CST2 to CST0 (H8S/2227 Group) or bits CST5 to CST0 (groups other than H8S/2227) in TSTR is set to 1, the TCNT counter for the corresponding channel starts counting. TCNT can operate as a free-running counter, periodic counter, and so on.

1. Example of count operation setting procedure

Figure 11.3 shows an example of the count operation setting procedure.

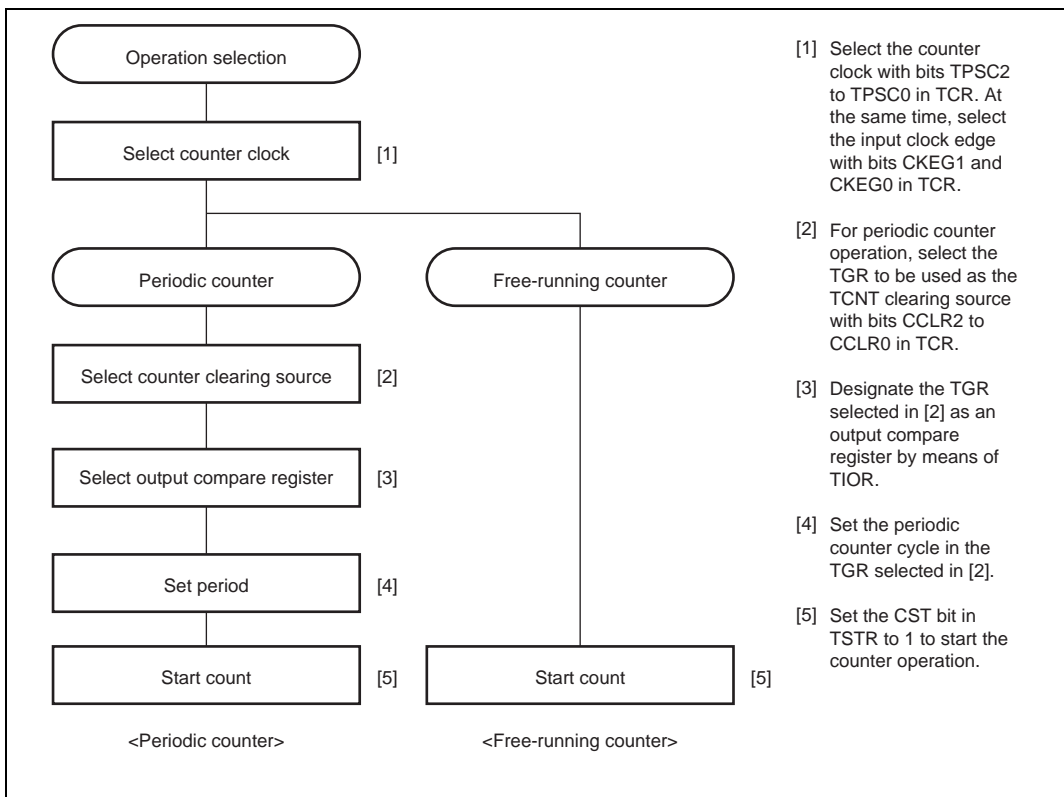


Figure 11.3 Example of Counter Operation Setting Procedure

2. Free-running count operation and periodic count operation

Immediately after a reset, the TPU's TCNT counters are all designated as free-running counters. When the relevant bit in TSTR is set to 1 the corresponding TCNT counter starts up-count operation as a free-running counter. When TCNT overflows (changes from H'FFFF to H'0000), the TCFV bit in TSR is set to 1. If the value of the corresponding TCIEV bit in TIER is 1 at this point, the TPU requests an interrupt. After overflow, TCNT starts counting up again from H'0000.

Figure 11.4 illustrates free-running counter operation.

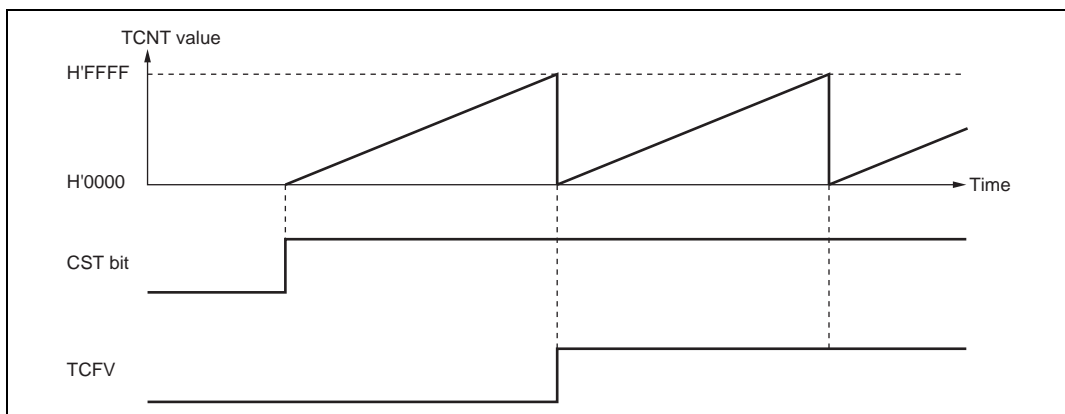


Figure 11.4 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The 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 CCLR2 to CCLR0 in TCR. After the settings have been made, TCNT starts count-up operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the count value matches the value in TGR, the TGF bit in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding TGIE bit in TIER is 1 at this point, the TPU requests an interrupt. After a compare match, TCNT starts counting up again from H'0000.

Figure 11.5 illustrates periodic counter operation.

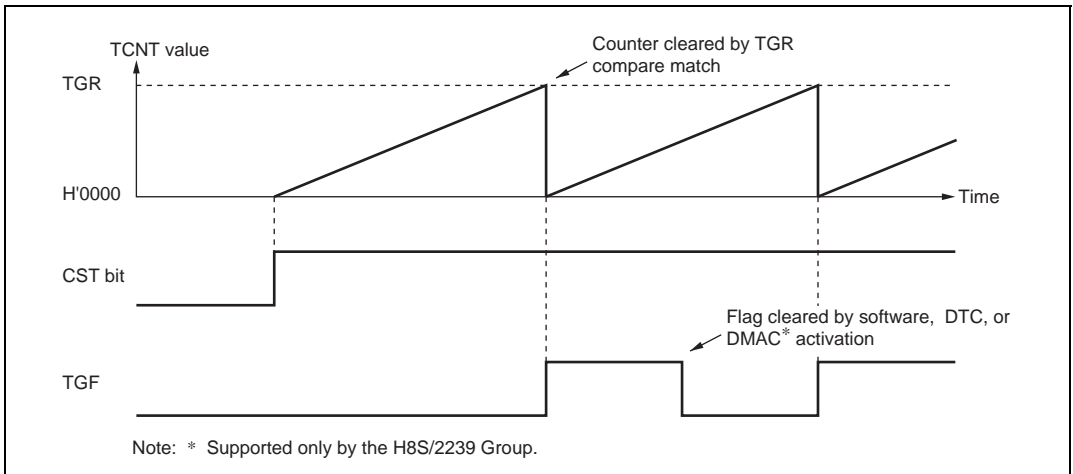


Figure 11.5 Periodic Counter Operation

Waveform Output by Compare Match: The TPU can perform 0, 1, or toggle output from the corresponding output pin using a compare match.

1. Example of setting procedure for waveform output by compare match

Figure 11.6 shows an example of the setting procedure for waveform output by a compare match.

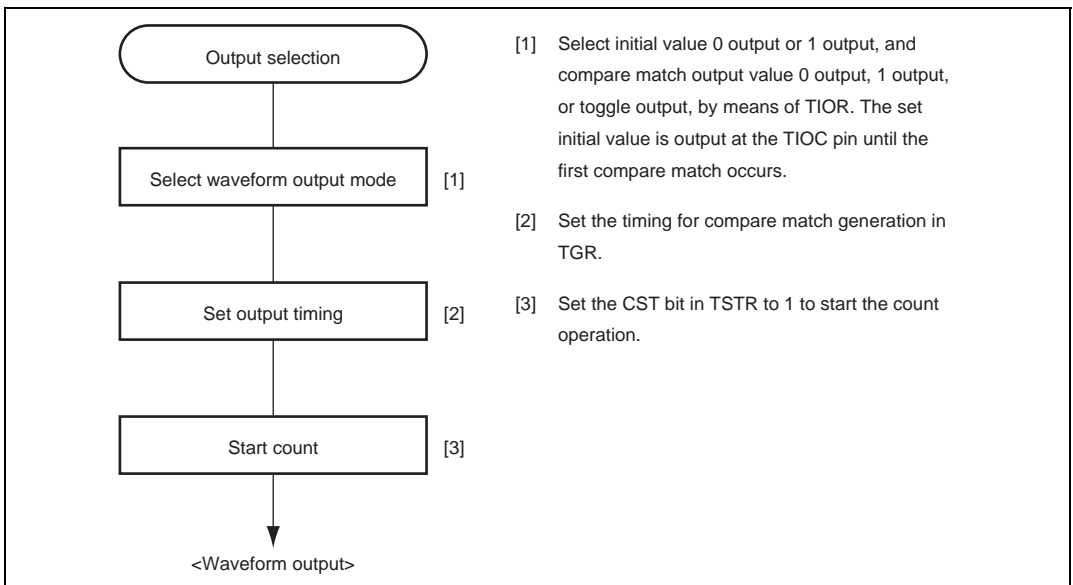


Figure 11.6 Example of Setting Procedure for Waveform Output by Compare Match

2. Examples of waveform output operation

Figure 11.7 shows an example of 0 output/1 output.

In this example, TCNT has been designated as a free-running counter, and settings have been made so that 1 is output by compare match A, and 0 is output by compare match B. When the set level and the pin level match, the pin level does not change.

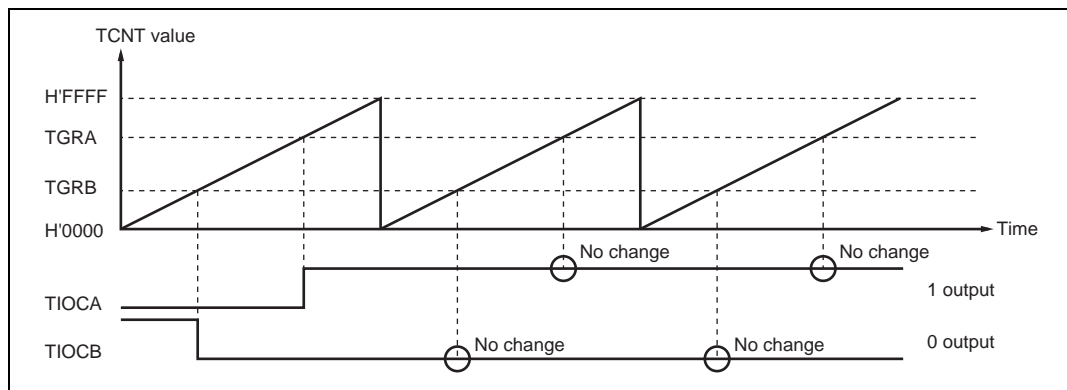


Figure 11.7 Example of 0 Output/1 Output Operation

Figure 11.8 shows an example of toggle output.

In this example 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 both compare match A and compare match B.

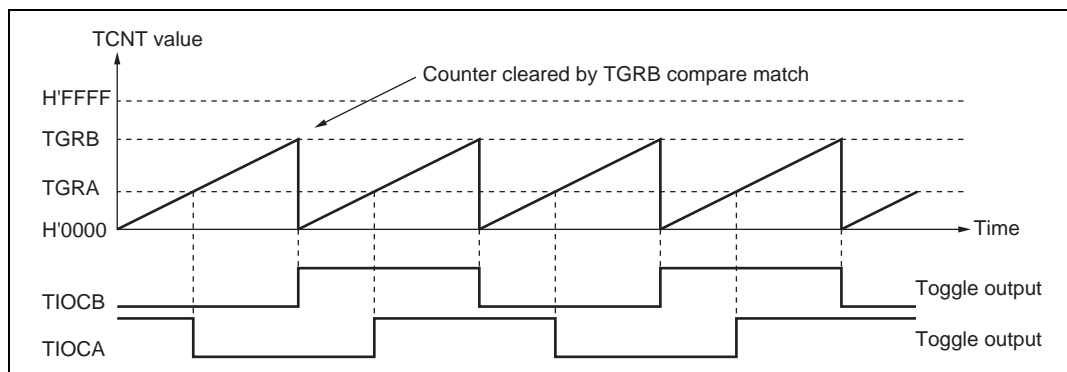


Figure 11.8 Example of Toggle Output Operation

Input Capture Function: The TCNT value can be transferred to TGR on detection of the TIOC pin input edge.

Rising edge, falling edge, or both edges can be selected as the detection edge. For channels 0, 1, 3*, and 4*, it is also possible to specify another channel's counter input clock or compare match signal as the input capture source.

Notes: When another channel's counter input clock is used as the input capture input for channels 0 and 3, $\phi/1$ should not be selected as the counter input clock used for input capture input. Input capture will not be generated if $\phi/1$ is selected.

* Not available in the H8S/2227 Group.

1. Example of setting procedure for input capture operation

Figure 11.9 shows an example of the setting procedure for input capture operation.

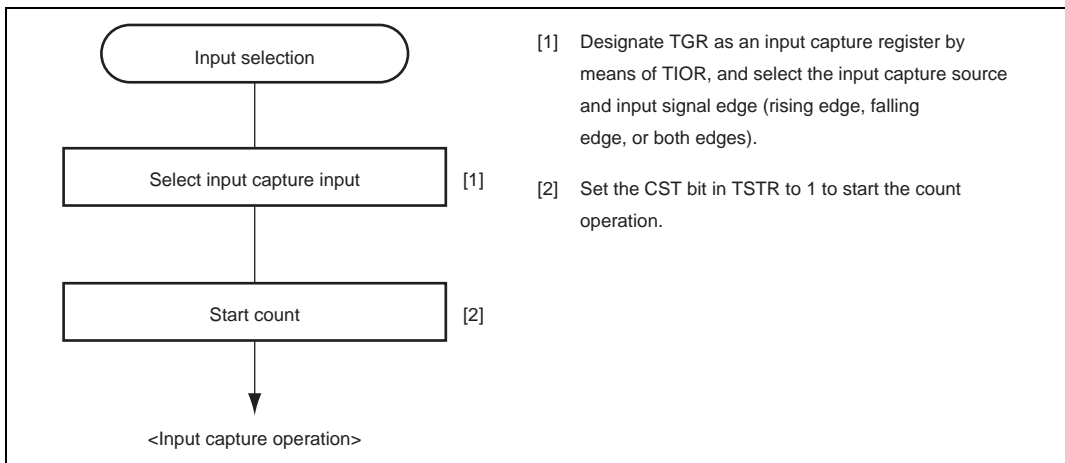


Figure 11.9 Example of Setting Procedure for Input Capture Operation

2. Example of input capture operation

Figure 11.10 shows an example of input capture operation.

In this example both rising and falling edges have been selected as the TIOCA pin input capture input edge, falling edge has been selected as the TIOCB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT.

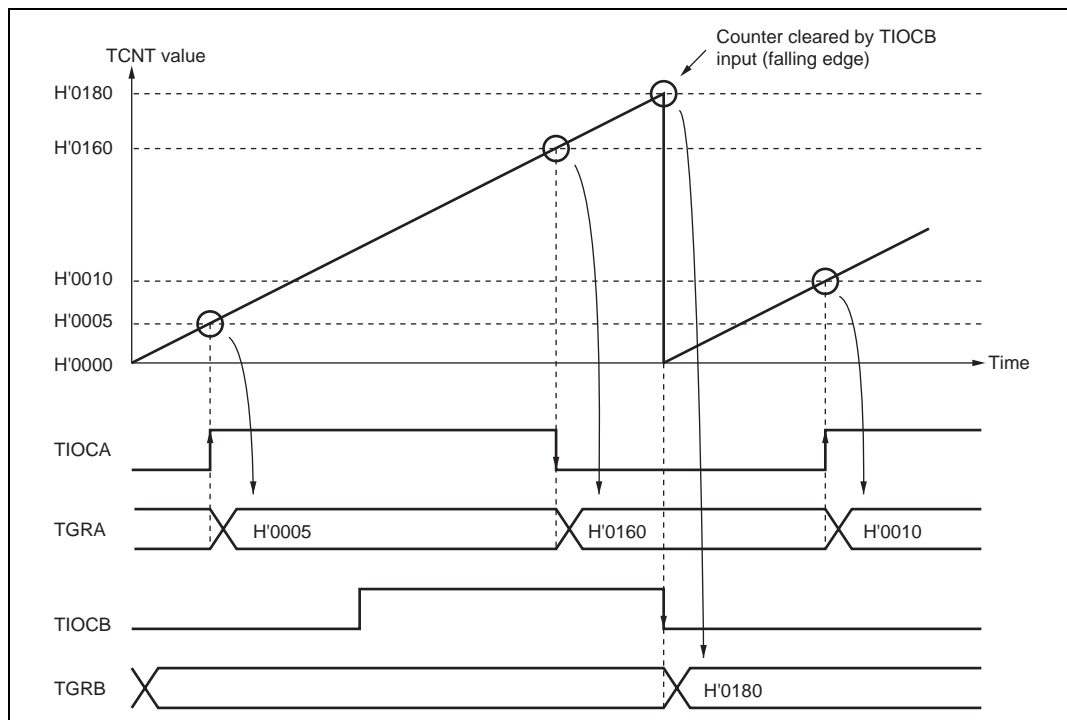


Figure 11.10 Example of Input Capture Operation

11.4.2 Synchronous Operation

In synchronous operation, the values in multiple TCNT counters can be rewritten simultaneously (synchronous presetting). Also, multiple of TCNT counters can be cleared simultaneously (synchronous clearing) by making the appropriate setting in TCR.

Synchronous operation enables TGR to be incremented with respect to a single time base.

Channels 0 to 2 (H8S/2227 Group) or 0 to 5 (groups other than H8S/2227) can all be designated for synchronous operation.

Example of Synchronous Operation Setting Procedure: Figure 11.11 shows an example of the synchronous operation setting procedure.

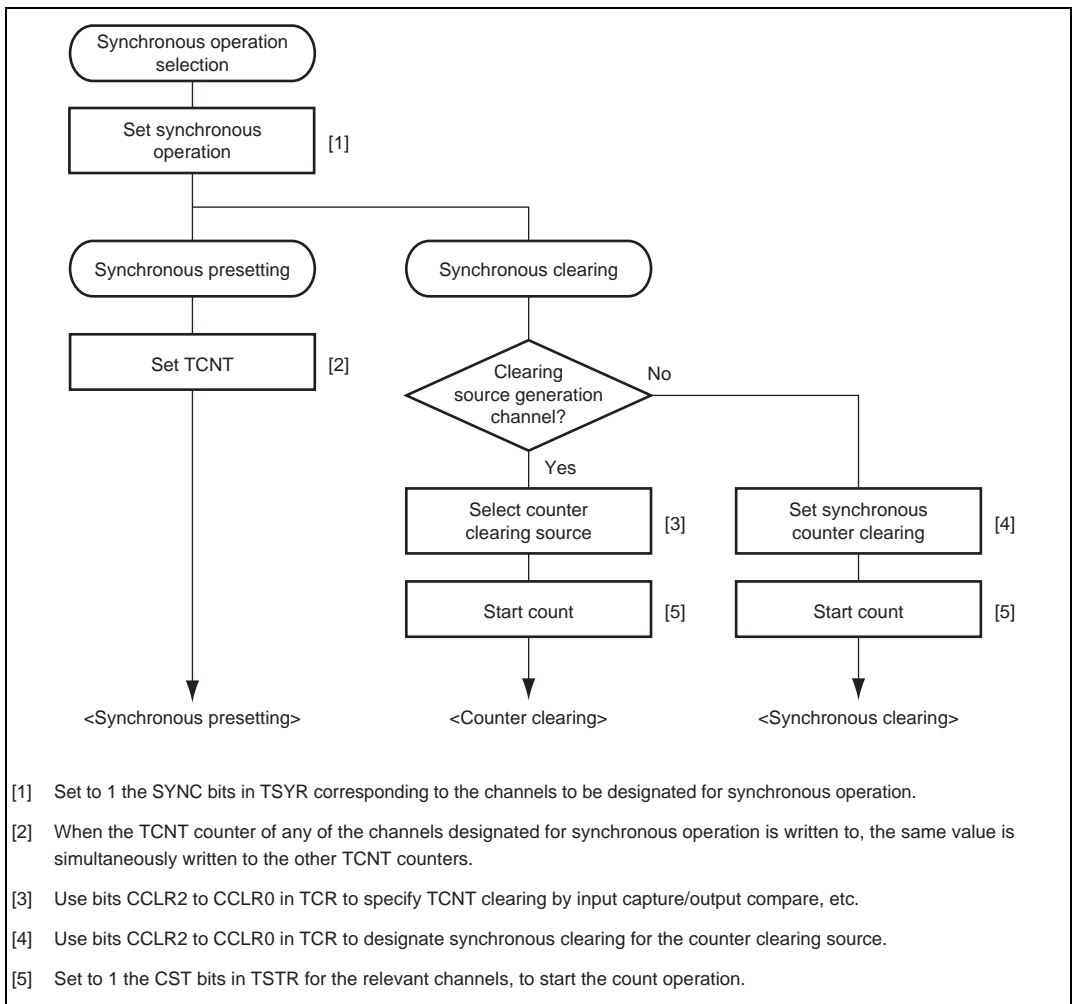


Figure 11.11 Example of Synchronous Operation Setting Procedure

Example of Synchronous Operation: Figure 11.12 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for channels 0 to 2, TGRB_0 compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 and 2 counter clearing source.

Three-phase PWM waveforms are output from pins TIOCA2, TIOCA1, and TIOCA0. At this time, synchronous presetting, and synchronous clearing by TGRB_0 compare match, is performed for channel 0 to 2 TCNT counters, and the data set in TGRB_0 is used as the PWM cycle.

For details on PWM modes, see section 11.4.5, PWM Modes.

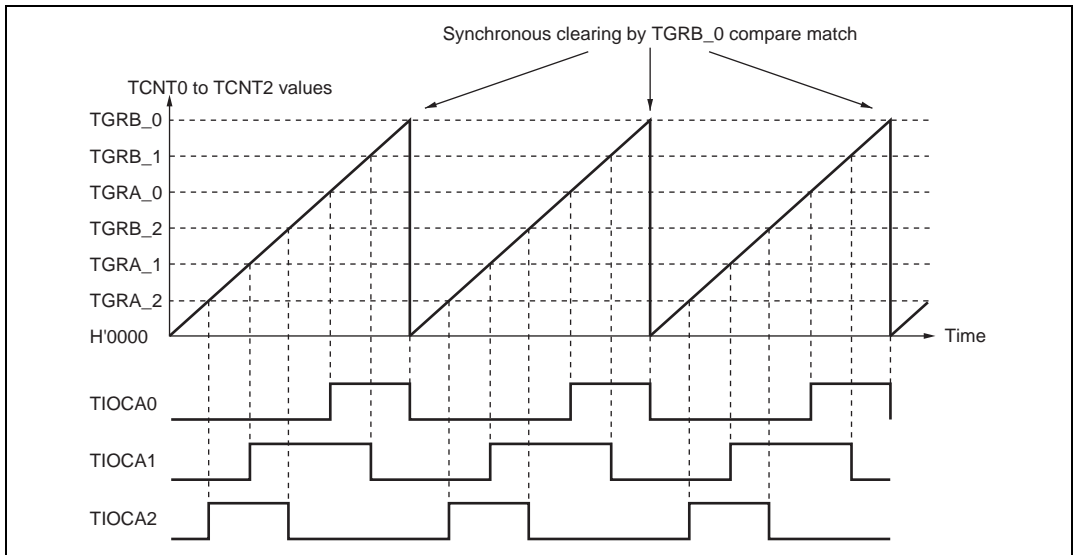


Figure 11.12 Example of Synchronous Operation

11.4.3 Buffer Operation

Buffer operation, provided for channels 0 and 3, enables TGRC and TGRD to be used as buffer registers.

Buffer operation differs depending on whether TGR has been designated as an input capture register or a compare match register.

Table 11.28 shows the register combinations used in buffer operation.

Table 11.28 Register Combinations in Buffer Operation

| Channel | Timer General Register | Buffer Register |
|---------|------------------------|-----------------|
| 0 | TGRA_0 | TGRC_0 |
| | TGRB_0 | TGRD_0 |
| 3* | TGRA_3 | TGRC_3 |
| | TGRB_3 | TGRD_3 |

Note: * Not available in the H8S/2227 Group.

- When TGR is an output compare register

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

This operation is illustrated in figure 11.13.

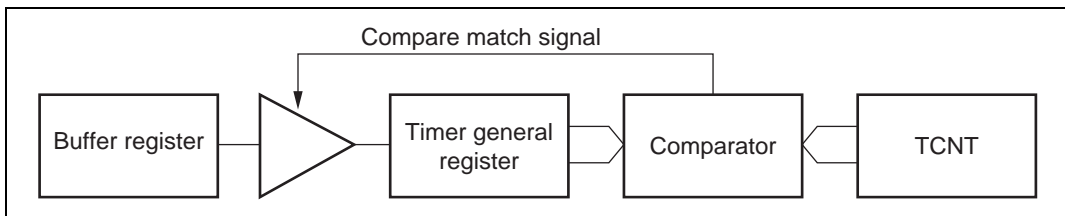


Figure 11.13 Compare Match Buffer Operation

- When TGR is an input capture register

When input capture occurs, the value in TCNT is transferred to TGR and the value previously held in the timer general register is transferred to the buffer register.

This operation is illustrated in figure 11.14.

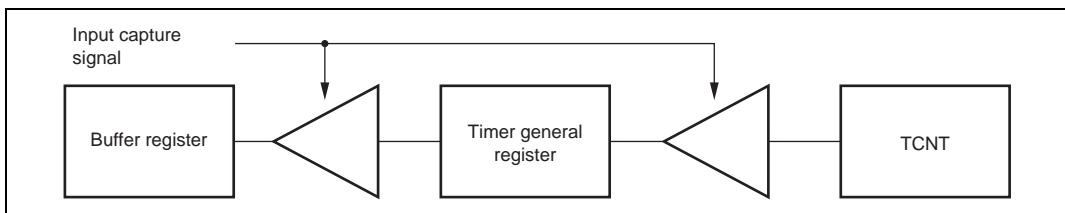


Figure 11.14 Input Capture Buffer Operation

Example of Buffer Operation Setting Procedure: Figure 11.15 shows an example of the buffer operation setting procedure.

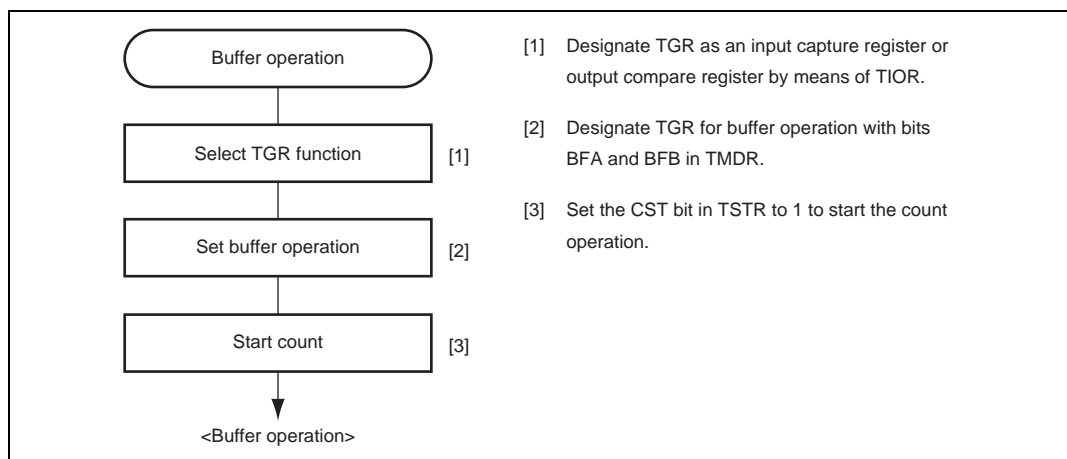


Figure 11.15 Example of Buffer Operation Setting Procedure

Examples of Buffer Operation:

1. When TGR is an output compare register

Figure 11.16 shows an operation example in which PWM mode 1 has been designated for channel 0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, 1 output at compare match A, and 0 output at compare match B.

As buffer operation has been set, when compare match A occurs the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time compare match A occurs.

For details on PWM modes, see section 11.4.5, PWM Modes.

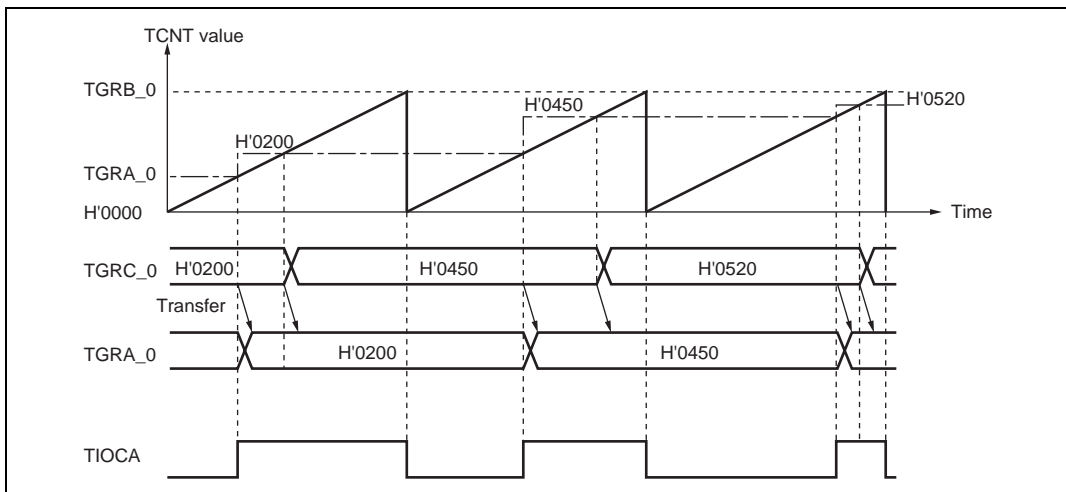


Figure 11.16 Example of Buffer Operation (1)

2. When TGR is an input capture register

Figure 11.17 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the TIOCA pin input capture input edge.

As buffer operation has been set, when the TCNT value is stored in TGRA upon occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

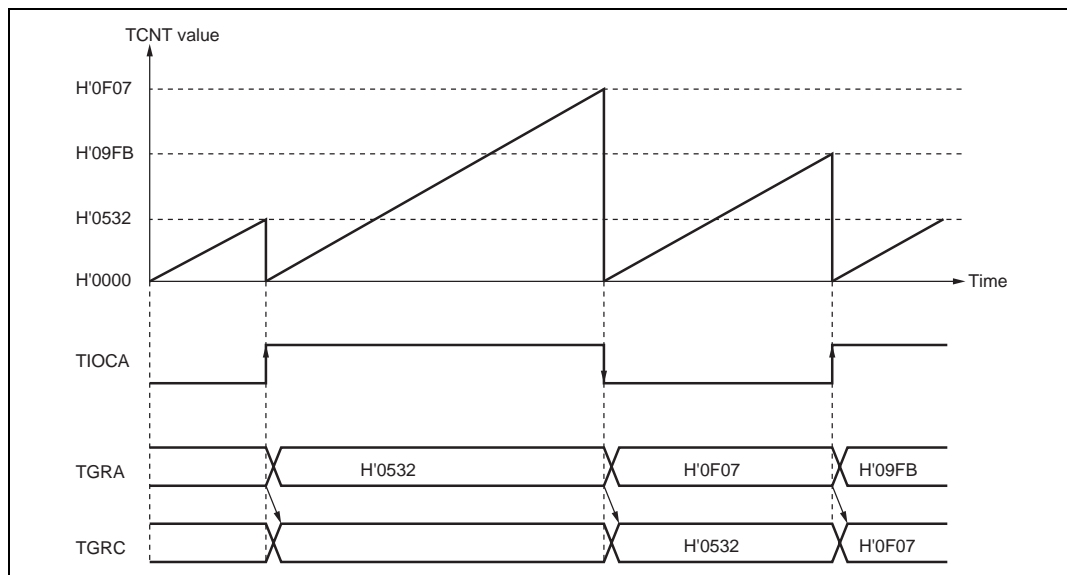


Figure 11.17 Example of Buffer Operation (2)

11.4.4 Cascaded Operation

In cascaded operation*, two 16-bit counters for different channels are used together as a 32-bit counter.

This function works by counting the channel 1 (channel 4) counter clock at overflow/underflow of TCNT_2 (TCNT_5) as set in bits TPSC2 to TPSC0 in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

Table 11.29 shows the register combinations used in cascaded operation.

Notes: When phase counting mode is set for channel 1 or 4, the counter clock setting is invalid and the counter operates independently in phase counting mode.

* Not available in the H8S/2227 Group.

Table 11.29 Cascaded Combinations

| Combination | Upper 16 Bits | Lower 16 Bits |
|------------------|---------------|---------------|
| Channels 1 and 2 | TCNT_1 | TCNT_2 |
| Channels 4 and 5 | TCNT_4 | TCNT_5 |

Example of Cascaded Operation Setting Procedure: Figure 11.18 shows an example of the setting procedure for cascaded operation.

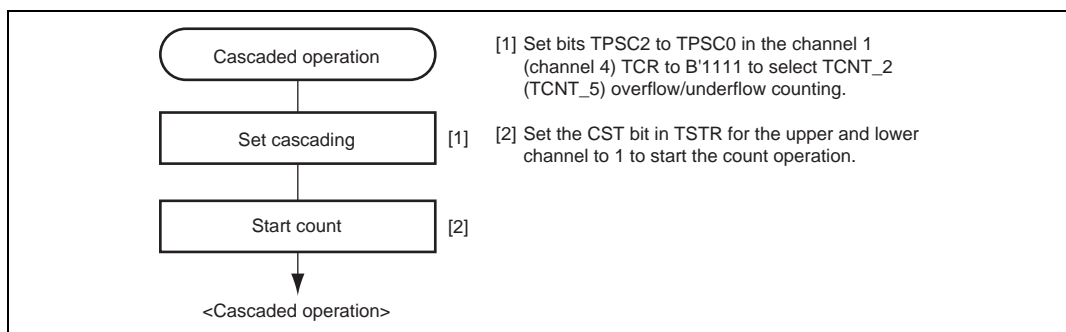


Figure 11.18 Cascaded Operation Setting Procedure

Examples of Cascaded Operation: Figure 11.19 illustrates the operation when counting upon TCNT_2 overflow/underflow has been set for TCNT_1, TGRA_1 and TGRA_2 have been designated as input capture registers, and the TIOC pin rising edge has been selected.

When a rising edge is input to the TIOCA1 and TIOCA2 pins simultaneously, the upper 16 bits of the 32-bit data are transferred to TGRA_1, and the lower 16 bits to TGRA_2.

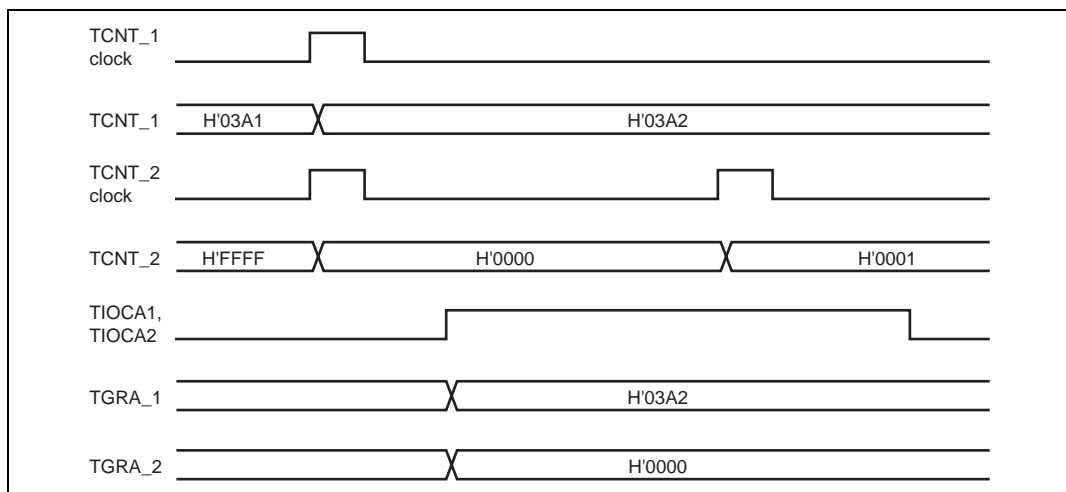


Figure 11.19 Example of Cascaded Operation (1)

Figure 11.20 illustrates the operation when counting upon TCNT_2 overflow/underflow has been set for TCNT_1, and phase counting mode has been designated for channel 2.

TCNT_1 is incremented by TCNT_2 overflow and decremented by TCNT_2 underflow.

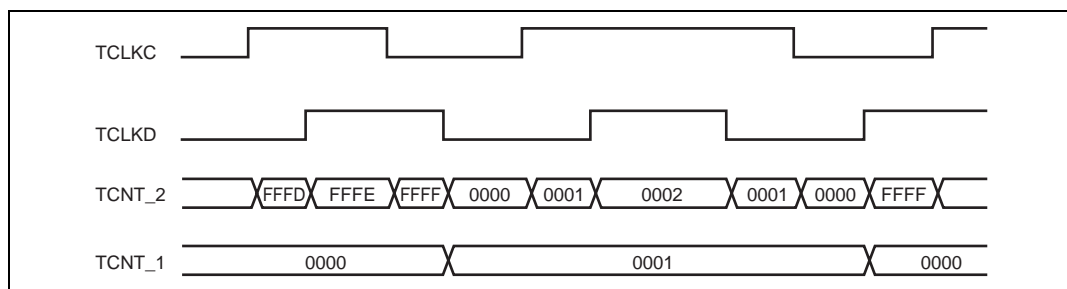


Figure 11.20 Example of Cascaded Operation (2)

11.4.5 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. 0, 1, or toggle output can be selected as the output level in response to compare match of each TGR.

Settings of TGR registers can output a PWM waveform in the range of 0% to 100% duty cycle.

Designating TGR 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. Synchronous operation is also possible.

There are two PWM modes, as described below.

- **PWM mode 1**

PWM output is generated from the TIOCA and TIOCC pins by pairing TGRA with TGRB and TGRC with TGRD. The outputs specified by bits IOA3 to IOA0 and IOC3 to IOC0 in TIOR are output from the TIOCA and TIOCC pins at compare matches A and C, respectively. The outputs specified by bits IOB3 to IOB0 and IOD3 to IOD0 in TIOR are output at compare matches B and D, respectively. The initial output value is the value set in TGRA or TGRC. If the set values of paired TGRs are identical, the output value does not change when a compare match occurs.

In PWM mode 1, a maximum 8-phase PWM output is possible.

- **PWM mode 2**

PWM output is generated using one TGR as the cycle register and the others as duty cycle registers. The output specified in TIOR is performed by means of compare matches. Upon counter clearing by a synchronization register compare match, the output value of each pin is the initial value set in TIOR. If the set values of the cycle and duty cycle registers are identical, the output value does not change when a compare match occurs.

In PWM mode 2, a maximum 15-phase PWM output is possible by combined use with synchronous operation.

The correspondence between PWM output pins and registers is shown in table 11.30.

Table 11.30 PWM Output Registers and Output Pins

| Channel | Registers | Output Pins | |
|---------|-----------|-------------|------------|
| | | PWM Mode 1 | PWM Mode 2 |
| 0 | TGRA_0 | TIOCA0 | TIOCA0 |
| | TGRB_0 | | TIOCB0 |
| | TGRC_0 | TIOCC0 | TIOCC0 |
| | TGRD_0 | | TIOCD0 |
| 1 | TGRA_1 | TIOCA1 | TIOCA1 |
| | TGRB_1 | | TIOCB1 |
| 2 | TGRA_2 | TIOCA2 | TIOCA2 |
| | TGRB_2 | | TIOCB2 |
| 3* | TGRA_3 | TIOCA3 | TIOCA3 |
| | TGRB_3 | | TIOCB3 |
| | TGRC_3 | TIOCC3 | TIOCC3 |
| | TGRD_3 | | TIOCD3 |
| 4* | TGRA_4 | TIOCA4 | TIOCA4 |
| | TGRB_4 | | TIOCB4 |
| 5* | TGRA_5 | TIOCA5 | TIOCA5 |
| | TGRB_5 | | TIOCB5 |

Notes: In PWM mode 2, PWM output is not possible for the TGR register in which the cycle is set.

* Not available in the H8S/2227 Group.

Example of PWM Mode Setting Procedure: Figure 11.21 shows an example of the PWM mode setting procedure.

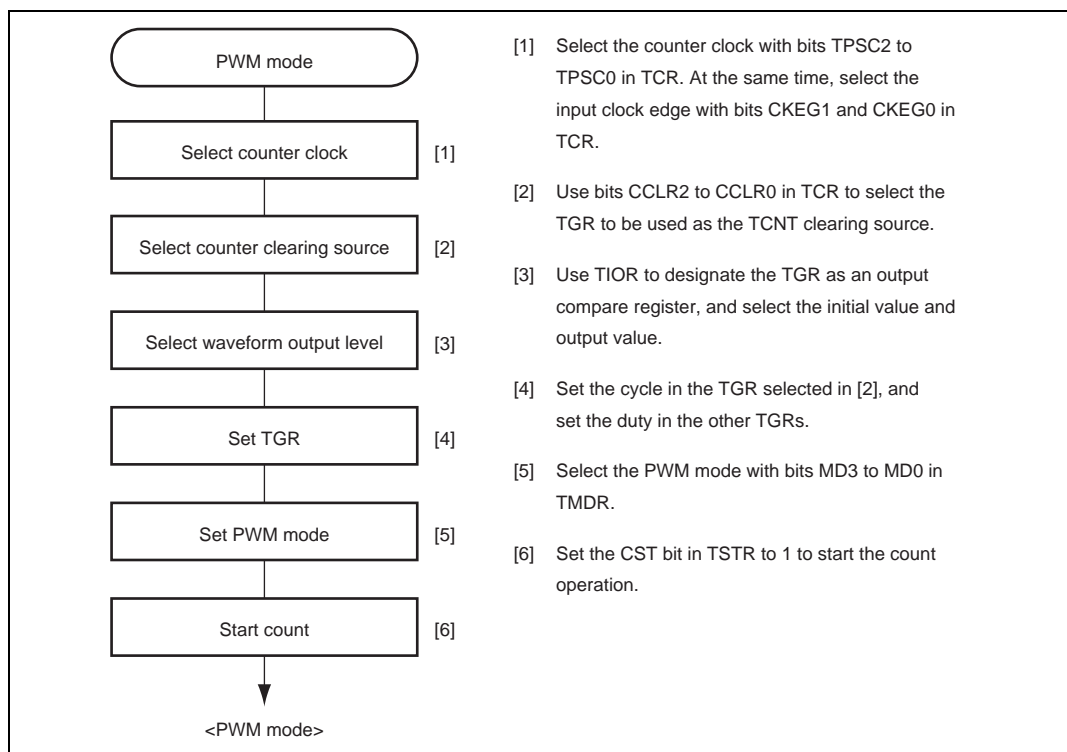


Figure 11.21 Example of PWM Mode Setting Procedure

Examples of PWM Mode Operation: Figure 11.22 shows an example of PWM mode 1 operation.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set for the TGRA initial output value and output value, and 1 is set as the TGRB output value.

In this case, the value set in TGRA is used as the cycle, and the values set in TGRB registers as the duty cycle.

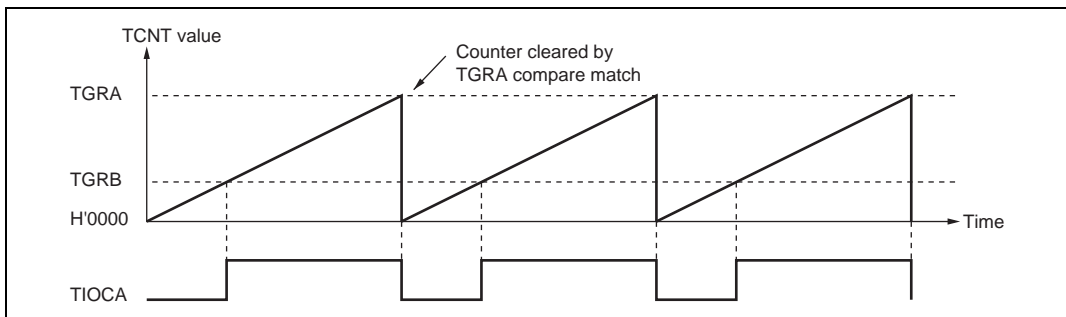


Figure 11.22 Example of PWM Mode Operation (1)

Figure 11.23 shows an example of PWM mode 2 operation.

In this example, synchronous operation is designated for channels 0 and 1, TGRB_1 compare match is set as the TCNT clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGR registers (TGRA_0 to TGRD_0, TGRA_1), to output a 5-phase PWM waveform.

In this case, the value set in TGRB_1 is used as the cycle, and the values set in the other TGRs as the duty cycle.

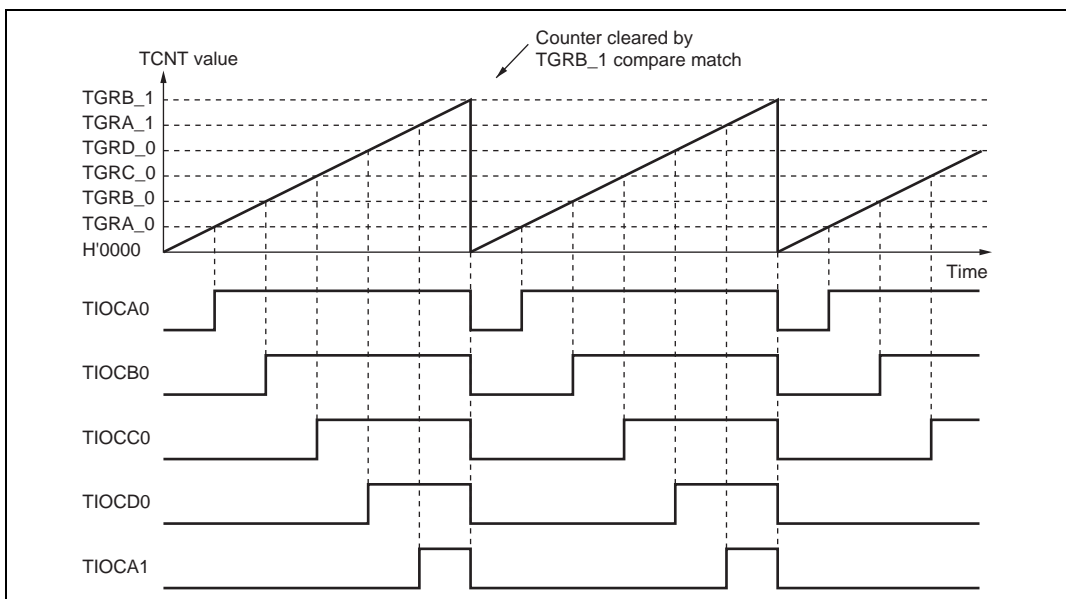


Figure 11.23 Example of PWM Mode Operation (2)

Figure 11.24 shows examples of PWM waveform output with 0% duty cycle and 100% duty cycle in PWM mode.

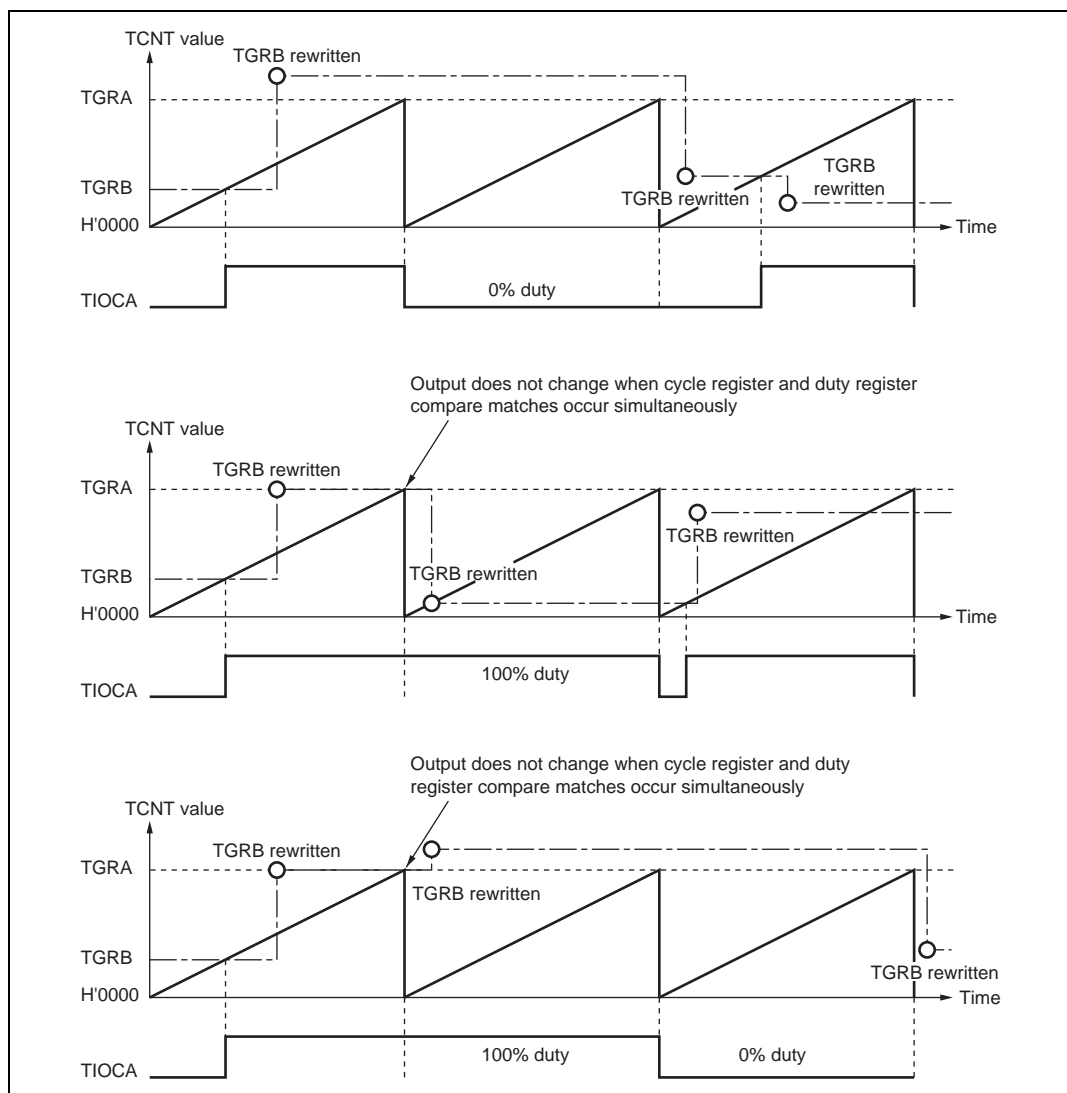


Figure 11.24 Example of PWM Mode Operation (3)

11.4.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and TCNT is incremented/decremented accordingly. In the H8S/2227 Group, this mode can be set for channels 1 and 2. In other groups, it can be set for channels 1, 2, 4, and 5.

When phase counting mode is set, an external clock is selected as the counter input clock and TCNT operates as an up/down-counter regardless of the setting of bits TPSC2 to TPSC0 and bits CKEG1 and CKEG0 in TCR. However, the functions of bits CCLR1 and CCLR0 in TCR, and of TIOR, TIER, and TGR are valid, and input capture/compare match and interrupt functions can be used.

This can be used for two-phase encoder pulse input.

When overflow occurs while TCNT is counting up, the TCFV flag in TSR is set; when underflow occurs while TCNT is counting down, the TCFU flag is set.

The TCFD bit in TSR is the count direction flag. Reading the TCFD flag provides an indication of whether TCNT is counting up or down.

Table 11.31 shows the correspondence between external clock pins and channels.

Table 11.31 Clock Input Pins in Phase Counting Mode

| Channels | External Clock Pins | |
|--|---------------------|---------|
| | A-Phase | B-Phase |
| When channel 1 or 5* is set to phase counting mode | TCLKA | TCLKB |
| When channel 2 or 4* is set to phase counting mode | TCLKC | TCLKD |

Note: * Not available in the H8S/2227 Group.

Example of Phase Counting Mode Setting Procedure: Figure 11.25 shows an example of the phase counting mode setting procedure.

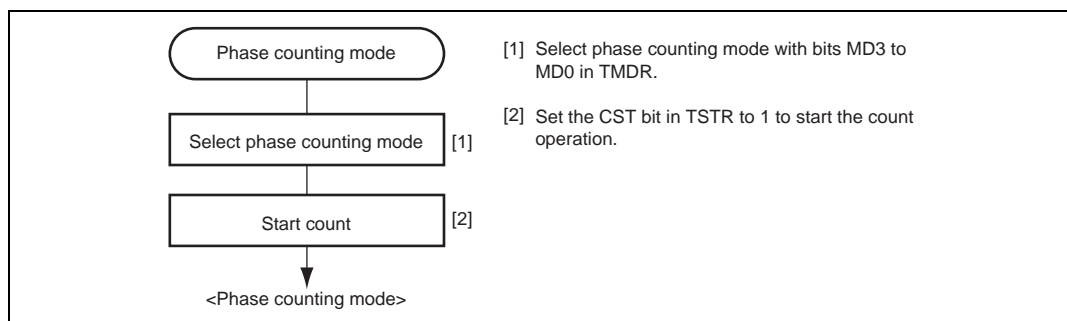


Figure 11.25 Example of Phase Counting Mode Setting Procedure

Examples of Phase Counting Mode Operation: In phase counting mode, TCNT counts up or down according to the phase difference between two external clocks. There are four modes, according to the count conditions.

1. Phase counting mode 1

Figure 11.26 shows an example of phase counting mode 1 operation, and table 11.32 summarizes the TCNT up/down-count conditions.

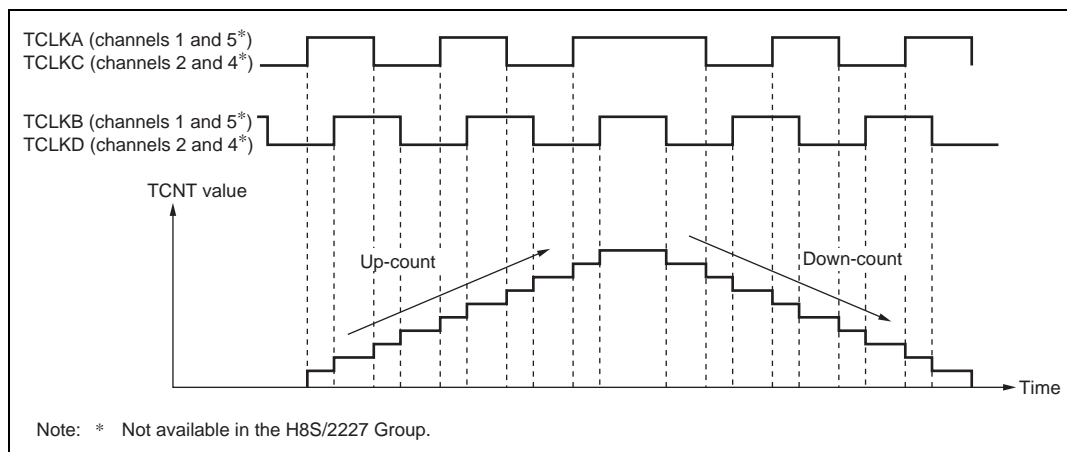











Figure 11.26 Example of Phase Counting Mode 1 Operation

Table 11.32 Up/Down-Count Conditions in Phase Counting Mode 1

| TCLKA (Channels 1 and 5*) TCLKC (Channels 2 and 4*) | TCLKB (Channels 1 and 5*) TCLKD (Channels 2 and 4*) | Operation |
|---|---|------------------|
| High level |  | Up-count |
| Low level |  | |
|  | Low level | |
|  | High level | |
| High level |  | Down-count |
| Low level |  | |
|  | High level | |
|  | Low level | |

Legend:

: Rising edge: Falling edge

Note: * Not available in the H8S/2227 Group.

2. Phase counting mode 2

Figure 11.27 shows an example of phase counting mode 2 operation, and table 11.33 summarizes the TCNT up/down-count conditions.

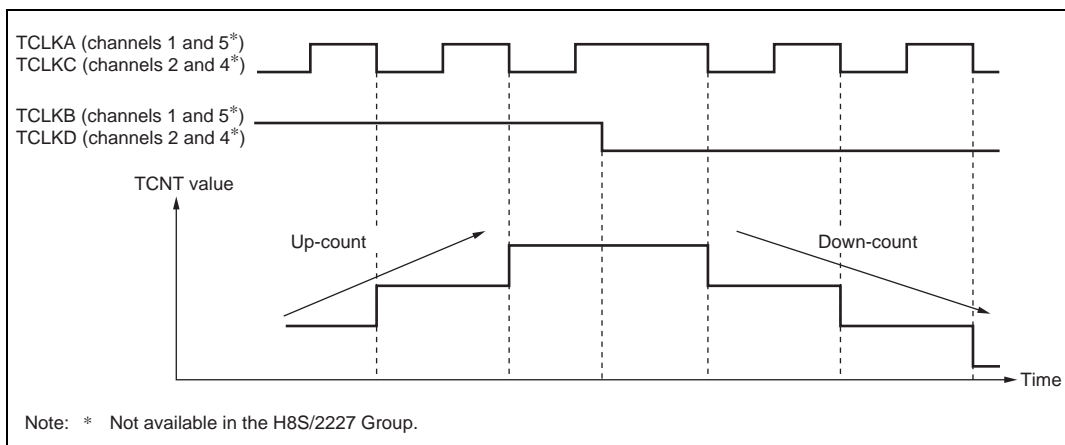


Figure 11.27 Example of Phase Counting Mode 2 Operation

Table 11.33 Up/Down-Count Conditions in Phase Counting Mode 2

| TCLKA (Channels 1 and 5*) TCLKC (Channels 2 and 4*) | TCLKB (Channels 1 and 5*) TCLKD (Channels 2 and 4*) | Operation |
|--|--|------------|
| High level | | Don't care |
| Low level | | |
| | Low level | Up-count |
| | High level | |
| High level | | Don't care |
| Low level | | |
| | High level | Down-count |
| | Low level | |

Legend:

: Rising edge

: Falling edge

Note: * Not available in the H8S/2227 Group.

3. Phase counting mode 3

Figure 11.28 shows an example of phase counting mode 3 operation, and table 11.34 summarizes the TCNT up/down-count conditions.

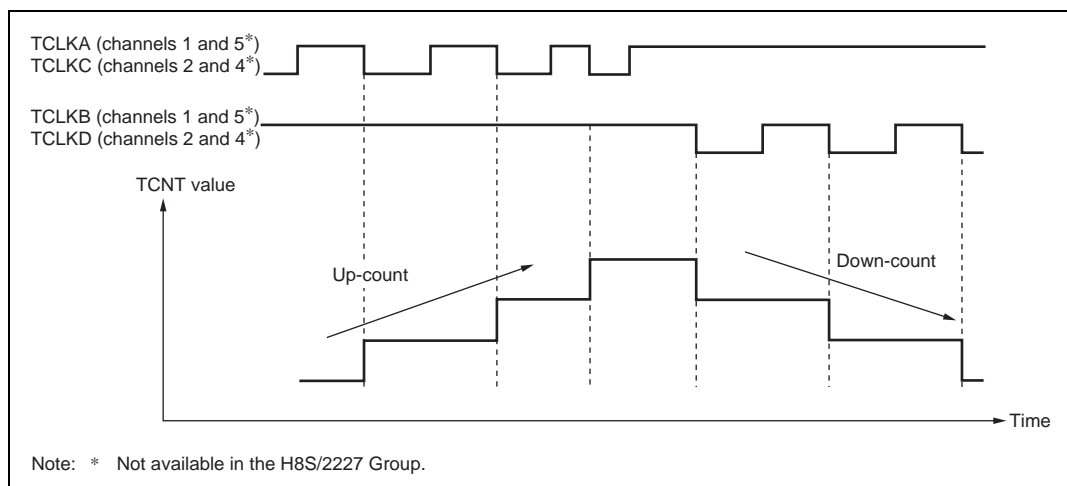


Figure 11.28 Example of Phase Counting Mode 3 Operation

Table 11.34 Up/Down-Count Conditions in Phase Counting Mode 3

| TCLKA (Channels 1 and 5*) TCLKC (Channels 2 and 4*) | TCLKB (Channels 1 and 5*) TCLKD (Channels 2 and 4*) | Operation |
|--|--|------------|
| High level | | Don't care |
| Low level | | Don't care |
| | Low level | Up-count |
| | High level | Up-count |
| High level | | Down-count |
| Low level | | Down-count |
| | High level | Don't care |
| | Low level | Don't care |

Legend:

: Rising edge

: Falling edge

Note: * Not available in the H8S/2227 Group.

4. Phase counting mode 4

Figure 11.29 shows an example of phase counting mode 4 operation, and table 11.35 summarizes the TCNT up/down-count conditions.

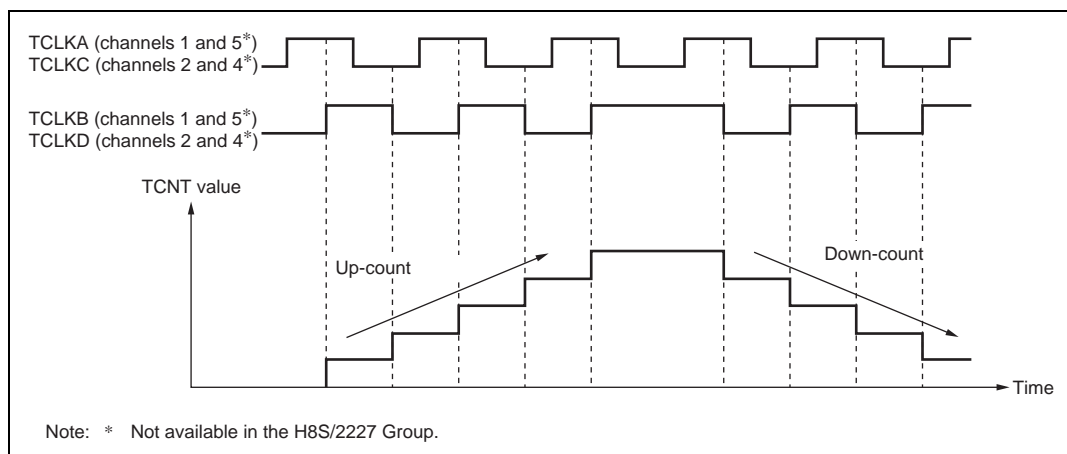


Figure 11.29 Example of Phase Counting Mode 4 Operation

Table 11.35 Up/Down-Count Conditions in Phase Counting Mode 4

| TCLKA (Channels 1 and 5*) TCLKC (Channels 2 and 4*) | TCLKB (Channels 1 and 5*) TCLKD (Channels 2 and 4*) | Operation |
|--|--|------------|
| High level | | Up-count |
| Low level | | |
| | Low level | Don't care |
| | High level | |
| High level | | Down-count |
| Low level | | |
| | High level | Don't care |
| | Low level | |

Legend:

: Rising edge

: Falling edge

Note: * Not available in the H8S/2227 Group.

Phase Counting Mode Application Example: Figure 11.30 shows an example in which phase counting mode is designated for channel 1, and channel 1 is coupled with channel 0 to input servo motor 2-phase encoder pulses in order to detect the position or speed.

Channel 1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to TCLKA and TCLKB.

Channel 0 operates with TCNT counter clearing by TGRC_0 compare match; TGRA_0 and TGRC_0 are used for the compare match function, and are set with the speed control cycle and position control cycle. TGRB_0 is used for input capture, with TGRB_0 and TGRD_0 operating in buffer mode. The channel 1 counter input clock is designated as the TGRB_0 input capture source, and detection of the pulse width of 2-phase encoder 4-multiplication pulses is performed.

TGRA_1 and TGRB_1 for channel 1 are designated for input capture, channel 0 TGRA_0 and TGRC_0 compare matches are selected as the input capture source, and the up/down-counter values for the control cycles are stored.

This procedure enables accurate position/speed detection to be achieved.

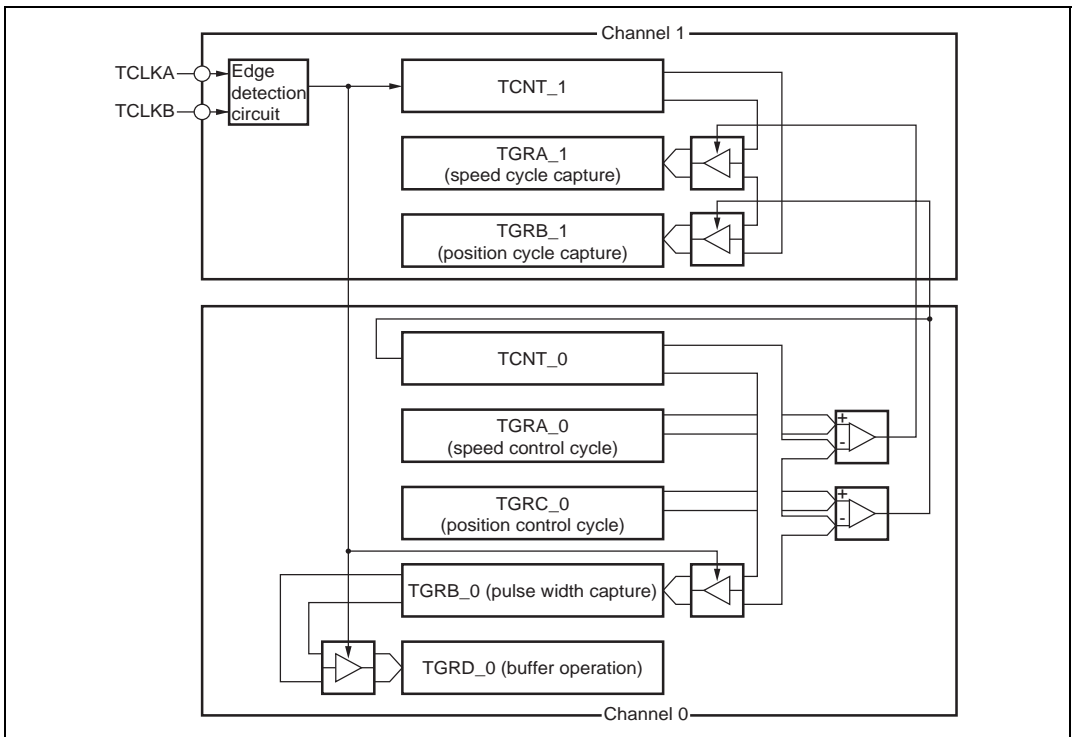


Figure 11.30 Phase Counting Mode Application Example

11.5 Interrupt Sources

There are three kinds of TPU interrupt source: TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own status flag and enable/disable bit, allowing generation of interrupt request signals to be enabled or disabled individually.

When an interrupt request is generated, the corresponding status flag in TSR is set to 1. If the corresponding enable/disable bit in TIER is set to 1 at this time, an interrupt is requested. The interrupt request is cleared by clearing the status flag to 0.

Relative channel priorities can be changed by the interrupt controller, but the priority order within a channel is fixed. For details, see section 5, Interrupt Controller.

Table 11.36 lists the TPU interrupt sources.

Table 11.36 TPU Interrupts

| Channel | Name | Interrupt Source | Interrupt Flag | DTC Activation | DMAC Activation ^{*1} |
|-----------------|-------|------------------------------------|----------------|----------------|-------------------------------|
| 0 | TGI0A | TGRA_0 input capture/compare match | TGFA_0 | Possible | Possible |
| | TGI0B | TGRB_0 input capture/compare match | TGFB_0 | Possible | Not possible |
| | TGI0C | TGRC_0 input capture/compare match | TGFC_0 | Possible | Not possible |
| | TGI0D | TGRD_0 input capture/compare match | TGFD_0 | Possible | Not possible |
| | TGI0V | TCNT_0 overflow | TCFV_0 | Not possible | Not possible |
| 1 | TGI1A | TGRA_1 input capture/compare match | TGFA_1 | Possible | Possible |
| | TGI1B | TGRB_1 input capture/compare match | TGFB_1 | Possible | Not possible |
| | TCI1V | TCNT_1 overflow | TCFV_1 | Not possible | Not possible |
| | TCI1U | TCNT_1 underflow | TCFU_1 | Not possible | Not possible |
| 2 | TGI2A | TGRA_2 input capture/compare match | TGFA_2 | Possible | Possible |
| | TGI2B | TGRB_2 input capture/compare match | TGFB_2 | Possible | Not possible |
| | TCI2V | TCNT_2 overflow | TCFV_2 | Not possible | Not possible |
| | TCI2U | TCNT_2 underflow | TCFU_2 | Not possible | Not possible |
| 3 ^{*2} | TGI3A | TGRA_3 input capture/compare match | TGFA_3 | Possible | Possible |
| | TGI3B | TGRB_3 input capture/compare match | TGFB_3 | Possible | Not possible |
| | TGI3C | TGRC_3 input capture/compare match | TGFC_3 | Possible | Not possible |
| | TGI3D | TGRD_3 input capture/compare match | TGFD_3 | Possible | Not possible |
| | TCI3V | TCNT_3 overflow | TCFV_3 | Not possible | Not possible |
| 4 ^{*2} | TGI4A | TGRA_4 input capture/compare match | TGFA_4 | Possible | Possible |
| | TGI4B | TGRB_4 input capture/compare match | TGFB_4 | Possible | Not possible |
| | TCI4V | TCNT_4 overflow | TCFV_4 | Not possible | Not possible |
| | TCI4U | TCNT_4 underflow | TCFU_4 | Not possible | Not possible |
| 5 ^{*2} | TGI5A | TGRA_5 input capture/compare match | TGFA_5 | Possible | Possible |
| | TGI5B | TGRB_5 input capture/compare match | TGFB_5 | Possible | Not possible |
| | TCI5V | TCNT_5 overflow | TCFV_5 | Not possible | Not possible |
| | TCI5U | TCNT_5 underflow | TCFU_5 | Not possible | Not possible |

Notes: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

1. Supported only by the H8S/2239 Group.
2. Not available in the H8S/2227 Group.

Input Capture/Compare Match Interrupt: An interrupt is requested if the TGIE bit in TIER is set to 1 when the TGF flag in TSR is set to 1 by the occurrence of a TGR input capture/compare match on a particular channel. The interrupt request is cleared by clearing the TGF flag to 0. In the H8S/2227 Group, the TPU has eight input capture/compare match interrupts, four for channel 0 and two each for channels 1 and 2. In other groups, the TPU has 16 input capture/compare match interrupts, four each for channels 0 and 3, and two each for channels 1, 2, 4, and 5.

Overflow Interrupt: An interrupt is requested if the TCIEV bit in TIER is set to 1 when the TCFV flag in TSR is set to 1 by the occurrence of TCNT overflow on a channel. The interrupt request is cleared by clearing the TCFV flag to 0. In the H8S/2227 Group, the TPU has three overflow interrupts, one each for channels 0 to 2. In other groups, the TPU has six overflow interrupts, one each for channels 0 to 5.

Underflow Interrupt: An interrupt is requested if the TCIEU bit in TIER is set to 1 when the TCFU flag in TSR is set to 1 by the occurrence of TCNT underflow on a channel. The interrupt request is cleared by clearing the TCFU flag to 0. The TPU of the H8S/2227 Group has two underflow interrupts, one each for channels 1 and 2. In other groups, the TPU has four underflow interrupts, one each for channels 1, 2, 4, and 5.

11.6 DTC Activation

The DTC can be activated by the TGR input capture/compare match interrupt for a channel. For details, see section 9, Data Transfer Controller (DTC).

In the H8S/2227 Group, a total of eight TPU input capture/compare match interrupts can be used as DTC activation sources, four for channel 0 and two each for channels 1 and 2. In other groups, a total of 16 TPU input capture/compare match interrupts can be used as DTC activation sources, four each for channels 0 and 3, and two each for channels 1, 2, 4, and 5.

11.7 DMAC Activation (H8S/2239 Group Only)

The DMAC can be activated by the TGRA input capture/compare match interrupt for a channel. For details, see section 8, DMA Controller (DMAC).

In the TPU, a total of six TGRA input capture/compare match interrupts can be used as DMAC activation sources, one for each channel.

11.8 A/D Converter Activation

The A/D converter can be activated by the TGRA input capture/compare match for a channel.

If the TTGE bit in TIER is set to 1 when the TGFA flag in TSR is set to 1 by the occurrence of a TGRA input capture/compare match on a particular channel, a request to start A/D conversion is sent to the A/D converter. If the TPU conversion start trigger has been selected on the A/D converter side at this time, A/D conversion is started.

In the TPU, a total of six TGRA input capture/compare match interrupts can be used as A/D converter conversion start sources, one for each channel.

11.9 Operation Timing

11.9.1 Input/Output Timing

TCNT Count Timing: Figure 11.31 shows TCNT count timing in internal clock operation, and figure 11.32 shows TCNT count timing in external clock operation.

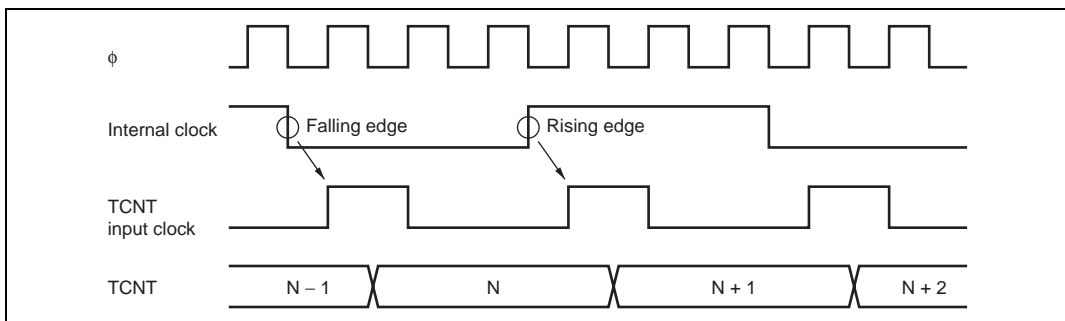


Figure 11.31 Count Timing in Internal Clock Operation

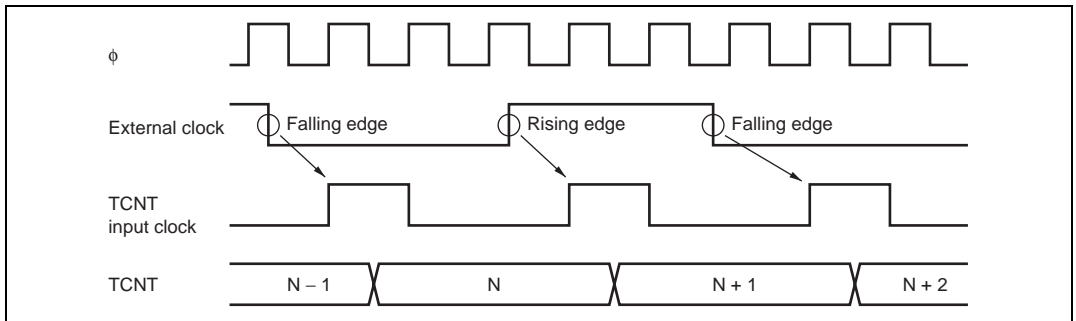


Figure 11.32 Count Timing in External Clock Operation

Output Compare Output Timing: A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the output value set in TIOR is output at the output compare output pin. After a match between TCNT and TGR, the compare match signal is not generated until the (TIOC pin) TCNT input clock is generated.

Figure 11.33 shows output compare output timing.

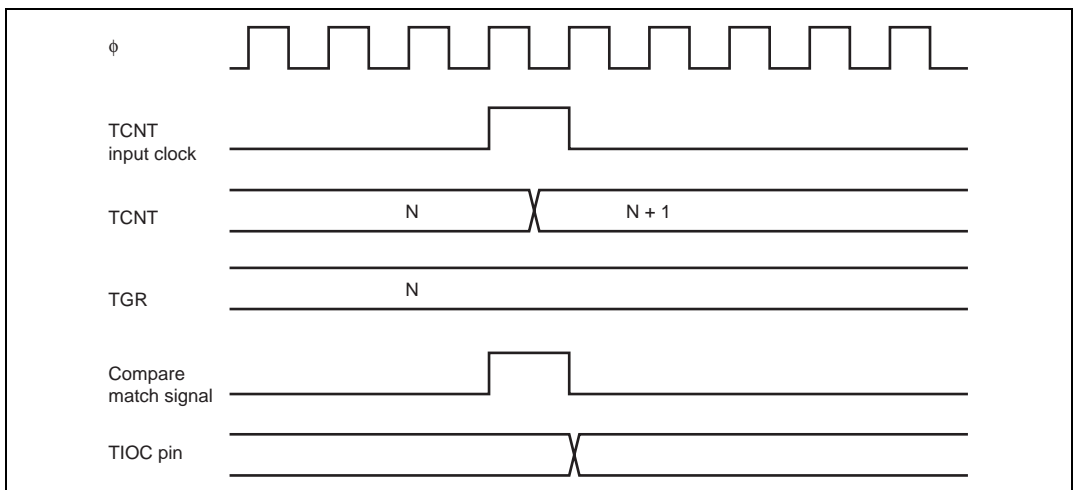


Figure 11.33 Output Compare Output Timing

Input Capture Signal Timing: Figure 11.34 shows input capture signal timing.

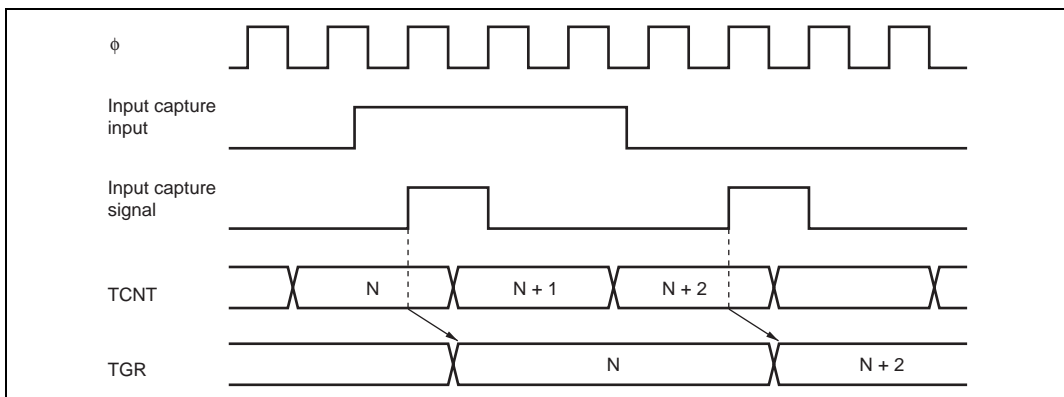


Figure 11.34 Input Capture Input Signal Timing

Timing for Counter Clearing by Compare Match/Input Capture: Figure 11.35 shows the timing when counter clearing by compare match occurrence is specified, and figure 11.36 shows the timing when counter clearing by input capture occurrence is specified.

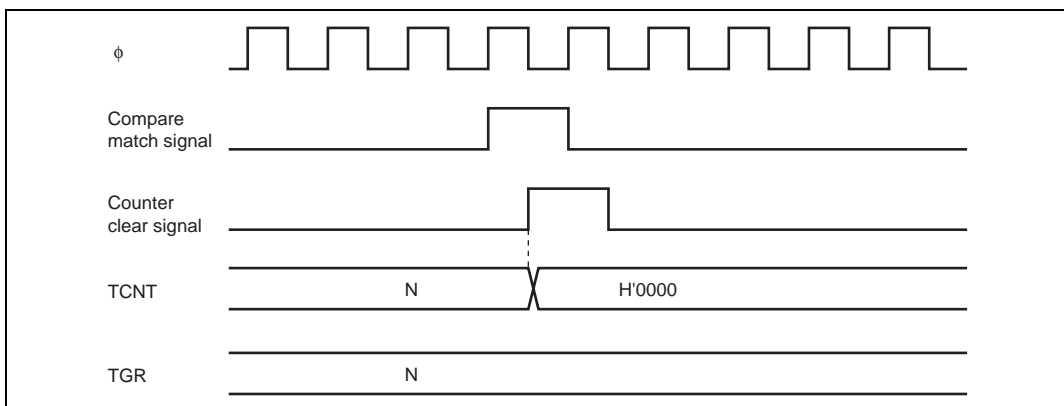


Figure 11.35 Counter Clear Timing (Compare Match)

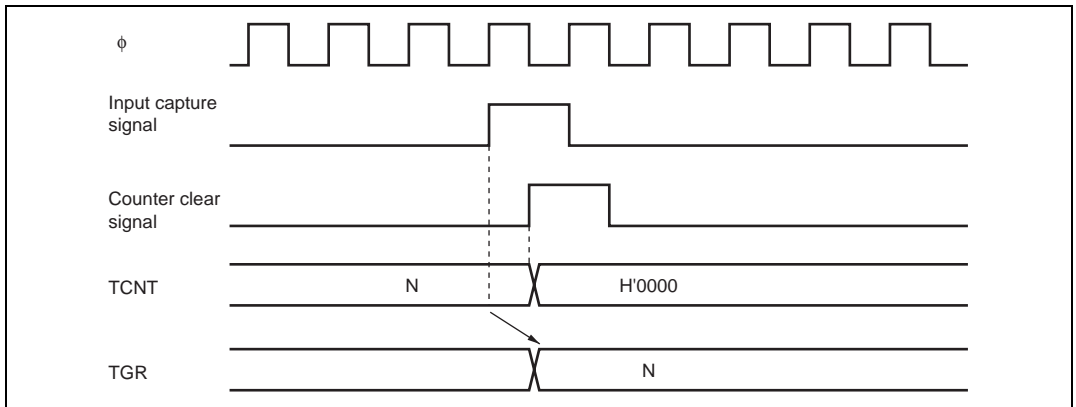


Figure 11.36 Counter Clear Timing (Input Capture)

Buffer Operation Timing: Figures 11.37 and 11.38 show the timings in buffer operation.

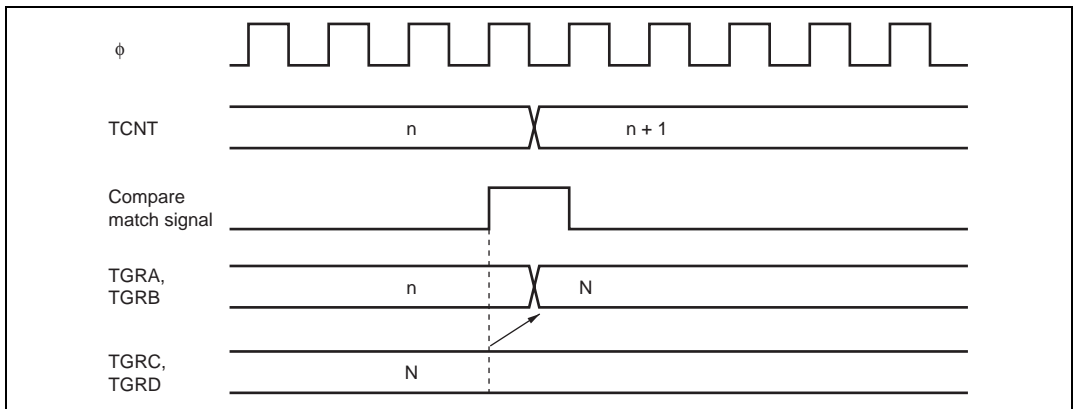


Figure 11.37 Buffer Operation Timing (Compare Match)

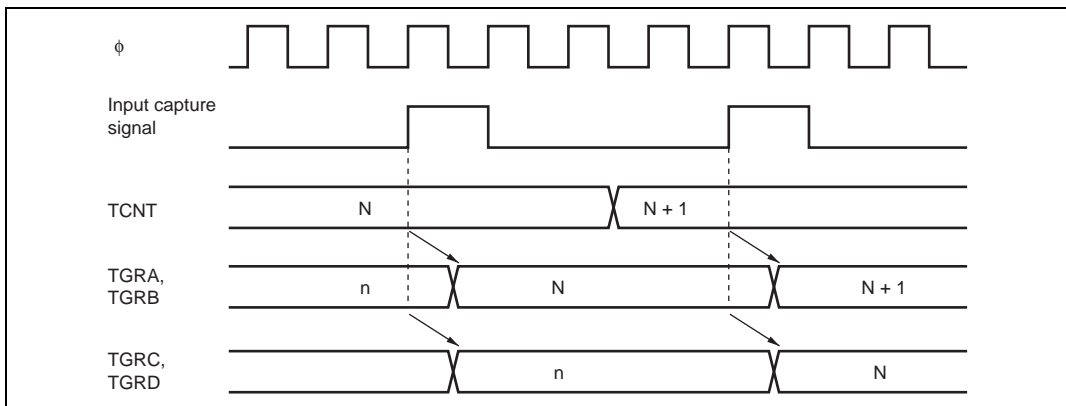


Figure 11.38 Buffer Operation Timing (Input Capture)

11.9.2 Interrupt Signal Timing

TGF Flag Setting Timing in Case of Compare Match: Figure 11.39 shows the timing for setting of the TGF flag in TSR by compare match occurrence, and the TGI interrupt request signal timing.

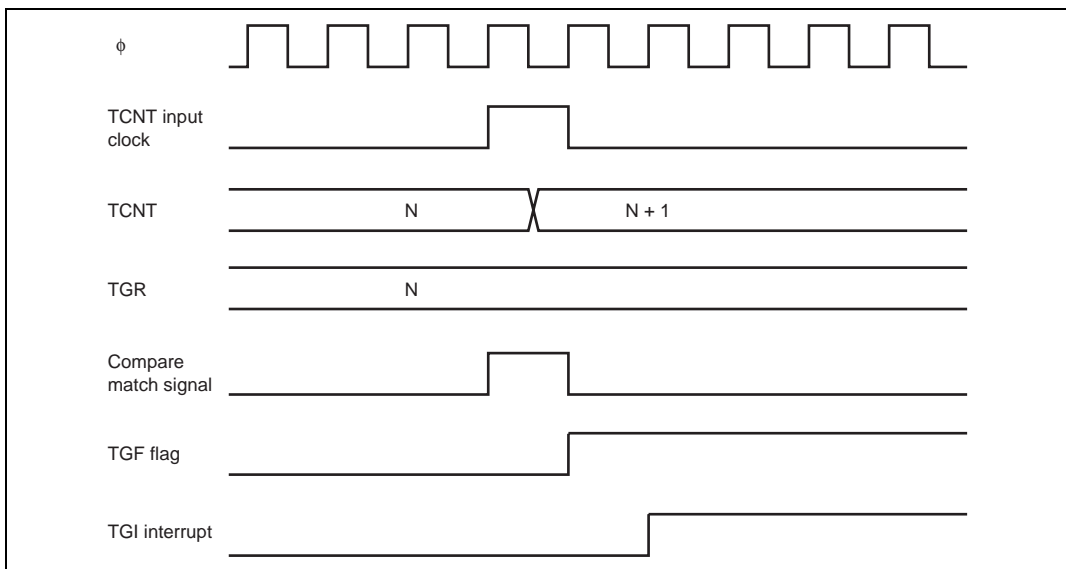


Figure 11.39 TGI Interrupt Timing (Compare Match)

TGF Flag Setting Timing in Case of Input Capture: Figure 11.40 shows the timing for setting of the TGF flag in TSR by input capture occurrence, and the TGI interrupt request signal timing.

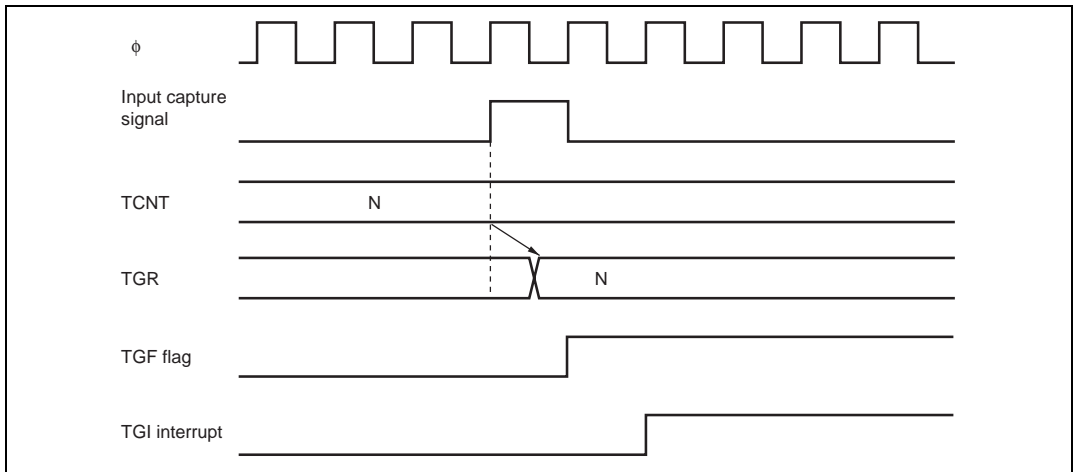


Figure 11.40 TGI Interrupt Timing (Input Capture)

TCFV Flag/TCFU Flag Setting Timing: Figure 11.41 shows the timing for setting of the TCFV flag in TSR by overflow occurrence, and the TCIV interrupt request signal timing.

Figure 11.42 shows the timing for setting of the TCFU flag in TSR by underflow occurrence, and the TCIU interrupt request signal timing.

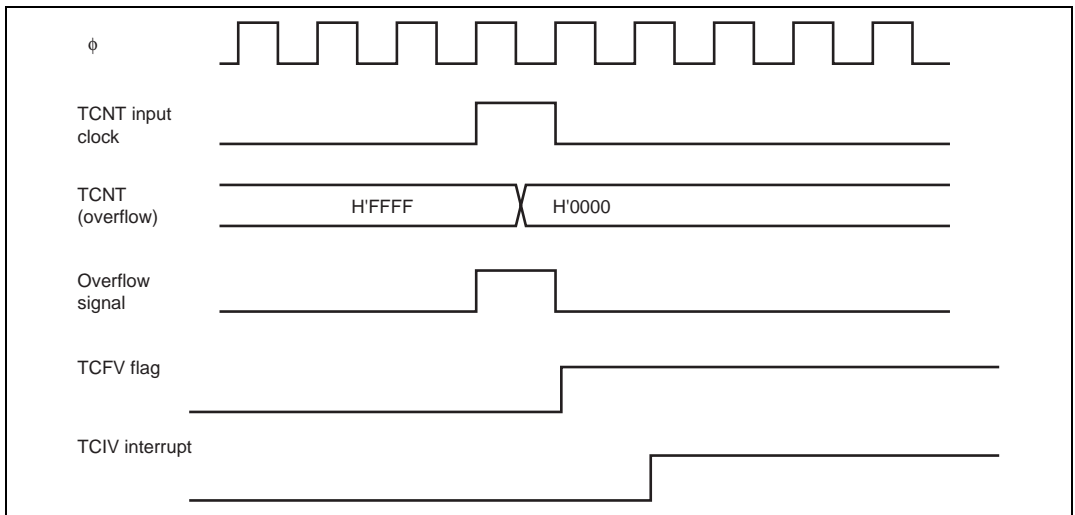


Figure 11.41 TCIV Interrupt Setting Timing

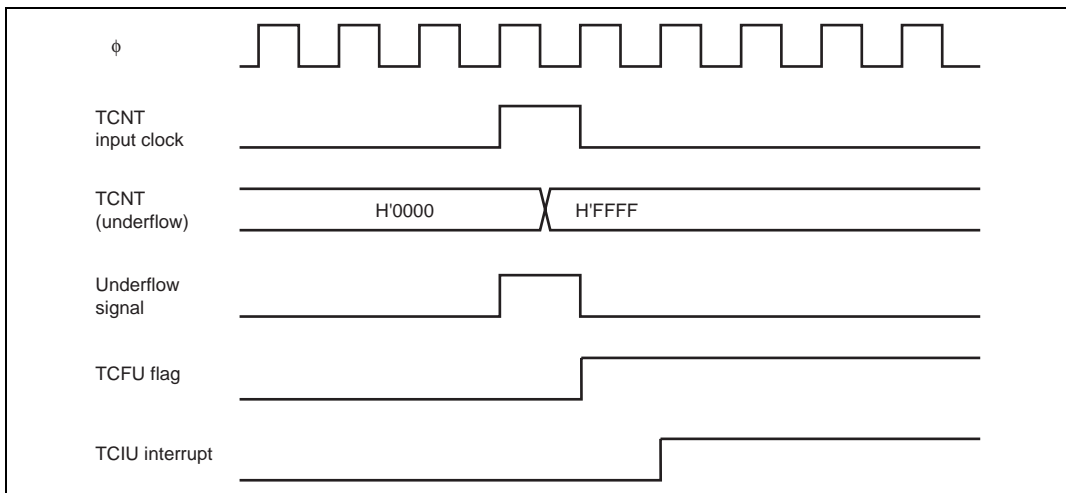


Figure 11.42 TCIU Interrupt Setting Timing

Status Flag Clearing Timing: After a status flag is read as 1 by the CPU, it is cleared by writing 0 to it. When the DTC or DMAC* is activated, the flag is cleared automatically. Figure 11.43 shows the timing for status flag clearing by the CPU, and figure 11.44 shows the timing for status flag clearing by the DTC or DMAC*.

Note: * Supported only by the H8S/2239 Group.

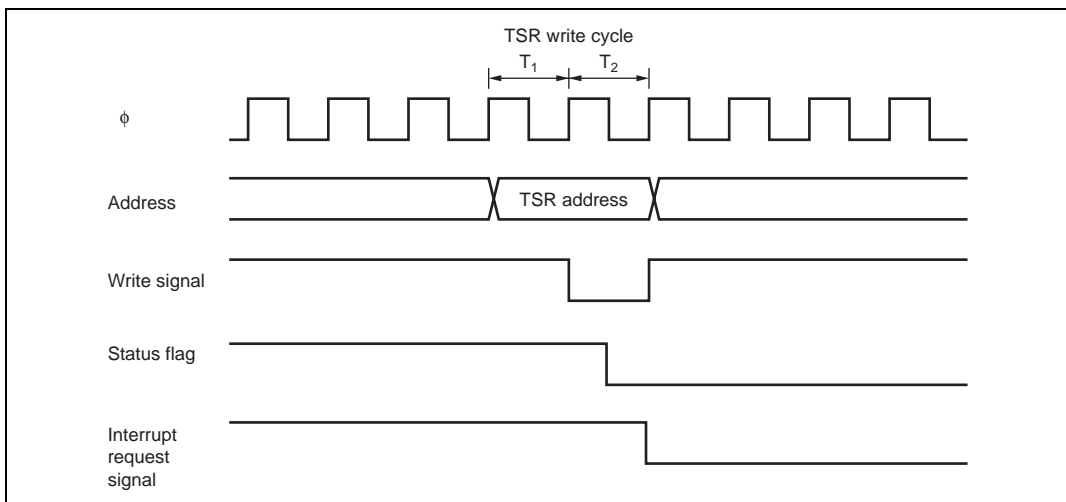


Figure 11.43 Timing for Status Flag Clearing by CPU

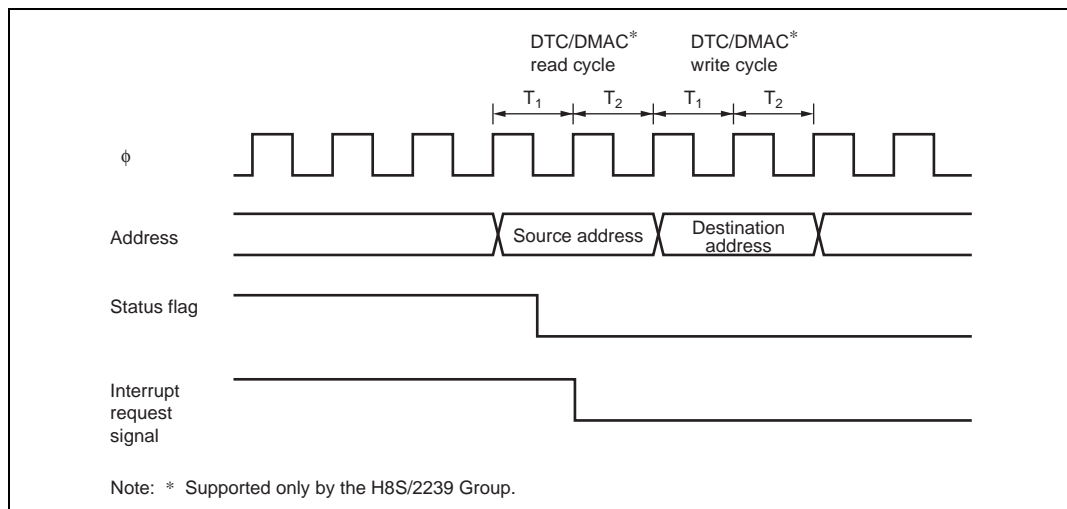


Figure 11.44 Timing for Status Flag Clearing by DTC/DMAC* Activation

Note: * Supported only by the H8S/2239 Group.

11.10 Usage Notes

11.10.1 Module Stop Mode Setting

TPU operation can be disabled or enabled using the module stop control register. The initial setting is for TPU operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 24, Power-Down Modes.

11.10.2 Input Clock Restrictions

The input clock pulse width must be at least 1.5 states in the case of single-edge detection, and at least 2.5 states in the case of both-edge detection. The TPU will not operate properly with a narrower pulse width.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least 1.5 states, and the pulse width must be at least 2.5 states. Figure 11.45 shows the input clock conditions in phase counting mode.

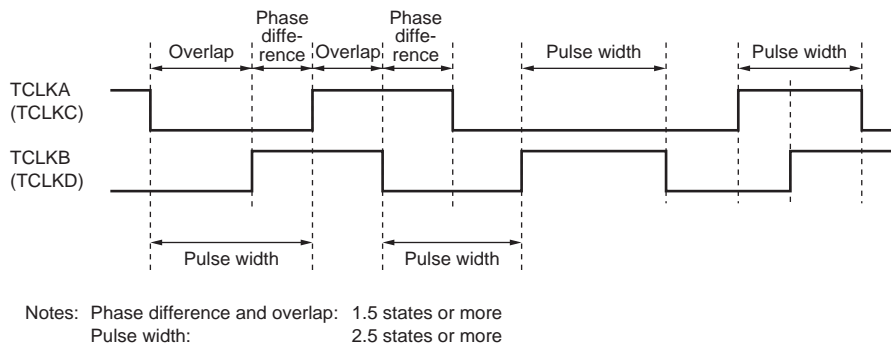


Figure 11.45 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

11.10.3 Caution on Cycle Setting

When counter clearing by compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which the count value matched by TCNT is updated). Consequently, the actual counter frequency is given by the following formula:

$$f = \frac{\phi}{(N + 1)}$$

Where f: Counter frequency
 ϕ : Operating frequency
 N: TGR set value

11.10.4 Contention between TCNT Write and Clear Operations

If the counter clearing signal is generated in the T_2 state of a TCNT write cycle, TCNT clearing takes precedence and the TCNT write is not performed. Figure 11.46 shows the timing in this case.

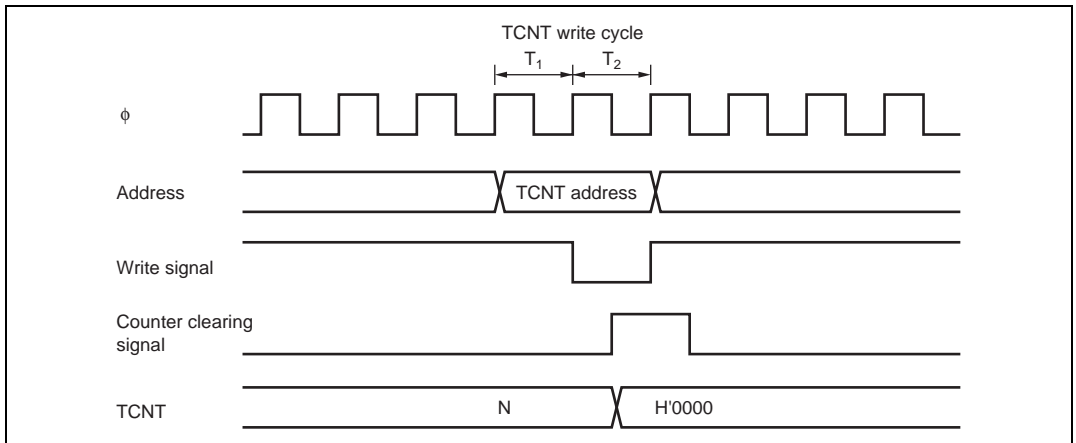


Figure 11.46 Contention between TCNT Write and Clear Operations

11.10.5 Contention between TCNT Write and Increment Operations

If incrementing occurs in the T_2 state of a TCNT write cycle, the TCNT write takes precedence and TCNT is not incremented. Figure 11.47 shows the timing in this case.

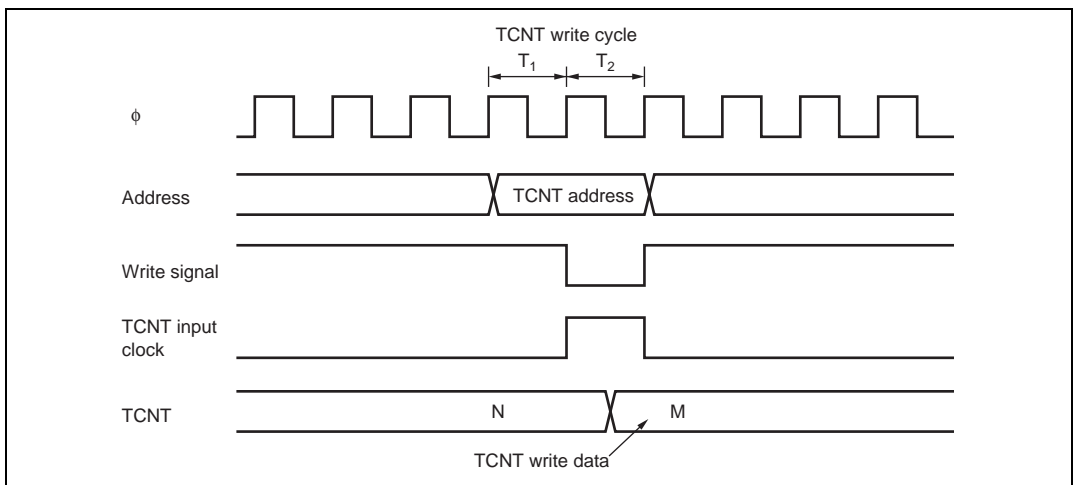


Figure 11.47 Contention between TCNT Write and Increment Operations

11.10.6 Contention between TGR Write and Compare Match

If a compare match occurs in the T_2 state of a TGR write cycle, the TGR write takes precedence and the compare match signal is disabled. A compare match also does not occur when the same value as before is written.

Figure 11.48 shows the timing in this case.

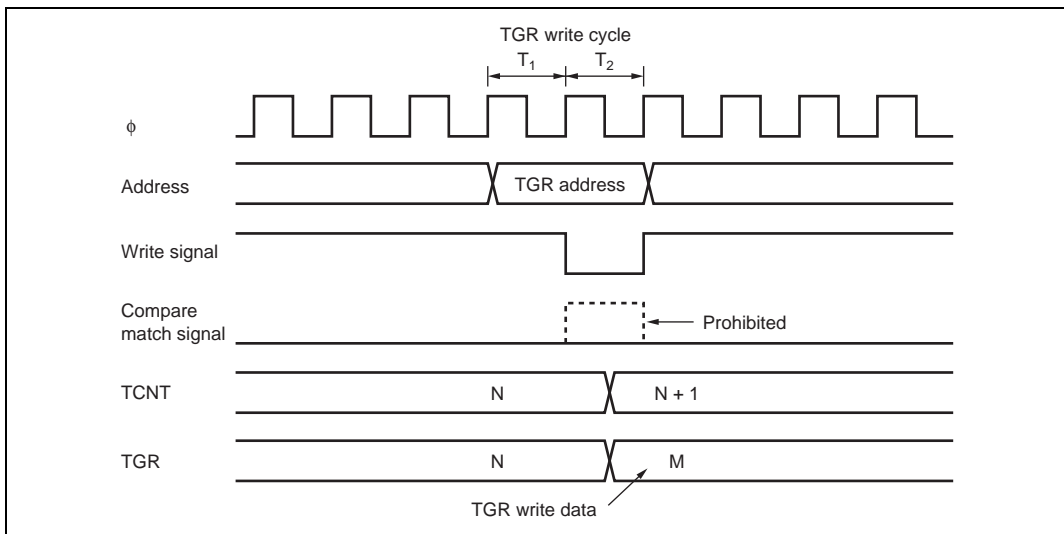


Figure 11.48 Contention between TGR Write and Compare Match

11.10.7 Contention between Buffer Register Write and Compare Match

If a compare match occurs in the T_2 state of a TGR write cycle, the data transferred to TGR by the buffer operation will be the data prior to the write.

Figure 11.49 shows the timing in this case.

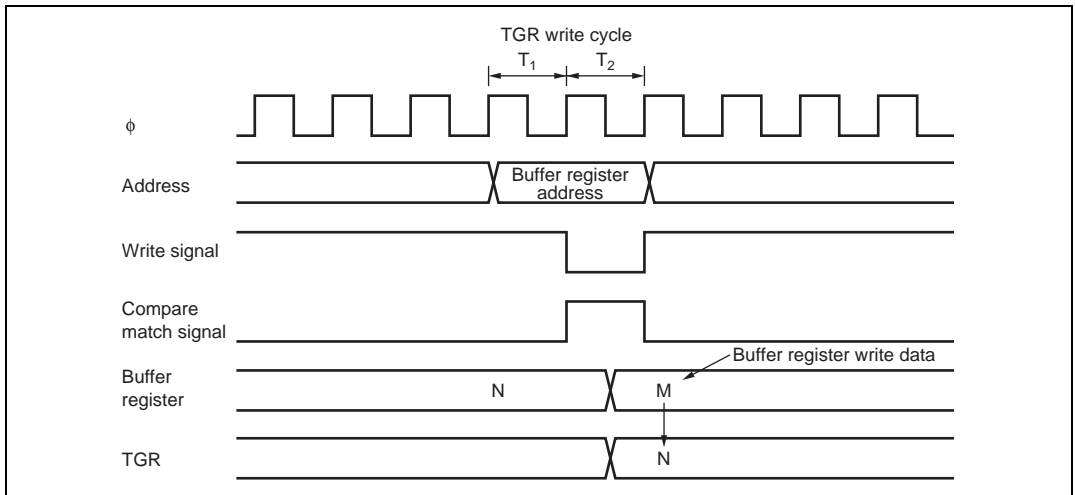


Figure 11.49 Contention between Buffer Register Write and Compare Match

11.10.8 Contention between TGR Read and Input Capture

If the input capture signal is generated in the T_1 state of a TGR read cycle, the data that is read will be the data after input capture transfer.

Figure 11.50 shows the timing in this case.

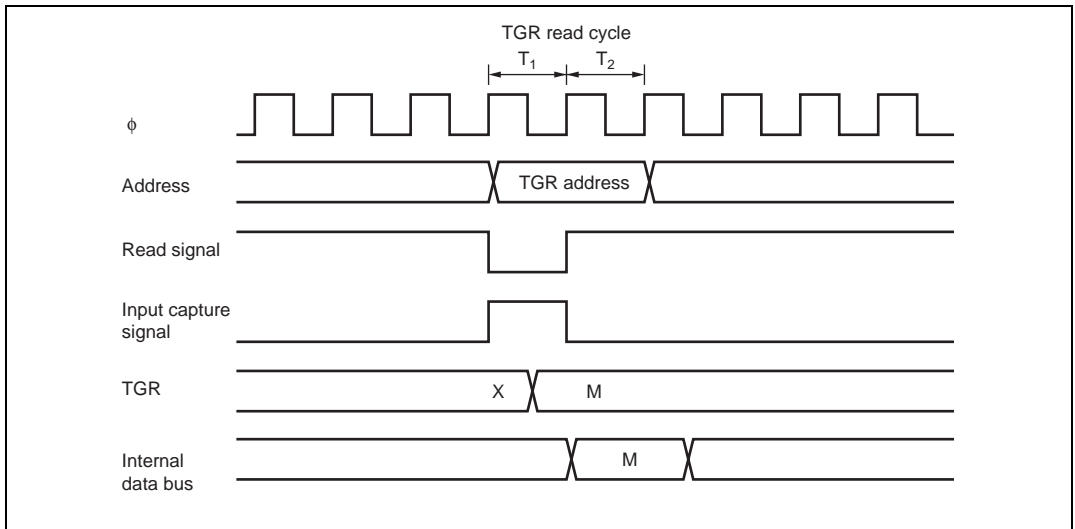


Figure 11.50 Contention between TGR Read and Input Capture

11.10.9 Contention between TGR Write and Input Capture

If the input capture signal is generated in the T_2 state of a TGR write cycle, the input capture operation takes precedence and the write to TGR is not performed.

Figure 11.51 shows the timing in this case.

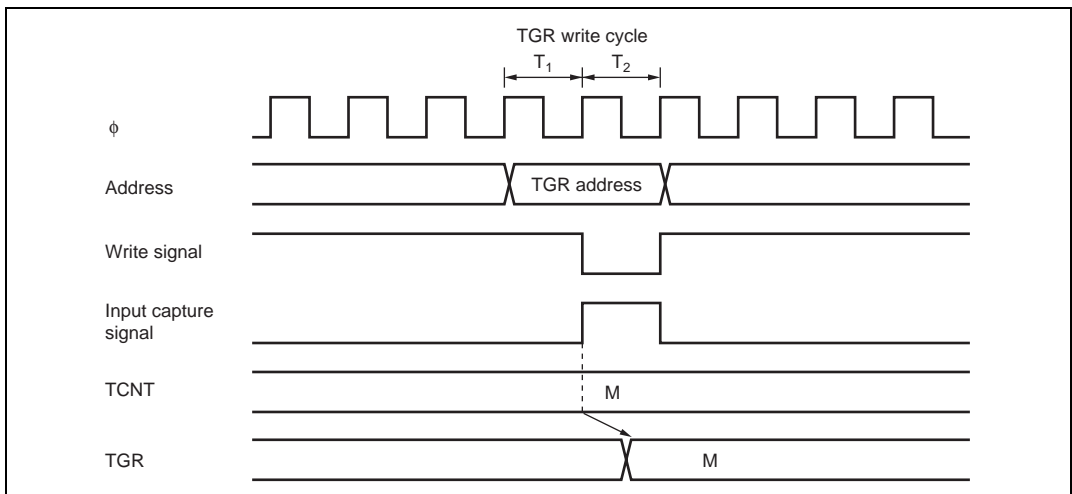


Figure 11.51 Contention between TGR Write and Input Capture

11.10.10 Contention between Buffer Register Write and Input Capture

If the input capture signal is generated in the T_2 state of a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 11.52 shows the timing in this case.

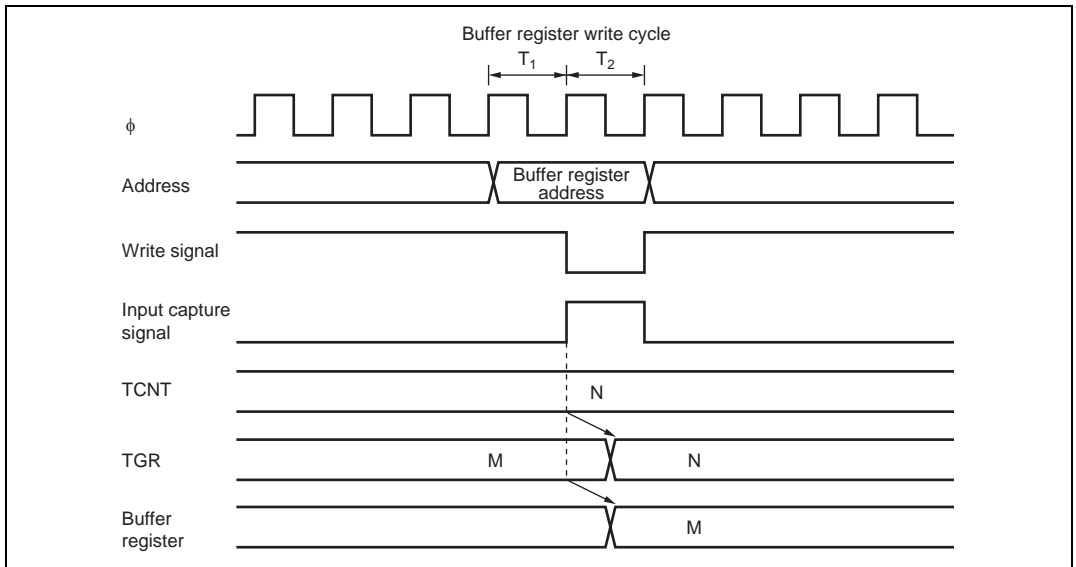


Figure 11.52 Contention between Buffer Register Write and Input Capture

11.10.11 Contention between Overflow/Underflow and Counter Clearing

If overflow/underflow and counter clearing occur simultaneously, the TCFV/TCFU flag in TSR is not set and TCNT clearing takes precedence.

Figure 11.53 shows the operation timing when a TGR compare match is specified as the clearing source, and H'FFFF is set in TGR.

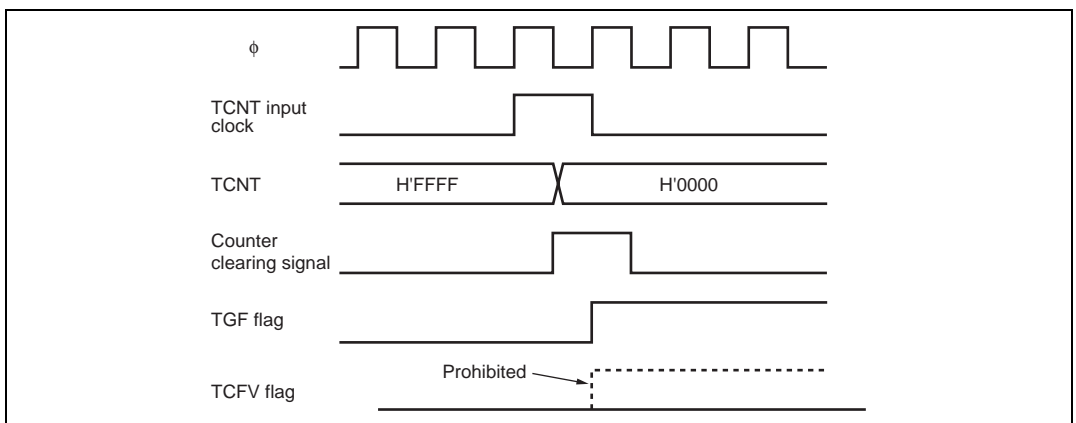


Figure 11.53 Contention between Overflow and Counter Clearing

11.10.12 Contention between TCNT Write and Overflow/Underflow

If there is an up-count or down-count in the T_2 state of a TCNT write cycle, when overflow/underflow occurs, the TCNT write takes precedence and the TCFV/TCFU flag in TSR is not set.

Figure 11.54 shows the operation timing when there is contention between TCNT write and overflow.

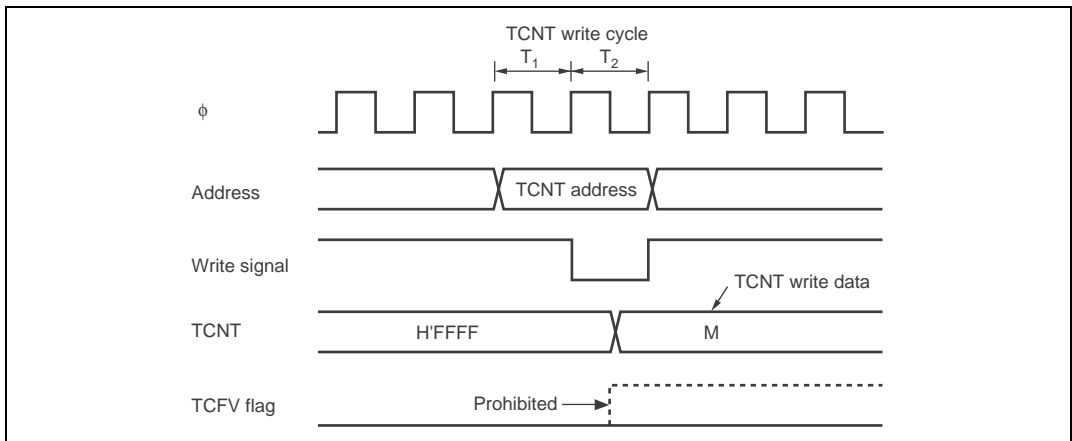


Figure 11.54 Contention between TCNT Write and Overflow

11.10.13 Multiplexing of I/O Pins

In this LSI, the TCLKA input pin is multiplexed with the TIOCC0 I/O pin, the TCLKB input pin with the TIOCD0 I/O pin, the TCLKC input pin with the TIOCB1 I/O pin, and the TCLKD input pin with the TIOCB2 I/O pin. When an external clock is input, compare match output should not be performed from a multiplexed pin.

11.10.14 Interrupts and Module Stop Mode

If module stop mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the DMAC* or DTC activation source. Interrupts should therefore be disabled before entering module stop mode.

Note: * Supported only by the H8S/2239 Group.

Section 12 8-Bit Timers

The H8S/2258 Group, H8S/2239 Group, and H8S/2238 Group have an on-chip 8-bit timer module with four channels (TMR_0, TMR_1, TMR_2, and TMR_3) operating on the basis of an 8-bit counter.

The H8S/2237 Group and H8S/2227 Group have an on-chip 8-bit timer module with two channels (TMR_0 and TMR_1) operating on the basis of an 8-bit counter.

The 8-bit timer module can be used to count external events and be used as a multifunction timer in a variety of applications, such as generation of counter reset, interrupt requests, and pulse output with an arbitrary duty cycle using a compare-match signal with two registers.

12.1 Features

- Selection of clock sources
Selected from three internal clocks ($\phi/8$, $\phi/64$, and $\phi/8192$) and an external clock.
- Selection of three ways to clear the counters
The counters can be cleared on compare-match A or B, or by an external reset signal.
- Timer output controlled by two compare-match signals
The timer output signal in each channel is controlled by two independent compare-match signals, enabling the timer to be used for various applications, such as the generation of pulse output or PWM output with an arbitrary duty cycle.
- Cascading of the two channels
 - TMR_0 and TMR_1 cascading
The module can operate as a 16-bit timer using TMR_0 as the upper half and channel TMR_1 as the lower half (16-bit count mode).
TMR_1 can be used to count TMR_0 compare-match occurrences (compare-match count mode).
 - TMR_2* and TMR_3* cascading
The module can operate as a 16-bit timer using TMR_2 as the upper half and channel TMR_3 as the lower half (16-bit count mode).
TMR_3 can be used to count TMR_2 compare-match occurrences (compare-match count mode).
- Multiple interrupt sources for each channel
Two compare-match interrupts and one overflow interrupt can be requested independently.
- Generation of A/D conversion start trigger
Channel 0 compare-match signal can be used as the A/D conversion start trigger.

- At initialization, the 8-bit timer operation is halted. Register access is enabled by canceling the module stop mode.

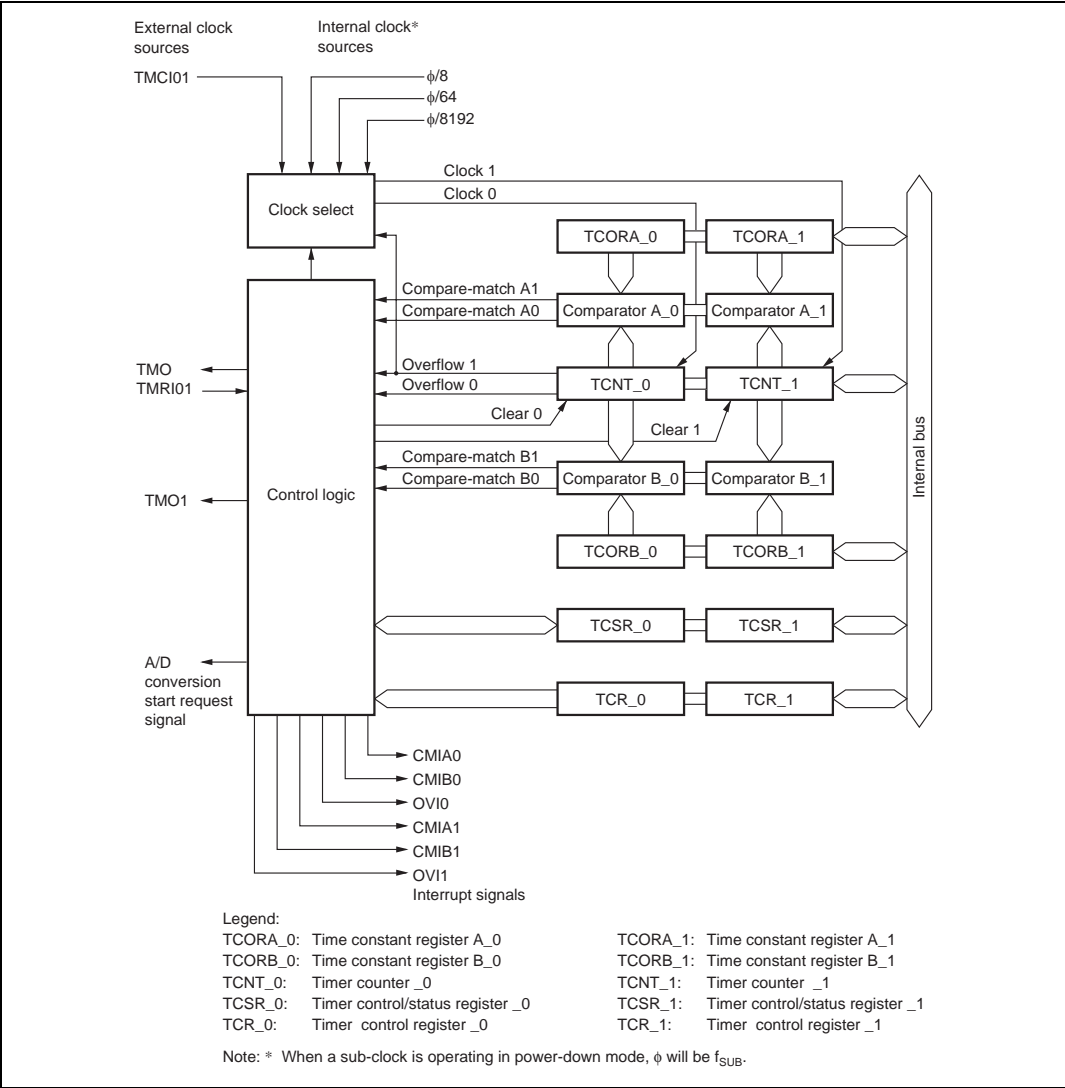


Figure 12.1 Block Diagram of 8-Bit Timer Module

12.2 Input/Output Pins

Table 12.1 summarizes the input and output pins of the 8-bit timer module.

Table 12.1 Pin Configuration

| Channel | Name | Symbol | I/O | Function |
|-------------------|-------------------|---------|--------|--------------------------------------|
| 0 | Timer output | TMO0 | Output | Output controlled by compare-match |
| 1 | Timer output | TMO1 | Output | Output controlled by compare-match |
| Common to 0 and 1 | Timer clock input | TMC101 | Input | External clock input for the counter |
| | Timer reset input | TMRI01 | Input | External reset input for the counter |
| 2 | Timer output | TMO2* | Output | Output controlled by compare-match |
| 3 | Timer output | TMO3* | Output | Output controlled by compare-match |
| Common to 2 and 3 | Timer clock input | TMC123* | Input | External clock input for the counter |
| | Timer reset input | TMRI23* | Input | External reset input for the counter |

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.3 Register Descriptions

The 8-bit timer has the following registers. For details on the module stop register, refer to section 24.1.2, Module Stop Registers A to C (MSTPCRA to MSTPCRC).

- Time constant register A_0 (TCORA_0)
- Time constant register B_0 (TCORB_0)
- Timer control register_0 (TCR_0)
- Timer control/status register_0 (TCSR_0)
- Timer counter_1 (TCNT_1)
- Time constant register A_1 (TCORA_1)
- Time constant register B_1 (TCORB_1)
- Timer control register_1 (TCR_1)
- Timer control/status register_1 (TCSR_1)
- Timer counter_2 (TCNT_2)*
- Time constant register A_2 (TCORA_2)*
- Time constant register B_2 (TCORB_2)*
- Timer control register_2 (TCR_2)*
- Timer control/status register_2 (TCSR_2)*
- Timer counter_3 (TCNT_3)*

- Time constant register A_3 (TCORA_3)*
- Time constant register B_3 (TCORB_3)*
- Timer control register_3 (TCR_3)*
- Timer control/status register_3 (TCSR_3)*

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.3.1 Timer Counter (TCNT)

Each TCNT is an 8-bit up-counter. TCNT_0 and TCNT_1 (TCNT_2 and TCNT_3)* comprise a single 16-bit register, so they can be accessed together by word access.

TCNT increments on pulses generated from an internal or external clock source. This clock source is selected by clock select bits CKS2 to CKS0 in TCR. TCNT can be cleared by an external reset input signal or compare-match signals A and B. Counter clear bits CCLR1 and CCLR0 in TCR select the method of clearing.

When TCNT overflows from H'FF to H'00, the overflow flag (OVF) in TCSR is set to 1. The initial value of TCNT is H'00.

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.3.2 Time Constant Register A (TCORA)

TCORA is an 8-bit readable/writable register. TCORA_0 and TCORA_1 (TCORA_2 and TCORA_3)* comprise a single 16-bit register, so they can be accessed together by word access.

TCORA is continually compared with the value in TCNT. When a match is detected, the corresponding compare-match flag A (CMFA) in TCSR is set. Note, however, that comparison is disabled during the T2 state of a TCORA write cycle.

The timer output from the TMO pin can be freely controlled by the compare-match signal A and the settings of output select bits OS1 and OS0 in TCSR.

The initial value of TCORA is H'FF.

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.3.3 Time Constant Register B (TCORB)

TCORB is an 8-bit readable/writable register. TCORB_0 and TCORB_1 (TCORB_2 and TCORB_3)* comprise a single 16-bit register, so they can be accessed together by word access.

TCORB is continually compared with the value in TCNT. When a match is detected, the corresponding compare-match flag B (CMFB) in TCSR is set. Note, however, that comparison is disabled during the T2 state of a TCORB write cycle.

The timer output from the TMO pin can be freely controlled by the compare-match signal B and the settings of output select bits OS1 and OS0 in TCSR.

The initial value of TCORB is H'FF.

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.3.4 Timer Control Register (TCR)

TCR selects the TCNT clock source and the time at which TCNT is cleared, and controls interrupt requests.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | CMIEB | 0 | R/W | Compare-Match Interrupt Enable B Selects whether the CMFB interrupt request (CMIB) is enabled or disabled when the CMFB flag in TCSR is set to 1. 0: CMFB interrupt request (CMIB) is disabled 1: CMFB interrupt request (CMIB) is enabled |
| 6 | CMIEA | 0 | R/W | Compare-Match Interrupt Enable A Selects whether the CMFA interrupt request (CMIA) is enabled or disabled when the CMFA flag in TCSR is set to 1. 0: CMFA interrupt request (CMIA) is disabled 1: CMFA interrupt request (CMIA) is enabled |
| 5 | OVIE | 0 | R/W | Timer Overflow Interrupt Enable Selects whether the OVF interrupt request (OVI) is enabled or disabled when the OVF flag in TCSR is set to 1. 0: OVF interrupt request (OVI) is disabled 1: OVF interrupt request (OVI) is enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 4 | CCLR1 | 0 | R/W | Counter Clear 1 and 0 |
| 3 | CCLR0 | 0 | R/W | These bits select the method by which TCNT is cleared. 00: Clearing is disabled 01: Cleared on compare-match A 10: Cleared on compare-match B 11: Cleared on rising edge of external reset input |
| 2 | CKS2 | 0 | R/W | Clock Select 2 to 0 |
| 1 | CKS1 | 0 | R/W | The input clock can be selected from three clocks divided from the system clock (ϕ). When use of an external clock is selected, three types of count can be selected: at the rising edge, the falling edge, and both rising and falling edges. 000: Clock input disabled 001: $\phi/8$ internal clock source, counted on the falling edge 010: $\phi/64$ internal clock source, counted on the falling edge 011: $\phi/8192$ internal clock source, counted on the falling edge 100: For channel 0: Counted on TCNT1 overflow signal ^{*1} For channel 1: Counted on TCNT0 compare-match A ^{*1} For channel 2: ^{*2} Counted on TCNT3 overflow signal ^{*1} For channel 3: ^{*2} Counted on TCNT2 compare-match A ^{*1} 101: External clock source, counted at rising edge 110: External clock source, counted at falling edge 111: External clock source, counted at both rising and falling edges |
| 0 | CKS0 | 0 | R/W | |

Notes: 1. If the count input of channel 0 (channel 2) is the TCNT1 (TCNT3) overflow signal and that of channel 1 (channel 3) is the TCNT1 (TCNT3) compare-match signal, no incrementing clock will be generated. Do not use this setting.

2. Not available in the H8S/2237 Group and H8S/2227 Group.

12.3.5 Timer Control/Status Register (TCSR)

TCSR indicates status flags and controls compare-match output.

- TCSR_0

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|--------|---|
| 7 | CMFB | 0 | R/(W)* | Compare-Match Flag B [Setting condition] When TCNT = TCORB [Clearing conditions] <ul style="list-style-type: none"> Read CMFB when CMFB = 1, then write 0 in CMFB When DTC is activated by CMIB interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 |
| 6 | CMFA | 0 | R/(W)* | Compare-Match Flag A [Setting condition] When TCNT = TCORA [Clearing conditions] <ul style="list-style-type: none"> Read CMFA when CMFA = 1, then write 0 in CMFA When DTC is activated by CMIA interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 |
| 5 | OVF | 0 | R/(W)* | Timer Overflow Flag [Setting condition] When TCNT overflows from H'FF to H'00 [Clearing condition] Read OVF when OVF = 1, then write 0 in OVF |
| 4 | ADTE | 0 | R/W | A/D Trigger Enable Enables or disables A/D converter start requests by compare-match A. 0: A/D converter start requests by compare-match A are disabled 1: A/D converter start requests by compare-match A are enabled |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | OS3 | 0 | R/W | Output Select 3 and 2 |
| 2 | OS2 | 0 | R/W | These bits specify how the timer output level is to be changed by a compare-match B of TCORB and TCNT. 00: No change when compare-match B occurs 01: 0 is output when compare-match B occurs 10: 1 is output when compare-match B occurs 11: Output is inverted when compare-match B occurs (toggle output) |
| 1 | OS1 | 0 | R/W | Output Select 1 and 0 |
| 0 | OS0 | 0 | R/W | These bits specify how the timer output level is to be changed by a compare-match A of TCORA and TCNT. 00: No change when compare-match A occurs 01: 0 is output when compare-match A occurs 10: 1 is output when compare-match A occurs 11: Output is inverted when compare-match A occurs (toggle output) |

Note: * Only 0 can be written to this bit, to clear the flag.

- TCSR_1 and TCSR_3^{*1}

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|---|
| 7 | CMFB | 0 | R/(W) ^{*2} | Compare-Match Flag B [Setting condition] When TCNT = TCORB [Clearing conditions] <ul style="list-style-type: none"> • Read CMFB when CMFB = 1, then write 0 in CMFB • When DTC is activated by CMIB interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 |
| 6 | CMFA | 0 | R/(W) ^{*2} | Compare-Match Flag A [Setting condition] When TCNT = TCORA [Clearing conditions] <ul style="list-style-type: none"> • Read CMFA when CMFA = 1, then write 0 in CMFA • When DTC is activated by CMIA interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 |
| 5 | OVF | 0 | R/(W) ^{*2} | Timer Overflow Flag [Setting condition] When TCNT overflows from H'FF to H'00 [Clearing condition] Read OVF when OVF = 1, then write 0 in OVF |
| 4 | — | 1 | — | Reserved This bit is always read as 1 and cannot be modified. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | OS3 | 0 | R/W | Output Select 3 and 2 |
| 2 | OS2 | 0 | R/W | These bits specify how the timer output level is to be changed by a compare-match B of TCORB and TCNT. 00: No change when compare-match B occurs 01: 0 is output when compare-match B occurs 10: 1 is output when compare-match B occurs 11: Output is inverted when compare-match B occurs (toggle output) |
| 1 | OS1 | 0 | R/W | Output Select 1 and 0 |
| 0 | OS0 | 0 | R/W | These bits specify how the timer output level is to be changed by a compare-match A of TCORA and TCNT. 00: No change when compare-match A occurs 01: 0 is output when compare-match A occurs 10: 1 is output when compare-match A occurs 11: Output is inverted when compare-match A occurs (toggle output) |

Notes: 1. Not available in the H8S/2237 Group and H8S/2227 Group.
2. Only 0 can be written to this bit, to clear the flag.

- TCSR_2^{*1}

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|---|
| 7 | CMFB | 0 | R/(W) ^{*2} | Compare-Match Flag B [Setting condition] When TCNT = TCORB [Clearing conditions] <ul style="list-style-type: none"> Read CMFB when CMFB = 1, then write 0 in CMFB When DTC is activated by CMIB interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 |
| 6 | CMFA | 0 | R/(W) ^{*2} | Compare-Match Flag A [Setting condition] When TCNT = TCORA [Clearing conditions] <ul style="list-style-type: none"> Read CMFA when CMFA = 1, then write 0 in CMFA When DTC is activated by CMIA interrupt while DISEL bit of MRB in DTC is 0 with the transfer counter not being 0 |
| 5 | OVF | 0 | R/(W) ^{*2} | Timer Overflow Flag [Setting condition] When TCNT overflows from H'FF to H'00 [Clearing condition] Read OVF when OVF = 1, then write 0 in OVF |
| 4 | — | 0 | R/W | Reserved This bit is a readable/writable bit, but the write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | OS3 | 0 | R/W | Output Select 3 and 2 |
| 2 | OS2 | 0 | R/W | These bits specify how the timer output level is to be changed by a compare-match B of TCORB and TCNT. 00: No change when compare-match B occurs 01: 0 is output when compare-match B occurs 10: 1 is output when compare-match B occurs 11: Output is inverted when compare-match B occurs (toggle output) |
| 1 | OS1 | 0 | R/W | Output Select 1 and 0 |
| 0 | OS0 | 0 | R/W | These bits specify how the timer output level is to be changed by a compare-match A of TCORA and TCNT. 00: No change when compare-match A occurs 01: 0 is output when compare-match A occurs 10: 1 is output when compare-match A occurs 11: Output is inverted when compare-match A occurs (toggle output) |

Notes: 1. Not available in the H8S/2237 Group and H8S/2227 Group.
2. Only 0 can be written to this bit, to clear the flag.

12.4 Operation

12.4.1 Pulse Output

Figure 12.2 shows an example of arbitrary duty pulse output.

1. Set TCR in CCR1 to 0 and CCLR0 to 1 to clear TCNT by a TCORA compare-match.
2. Set OS3 to OS0 bits in TCSR to B'0110 to output 1 by a compare-match A and 0 by compare-match B.

By the above settings, waveforms with the cycle of TCORA and the pulse width of TCRB can be output without software intervention.

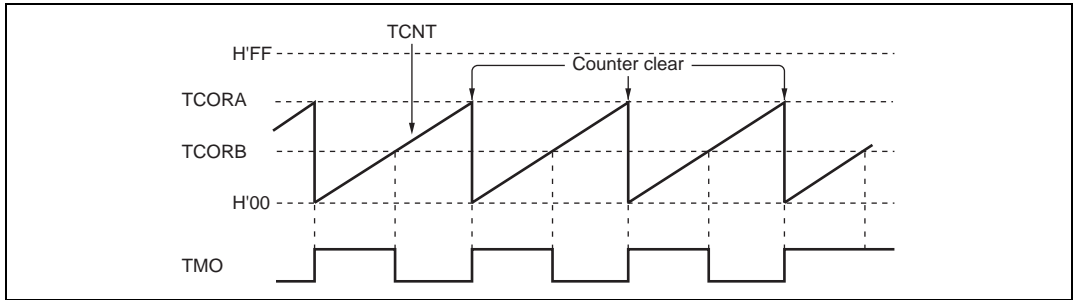


Figure 12.2 Example of Pulse Output

12.5 Operation Timing

12.5.1 TCNT Incrementation Timing

Figure 12.3 shows the TCNT count timing with internal clock source. Figure 12.4 shows the TCNT incrementation timing with external clock source. The pulse width of the external clock for incrementation at signal edge must be at least 1.5 system clock (ϕ) periods, and at least 2.5 states for incrementation at both edges. The counter will not increment correctly if the pulse width is less than these values.

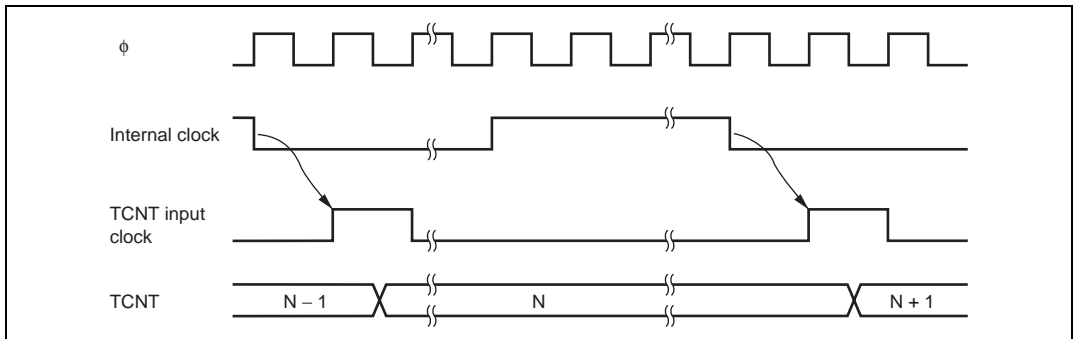


Figure 12.3 Count Timing for Internal Clock Input

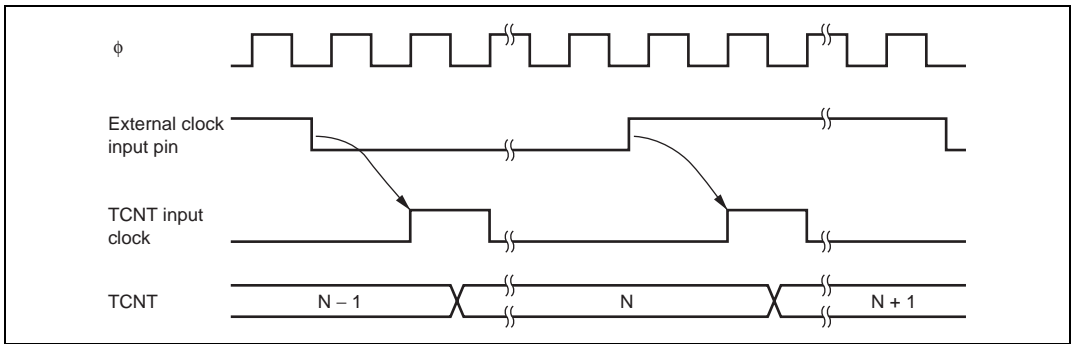


Figure 12.4 Count Timing for External Clock Input

12.5.2 Timing of CMFA and CMFB Setting when a Compare-Match Occurs

The CMFA and CMFB flags in TCSR are set to 1 by a compare-match signal generated when the TCOR and TCNT values match. The compare-match signal is generated at the last state in which the match is true, just before the timer counter is updated. Therefore, when TCOR and TCNT match, the compare-match signal is not generated until the next incrementation clock input. Figure 12.5 shows the timing of CMF flag setting.

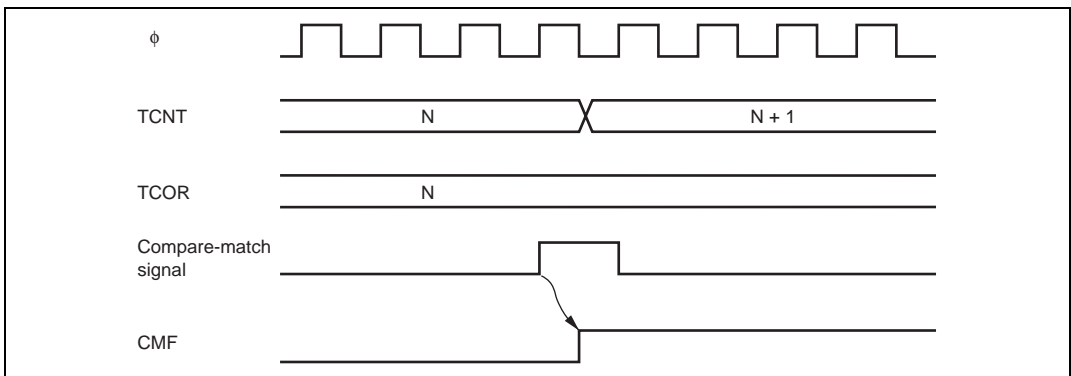


Figure 12.5 Timing of CMF Setting

12.5.3 Timing of Timer Output when a Compare-Match Occurs

When a compare-match occurs, the timer output changes as specified by the output select bits (OS3 to OS0) in TCSR. Figure 12.6 shows the timing when the output is set to toggle at compare-match A.

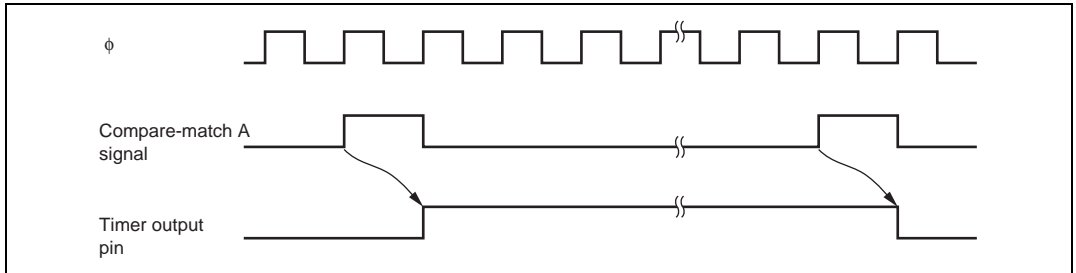


Figure 12.6 Timing of Timer Output

12.5.4 Timing of Compare-Match Clear when a Compare-Match Occurs

TCNT is cleared when compare-match A or B occurs, depending on the setting of the CCLR1 and CCLR0 bits in TCR. Figure 12.7 shows the timing of this operation.

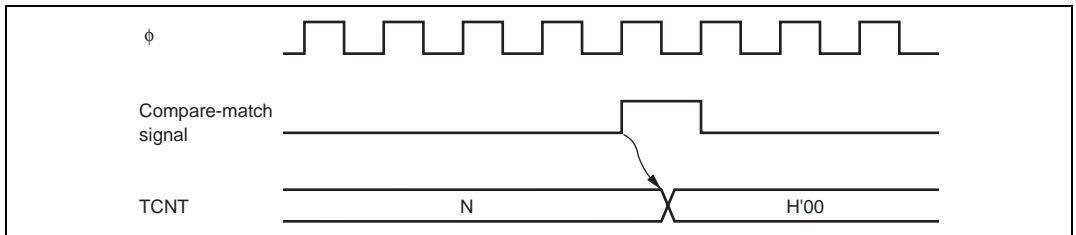


Figure 12.7 Timing of Compare-Match Clear

12.5.5 TCNT External Reset Timing

TCNT is cleared at the rising edge of an external reset input, depending on the settings of the CCLR1 and CCLR0 bits in TCR. The width of the clearing pulse must be at least 1.5 states. Figure 12.8 shows the timing of this operation.

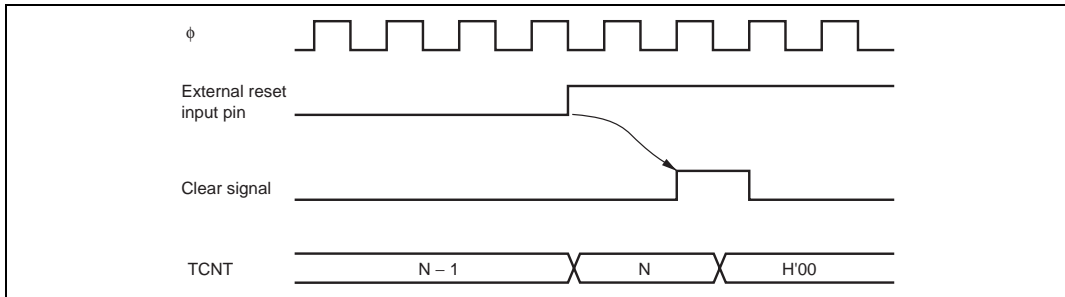


Figure 12.8 Timing of Clearing by External Reset Input

12.5.6 Timing of Overflow Flag (OVF) Setting

OVF in TCSR is set to 1 when the timer count overflows (changes from $H'FF$ to $H'00$). Figure 12.9 shows the timing of this operation.

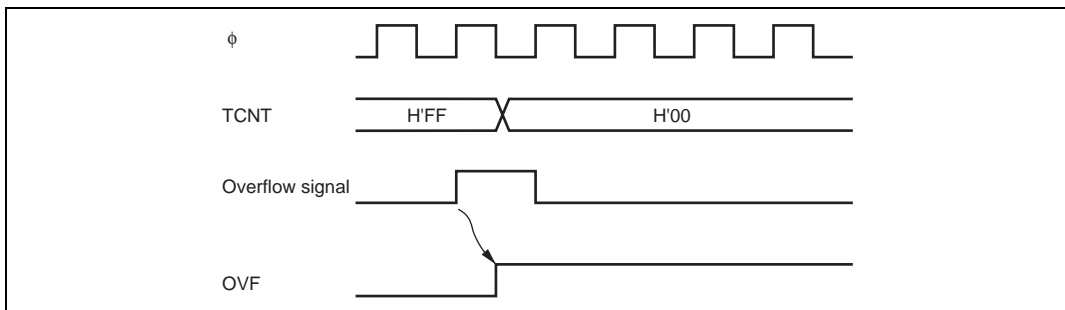


Figure 12.9 Timing of OVF Setting

12.6 Operation with Cascaded Connection

If bits CKS2 to CKS0 in one of TCR_0 and TCR_1 (TCR_2 and TCR_3)* are set to B'100, the 8-bit timers of the two channels are cascaded. With this configuration, a single 16-bit timer can be used (16-bit timer mode) or compare-matches of 8-bit channel 0 (channel 2)* can be counted by the timer of channel 1 (channel 3)* (compare-match count mode). In the case that channel 0 is connected to channel 1 in cascade, the timer operates as described below.

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.6.1 16-Bit Count Mode

When bits CKS2 to CKS0 in TCR_0 are set to B'100, the timer functions as a single 16-bit timer with channel 0 occupying the upper 8 bits and channel 1 occupying the lower 8 bits.

- Setting of compare-match flags
 - The CMF flag in TCSR_0 is set to 1 when a 16-bit compare-match occurs.
 - The CMF flag in TCSR_1 is set to 1 when a lower 8-bit compare-match occurs.
- Counter clear specification
 - If the CCLR1 and CCLR0 bits in TCR_0 have been set for counter clear at compare-match, the 16-bit counter (TCNT_0 and TCNT_1 together) is cleared when a 16-bit compare-match occurs. The 16-bit counter (TCNT_0 and TCNT_1 together) is cleared even if counter clear by the TMRI01 pin has also been set.
 - The settings of the CCLR1 and CCLR0 bits in TCR_1 are ignored. The lower 8 bits cannot be cleared independently.
- Pin output
 - Control of output from the TMO0 pin by bits OS3 to OS0 in TCSR_0 is in accordance with the 16-bit compare-match conditions.
 - Control of output from the TMO1 pin by bits OS3 to OS0 in TCSR_1 is in accordance with the lower 8-bit compare-match conditions.

12.6.2 Compare-Match Count Mode

When bits CKS2 to CKS0 in TCR_1 are B'100, TCNT_1 counts compare-match A for channel 0. Channels 0 and 1 are controlled independently. Conditions such as setting of the CMF flag, generation of interrupts, output from the TMO pin, and counter clearing are in accordance with the settings for each channel.

12.7 Interrupt Sources

12.7.1 Interrupt Sources and DTC Activation

The 8-bit timer can generate three types of interrupt: CMIA, CMIB, and OVI. Table 12.2 shows the interrupt sources and priority. Each interrupt source can be enabled or disabled independently by interrupt enable bits in TCR. Independent signals are sent to the interrupt controller for each interrupt. It is also possible to activate the DTC by means of CMIA and CMIB interrupts.

Table 12.2 8-Bit Timer Interrupt Sources

| Interrupt source | Description | Flag | DTC Activation | Interrupt Priority |
|------------------|-----------------------|------|----------------|--------------------|
| CMIA0 | TCORA_0 compare-match | CMFA | Possible | High |
| CMIB0 | TCORB_0 compare-match | CMFB | Possible | ↑ |
| OVI0 | TCNT_0 overflow | OVF | Not possible | Low |
| CMIA1 | TCORA_1 compare-match | CMFA | Possible | High |
| CMIB1 | TCORB_1 compare-match | CMFB | Possible | ↑ |
| OVI1 | TCNT_1 overflow | OVF | Not possible | Low |
| CMIA2* | TCORA_2 compare-match | CMFA | Possible | High |
| CMIB2* | TCORB_2 compare-match | CMFB | Possible | ↑ |
| OVI2* | TCNT_2 overflow | OVF | Not possible | Low |
| CMIA3* | TCORA_3 compare-match | CMFA | Possible | High |
| CMIB3* | TCORB_3 compare-match | CMFB | Possible | ↑ |
| OVI3* | TCNT_3 overflow | OVF | Not possible | Low |

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

12.7.2 A/D Converter Activation

The A/D converter can be activated only by channel 0 compare match A.

If the ADTE bit in TCSR0 is set to 1 when the CMFA flag is set to 1 by the occurrence of channel 0 compare match A, a request to start A/D conversion is sent to the A/D converter. If the 8-bit timer conversion start trigger has been selected on the A/D converter side at this time, A/D conversion is started.

12.8 Usage Notes

12.8.1 Contention between TCNT Write and Clear

If a timer counter clock pulse is generated during the T_2 state of a TCNT write cycle, the clear takes priority, so that the counter is cleared and the write is not performed. Figure 12.10 shows this operation.

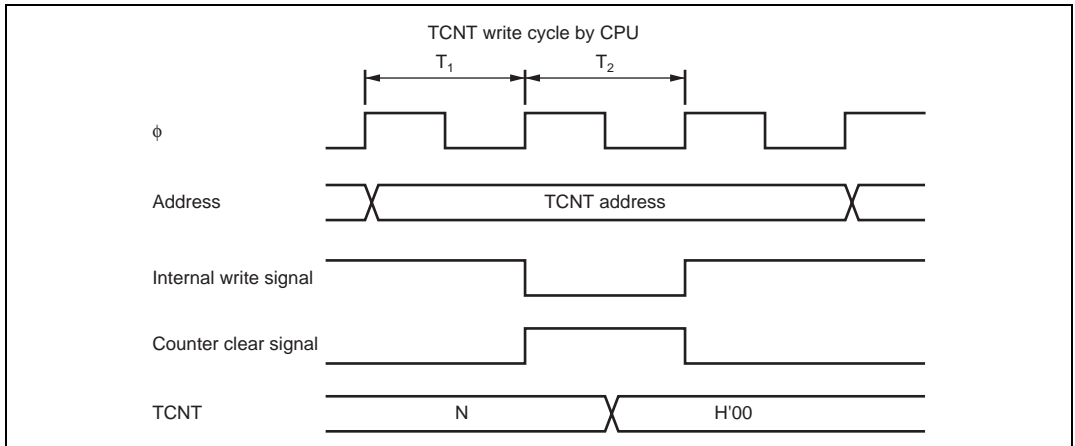


Figure 12.10 Contention between TCNT Write and Clear

12.8.2 Contention between TCNT Write and Increment

If a timer counter clock pulse is generated during the T_2 state of a TCNT write cycle, the write takes priority and the counter is not incremented. Figure 12.11 shows this operation.

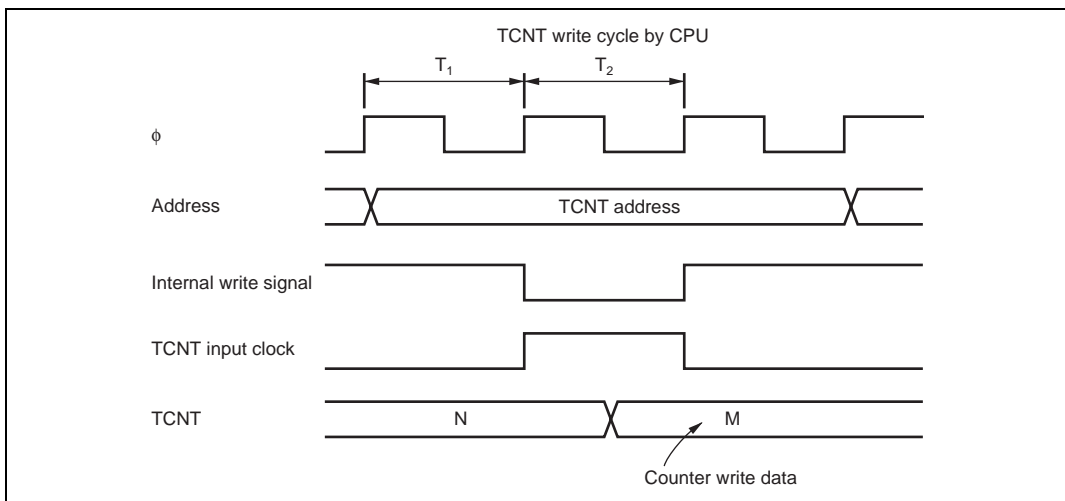


Figure 12.11 Contention between TCNT Write and Increment

12.8.3 Contention between TCOR Write and Compare-Match

During the T_2 state of a TCOR write cycle, the TCOR write has priority even if a compare-match occurs and the compare-match signal is disabled. Figure 12.12 shows this operation.

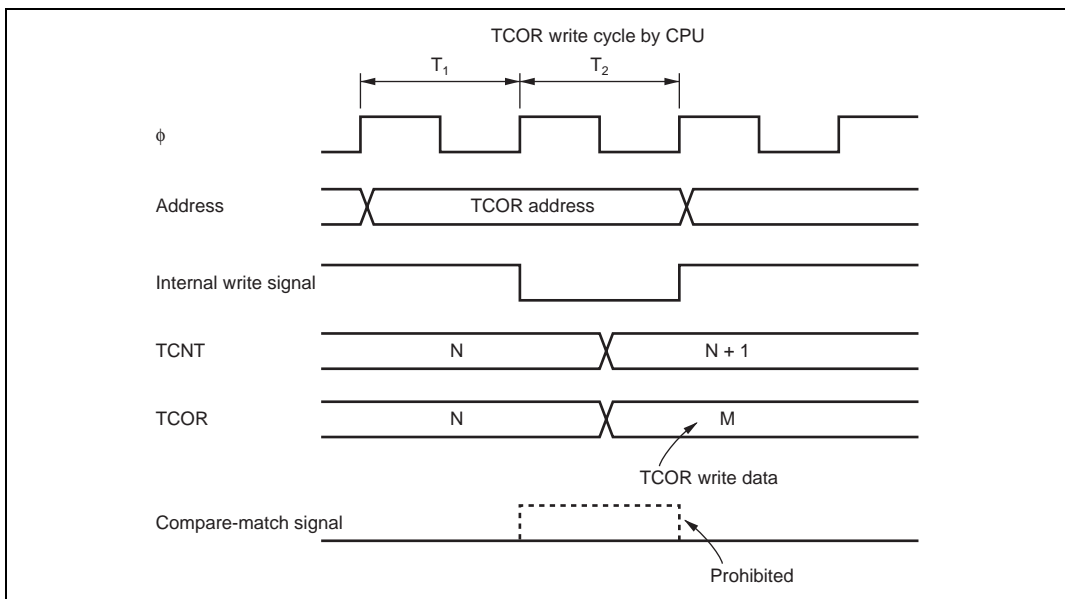


Figure 12.12 Contention between TCOR Write and Compare-Match

12.8.4 Contention between Compare-Matches A and B

If compare-matches A and B occur at the same time, the 8-bit timer operates in accordance with the priorities for the output states set for compare-match A and compare-match B, as shown in table 12.3.

Table 12.3 Timer Output Priorities

| Output Setting | Priority |
|----------------|----------|
| Toggle output | High |
| 1 output | ↑ |
| 0 output | |
| No change | |
| | Low |

12.8.5 Switching of Internal Clocks and TCNT Operation

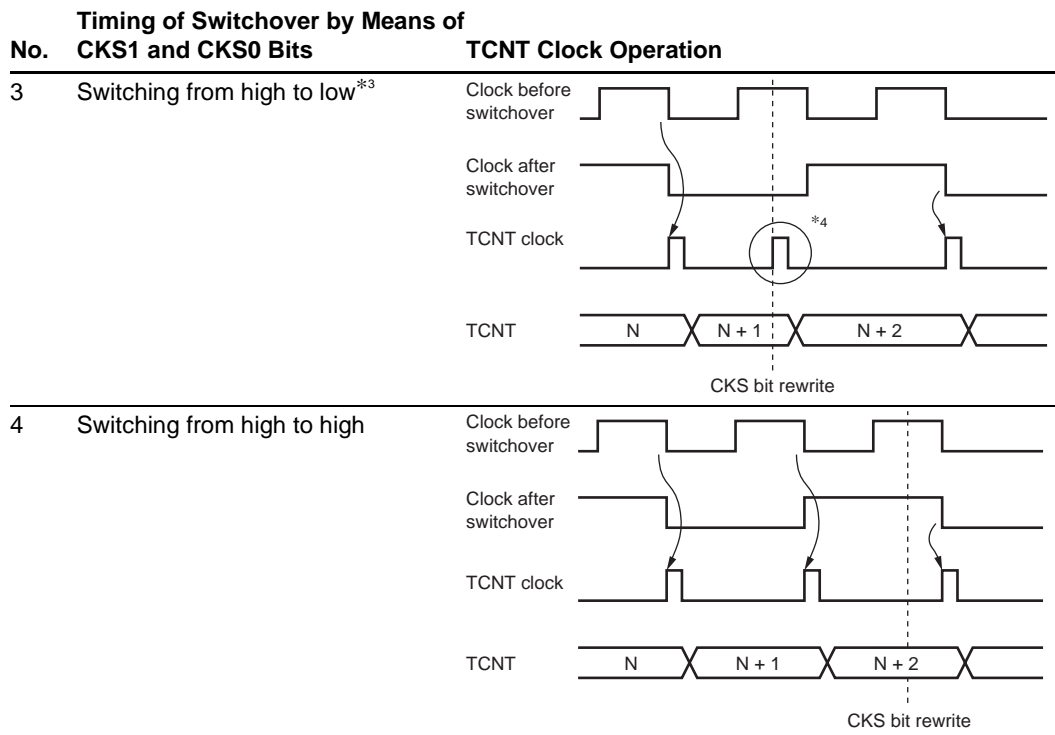
TCNT may increment erroneously when the internal clock is switched over. Table 12.4 shows the relationship between the timing at which the internal clock is switched (by writing to the CKS1 and CKS0 bits) and the TCNT operation

When the TCNT clock is generated from an internal clock, the falling edge of the internal clock pulse is detected. If clock switching causes a change from high to low level, as shown in no. 3 in table 12.4, a TCNT clock pulse is generated on the assumption that the switchover is a falling edge. This increments TCNT.

Erroneous incrementation can also happen when switching between internal and external clocks.

Table 12.4 Switching of Internal Clock and TCNT Operation

| No. | Timing of Switchover by Means of CKS1 and CKS0 Bits | TCNT Clock Operation |
|-----|--|------------------------|
| 1 | Switching from low to low ^{*1} | <p>CKS bit rewrite</p> |
| 2 | Switching from low to high ^{*2} | <p>CKS bit rewrite</p> |



- Notes:
1. Includes switching from low to stop, and from stop to low.
 2. Includes switching from stop to high.
 3. Includes switching from high to stop.
 4. Generated on the assumption that the switchover is a falling edge; TCNT is incremented.

12.8.6 Contention between Interrupts and Module Stop Mode

If module stop mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the DTC activation source. Interrupts should therefore be disabled before entering module stop mode.

12.8.7 Mode Setting of Cascaded Connection

When the 16-bit count mode and the compare-match count mode are set at the same time, input clocks for TCNT_0 and TCNT_1 (TCNT_2 and TCNT_3)* are not generated and the timer stops incrementation. This setting is prohibited.

Note: * Not available in the H8S/2237 Group and H8S/2227 Group.

Section 13 Watchdog Timer (WDT)

The watchdog timer (WDT) is an 8-bit timer that can generate an internal reset signal for this LSI if a system crash prevents the CPU from writing to the timer counter, thus allowing it to overflow.

When this watchdog function is not needed, the WDT can be used as an interval timer. In interval timer operation, an interval timer interrupt is generated each time the counter overflows.

The block diagram of the WDT is shown in figure 13.1.

13.1 Features

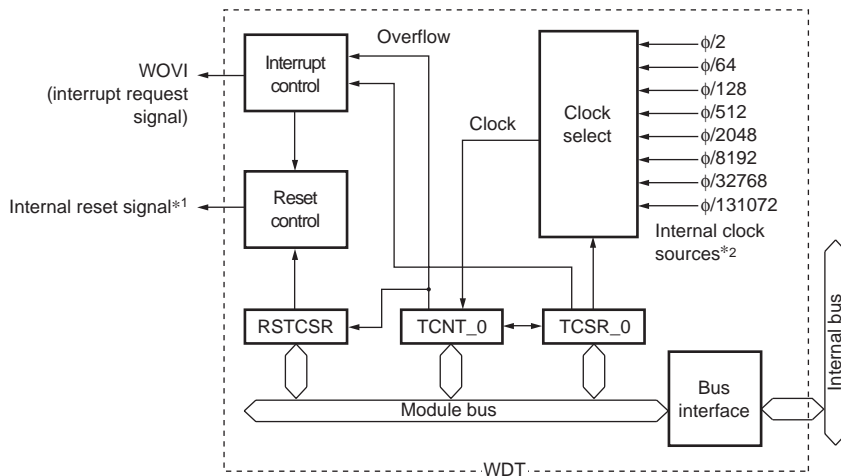
- Selectable from 8 counter input clocks for WDT_0
Selectable from 16 counter input clocks for WDT_1
- Switchable between watchdog timer mode and interval timer mode

In watchdog timer mode

- Choosable between power-on reset or manual reset as internal reset
- If the counter in WDT_0 overflows, it is possible to select whether this LSI is internally reset or not
- If the counter in WDT_1 overflows, it is possible to select whether this LSI is internally reset or the internal NMI interrupt is generated

In interval timer mode

- If the counter overflows, the WDT generates an interval timer interrupt (WOVI)
- The selected clock can be output from the BUZZ output pin (WDT_1)

**Legend:**

TCSR_0: Timer control/status register0

TCNT_0: Timer counter0

RSTCSR: Reset control/status register

Notes: 1. The type of internal reset signal depends on a register setting.

The power-on reset or manual reset can be selected as the internal reset.

2. When a sub-clock is operating in power-down mode, φ will be φ_{SUB}.**Figure 13.1 Block Diagram of WDT_0 (1)**

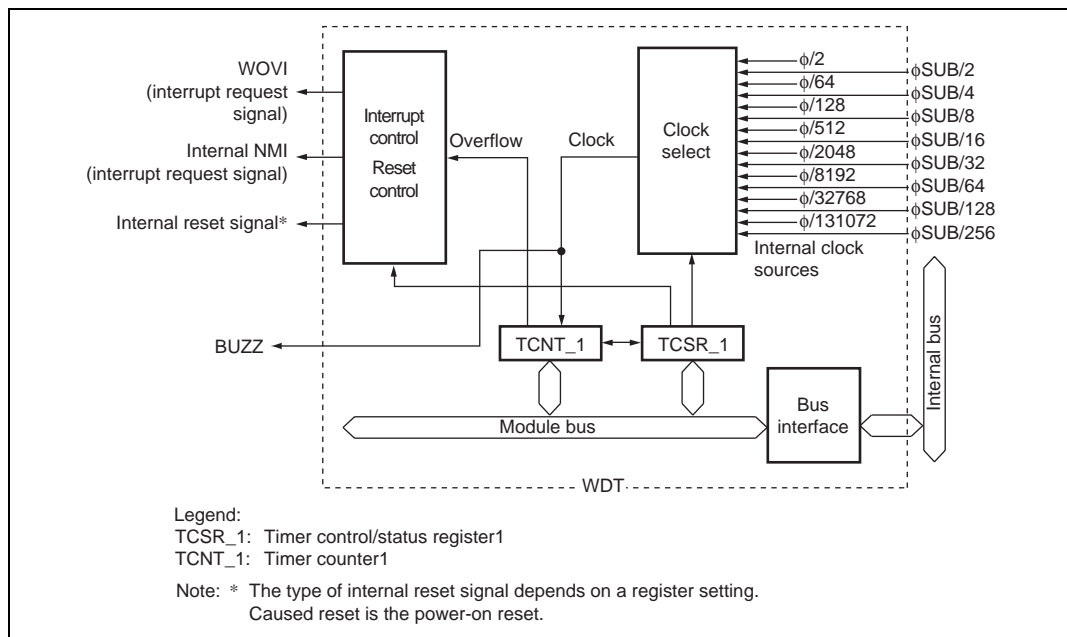


Figure 13.1 Block Diagram of WDT_1 (2)

13.2 Input/Output Pins

Table 13.1 Pin Configuration

| Name | Symbol | I/O | Function |
|---------------|--------|--------|------------------------------------|
| Buzzer Output | BUZZ | Output | Output the clock selected by WDT_1 |

13.3 Register Descriptions

The WDT has the following three registers. To prevent accidental overwriting, TCSR, TCNT, and RSTCSR have to be written to by a different method to normal registers. For details, refer to section 13.6.1, Notes on Register Access. For details on the system control register and pin function control register, refer to section 3.2.2, System Control Register (SYSCR) and section 7.3.6, Pin Function Control Register (PFCR), respectively.

- Timer counter (TCNT)
- Timer control/status register (TCSR)
- Reset control/status register (RSTCSR)

13.3.1 Timer Counter (TCNT)

TCNT is an 8-bit readable/writable up-counter. TCNT is initialized to H'00 when the TME bit in TCSR is cleared to 0.

To initialize TCNT to H'00 while the timer is operating, write H'00 to TCNT directly. See 13.6.7, Notes on Initializing TCNT by Using the TME Bit.

13.3.2 Timer Control/Status Register (TCSR)

TCSR functions include selecting the clock source to be input to TCNT and the timer mode.

- TCSR_0

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|---------------------|---|
| 7 | OVF | 0 | R/(W)* ¹ | <p>Overflow Flag</p> <p>Indicates that TCNT has overflowed. Only a 0 can be written to this bit, to clear the flag.</p> <p>[Setting condition]</p> <p>When TCNT overflows (changes from H'FF to H'00)</p> <p>When internal reset request generation is selected in watchdog timer mode, OVF is cleared automatically by the internal reset.</p> <p>[Clearing condition]</p> <p>Cleared by reading TCSR*² when OVF = 1, then writing 0 to OVF</p> |
| 6 | WT/IT | 0 | R/W | <p>Timer Mode Select</p> <p>Selects whether the WDT is used as a watchdog timer or interval timer.</p> <p>0: Interval timer mode (an interval timer interrupt (WOVI) is requested to CPU)</p> <p>1: Watchdog timer mode (internal reset selectable)</p> |
| 5 | TME | 0 | R/W | <p>Timer Enable</p> <p>When this bit is set to 1, TCNT starts counting. When this bit is cleared, TCNT stops counting and is initialized to H'00.</p> |
| 4, 3 | — | All 1 | — | <p>Reserved</p> <p>These bits are always read as 1 and cannot be modified.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | CKS2 | 0 | R/W | Clock Select 0 to 2 |
| 1 | CKS1 | 0 | R/W | Selects the clock source to be input to TCNT. The overflow frequency ^{*3} for $\phi = 10$ MHz is enclosed in parentheses. 000: Clock $\phi/2$ (frequency: 51.2 μ s) 001: Clock $\phi/64$ (frequency: 1.6 ms) 010: Clock $\phi/128$ (frequency: 3.2 ms) 011: Clock $\phi/512$ (frequency: 13.2 ms) 100: Clock $\phi/2048$ (frequency: 52.4 ms) 101: Clock $\phi/8192$ (frequency: 209.8 ms) 110: Clock $\phi/32768$ (frequency: 838.8 ms) 111: Clock $\phi/131072$ (frequency: 3.36 s) |
| 0 | CKS0 | 0 | R/W | |

- Notes:
1. Only 0 can be written, for flag clearing.
 2. When the OVF flag is polled with the interval timer interrupt disabled, read the OVF bit while it is 1 at least twice.
 3. The overflow period is the time from when TCNT starts counting up from H'00 until overflow occurs.

• TCSR_1

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-------------|---|
| 7 | OVF | 0 | R/(W) *1 | <p>Overflow Flag</p> <p>Indicates that TCNT has overflowed. Only a 0 can be written to this bit, to clear the flag.</p> <p>[Setting condition]</p> <p>When TCNT overflows (changes from H'FF to H'00)</p> <p>When internal reset request generation is selected in watchdog timer mode, OVF is cleared automatically by the internal reset.</p> <p>[Clearing condition]</p> <p>Cleared by reading TCSR*² when OVF = 1, then writing 0 to OVF</p> |
| 6 | WT/IT | 0 | R/W | <p>Timer Mode Select</p> <p>Selects whether the WDT is used as a watchdog timer or interval timer.</p> <p>0: Interval timer mode (an interval timer interrupt (WOVI) is requested to CPU)</p> <p>1: Watchdog timer mode (a power-on reset or NMI interrupt is requested to CPU)</p> |
| 5 | TME | 0 | R/W | <p>Timer Enable</p> <p>When this bit is set to 1, TCNT starts counting. When this bit is cleared, TCNT stops counting and is initialized to H'00.</p> |
| 4 | PSS | 0 | R/W | <p>Prescaler Select</p> <p>Selects the clock source input to TCNT of WDT_1</p> <p>0: TCNT counts divided clock of ϕ-base prescaler (PSM)</p> <p>1: TCNT counts divided clock of ϕ_{SUB}-base prescaler (PSS)</p> |
| 3 | RST/NMI | 0 | R/W | <p>Reset or NMI (RST/NMI)</p> <p>When TCNT overflows in watchdog timer mode, either a power-on reset or NMI interrupt is selected.</p> <p>0: An NMI interrupt is requested</p> <p>1: Reset is requested</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | CKS2 | 0 | R/W | Clock Select 0 to 2 |
| 1 | CKS1 | 0 | R/W | Selects the clock source to be input to TCNT. The overflow frequency ^{*3} for $\phi = 10$ MHz is enclosed in parentheses. |
| 0 | CKS0 | 0 | R/W | |
| | | | | When PSS = 0: |
| | | | | 000: Clock $\phi/2$ (frequency: 51.2 μ s) |
| | | | | 001: Clock $\phi/64$ (frequency: 1.6 ms) |
| | | | | 010: Clock $\phi/128$ (frequency: 3.2 ms) |
| | | | | 011: Clock $\phi/512$ (frequency: 13.2 ms) |
| | | | | 100: Clock $\phi/2048$ (frequency: 52.4 ms) |
| | | | | 101: Clock $\phi/8192$ (frequency: 209.8 ms) |
| | | | | 110: Clock $\phi/32768$ (frequency: 838.8 ms) |
| | | | | 111: Clock $\phi/131072$ (frequency: 3.36 s) |
| | | | | When PSS = 1: |
| | | | | 000: Clock $\phi_{\text{SUB}}/2$ (frequency: 15.6 ms) |
| | | | | 001: Clock $\phi_{\text{SUB}}/4$ (frequency: 31.3 ms) |
| | | | | 010: Clock $\phi_{\text{SUB}}/8$ (frequency: 62.5 ms) |
| | | | | 011: Clock $\phi_{\text{SUB}}/16$ (frequency: 125 ms) |
| | | | | 100: Clock $\phi_{\text{SUB}}/32$ (frequency: 250 ms) |
| | | | | 101: Clock $\phi_{\text{SUB}}/64$ (frequency: 500 ms) |
| | | | | 110: Clock $\phi_{\text{SUB}}/128$ (frequency: 1 s) |
| | | | | 111: Clock $\phi_{\text{SUB}}/256$ (frequency: 2 s) |

- Notes:
1. Only 0 can be written, for flag clearing.
 2. When the OVF flag is polled with the interval timer interrupt disabled, read the OVF bit while it is 1 at least twice
 3. The overflow period is the time from when TCNT starts counting up from H'00 until overflow occurs.

13.3.3 Reset Control/Status Register (RSTCSR) (only WDT_0)

RSTCSR controls the generation of the internal reset signal when TCNT overflows, and selects the type of internal reset signal. RSTCSR is initialized to H'1F by a reset signal from the RES pin, and not by the WDT internal reset signal caused by overflows.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|--------|---|
| 7 | WOVF | 0 | R/(W)* | <p>Watchdog Overflow Flag</p> <p>This bit is set when TCNT overflows in watchdog timer mode. This bit cannot be set in interval timer mode, and only 0 can be written, to clear the flag.</p> <p>[Setting condition]</p> <p>Set when TCNT overflows (changed from H'FF to H'00) in watchdog timer mode</p> <p>[Clearing condition]</p> <p>Cleared by reading RSTCSR when WOVF = 1, and then writing 0 to WOVF</p> |
| 6 | RSTE | 0 | R/W | <p>Reset Enable</p> <p>Specifies whether or not a reset signal is generated in the chip if TCNT overflows during watchdog timer operation.</p> <p>0: Reset signal is not generated even if TCNT overflows (Though this LSI is not reset, TCNT and TCSR in WDT are reset)</p> <p>1: Reset signal is generated if TCNT overflows</p> |
| 5 | RSTS | 0 | R/W | <p>Reset Select</p> <p>This bit selects the type of the internal reset that is generated by TCNT overflowing in watchdog timer mode.</p> <p>0: Power-on reset</p> <p>1: Manual reset</p> |
| 4 to 0 | — | All 1 | — | <p>Reserved</p> <p>These bits are always read as 1 and cannot be modified.</p> |

Note: * Only 0 can be written, to clear the flag.

13.4 Operation

13.4.1 Watchdog Timer Mode

To use the WDT as a watchdog timer, set the $\overline{WT/IT}$ bit in TCSR and the TME bit to 1.

Software must prevent TCNT overflows by rewriting the TCNT value (normally by writing H'00) before overflows occurs. Thus, TCNT does not overflow while the system is operating normally.

When the WDT is used as a watchdog timer and the RSTE bit in RSTCSR of WDT_0 is set to 1, and if TCNT overflows without being rewritten because of a system malfunction or other error, an internal reset signal for this LSI is output for 518 system clocks.

When the $\overline{RST/NMI}$ bit in TCSR of WDT_1 is set to 1, and if TCNT overflows, the internal reset signal is output for 516 system clock periods. When the $\overline{RST/NMI}$ bit is cleared to 0, an NMI interrupt request is generated (for 515 or 516 system clock periods when the clock source is set to ϕ_{SUB} (PSS = 1)).

An internal reset request from the watchdog timer and a reset input from the \overline{RES} pin are both treated as having the same vector. If a WDT internal reset request and the \overline{RES} pin reset occur at the same time, the \overline{RES} pin reset has priority and the WOVF bit in RSTCSR is cleared to 0.

An NMI request from the watchdog timer and an interrupt request from the NMI pin are both treated as having the same vector. So, avoid handling an NMI request from the watchdog timer and an interrupt request from the NMI pin at the same time.

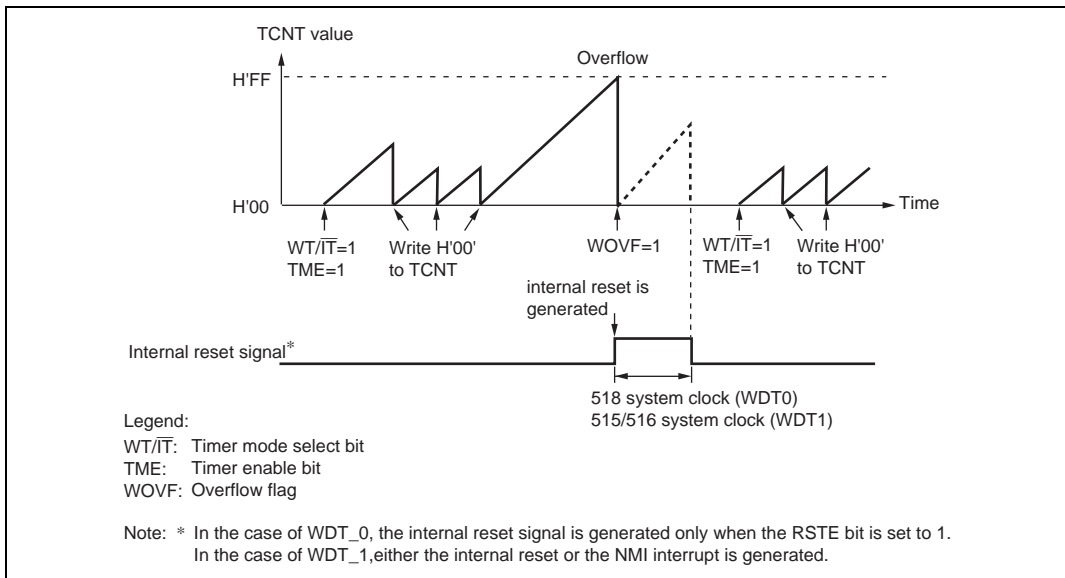


Figure 13.2 Watchdog Timer Mode Operation

13.4.2 Interval Timer Mode

To use the WDT as a watchdog timer, set the $\overline{\text{WT/IT}}$ and TME bits in TCSR to 1.

When the WDT is used as an interval timer, an interval timer interrupt (WOVI) is generated each time the TCNT overflows. (The NMI interrupt is not generated.) Therefore, an interrupt can be generated at intervals.

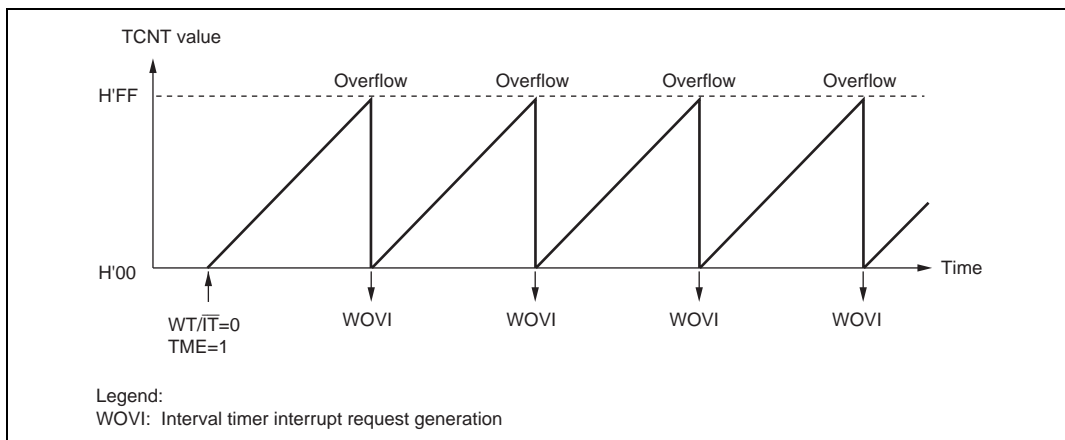


Figure 13.3 Interval Timer Mode Operation

13.4.3 Timing of Setting Overflow Flag (OVF)

The OVF flag is set to 1 if TCNT overflows during interval timer operation. At the same time, an interval timer interrupt (WOVI) is requested. This timing is shown in figure 13.4.

When NMI request is chosen in watchdog timer mode for WDT_1, TCNT overflow sets the OVF flag to 1. At the same time, NMI interrupt is requested.

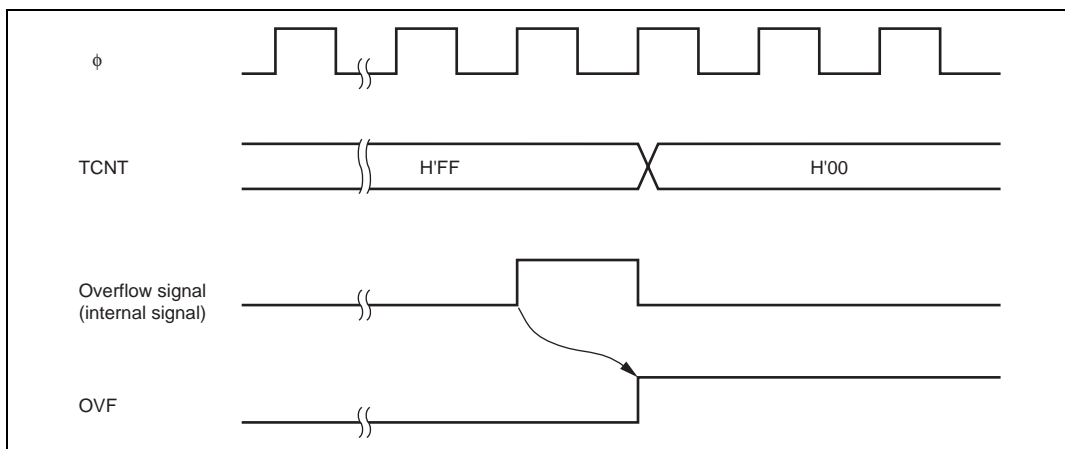


Figure 13.4 Timing of OVF Setting

13.4.4 Timing of Setting Watchdog Timer Overflow Flag (WOVF)

With WDT_0 the WOVF bit in RSTCSR is set to 1 if TCNT overflows in watchdog timer mode. If TCNT overflows while the RSTE bit in RSTCSR is set to 1, an internal is generated for the entire chip. (The WOVF interrupt is not generated.) This timing is illustrated in figure 13.5.

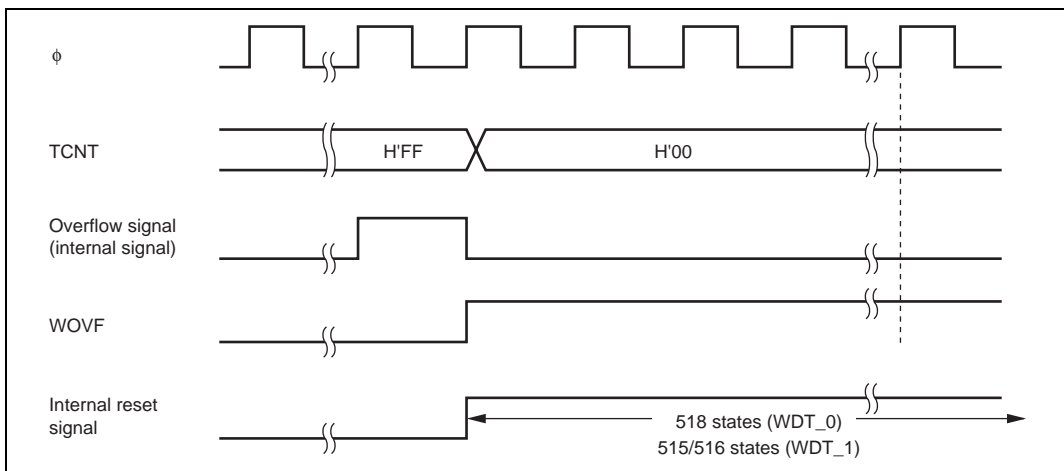


Figure 13.5 Timing of WOVF Setting

13.5 Interrupt Sources

During interval timer mode operation, an overflow generates an interval timer interrupt (WOVI). The interval timer interrupt is requested whenever the OVF flag is set to 1 in TCSR. OVF must be cleared to 0 in the interrupt handling routine.

If an NMI interrupt request has been chosen in the watchdog timer mode, an NMI interrupt request is generated when a TCNT overflow occurs.

Table 13.2 WDT Interrupt Source

| Name | Interrupt Source | Interrupt Flag |
|------|-------------------------------------|----------------|
| WOVI | TCNT overflow (interval timer mode) | OVF |
| NMI | TCNT overflow (watchdog timer mode) | OVF |

13.6 Usage Notes

13.6.1 Notes on Register Access

The write method for TCNT, TCSR, and RSTCSR differs from that of normal registers so that they cannot be easily rewritten. Use the following procedures to read and write these registers.

(1) Writing to TCNT and TCSR

Word transfer instructions must be used to write to TCNT and TCSR. These registers cannot be written with byte transfer instructions. This is shown in figure 13.6.

For writing, TCNT and TCSR are allocated to the same address. To write to TCNT, transfer a word in which the upper byte is H'5A and the lower byte is the write data. To write to TCSR, transfer a word in which the upper byte is H'A5 and the lower byte is the write data. When these transfer operations are performed, the lower byte data is written to TCNT or TCSR.

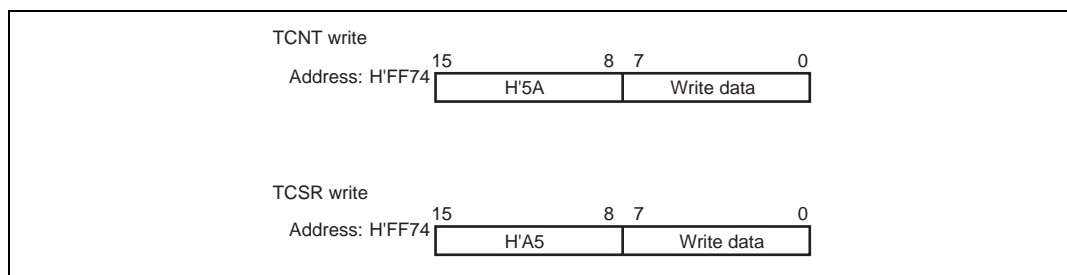


Figure 13.6 Writing to TCNT, TCSR

(2) Writing to RSTCSR

Use word transfer operations to write to RSTCSR. This register cannot be written using byte transfer instructions. This is shown in figure 13.7.

The method used to write a 0 to the WOVF bit and the method used to write the RSTE and RSTS bits are different.

To write a 0 to the WOVF bit, set the upper byte to H'A5 and the lower byte to H'00 and transfer that data. This will clear the WOVF bit to 0. This operation does not affect the RSTE and RSTS bits. To write the RSTE and RSTS bits, set the upper byte to H'5A and the lower byte to the data to be written and transfer that data. This will write the data in bits 6 and 5 of the lower byte to the RSTE and RSTS bits. This operation does not affect the WOVF bit.

When writing 0 to the WOVF bit



When writing to the RSTE and RSTS bits

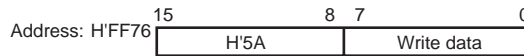


Figure 13.7 Writing to RSTCSR

(3) Reading from TCNT, TCSR, and RSTCSR

These registers can be read in the same way normal registers are read. TCSR is allocated at address H'FF74, TCNT at address H'FF75, and RSTCSR at address H'FF77.

13.6.2 Contention between Timer Counter (TCNT) Write and Increment

If a timer counter clock pulse is generated during the T_2 state of a TCNT write cycle, the write takes priority and the timer counter is not incremented. Figure 13.8 shows this operation.

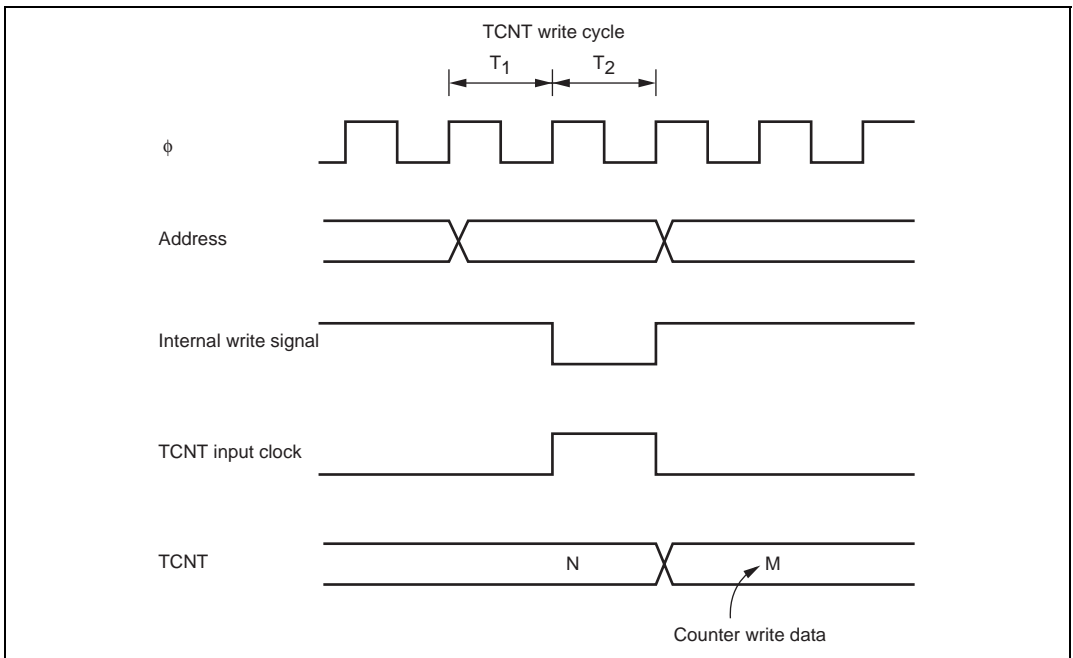


Figure 13.8 Contention between TCNT Write and Increment

13.6.3 Changing Value of PSS or CKS2 to CKS0

If the PSS or CKS0 to CKS2 bits in TCSR are written to while the WDT is operating, errors could occur in the incrementation. Software must be used to stop the watchdog timer (by clearing the TME bit to 0) before changing the value of the PSS or CKS0 to CKS2 bits.

13.6.4 Switching between Watchdog Timer Mode and Interval Timer Mode

If the mode is switched from watchdog timer to interval timer while the WDT is operating, errors could occur in the incrementation. Software must be used to stop the watchdog timer (by clearing the TME bit to 0) before switching the mode.

13.6.5 Internal Reset in Watchdog Timer Mode

This LSI is not reset internally if TCNT overflows while the RSTE bit is cleared to 0 during watchdog timer operation, however TCNT_0 and TCSR_0 of the WDT_0 are reset.

TCNT, TCSR, or RSTCR cannot be written to for 132 states following an overflow. During this period, any attempt to read the WOVF flag is not acknowledged. Accordingly, wait 132 states after overflow to write 0 to the WOVF flag for clearing.

13.6.6 OVF Flag Clearing in Interval Timer Mode

When the OVF flag setting conflicts with the OVF flag reading in interval timer mode, writing 0 to the OVF bit may not clear the flag even though the OVF bit has been read while it is 1. If there is a possibility that the OVF flag setting and reading will conflict, such as when the OVF flag is polled with the interval timer interrupt disabled, read the OVF bit while it is 1 at least twice before writing 0 to the OVF bit to clear the flag.

13.6.7 Notes on Initializing TCNT by Using the TME Bit

When the ϕ SUB (subclock) division clock is selected as the TCNT input clock (PSS in TCSR set to 1) and, after TME in TCSR is cleared to 0 to initialize the counter (TCNT) while the counter (TCNT) is operating in the high-speed mode or medium-speed mode, TCNT is restarted by setting TME to 1 once again, TCNT may not be correctly initialized.

In such cases, use either of the following methods to initialize TCNT:

- (1) Write H'00 to TCNT.
- (2) In subtractive mode, clear the TME bit to 0.

Section 14 IEBus™ Controller (IEB) [H8S/2258 Group]

This LSI has an on-chip one-channel IEBus™ controller (IEB). The Inter Equipment Bus™ (IEBus™)*¹ is a small-scaled digital data transfer system for inter equipment data transfer.

This LSI does not have an on-chip IEBus driver/receiver, so it is necessary to mount a dedicated driver/receiver*² externally.

Notes: 1. IEBus is a trademark of NEC Electronics Corporation.
2. Bus interface driver/receiver IC: HA12187FP is recommended.

14.1 Features

- IEBus protocol control (layer 2) supported
 - Half duplex asynchronous communications
 - Multi-master system
 - Broadcast communications function
 - Selectable mode (three types) with different transfer speeds
- Data transfer by the data transfer controller (DTC)
 - Transfer buffer: 1 byte
 - Reception buffer: 1 byte
 - Up to 128 bytes of consecutive transfer/reception (maximum number of transfer bytes in mode 2)
- Operating frequency
 - 12 MHz, 12.58 MHz (IEB uses 1/2 divided external clock)

Note: $\pm 1.5\%$ when mode 0 or 1 is used, $\pm 0.5\%$ when mode 2 is used
- Noise resistance is improved by mounting the IEBus driver/receiver (layer 1) externally
- Module stop mode can be set

Figure 14.1 shows an IEB block diagram.

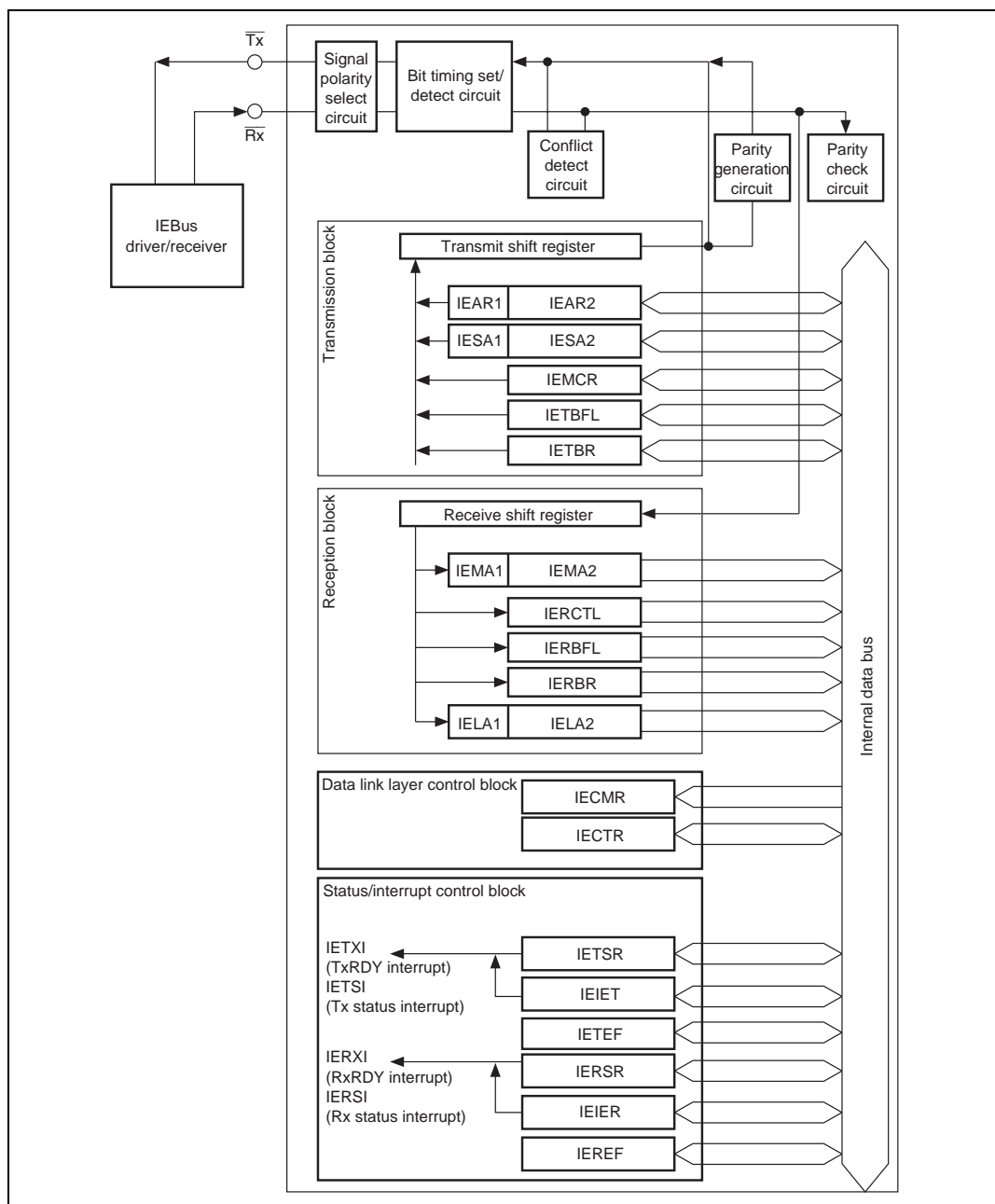


Figure 14.1 Block Diagram of IEB

14.1.1 IEBus Communications Protocol

The overview of the IEBus is described below.

- Communications method: Half duplex asynchronous communications
- Multi-master system
All units connected to the IEBus can transfer data to other units.
- Broadcast communications function (one-to-many communications)
 - Group broadcast communications: Broadcast communications to group unit
 - General broadcast communications: Broadcast communications to all units
- Mode is selectable (three modes with different transfer speeds).

Table 14.1 Mode Types

| Mode | $\phi = 12 \text{ MHz}$ | $\phi = 12.58 \text{ MHz}$ | Maximum Number of Transfer Bytes (byte/frame) |
|------|-------------------------|----------------------------|---|
| 0 | About 3.9 kbps | About 4.1 kbps | 16 |
| 1 | About 17 kbps | About 18 kbps | 32 |
| 2 | About 26 kbps | About 27 kbps | 128 |

- Access control: CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
Priority of bus mastership is as follows.
 - Broadcast communications (one-to-many communications) have priority rather than normal communications (one-to-one communications).
 - Smaller master address has priority.
- Communications scale
 - Number of units: Up to 50
 - Cable length: Up to 150 m (when using a twisted pair cable)

Note: The communications scale of the actual system depends on the externally mounted IEBus driver/receiver characteristics and the characteristics of the cable to be used.

(1) Determination of Bus Mastership (Arbitration)

A unit connected to the IEBus performs an operation for getting the bus to control other units. This operation is called arbitration. In arbitration, when the multiple units start transfer simultaneously, the bus mastership is given to one unit among them.

Only one unit can get bus mastership through arbitration, so the following priority for bus mastership is defined.

(a) Priority according to communications type

Broadcast communications (one-to-many communications) has priority over normal communications (one-to-one communications).

(b) Priority according to master address

A unit with the smallest master address has priority among units with the same communications type.

Example: The master address is configured with 12 bits. A unit with H'000 has the highest priority, and a unit with H'FFF has the lowest priority.

Note: When a unit loses arbitration, the unit can automatically enter retransfer mode (0 to 7 retransfer times can be selected by bits RN2 to RN0 in IEMCR).

(2) Communications Mode

The IEBus has three communications modes with different transfer speeds. Table 14.2 shows the transfer speed in each communications mode and the maximum number of transfer bytes in one communications frame.

Table 14.2 Transfer speed and Maximum Number of Transfer Bytes in Each Communications Mode

| Communications Mode | Maximum Number of Transfer Bytes (byte/frame) | Effective Transfer Speed* ¹ (kbps) | |
|---------------------|---|---|---------------------------------|
| | | $\phi = 12 \text{ MHz}^{*2}$ | $\phi = 12.58 \text{ MHz}^{*2}$ |
| 0 | 16 | About 3.9 | About 4.1 |
| 1 | 32 | About 17 | About 18 |
| 2 | 128 | About 26 | About 27 |

Notes: Each unit connected to the IEBus should select a communications mode prior to performing communications. Note that correct communications is not guaranteed if the master and slave units do not adopt the same communications mode.

In the case of communications between a unit with $\phi = 12 \text{ MHz}$ and a unit with $\phi = 12.58 \text{ MHz}$, correct communications is not possible even if the same communications mode is adopted. Communications must be performed at the same oscillation frequency.

1. An effective transfer speed when the maximum number of transfer bytes is transmitted.
2. Oscillation frequency when this LSI is used

(3) Communications Address

In the IEBus, a 12-bit specific communications addresses are allocated to individual units. A communications address is configured as follows.

- Upper four bits: group number (number identifying a group to which the unit belongs)
- Lower eight bits: unit number (number identifying individual units in a group)

(4) Broadcast Communications

In normal transfer, a single master unit communicates with a single slave unit. So, one-to-one transfer or reception is performed. In broadcast communications, a single master unit communicates with multiple slave units. Since there are multiple slave units, acknowledgement is not returned from the slave units during communications.

A broadcast bit decides whether broadcast or normal communications is performed. (For details of the broadcast bit, see section 14.1.2 (1) (b), Broadcast Bit.

There are two types of broadcast communications.

(a) Group broadcast communications

Broadcast communications is performed to units with the same group number, meaning that those units have the same upper four bits of the communications address.

(b) General broadcast communications

Broadcast communications is performed to all units regardless of the group number.

Group broadcast and general broadcast communications are identified by a slave address. (For details on the slave address, see section 14.1.2 (3), Slave Address Field.)

14.1.2 Communications Protocol

Figure 14.2 shows an IEBus transfer signal format.

Communications data is transferred as a series of signals referred to as a communications frame. The number of data which can be transmitted in a single communications frame and the transfer speed differ according to communications mode.

(When $\phi = 12 \text{ MHz}$)

| Field name | Header | | Master address field | | Slave address field | | | Control field | | | Message length field | | | Data field | | | | | |
|----------------|----------------------------------|---------------|----------------------|---|---------------------|---|---|---------------|---|---|----------------------|---|---|---|---|---|--|--|--|
| Number of bits | 1 | 1 | 12 | 1 | 12 | 1 | 1 | 4 | 1 | 1 | 8 | 1 | 1 | 8 | 1 | 1 | | | |
| Transfer time | Start bit | Broadcast bit | Master address | P | Slave address | P | A | Control bits | P | A | Message length bits | P | A | Data bits | P | A | | | |
| Mode 0 | Approximately 7330 μs | | | | | | | | | | | | | Approximately 1590 $\times N \mu\text{s}$ | | | | | |
| Mode 1 | Approximately 2090 μs | | | | | | | | | | | | | Approximately 410 $\times N \mu\text{s}$ | | | | | |
| Mode 2 | Approximately 1590 μs | | | | | | | | | | | | | Approximately 300 $\times N \mu\text{s}$ | | | | | |

P: Parity bit (1 bit)

A: Acknowledge bit (1 bit)

When A = 0: ACK

When A = 1: NAK

N: Number of bytes

Note: The value of acknowledge bit is ignored in broadcast communications.

Figure 14.2 Transfer Signal Format**(1) Header**

Header is comprised of a start bit and a broadcast bit.

(a) Start Bit

The start bit is a signal for informing a start of data transfer to other units. A unit, which attempts to start data transfer, outputs a low-level signal (start bit) for a specified period and then outputs the broadcast bit.

If another unit is already outputting a start bit when a unit attempts to output a start bit, the unit waits for completion of output of the start bit from the other unit without outputting the start bit, and then outputs the broadcast bit synchronized with the completion timing.

Other units enter the receive state after detecting the start bit.

(b) Broadcast Bit

The broadcast bit is a bit to identify the type of communications: broadcast or normal.

When this bit is cleared to 0, it indicates the broadcast communications. When it is set to 1, it indicates the normal communications. Broadcast communications includes group broadcast and general broadcast, which are identified by a value of the slave address. (For details of the slave address, see section 14.1.2 (3), Slave Address Field.)

Since there are multiple slave units, which are communications destination units, in the case of broadcast communications, the acknowledge bit is not returned from each field described in (2) and below.

When more than one unit starts transfer of communications frame at the same timing, broadcast communications has priority over normal communications, and arbitration occurs.

(2) Master Address Field

The master address field is a field for transmitting the unit address (master address) to other units. The master address field is comprised of master address bits and a parity bit.

The master address has 12 bits and are output MSB first.

When more than one unit starts transfer of the broadcast bit having the same value at the same timing, arbitration is decided by the master address field.

In the master address field, self-output data and data on the bus are compared for every one-bit transfer. If the self-output master address and data on the bus are different, the unit that loses arbitration, stops transfer, and enters the receive state.

Since the IEBus is configured with wired AND, a unit having the smallest master address of the units in arbitration (arbitration master) wins in arbitration.

Finally, only a single unit remains in the transfer state as a master unit after outputting 12-bit master address.

Next, this master unit outputs a parity bit*, defines the master address to other units, and then enters the slave address field output state.

Note: * Since even parity is used, when the number of one bits in the master address is odd, the parity bit is 1.

(3) Slave Address Field

The slave address field is a field to transmit an address (slave address) of a unit (slave unit) to which a master transmit data. The slave address field is comprised of slave address bits, a parity bit, and an acknowledge bit.

The slave address has 12 bits and is output MSB first. The parity bit is output after the 12-bit slave address is transmitted in order to avoid receiving the slave address accidentally. The master unit then detects the acknowledgement from the slave unit in order to confirm that the slave unit exists on the bus. When the acknowledgement is detected, the master unit enters the control field output state. However, the master unit enters the control field output state without detecting the acknowledgement in broadcast communications.

The slave unit returns the acknowledgement when the slave addresses match and the parities of the master and slave addresses are correct. When either of the parities of the master and slave addresses is wrong, the slave unit decides that the master or slave address is not correctly received and does not return the acknowledgement. In this case, the master unit enters the waiting (monitor) state, and communications end.

In the case of broadcast communications, the slave address is used to identify the type of broadcast communications (group or general) as follows:

- When the slave address is H'FFF: General broadcast communications
- When the slave address is other than H'FFF: Group broadcast communications

Note: The group number is the upper 4-bit value of the slave address in group broadcast communications.

(4) Control Field

The control field is a field for transmitting the type and direction of the following data field. The control field is comprised of control bits, a parity bit, and an acknowledge bit.

The control bits include four bits and are output MSB first.

The parity bit is output following the control bits. When the parity is correct, and the slave unit can implement the function required from the master unit, the slave unit returns the acknowledgement and enters the message length field output state. However, if the slave unit cannot implement the requirements from the master unit even though the parity is correct, or if the parity is not correct, the slave unit does not return the acknowledgement, and returns to the waiting (monitor) state.

The master unit enters the subsequent message length field output state after confirming the acknowledgement.

When the acknowledgement is not confirmed, the master unit enters the waiting (monitor) state, and communications end. However, in the case of broadcast communications, the master unit enters the following message length field output state without confirming the acknowledgement. For details of the contents of the control bit, see table 14.4.

(5) Message Length Field

The message length field is a field for specifying the number of transfer bytes. The message length field is comprised of message length bits, a parity bit, and an acknowledge bit.

The message length has eight bits and is output MSB first. Table 14.3 shows the number of transfer bytes.

Table 14.3 Contents of Message Length Bits

| Message Length bits (Hexadecimal) | Number of Transfer Bytes |
|-----------------------------------|--------------------------|
| H'01 | 1 byte |
| H'02 | 2 bytes |
| . | . |
| . | . |
| H'FF | 255 bytes |
| H'00 | 256 bytes |

Note: * If a number greater than the maximum number of transfer bytes in one frame is specified, communications are performed in multiple frames depending on the communications mode. In this case, the message length bits indicate the number of remaining communications data after the first transfer. In this LSI, after the first transfer, the message length bits must be specified to the number of remaining communications data by a program, since these bits are not automatically specified by the hardware.

This field operation differs depending on the value of bit 3 in the control field: master transmission (bit 3 in the control bits is 1) or master reception (bit 3 in the control bits is 0).

(a) Master Transmission

The master unit outputs the message length bits and parity bit. When the parity is correct, the slave unit returns the acknowledgement and enters the following data field. Note that the slave unit does not return the acknowledgement in broadcast communications.

In addition, when the parity is not correct, the slave unit decides that the message length field is not correctly received, does not return the acknowledgement, and returns to the waiting (monitor) state. In this case, the master unit also returns to the waiting state, and communications end.

(b) Master Reception

The slave unit outputs the message length bits and parity bit. When the parity is correct, the master unit returns the acknowledgement.

When the parity is not correct, the master unit decides that the message length bits are not correctly received, does not return the acknowledgement, and returns to the waiting state. In this case, the slave unit also returns to the waiting state, and communications end.

(6) Data Field

The data field is a field for data transmission/reception to the slave unit. The master unit transmits/receives data to/from the slave unit using the data field. The data field is comprised of data bits, a parity bit, and an acknowledge bit.

The data bits include eight bits and are output MSB first.

The parity bit and acknowledge bit following the data bits are output from the master unit and slave unit, respectively.

Broadcast communications are performed only for the transmission of the master unit. In this case, the acknowledge bit is ignored. Operations in master transmission and master reception are described below.

(a) Master Transmission

The master unit transmits the data bits and parity bit to the slave unit to write data from the master unit to the slave unit. The slave unit receives the data bits and parity bit, and returns the acknowledgement if the parity bit is correct and the receive buffer is empty. If the parity bit is not correct or the receive buffer is not empty, the slave unit rejects acceptance of corresponding data and does not return the acknowledgement.

When the slave unit does not return the acknowledgement, the master unit retransmits the same data. This operation is repeated until either the acknowledgement from the slave unit is detected or the maximum number of data transfer bytes is exceeded.

When the parity is correct and the acknowledgement is output from the slave unit, the master unit transmits the subsequent data if data remains and the maximum number of transfer bytes is not exceeded.

In the case of broadcast communications, the slave unit does not return the acknowledgement, and the master unit transfers data byte by byte.

(b) Master Reception

The master unit outputs synchronous signals corresponding to all data bits to be read from the slave unit.

The slave unit outputs the data bits and parity bit on the bus in accordance with the synchronous signals from the master unit.

The master unit reads the parity bit output from the slave unit, and checks the parity. If the parity is not correct, or the receive buffer is not empty, the master unit rejects acceptance of the data, and does not return the acknowledgement. The master unit reads the same data repeatedly if the number of data does not exceed the maximum number of transfer bytes in one frame. If the parity is correct and the receive buffer is empty, the master unit accepts data and returns the

acknowledgement. The master unit reads in the subsequent data if the number of data does not exceed the maximum number of transfer bytes in one frame.

(7) Parity Bit

The parity bit is used to confirm that transfer data has no error.

The parity bit is added to respective data of the master address, slave address, control, message length, and data bits.

The even parity is used. When the number of one bits in data is odd, the parity bit is 1. When the number of one bits in data is even, the parity bit is 0.

(8) Acknowledge Bit

In normal communications (a single unit to a single unit communications), the acknowledge bit is added to the following position in order to confirm that data is correctly accepted.

- At the end of the slave address field
- At the end of the control field
- At the end of the message length field
- At the end of the data field

The acknowledge bit is defined below.

- 0: indicates that the transfer data is acknowledged. (ACK)
- 1: indicates that the transfer data is not acknowledged. (NAK)

Note that the acknowledge bit is ignored in the case of broadcast communications.

(a) Acknowledge bit at the End of the Slave Address Field

The acknowledge bit at the end of the slave address field becomes NAK in the following cases and transfer is stopped.

- When the parity of the master address or slave address bits is incorrect
- When a timing error (an error in bit format) occurs
- When there is no slave unit

(b) Acknowledge bit at the End of the Control Field

The acknowledge bit at the end of the control field becomes NAK in the following cases and transfer is stopped.

- When the parity of the control bits is incorrect

- When bit 3 in the control bits is 1 (data write) although the slave receive buffer* is not empty
- When the control bits are set to the data read (H'3, H'7) although the slave transmit buffer* is empty
- When another unit which locked the slave unit requests H'3, H'6, H'7, H'A, H'B, H'E, or H'F in the control bits although the slave unit has been locked
- When the control bits are the locked address read (H'4, H'5) although the unit is not locked
- When a timing error occurs
- When the control bits are undefined

Note: * See section 14.1.3 (1), Slave Status Read (Control Bits: H'0, H'6).

(c) Acknowledge Bit at the End of the Message Length Field

The acknowledge bit at the end of the message length field becomes NAK in the following cases and transfer is stopped.

- When the parity of the message length bits is incorrect
- When a timing error occurs

(d) Acknowledge Bit at the End of the Data Field

The acknowledge bit at the end of the data field becomes NAK in the following cases and transfer is stopped.

- When the parity of the data bits is incorrect*
- When a timing error occurs after the previous transfer of the acknowledge bit
- When the receive buffer becomes full and cannot accept further data

Note: * In this case, data field is transferred repeatedly until the number of data reaches the maximum number of transfer bytes if the number of data does not exceed the maximum number of transfer bytes in one frame.

14.1.3 Transfer Data (Data Field Contents)

The data field contents are specified by the control bits.

Table 14.4 Control Bit Contents

| Setting Value | Bit 3 ^{*1} | Bit 2 | Bit 1 | Bit 0 | Function ^{*2} |
|---------------|---------------------|-------|-------|-------|--------------------------------------|
| H'0 | 0 | 0 | 0 | 0 | Reads slave status (SSR) |
| H'1 | 0 | 0 | 0 | 1 | Undefined — do not use |
| H'2 | 0 | 0 | 1 | 0 | Undefined — do not use |
| H'3 | 0 | 0 | 1 | 1 | Reads data and locks |
| H'4 | 0 | 1 | 0 | 0 | Reads locked address (lower 8 bits) |
| H'5 | 0 | 1 | 0 | 1 | Reads locked address (upper 4 bits) |
| H'6 | 0 | 1 | 1 | 0 | Reads slave status (SSR) and unlocks |
| H'7 | 0 | 1 | 1 | 1 | Reads data |
| H'8 | 1 | 0 | 0 | 0 | Undefined — do not use |
| H'9 | 1 | 0 | 0 | 1 | Undefined — do not use |
| H'A | 1 | 0 | 1 | 0 | Writes command and locks |
| H'B | 1 | 0 | 1 | 1 | Writes data and locks |
| H'C | 1 | 1 | 0 | 0 | Undefined — do not use |
| H'D | 1 | 1 | 0 | 1 | Undefined — do not use |
| H'E | 1 | 1 | 1 | 0 | Writes command |
| H'F | 1 | 1 | 1 | 1 | Writes data |

Notes: 1. According to the value of bit 3 (MSB), the transfer directions of the message length bits in the following message length field and data in the data field vary.

When bit 3 is 1: Data is transferred from the master unit to the slave unit.

When bit 3 is 0: Data is transferred from the slave unit to the master unit.

2. H'3, H'6, H'A, and H'B are control bits to specify lock setting and cancellation.

When the undefined values of H'1, H'2, H'8, H'9, H'C, and H'D are transmitted, the acknowledge bit is not returned.

When the control bits received from another unit which locked are not included in table 14.5, the slave unit which has been locked by the master unit rejects acceptance of the control bits and does not return the acknowledge bit.

Table 14.5 Control Field for Locked Slave Unit

| Setting Value | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Function |
|---------------|-------|-------|-------|-------|-------------------------------------|
| H'0 | 0 | 0 | 0 | 0 | Reads slave status |
| H'4 | 0 | 1 | 0 | 0 | Reads locked address (upper 8 bits) |
| H'5 | 0 | 1 | 0 | 1 | Reads locked address (lower 4 bits) |

(1) Slave Status Read (Control Bits: H'0, H'6)

The master unit can decide the reason the slave unit does not return the acknowledgement (ACK) by reading the slave status (H'0, H'6). The slave status indicates the result of the last communications that the slave unit performs. All slave units can provide slave status information. Figure 14.3 shows bit configuration of the slave status.

| MSB | | | | LSB | | | |
|-----------------|-------|------------------------------------|---|-------|-------|-------|-------|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| Bit | Value | Description | | | | | |
| Bit 7, bit 6 | 00 | Mode 0 | Indicates the highest mode supported by a unit.*1 | | | | |
| | 01 | Mode 1 | | | | | |
| | 10 | Mode 2 | | | | | |
| | 11 | For future use | | | | | |
| Bit 5 | 0 | Fixed 0 | | | | | |
| Bit 4*2 | 0 | Slave transmission halted | | | | | |
| | 1 | Slave transmission enabled | | | | | |
| Bit 3 | 0 | Fixed 0 | | | | | |
| Bit 2 | 0 | Unit is unlocked | | | | | |
| | 1 | Unit is locked | | | | | |
| Bit 1*3 | 0 | Slave receive buffer is empty | | | | | |
| | 1 | Slave receive buffer is not empty | | | | | |
| Bit 0*4 | 0 | Slave transmit buffer is empty | | | | | |
| | 1 | Slave transmit buffer is not empty | | | | | |

Notes: 1. Since this LSI can support up to mode 2, bits 6 and 7 are fixed to 10.

2. The value of bit 4 can be selected by the STE bit in the IEBus master unit address register 1 (IEAR1).

3. The slave receive buffer is a buffer which is accessed during data write (control bits: H'8, H'A, H'B, H'E, H'F).

In this LSI, the slave receive buffer corresponds to the IEBus receive buffer register (IERBR); and bit 2 is the value of the RxRDY flag in the IEBus receive status register (IERSR).

4. The slave transmit buffer is a buffer which is accessed during data read (control bits: H'3, H'7).

In this LSI, the slave transmit buffer corresponds to the IEBus transmit buffer register (IETBR) when SRQ = 1 in the IEBus general flag register (IEFLG); and bit 1 is a value which reverses the TxRDY flag in the IEBus transmit/runaway status register (IETSR).

Figure 14.3 Bit Configuration of Slave Status (SSR)

(2) Data Command Transfer (Control Bits: Read (H'3, H'7), Write (H'A, H'B, H'E, H'F))

In the case of data read (H'3, H'7), data in the data buffer of the slave unit is read in the master unit. In the case of data write (H'B or H'F) or command write (H'A or H'E), data received in the slave unit is processed in accordance with the operation specification of the slave unit.

Notes: 1. The user can select data and commands freely in accordance with the system.
 2. H'3, H'A, or H'B may lock depending on the communications condition and status.

(3) Locked Address Read (Control Bits: H'4, H'5)

In the case of the locked address read (H'4 or H'5), the address (12 bits) of the master unit which issues lock instruction is configured in bytes shown in figure 14.4.

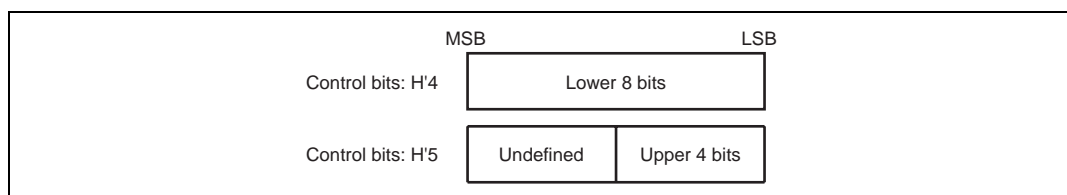


Figure 14.4 Locked Address Configuration

(4) Locking/Unlocking (Control Bits: Setting (H'3, H'A, H'B), Cancellation: (H'6))

The lock function is used for message transfer over multiple communications frames. Locked unit receives data only from the unit which has locked.

Locking and unlocking are described below.

- Locking

When the acknowledge bit of 0 in the message length field is transmitted/received with the control bits indicating the lock operation, and then the communications frame is completed before completion of data transmission/reception for the number of bytes specified by the message length bits, the slave unit is locked by the master unit. In this case, the bit (bit 2) relevant to lock in the byte data indicating the slave status is set to 1.

Lock is set only when the number of data exceeds the maximum number of transfer bytes in one frame. Lock is not set by other error termination.

- Unlocking

When the control bits indicate the lock (H'3, H'A, or H'B) or unlock (H'6) operation and the byte data for the number of bytes specified by the message length bits are transmitted/received

in a single communications frame, the slave unit is unlocked by the master unit. In this case, a bit (bit 2) relevant to lock in the byte indicating the slave status is cleared to 0.

Note that locking and unlocking are not performed in broadcast communications.

Note: * There are three methods to unlock by a locked unit itself.

- Perform hardware reset
- Enter module stop mode
- Issue unlock command by the IEBus command register (IECMR)

Note that the LCK flag in IEFLG can be used to check whether the unit is locked/unlocked.

14.1.4 Bit Format

Figure 14.5 shows the bit format (conceptual diagram) configuring the IEBus communications frame.

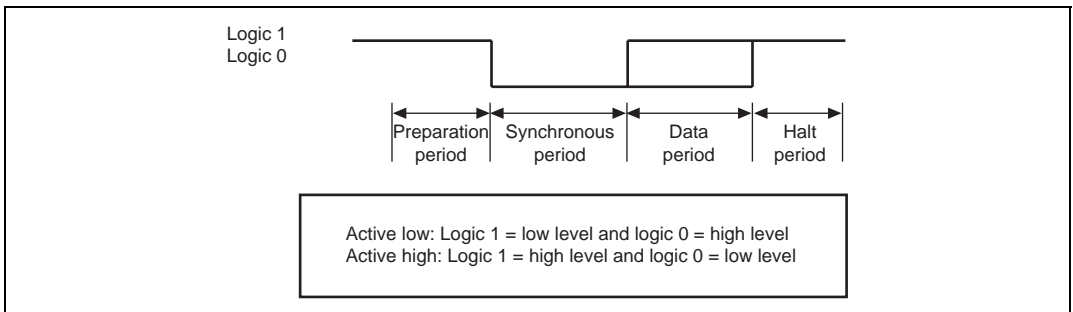


Figure 14.5 IEBus Bit Format (Conceptual Diagram)

Each period of bit format for use of active high signals is described below.

- Preparation period: first logic 1 period (high level)
- Synchronous period: subsequent logic 0 period (low level)
- Data period: period indicating bit value (logic 1: high level, logic 0: low level)
- Halt period: last logic 1 cycle (high level)

For use of active low signals, levels are reversed from the active high signals.

The synchronous and data periods have approximately the same length.

The IEBus is synchronized bit by bit. The specifications for the time of all bits and the periods allocated to the bits differ depending on the type of transfer bits and the unit (master or slave unit).

14.2 Input/Output Pins

Table 14.6 shows the IEB pin configuration.

Table 14.6 Pin Configuration

| Name | Abbreviation | I/O | Function |
|-------------------------|------------------------|--------|--------------------------|
| IEBus transmit data pin | $\overline{\text{Tx}}$ | Output | Transmit data output pin |
| IEBus receive data pin | $\overline{\text{Rx}}$ | Input | Receive data input pin |

14.3 Register Descriptions

The IEB has the following registers. For the module stop control register, see section 24.1.2, Module Stop Control Registers A to C (MSTPCRA to MSTPCRC).

- IEBus control register (IECTR)
- IEBUS command register (IECMR)
- IEBus master control register (IEMCR)
- IEBus master unit address register 1 (IEAR1)
- IEBus master unit address register 2 (IEAR2)
- IEBus slave address setting register 1 (IESA1)
- IEBus slave address setting register 2 (IESA2)
- IEBus transmit message length register (IETBFL)
- IEBus transmit buffer register (IETBR)
- IEBus reception master address register 1 (IEMA1)
- IEBus reception master address register 2 (IEMA2)
- IEBus receive control field register (IERCTL)
- IEBus receive message length register (IERBFL)
- IEBus receive buffer register (IERBR)
- IEBus lock address register 1 (IELA1)
- IEBus lock address register 2 (IELA2)
- IEBus general flag register (IEFLG)
- IEBus transmit/runaway status register (IETSR)
- IEBus transmit/runaway interrupt enable register (IEIET)
- IEBus transmit error flag register (IETEF)
- IEBus receive status register (IERSR)
- IEBus receive interrupt enable register (IEIER)

- IEBus receive error flag register (IEREF)

14.3.1 IEBus Control Register (IECTR)

IECTR controls IEB operation (switches IEBus pin/port functions, selects input/output level, and enables receive operation).

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | IEE | 0 | R/W | <p>IEB Pin Switch</p> <p>Switches IEB pin and port functions.</p> <p>0: The PG3/$\overline{Rx}/\overline{CS1}$ and PG2/$\overline{Tx}/\overline{CS2}$ pins function as the PG3/$\overline{CS1}$ and PG2/$\overline{CS2}$ pins.</p> <p>1: The PG3/$\overline{Rx}/\overline{CS1}$ and PG2/$\overline{Tx}/\overline{CS2}$ pins function as the \overline{Tx} and \overline{Rx} pins.</p> |
| 6 | IOL | 0 | R/W | <p>Input/Output Level</p> <p>Selects input/output pin level (polarity) for the \overline{Rx} and \overline{Tx} pins.</p> <p>0: Pin input/output is set to active low. (Logic 1 is low level and logic 0 is high level.)</p> <p>1: Pin input/output is set to active high. (Logic 1 is high level and logic 0 is low level.)</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | DEE | 0 | R/W | <p>Broadcast Receive Error Interrupt Enable</p> <p>Since the acknowledgement is not returned between the master and slave units in broadcast reception, the master unit cannot decide whether the slave unit is in the receive enabled state. If this bit is set to 1, a reception error interrupt occurs (note that there is not the corresponding bit in the IEBus receive error flag register to this error) when the receive buffer is not in the receive enabled state during receiving the control field in broadcast reception (when the RE bit is not set to 1 or the RxRDY flag is set.). At this time, the master address is stored in IEMA1 and IEMA2. The receive data is not stored in the IERCTL.</p> <p>While this bit is 0, a reception error interrupt does not occur when the receive buffer is not in the receive enabled state, and the reception stops and enters the wait state. The master address is not saved.</p> <p>0: A broadcast receive error is not generated up to the control field.</p> <p>1: A broadcast receive error is generated up to the control field.</p> |
| 4 | CKS | 0 | R/W | <p>Input Clock Select</p> <p>Always set this bit to 0 in this LSI. Selects clock used by the IEB.</p> |
| 3 | RE | 0 | R/W | <p>Receive Enable</p> <p>Enables/disables IEB reception. This bit must be set at the initial setting before frame reception. Changing this bit before receiving the control field is valid, however, changing this bit after receiving the control field is invalid and the value before the change is validated.</p> <p>0: Reception is disabled.</p> <p>1: Reception is enabled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 2 | LUEE | 0 | R/W | <p>Last Byte Underrun Enable</p> <p>Sets whether to generate an underrun error when the last data field byte is transferred in data transmission.</p> <p>If the IEB reads from IETBR when the TxRDY flag is set (the transmit buffer register (IETBR) is empty), an underrun error occurs. In transmission using the DTC, an underrun error occurs at the last byte transmission if the CPU did not clear the TxRDY flag, because the DTC does not clear the TxRDY flag. When the DTC is used, set this bit to 0 to mask an underrun error generated at the last byte transmission. When the DTC is not used, set this bit to 1 to generate an underrun error at the last byte transmission.</p> <p>0: An underrun error does not occur at the last byte transmission (when using the DTC)</p> <p>1: An underrun error does not occur at the last byte transmission (when not using the DTC)</p> |
| 1, 0 | — | All 0 | — | <p>Reserved</p> <p>This bit is always read as 0 and cannot be modified.</p> |

14.3.2 IEBus Command Register (IECMR)

IECMR issues commands to control IEB communications. Since this register is a write-only register, bit-manipulation instructions should not be used when writing. See section 2.9.4, Access Methods for Registers with Write-Only Bits.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 3 | — | All 0 | — | Reserved The read value is undefined. In order to avoid malfunction, do not use bit manipulation instructions. These bits cannot be modified. |
| 2 | CMD2 | 0 | W | Command Bits |
| 1 | CMD1 | 0 | W | These bits issue a command to control IEB communications. When the CMX flag in IEFLG is set after the command issuance, the command is indicated to be in execution. When the CMX flag becomes 0, the operation state is entered. These bits are read as 0. The read value is undefined. Do not use a bit manipulation instruction that causes malfunction. 000: No operation. Operation is not affected. 001: Unlock (required from other units) ^{*1} 010: Requires communications as the master 011: Stops master communications ^{*2} 100: Undefined bits. Operation is not affected by this command. 101: Requires data transfer from the slave. 110: Stops data transfer from the slave ^{*3} . 111: Undefined bits. Operation is not affected by this command. |
| 0 | CMD0 | 0 | W | |
| | | | | |
| | | | | |
| | | | | |

- Notes:
1. Do not execute this command in slave communications. Execute this command after slave communications ends or in master communications. If this command is issued in slave communications, this command is ignored.
 2. This command is valid during master communications (MRQ = 1). In other states, this command issuance is ignored. If this command is issued in master communications, the communications controller immediately enters the wait state. At this time, the issued master transmission request ends (MRQ = 0).
 3. This command is valid during slave communications (SRQ = 1). In other states, this command issuance is ignored. Once this command was issued in slave transmission, the SRQ flag is 0 before slave transmission. Therefore, a transmit request from the master is not responded. If a transmit request is issued during slave transmission, the transmission stops and the wait state is entered (SRQ = 0).

14.3.3 IEBus Master Control Register (IEMCR)

IEMCR sets communications conditions for master communications (selection of broadcast or normal communications, retransmission counts at arbitration loss, and control bits value). It is not necessary to set this register for slave communications.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | SS | 1 | R/W | Broadcast/Normal Communications Select Selects broadcast or normal communications for master communications. 0: Broadcast communications 1: Normal communications |
| 6 | RN2 | 0 | R/W | Retransmission Counts |
| 5 | RN1 | 0 | R/W | Set the number of times retransmission is performed when arbitration is lost in master communications. If arbitration is lost for a specified number of times, the TxE flag in IETSR and the AL flag in IETEF are set and transmission ends with a transmit error. If arbitration is won during retransmission, the retransmission count is automatically restored to the initial setting after master address transfer. 000: 0 001: 1 010: 2 011: 3 100: 4 101: 5 110: 6 111: 7 |
| 4 | RN0 | 0 | R/W | |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|--------------------|---------------|-----|--|
| 3 | CTL3 ^{*1} | 0 | R/W | Control bits |
| 2 | CTL2 | 0 | R/W | Set the control bits in the control field for master transmission. |
| 1 | CTL1 | 0 | R/W | |
| 0 | CTL0 | 0 | R/W | 0000: Reads slave status 0001: Undefined. Setting prohibited. 0010: Undefined. Setting prohibited. 0011: Reads data and locks ^{*2} 0100: Reads locked address (lower 8 bits) 0101: Reads locked address (upper 4 bits) 0110: Reads slave status and unlocks ^{*2} 0111: Reads data 1000: Undefined. Setting prohibited. 1001: Undefined. Setting prohibited. 1010: Writes command and locks ^{*2} 1011: Writes data and locks ^{*2} 1100: Undefined. Setting prohibited. 1101: Undefined. Setting prohibited. 1110: Writes command 1111: Writes data |

Notes: 1. CTL3 decides the data transfer direction of the message length bits in the message length field and data bits in the data field:

CTL3 = 1: Transfer is performed from master unit to slave unit

CTL3 = 0: Transfer is performed from slave unit to master unit

2. Control bits to lock and unlock

14.3.4 IEBus Master Unit Address Register 1 (IEAR1)

IEAR1 sets the lower 4 bits of the master unit address and communications mode. In master communications, the master unit address becomes the master address field value. In slave communications, the master unit address is compared with the received slave address field.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | IAR3 | 0 | R/W | Lower 4 Bits of IEBus Master Unit Address |
| 6 | IAR2 | 0 | R/W | Set the lower 4 bits of the master unit address. |
| 5 | IAR1 | 0 | R/W | |
| 4 | IAR0 | 0 | R/W | |
| 3 | IMD1 | 0 | R/W | IEBus Communications Mode |
| 2 | IMD0 | 0 | R/W | Set IEBus communications mode. 00: Communications mode 0 01: Communications mode 1 10: Communications mode 2 11: Setting prohibited |
| 1 | — | 0 | — | Reserved This bit is always read as 0 and cannot be modified. |
| 0 | STE | 0 | R/W | Slave Transmission Setting Sets bit 4 in the slave status register. Transmitting the slave status register informs the master unit that the slave transmission enabled state is entered by setting this bit to 1. Note that this bit only sets the slave status register value and does not affect slave transmission directly. 0: Bit 4 in the slave status register is 0 (slave transmission stop state) 1: Bit 4 in the slave status register is 1 (slave transmission enabled state) |

14.3.5 IEBus Master Unit Address Register 2 (IEAR2)

IEAR2 sets the upper 8 bits of the master unit address. In master communications, this register becomes the master address field value. In slave communications, this register is compared with the received slave address field.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | IAR11 | 0 | R/W | Upper 8 Bits of IEBus Master Unit Address |
| 6 | IAR10 | 0 | R/W | Set the upper 8 bits of the master unit address. |
| 5 | IAR9 | 0 | R/W | |
| 4 | IAR8 | 0 | R/W | |
| 3 | IAR7 | 0 | R/W | |
| 2 | IAR6 | 0 | R/W | |
| 1 | IAR5 | 0 | R/W | |
| 0 | IAR4 | 0 | R/W | |

14.3.6 IEBus Slave Address Setting Register 1 (IESA1)

IESA1 sets the lower 4 bits of the communications destination slave unit address. For slave communications, it is not necessary to set this register.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | ISA3 | 0 | R/W | Lower 4 Bits of IEBus Slave Address |
| 6 | ISA2 | 0 | R/W | These bits set the lower 4 bits of the |
| 5 | ISA1 | 0 | R/W | communications destination slave unit address |
| 4 | ISA0 | 0 | R/W | |
| 3 to 0 | — | All 0 | — | Reserved |
| | | | | These bits are always read as 0 and cannot be modified. |

14.3.7 IEBus Slave Address Setting Register 2 (IESA2)

IESA2 sets the upper 8 bits of the communications destination slave unit address. For slave communications, it is not necessary to set this register.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | ISA11 | 0 | R/W | Upper 8 Bits of IEBus Slave Address |
| 6 | ISA10 | 0 | R/W | Set upper 8 bits of the communications destination slave unit address |
| 5 | ISA9 | 0 | R/W | |
| 4 | ISA8 | 0 | R/W | |
| 3 | ISA7 | 0 | R/W | |
| 2 | ISA6 | 0 | R/W | |
| 1 | ISA5 | 0 | R/W | |
| 0 | ISA4 | 0 | R/W | |

14.3.8 IEBus Transmit Message Length Register (IETBFL)

IETBFL sets the message length for master or slave transmission.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | TBFL7 | 0 | R/W | Transmit Message Length |
| 6 | TBFL6 | 0 | R/W | Set the message length for master or slave transmission. |
| 5 | TBFL5 | 0 | R/W | |
| 4 | TBFL4 | 0 | R/W | If a value exceeding the maximum transmit bytes for one frame is set in IETBFL, communications are performed with two or more frames in some communications modes. In this case, in or after the second frame, the message length value should be the number of bytes of the remaining communications data, however, the initial IETBFL setting remains unchanged. Therefore, for the second frame or after, re-set the number of bytes of the remaining communications data. |
| 3 | TBFL3 | 0 | R/W | |
| 2 | TBFL2 | 0 | R/W | |
| 1 | TBFL1 | 0 | R/W | |
| 0 | TBFL0 | 0 | R/W | |

14.3.9 IEBus Transmit Buffer Register (IETBR)

IETBR is a 1-byte buffer to which data to be transmitted in master or slave transmission is written. IETBR is empty when the TxRDY flag in IETSR is 1. Check the TxRDY flag before setting transmit data in IETBR.

Data written in IETBR is transmitted in the data field in master or slave transmission. Figure 14.6 shows the correspondence between the communications signal format and registers for IEBus data transfer.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | TBR7 | 0 | R/W | Data to be transmitted is written to this 1-byte buffer. |
| 6 | TBR6 | 0 | R/W | |
| 5 | TBR5 | 0 | R/W | |
| 4 | TBR4 | 0 | R/W | |
| 3 | TBR3 | 0 | R/W | |
| 2 | TBR2 | 0 | R/W | |
| 1 | TBR1 | 0 | R/W | |
| 0 | TBR0 | 0 | R/W | |

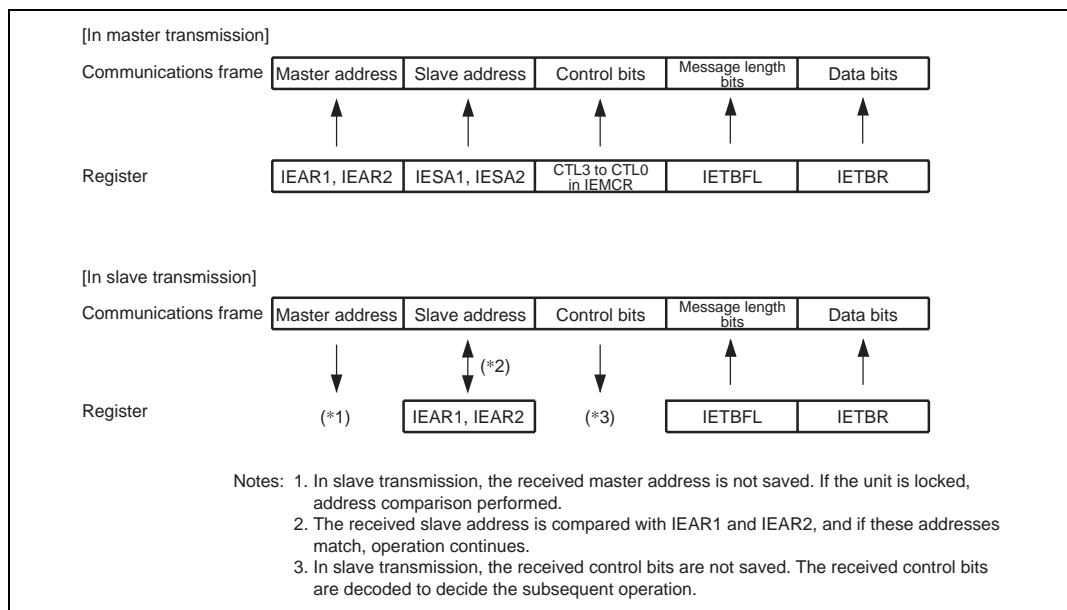


Figure 14.6 Transmission Signal Format and Registers in Data Transfer

14.3.10 IEBus Reception Master Address Register 1 (IEMA1)

IEMA1 indicates the lower four bits of the communications destination master unit address in slave/broadcast reception. This register is enabled when slave/broadcast reception starts, and the contents are changed at the timing of setting the RxS flag in IERSR.

If a broadcast receive error interrupt is selected by the DEE bit in IECTR and the receive buffer is not in the receive enabled state on control field reception, a receive error interrupt is generated and the lower 4 bits of the master address are stored in IEMA1. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | IMA3 | 0 | R | Lower 4 Bits of IEBus Reception Master Address |
| 6 | IMA2 | 0 | R | Indicate the lower 4 bits of the communications destination master unit address in slave/broadcast reception. |
| 5 | IMA1 | 0 | R | |
| 4 | IMA0 | 0 | R | |
| 3 to 0 | — | All 0 | R | Reserved These bits are always read as 0. |

14.3.11 IEBus Reception Master Address Register 2 (IEMA2)

IEMA2 indicates the upper 8 bits of the communications destination master unit address in slave/broadcast reception. This register is enabled when slave/broadcast reception starts, and the contents are changed at the timing of setting the RxS flag in IERSR.

If a broadcast receive error interrupt is selected with the DEE bit in IECTR and the receive buffer is not in the receive enabled state at control field reception, a receive error interrupt is generated and the upper 8 bits of the master address are stored in IEMA2. This register cannot be modified by a write.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | IMA11 | 0 | R | Upper 8 Bits of IEBus Reception Master Address |
| 6 | IMA10 | 0 | R | Indicate the upper 8 bits of the communications destination master unit address in slave/broadcast reception. |
| 5 | IMA9 | 0 | R | |
| 4 | IMA8 | 0 | R | |
| 3 | IMA7 | 0 | R | |
| 2 | IMA6 | 0 | R | |
| 1 | IMA5 | 0 | R | |
| 0 | IMA4 | 0 | R | |

14.3.12 IEBus Receive Control Field Register (IERCTL)

IERCTL indicates the control field value in slave/broadcast reception. This register is enabled when slave/broadcast receive starts, and the contents are changed at the timing of setting the RxS flag in IERSR.

This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. |
| 3 | RCTL3 | 0 | R | IEBus Receive Control Field |
| 2 | RCTL2 | 0 | R | Indicate the control field value in slave/broadcast reception. |
| 1 | RCTL1 | 0 | R | |
| 0 | RCTL0 | 0 | R | |

14.3.13 IEBus Receive Message Length Register (IERBFL)

IERBFL indicates the message length field in slave/broadcast reception. This register is enabled when slave/broadcast receive starts, and the contents are changed at the timing of setting the RxS flag in IERSR.

This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | RBFL7 | 0 | R | IEBus Receive Message Length |
| 6 | RBFL6 | 0 | R | Indicate the contents of message length field in slave/broadcast reception. |
| 5 | RBFL5 | 0 | R | |
| 4 | RBFL4 | 0 | R | |
| 3 | RBFL3 | 0 | R | |
| 2 | RBFL2 | 0 | R | |
| 1 | RBFL1 | 0 | R | |
| 0 | RBFL0 | 0 | R | |

14.3.14 IEBus Receive Buffer Register (IERBR)

IERBR is a 1-byte read-only buffer that stores data received in master or slave reception. This register can be read when the RxRDY flag in IERSR is set to 1. This register indicates the data field value both in master and slave receptions. This register cannot be modified.

Figure 14.7 shows the relationship between transmission signal format and registers in IEBus data reception.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | RBR7 | 0 | R | One-byte read-only buffer that stores data received in master or slave reception |
| 6 | RBR6 | 0 | R | |
| 5 | RBR5 | 0 | R | |
| 4 | RBR4 | 0 | R | |
| 3 | RBR3 | 0 | R | |
| 2 | RBR2 | 0 | R | |
| 1 | RBR1 | 0 | R | |
| 0 | RBR0 | 0 | R | |

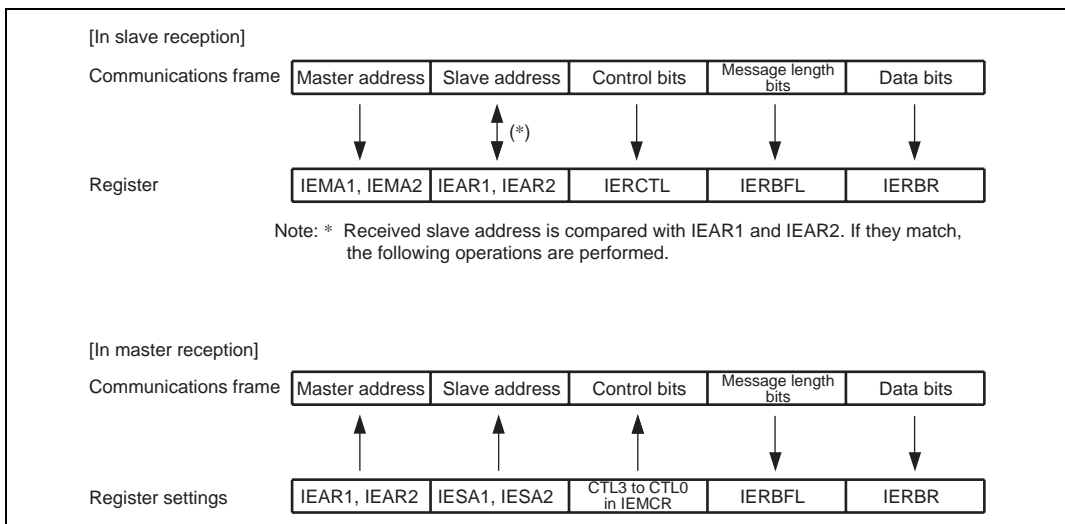


Figure 14.7 Relationship between Transmission Signal Format and Registers in IEBus Data Reception

14.3.15 IEBus Lock Address Register 1 (IELA1)

IELA1 specifies the lower 8 bits of a locked address when a unit is locked. Data in this register is valid when the LCK flag in IEFLG is set to 1. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | ILA7 | 0 | R | Lower 8 Bits of IEBus Lock Address |
| 6 | ILA6 | 0 | R | Store the lower 8 bits of the master unit address when a unit is locked. |
| 5 | ILA5 | 0 | R | |
| 4 | ILA4 | 0 | R | |
| 3 | ILA3 | 0 | R | |
| 2 | ILA2 | 0 | R | |
| 1 | ILA1 | 0 | R | |
| 0 | ILA0 | 0 | R | |

14.3.16 IEBus Lock Address Register 2 (IELA2)

IELA2 is an 8-bit read-only register that specifies the upper 4 bits of a locked address when a unit is locked. Data in this register is valid when the LCK flag in IEFLG is set to 1. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | All 0 | R | Reserved These bits are always read as 0. |
| 3 | ILA11 | 0 | R | Upper 4 Bits of IEBus Locked Address |
| 2 | ILA10 | 0 | R | Store the upper 4 bits of the master unit address when a unit is locked. |
| 1 | ILA9 | 0 | R | |
| 0 | ILA8 | 0 | R | |

14.3.17 IEBus General Flag Register (IEFLG)

IEFLG indicates the IEB command execution status, lock status and slave address match, and broadcast reception detection. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | CMX | 0 | R | <p>Command Execution Status</p> <p>Indicates the command execution status.</p> <p>1: A command is being executed [Setting condition]</p> <p>When a master communications request or slave transmit request command is issued while the MRQ, SRQ, or SRE flag is set to 1</p> <p>0: A command execution is completed [Clearing condition]</p> <p>When a command execution has been completed</p> |
| 6 | MRQ | 0 | R | <p>Master Communications Request</p> <p>Indicates whether or not the unit is in communications request state as a master unit.</p> <p>1: The unit is in communications request state as a master unit [Setting condition]</p> <p>When the CMX flag is cleared to 0 after the master communications request command is issued</p> <p>0: The unit is not in communications request status as a master unit [Clearing condition]</p> <p>When the master communications have been completed</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 5 | SRQ | 0 | R | <p>Slave Transmission Request</p> <p>Indicates whether or not the unit is in transmit request status as a slave unit.</p> <p>1: The unit is in transmit request status as a slave unit [Setting condition]</p> <p>When the CMX flag is cleared to 0 after the slave transmit request command is issued.</p> <p>0: The unit is not in transmit request status as a slave unit [Clearing condition]</p> <p>When a slave transmission has been completed.</p> |
| 4 | SRE | 0 | R | <p>Slave Receive Status</p> <p>Indicates the execution status in slave/broadcast reception.</p> <p>1: Slave/broadcast reception is being executed [Setting condition]</p> <p>When the slave/broadcast reception is started while the RE bit in IECTR is set to 1.</p> <p>0: Slave/broadcast reception is not being executed [Clearing condition]</p> <p>When the slave/broadcast reception has been completed.</p> |
| 3 | LCK | 0 | R | <p>Lock Status Indication</p> <p>Set to 1 when a unit is locked by a lock request from the master unit. IELA1 and IELA2 values are valid only when this flag is set to 1.</p> <p>1: A unit is locked [Setting condition]</p> <p>When data for the number of bytes specified by the message length is not received after the control bits that make the unit locked are received from the master unit. (The LCK flag is set to 1 only when the message length exceeds the maximum number of transfer bytes in one frame. This flag is not set by completion of other errors.)</p> <p>0: A unit is unlocked [Clearing condition]</p> <p>When an unlock condition is satisfied or when an unlock command is issued.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | — | 0 | R | Reserved This bit is always read as 0. |
| 1 | RSS | 0 | R | Receive Broadcast Bit Status Indicates the received broadcast bit value. This flag is valid when the slave/broadcast reception is started. (This flag is changed at the timing of setting the RxS flag in IERSR.) The previous value remains unchanged until the next slave/broadcast reception is started. |
| 0 | GG | 0 | R | General Broadcast Reception Acknowledgement Set to 1 when the slave address is acknowledged as H'FFF in broadcast reception. As well as the receive broadcast bit, this flag is valid when the slave/broadcast reception is started. (This flag is changed at the timing of setting the RxS flag in IERSR.) The previous value remains unchanged until the next slave/broadcast reception is started. This flag is cleared to 0 in slave normal reception. [Setting condition] When H'FFF is acknowledged in the slave field in broadcast reception [Clearing conditions] <ul style="list-style-type: none"> • A unit is in slave reception • When H'FFF is not acknowledged in slave field in broadcast reception |

14.3.18 IEBus Transmit/Runaway Status Register (IETSR)

IETSR detects transmit data ready, transmit start, transmit normal completion, transmit completion with an error, or runaway states. Each status flags in IETSR corresponds to a bit in the IEBus transmit/runaway interrupt enable register (IEIET) that enables or disables each interrupt.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | TxRDY | 1 | R/W | <p>Transmit Data Ready</p> <p>Indicates that the next data can be written to IETBR since IETBR is empty. This flag is automatically cleared by DTC* data transfer. When data is transmitted by the CPU, this flag must be cleared by software. This flag is cleared by writing 0 after reading a 1 from this flag.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> Immediately after reset When data can be written to IETBR (: When IEB has loaded data from IETBR to the transmit shift register) <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When writing 0 after reading TxRDY = 1 When data is written to TBR by the DTC by a TxRDY request. <p>Note: This flag is not cleared on the end byte of DTC transfer.</p> |
| 6 to 4 | — | All 0 | — | <p>Reserved</p> <p>These bits are always read as 0 and cannot be modified.</p> |
| 3 | IRA | 0 | R/W | <p>IEBus Runaway State</p> <p>Indicates that the on-chip microprogram for IEBus control is in the runaway states. This flag is set to 1 when a runaway occurs during either IEBus transmission or reception. (This flag is not a transfer specific flag and is also set for a reception runaway.)</p> <p>[Setting condition]</p> <p>When the on-chip microprogram is in the runaway states</p> <p>[Clearing condition]</p> <p>When writing 0 after reading IRA = 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | TxS | 0 | R/W | <p>Transmit Start Detection</p> <p>Indicates that the IEB starts transmission.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> Master transmission: When the arbitration is won and when the master address field transmission is completed Slave transmission: When the control bits of H'3 (0011) or H'7 (0111) is received from the master unit meaning that data transfer is requested <p>[Clearing condition]</p> <p>When writing 0 after reading TxS = 1</p> |
| 1 | TxF | 0 | R/W | <p>Transmit Normal Completion</p> <p>Indicates that data for the number of bytes specified by the message length bits has been transmitted with no error.</p> <p>[Setting condition]</p> <p>When data for the number of bytes specified by the message length bits has been transmitted normally</p> <p>[Clearing condition]</p> <p>When writing 0 after reading TxF = 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | TxE | 0 | R/W | <p>Transmit Error Completion</p> <p>Indicates that data for the number of bytes specified by the message length bits is not completed and that the data transmission is terminated. The source of this error can be checked by the contents of IETEF. This flag is set at the timing that an error indicated by IETEF occurs. The TxE flag can be cleared even when the error source flag in IETEF is set to 1 because the TxE flag is not logically ORed with the flags in IETEF.</p> <p>In master reception, an error (arbitration loss, timing error, or NAK reception) generated after a master communications command is issued before master reception starts will be detected as a transmit error.</p> <p>[Setting condition]</p> <p>When the data for the number of bytes specified by the message length bits is not completed and when the transmission is terminated</p> <p>[Clearing condition]</p> <p>When writing 0 after reading TxE = 1</p> |

14.3.19 IEBus Transmit/Runaway Interrupt Enable Register (IEIET)

IEIET enables/disables IETSR transmit ready, transmit start, transmit normal completion, transmit completion with an error, and runaway interrupts.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | TxRDYE | 0 | R/W | Transmit Data Ready Interrupt Enable Enables/disables a transmit data ready interrupt. 0: Disables a transmit data ready (TxRDY) interrupt 1: Enables a transmit data ready (TxRDY) interrupt |
| 6 to 4 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 3 | IRAE | 0 | R/W | IEBus Runaway State Interrupt Enable Enables/disables an IEBus runaway state interrupt. 0: Disables an IEBus runaway state interrupt (IRA) 1: Enables an IEBus runaway state interrupt (IRA) |
| 2 | TxSE | 0 | R/W | Transmit Start Interrupt Enable Enables/disables a transmit start (TxS) interrupt. 0: Disables a transmit start (TxS) interrupt 1: Enables a transmit start (TxS) interrupt |
| 1 | TxFE | 0 | R/W | Transmit Normal Completion Interrupt Enable Enables/disables a transmit normal completion (TxF) interrupt. 0: Disables a transmit normal completion (TxF) interrupt 1: Enables a transmit normal completion (TxF) interrupt |
| 0 | TxEE | 0 | R/W | Transmit Error Termination Interrupt Enable Enables/disables a transmit error termination (TxE) interrupt. 0: Disables a transmit error termination (TxE) interrupt 1: Enables a transmit error termination (TxE) interrupt |

14.3.20 IEBus Transmit Error Flag Register (IETEF)

IETEF checks the source of a TxE interrupt indicated in IETSR. This register detects an overflow of a maximum number of bytes in one frame, arbitration loss, underrun error, timing error, and NAK reception.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 5 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 4 | AL | 0 | R/W | Arbitration Loss The IEB retransmits from the start bit for the number of times specified by bits RN2 to Rn0 in IEMCR if the arbitration has been lost in master communications. If the arbitration has been lost for the specified number of times, the AL and TxE flags are set to enter the wait state. If the arbitration has been won within retransmit for the specified number of times, this flag is not set to 1. This flag is set only when the arbitration has been lost and the wait state is entered. [Setting condition] When the arbitration has been lost during data transmission and the transmission has been terminated [Clearing condition] When writing 0 after reading AL = 1 |
| 3 | UE | 0 | R/W | Underrun Error Indicates that an underrun error has occurred during data transmission. The IEB detects an underrun error occurrence when the IEB fetches data from IETBR while the TxRDY flag is set to 1, and the IEB sets the TxE flag and enters the wait state. Accordingly, when the TxRDY flag is not cleared even if data is written to IETBR, an underrun error occurs and data transmission is terminated. Note that the TxRDY flag must be cleared in data transmission by the CPU. [Setting condition] When the IEB loads data from IETBR to the transmit shift register while the TxRDY flag is set to 1 [Clearing condition] When writing 0 after reading UE = 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| Bit | Bit Name | Initial Value | R/W | Description |
| 2 | TTME | 0 | R/W | <p>Timing Error</p> <p>Set to 1 if data is not transmitted at the timing specified by the IEBus protocol during data transmission. The IEB sets the TxE flag and enters the wait state.</p> <p>[Setting condition]</p> <p>When a timing error occurs during data transmission</p> <p>[Clearing condition]</p> <p>When writing 0 after reading TTME = 1</p> |
| 1 | RO | 0 | R/W | <p>Overflow of Maximum Number of Transmit Bytes in One Frame</p> <p>Indicates that the maximum number of bytes defined by communications mode have been transmitted because a NAK has been received from the receive unit and retransmit has been performed, or that transmission has not been completed because the message length value exceeds the maximum number of transmit bytes in one frame. The IEB sets the TxE flag and enters the wait state.</p> <p>[Setting condition]</p> <p>When the transmit has not been completed although the maximum number of bytes defined by communications mode have been transmitted</p> <p>[Clearing condition]</p> <p>When writing 0 after reading RO = 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | ACK | 0 | R/W | <p>Acknowledge bit Status</p> <p>Indicates the data received in the acknowledge bit of the data field.</p> <ul style="list-style-type: none"> Acknowledge bit other than in the data field The IEB terminates the transmission and enters the wait state if a NAK is received. In this case, this bit and the TxE flag are set to 1. Acknowledge bit in the data field The IEB retransmits data up to the maximum number of bytes defined by communications mode until an ACK is received from the receive unit if a NAK is received from the receive unit during data field transmission. In this case, when an ACK is received from the receive unit during retransmission, this flag is not set and transmission will be continued. When transmission is terminated without receiving an ACK, this flag is set to 1. <p>Note: This flag is invalid in broadcast communications.</p> <p>[Setting condition] When the acknowledge bit of 1 (NAK) is detected</p> <p>[Clearing condition] When writing 0 after reading ACK = 1</p> |

14.3.21 IEBus Receive Status Register (IERSR)

IERSR detects receive data ready, receive start, transmit/receive normal completion, or receive completion with an error. Each status flag in IERSR corresponds to a bit in the IEIER that enables/disables each interrupt.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | RxRDY | 1 | R/W | <p>Receive Data Ready</p> <p>Indicates that the receive data is stored in IERBR and that the receive data can be read. This flag is automatically cleared by DTC* data transfer. When data is transmitted by the CPU, this flag must be cleared by software.</p> <p>[Setting condition]</p> <p>When data reception has been completed normally and receive data has been loaded to IERBR.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When writing 0 after reading RxRDY = 1 • When IERBR data is read by the DTC by a RxRDY request. <p>Note: This flag cannot be cleared on the end byte of the DTC transfer.</p> |
| 6 to 3 | — | All 0 | — | <p>Reserved</p> <p>These bits are always read as 0 and cannot be modified.</p> |
| 2 | RxS | 0 | R/W | <p>Receive Start Detection</p> <p>Indicates that the IEB starts reception.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • Master reception: When the message length field has been received from the slave unit correctly after the arbitration is won and the control field transmission is completed • Slave reception: When the message length field has been received from the master unit correctly <p>[Clearing condition]</p> <p>When writing 0 after reading RxS = 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | RxF | 0 | R/W | <p>Receive Normal Completion</p> <p>Indicates that data for the number of bytes specified by the message length bits has been received and with no error.</p> <p>[Setting condition]</p> <p>When data for the number of bytes specified by the message length bits has been received normally.</p> <p>[Clearing condition]</p> <p>When writing 0 after reading RxF = 1</p> |
| 0 | RxE | 0 | R/W | <p>Receive Error Completion</p> <p>Indicates that data for the number of bytes specified by the message length bits is not completed and that the data reception is terminated. The source of this error can be checked by the contents of IEREF. This flag is set at the timing that an error indicated by IEREF occurs. The RxE flag can be cleared even when the error source flag in IEREF is set to 1 because the RxE flag is not logically ORed with the flags in IEREF.</p> <p>[Setting condition]</p> <p>When the data for the number of bytes specified by the message length bits is not completed and when the reception is terminated.</p> <p>[Clearing condition]</p> <p>When writing 0 after reading RxE = 1</p> |

14.3.22 IEBus Receive Interrupt Enable Register (IEIER)

IEIER enables/disables IERSR reception ready, receive start, transmit/receive normal completion, and receive completion with an error interrupts.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | RxRDYE | 0 | R/W | Receive Data Ready Interrupt Enable Enables/disables a receive data ready interrupt. 0: Disables a receive data ready (RxRDY) interrupt 1: Enables a receive data ready (RxRDY) interrupt |
| 6 to 3 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 2 | RxSE | 0 | R/W | Receive Start Interrupt Enable Enables/disables a receive start (RxS) interrupt. 0: Disables a receive start (RxS) interrupt 1: Enables a receive start (RxS) interrupt |
| 1 | RxFE | 0 | R/W | Receive Normal Completion Enable Enables or disables a receive normal completion (RxF) interrupt. 0: Disables a receive normal completion (RxF) interrupt 1: Enables a receive normal completion (RxF) interrupt |
| 0 | RxEE | 0 | R/W | Receive Error Termination Interrupt Enable Enables or disables a receive error termination (RxE) interrupt. 0: Disables a receive error termination (RxE) interrupt 1: Enables a receive error termination (RxE) interrupt |

14.3.23 IEBus Receive Error Flag Register (IEREF)

IEREF checks the source of an RxE interrupt indicated in IERSR. This register detects an overrun error, timing error, overflow of a maximum number of bytes in one frame, and parity error.

These flags become valid when the receive start flag (RxS) is set to 1. If an error occurs before the RxS flag is set to 1, the IEB terminates the communications and enters the wait state. In this case, these flags will not be set and the RxE flag is not set.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 4 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |
| 3 | OVE | 0 | R/W | <p>Overrun Control Flag</p> <p>Used to control the overrun during data reception. The IEB sets the OVE and RxE flags when the IEB receives the next byte data while the receive data has not been read (the RxRDY flag is not cleared) and when the parity bit reception has been started. If this flag remains set until acknowledge bit transfer, the IEB assumes that an overrun error has occurred and returns a NAK to the communications destination unit.</p> <p>The communications destination unit retransmits data up to the maximum number of transmit bytes. The IEB, however, returns a NAK when this flag remains set because the IEB assumes that the overrun error has not been cleared.</p> <p>If this flag is cleared to 0, the IEB decides that the overrun error has been cleared, returns an ACK, and receives the next data.</p> <p>In broadcast reception, if this flag is set during acknowledge bit transmission, the IEB immediately enters the wait state.</p> <p>[Setting condition]</p> <p>When the next byte data is received while the RxRDY flag is not cleared and when the parity bit of the data is received.</p> <p>[Clearing condition]</p> <p>When writing 0 after reading OVE = 1</p> |
| 2 | RTME | 0 | R/W | <p>Timing Error</p> <p>Set to 1 if data is not received at the timing specified by the IEBus protocol during data reception. The IEB sets the RxE flag and enters the wait state.</p> <p>[Setting condition]</p> <p>When a timing error occurs during data reception</p> <p>[Clearing condition]</p> <p>When writing 0 after reading RTME = 1</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | DLE | 0 | R/W | <p>Overflow of Maximum Number of Receive Bytes in One Frame</p> <p>Indicates that the maximum number of bytes defined by communications mode have been received because a parity error or overrun error occurred, or that the reception has not be completed because the message length value exceeds the maximum number of receive bytes in one frame. The IEB sets the RxE flag and enters the wait state.</p> <p>[Setting condition]</p> <p>When the reception has not been completed although the maximum number of bytes defined by communications mode have been received.</p> <p>[Clearing condition]</p> <p>When writing 0 after reading DLE = 1</p> |
| 0 | PE | 0 | R/W | <p>Parity Error</p> <p>Indicates that a parity error has occurred during data field reception. If a parity error occurs before data field reception, the IEB immediately enters the wait state and the PE flag is not set.</p> <p>If a parity error occurs when the maximum number of receive bytes in one frame has not been received, the PE flag is not set. When a parity error occurs, the IEB returns a NAK to the communications destination unit via the acknowledge bit. In this case, the communications destination unit continues retransfer up to the maximum number of receive bytes in one frame and if the reception has been completed normally by clearing the parity error, the PE flag is not set. If the parity error is not cleared when the reception is terminated before receiving data for the number of bytes specified by the message length, the PE flag is set.</p> <p>In broadcast reception, if a parity error occurs during data field reception, the IEB enters the wait state immediately after setting the PE flag.</p> <p>[Setting condition]</p> <p>When the parity bit of last data of the data field is not correct after the maximum number of receive bytes has been received</p> <p>[Clearing condition]</p> <p>When writing 0 after reading PE = 1</p> |

14.4 Operation Descriptions

14.4.1 Master Transmit Operation

This section describes an example of master transmission using the DTC after slave reception.

(1) IEB Initialization

(a) Setting the IEBus Control Register (IECTR)

Enable the IEBus pins, select the signal polarity, and select a clock supplied to the IEB. Clear the LUEE bit to 0 since the transfer is performed by the DTC.

(b) Setting the IEBus Master Unit Address Registers 1 and 2 (IEAR1 and IEAR2)

Specify the master unit address and specify the communications mode in IEAR1.

(c) Setting the IEBus Slave Address Setting Registers 1 and 2 (IESA1 and IESA2)

Specify the communications destination slave unit address.

(d) Setting the IEBus Master Control register (IEMCR)

Select broadcast/normal communications, specify the number of retransfer counts at arbitration loss, and specify the control bits.

(e) Setting the IEBus Transmit Message Length Register (IETBFL)

Specify the message length bits.

(f) Setting the IEBus Transmit/Runaway Interrupt Enable Register (IEIET)

Enable TxRDY (IETxI), TxS, TxF, and TxE (IETSI) interrupts.

The above registers can be specified in any order. (The register specification order does not affect the IEB operation.)

(2) DTC Initialization

1. Set the start address of the RAM which stores the register information necessary for the DTC transfer in the vector address (H'000004D4) to be accessed when a DTC transfer request is generated.
2. Set the following data from the start address of the RAM.
 - Transfer source address (SAR): Start address of the RAM which stores data to be transmitted in the data field.
 - Transfer destination address (DAR): Address (H'FFF808) of the IEBus transmit buffer register (IETBR)
 - Transfer count (CRA): The same value as the IETBFL contents
3. Set DTCEG5 in the DTC enable register G (DTCERG) to enable the TxRDY interrupt (IETxI).

Because the TxRDY flag is retained after a reset, the DTC transfer starts when the IETxI is enabled and the first data for the data field is written to IETBR. The DTC negates the TxRDY flag and the first byte of DTC transfer is completed.

(3) Master Transmission Flow

Figure 14.8 shows the master transmission flow. Numbers in the following description correspond to the number in figure 14.8.

1. After the IEB and DTC have been initialized, a master communications request command is issued from IECMR. During slave reception, the command execution status flag (CMX) in IEFLG is set and the master communications request will not be issued.
2. When the slave reception has been completed, the CMX flag is cleared, the master communications command is executed, and the MRQ flag is set.
3. The transmit start detection flag (TxS) in IETSR is set when arbitration is won and the master address has been transmitted. In this case, one of the transmit status interrupts (IETSI) is requested to the CPU, and the TxS flag is cleared in the interrupt handling routine.
4. The IEB loads data to be transmitted in the data field from IETBR when the control and message length fields have been transmitted and an ACK is received in each field. After that, the TxRDY flag is set. A DTC transfer request is generated by IETxI and the second byte is written to the transmit buffer.
5. Similarly, the data field load and transmission are repeated.
6. The DTC completes the data transfer for the number of specified bytes when data to be transmitted in the last byte is written to. At this time, the DTC does not clear the TxRDY flag. It, however, clears bit DTCEG5 in the DTC enable register G (DTCERG) so as not to generate more DTC transfer request.
7. A TxRDY interrupt (IETxI) is issued to the CPU when the DTC transfer is completed. In this interrupt handling routine, the TxRDY flag can be cleared. However, since a TxRDY interrupt will be generated again after the last byte transfer, the TxRDY flag remains set. (Note that the LUEE bit must be cleared to 0 because an underrun error occurs to terminate the transfer if the LUEE bit in IECTR is set to 1.) Note, however, that the TxRDY interrupt must be disabled because the TxRDY interrupt is always generated.
8. A transmit normal completion (TxF) interrupt (IETSI) occurs after the last data transfer is completed. In this case, the CPU clears the TxF flag and completes the normal completion interrupt and clears the MRQ flag to 0.

Note: As a transmit status interrupt (IETSI), the transmit error termination (TxE) interrupt as well as the transfer start detection (TxS) and transmit normal completion (TxF) interrupts must be enabled. If an error termination interrupt is disabled, no interrupt is generated even if the transmission is terminated by an error.

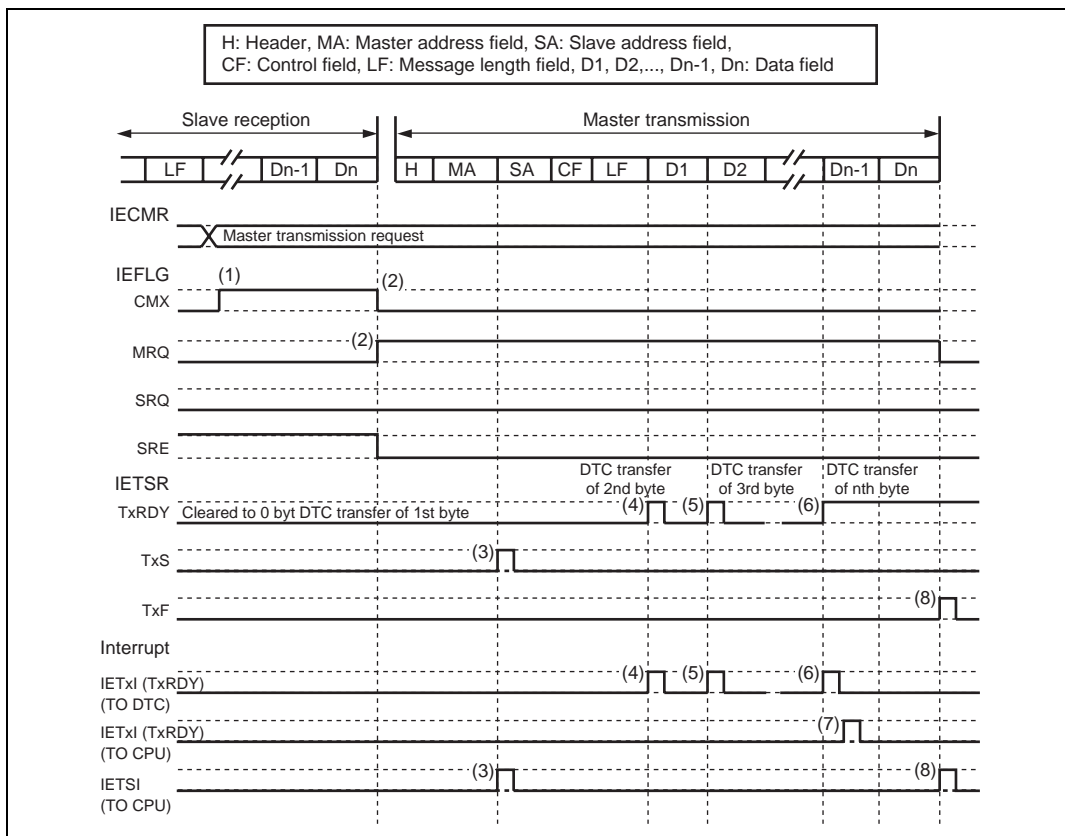


Figure 14.8 Master Transmit Operation Timing

14.4.2 Slave Receive Operation

This section describes an example of performing a slave reception using the DTC.

(1) IEB Initialization

(a) Setting the IEBus Control Register (IECTR)

Enable the IEBus pins, select the signal polarity, and select a clock supplied to the IEB. Set the RE bit to 1 to perform reception. The LUEE bit does not need to be specified.

(b) Setting the IEBus Master Unit Address Registers 1 and 2 (IEAR1 and IEAR2)

Specify the master unit address and specify the communications mode in IEAR1. Compare with the slave address in the communications frame and receive the frame if matched.

(c) Setting the IEBus Receive Interrupt Enable Register (IEIER)

Enable RxRDY (IERxI), RxS, and RxE (IERSI) interrupts.

The above registers can be specified in any order. (The register specification order does not affect the IEB operation.)

(2) DTC Initialization

1. Set the start address of the RAM which stores the register information necessary for the DTC transfer in the vector address (H'000004D2) to be accessed when a DTC transfer request is generated.
2. Specify the following from the start address of the RAM.
 - Transfer source address (SAR): Address (H'FFF80D) of the IEBus receive buffer register (IERBR).
 - Transfer destination address (DAR): Start address of the RAM which stores data received from the data field.
 - Transfer count (CRA): Maximum number of transfer bytes in one frame in the transfer mode.
3. Set DTCEG6 in the DTC enabler register G (DTCERG) to enable the RxRDY interrupt (IETxI).

Because the above settings are performed before the frame reception, the length of data to be received cannot be decided. Accordingly, the maximum number of transfer bytes in one frame is specified as the DTC transfer count.

If the DTC is specified after reception starts, the above settings are performed in the receive start (RxS) interrupt handling routine. In this case, the transfer count must be the same value as the contents of the IEBus receive message length register (IERBFL).

(3) Slave Reception Flow

Figure 14.9 shows the slave reception flow. Numbers in the following description correspond to the number in figure 14.9. In this example, the DTC is specified when the frame reception starts.

1. After the broadcast reception has been completed, the slave reception is performed. The receive broadcast bit status flag (RSS) in IEFLG retains the previous frame information (set to 1) until the receive start detection flag (RxS) is set to 1. If the RSS flag changes at the timing of header reception, the interrupt handling of the broadcast reception completion must be completed before the header reception. Accordingly, the RSS flag is stipulated that it changes at the timing of starting reception.
2. If data is received up to the message length field, a receive start detection (RxS) interrupt (receive status interrupt (IERSI)) will occur and the SRE flag is set to 1. In this case, the DTC initialization described in (2) is performed. After initialization, the RxS flag is cleared to 0.

3. When the first data is received, the RxRDY flag is set to 1. A DTC transfer request by IERxI occurs, and the DTC loads data from the IEBus receive buffer register (IERBR) and clears the RxRDY flag.
4. Similarly, the data field reception and load are repeated.
5. When the last data is received, the DTC completes the data transfer for the specified number of bytes after loading the receive data to the RAM. In this case, the DTC does not clear the RxRDY flag. It, however, clears the DTC enable register G (DTCEG). Accordingly, hereafter, no transfer request will be issued to the DTC.
6. When the DTC transfer has been completed, an RxRDY interrupt (IERxI) is issued to the CPU. In this interrupt handling routine, the RxRDY flag is cleared.
7. When the last data is received, a receive normal completion (RxF) interrupt (IERSI) occurs. In this case, the CPU clears the RxF flag in order to complete the normal completion interrupt. The SRE flag is cleared to 0.

- Notes:
1. As a receive status interrupt (IERSI), the receive error termination (RxEx) interrupt as well as the receive start detection (RxS) and receive normal completion (RxF) interrupts must be enabled. If an error termination interrupt is disabled, no interrupt is generated even if the reception is terminated by an error.
 2. The interrupt occurs after the DTC transfer has been completed. Accordingly, the interrupt described in item 6 actually occurs after item 7 above.

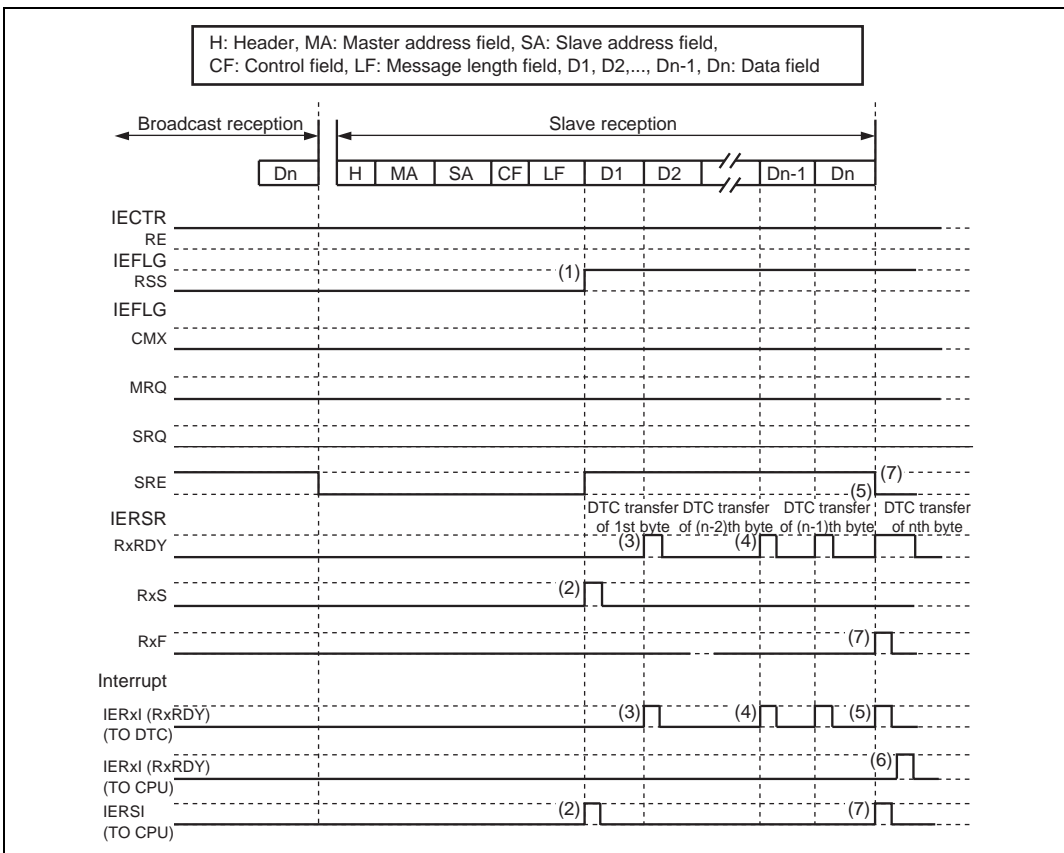


Figure 14.9 Slave Reception Operation Timing

(4) When an Error Occurs in Broadcast Reception (DEE = 1)

Figure 14.10 shows an example in which a receive error occurs because the receive preparation cannot be completed (the RxRDY flag is not cleared) until the control field is received in broadcast reception after the slave reception while the DEE bit is set to 1.

Note: The same as the case in which the RE bit is not set before the control field reception.

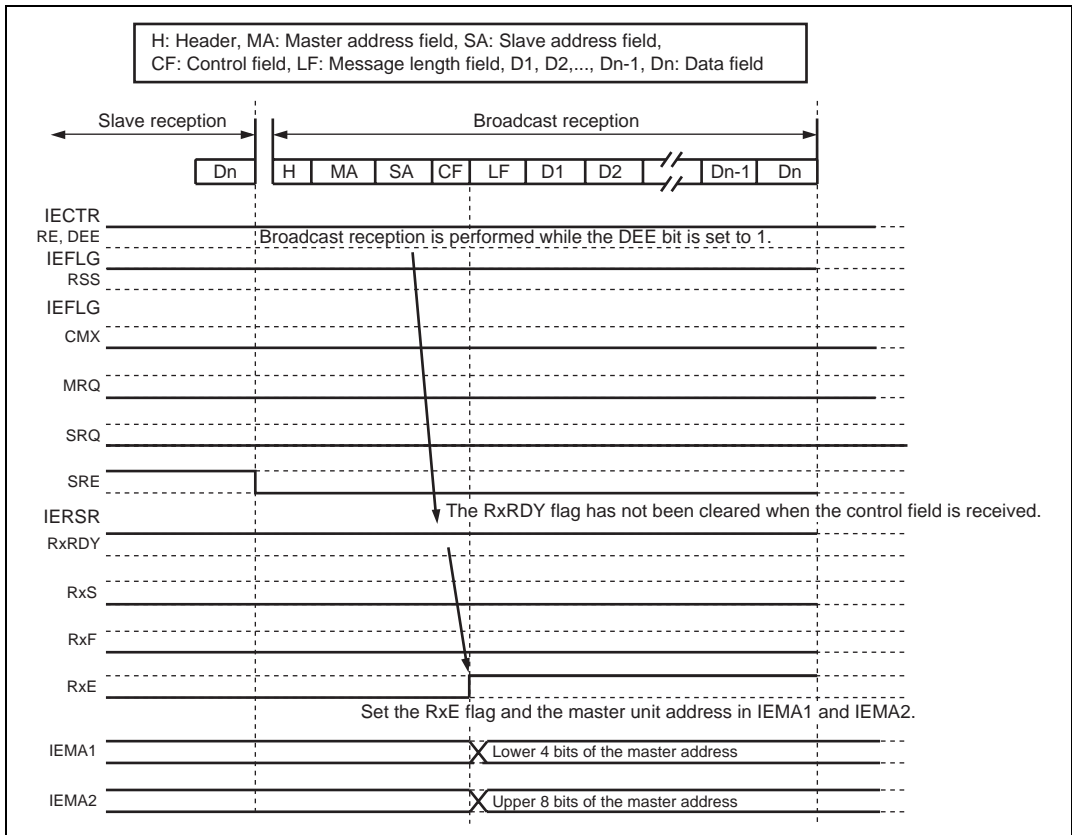


Figure 14.10 Error Occurrence in the Broadcast Reception (DEE = 1)

14.4.3 Master Reception

This section shows an example of performing a master reception using the DTC after slave reception.

(1) IEB Initialization

(a) Setting the IEBus Control Register (IECTR)

Enable the IEBus pins, select the signal polarity, and select a clock supplied to the IEB. Set the RE bit to 1 to perform reception. The LUEE bit does not need to be specified.

(b) Setting the IEBus Master Unit Address Registers 1 and 2 (IEAR1 and IEAR2)

Specify the master unit address and specify the communications mode in IEAR1. Compare with the slave address in the communications frame and receive the frame if matched.

- (c) Setting the IEBus Slave Address Setting Registers 1 and 2 (IESA1 and IESA2)
Specify the communications destination slave unit address.
- (d) Setting the IEBus Master Control Register (IEMCR)
Select broadcast/normal communications, specify the number of retransfer counts at arbitration loss, and specify the control bits.
- (e) Setting the IEBus Receive Interrupt Enable Register (IEIER)
Enable the RxRDY (IERxI), RxS, RxF, and RxE (IERSI) interrupts.

The above registers can be specified in any order. (The register specification order does not affect the IEB operation.)

(2) DTC Initialization

1. Set the start address of the RAM which stores the register information necessary for the DTC transfer in the vector address (H'000004D2) to be accessed when a DTC transfer request is generated.
2. Set the following data from the start address of the RAM.
 - Transfer source address (SAR): Address (H'FFF80D) of the IEBus receive buffer register (IERBR).
 - Transfer destination address (DAR): Start address of the RAM which stores data to be received from the data field.
 - Transfer count (CRA): Maximum number of transfer bytes in one frame in the transfer mode.
3. Set bit DTCEG6 in the DTC enabler register G (DTCERG), and enable the RxRDY interrupt (IERxI).

Because the above settings are performed before frame reception, the length of data to be received cannot be determined. Accordingly, the maximum number of transfer bytes in one frame is specified as the DTC transfer count.

If the DTC is specified after reception starts, the above settings are performed in the receive start detection (RxS) interrupt handling routine. In this case, the transfer count must be the same value as the contents of the IEBus receive message length register (IERBFL).

(3) Master Reception Flow

Figure 14.11 shows the master reception flow. Numbers in the following description correspond to the number in figure 14.11. In this example, the DTC is specified when the frame reception starts.

1. After the IEB has been initialized, a master communications request command is issued from IECMR. During slave reception, the command execution status flag (CMX) in IEFLG is set and the master communications request will not be issued.
2. The CMX flag is cleared when the slave reception is completed, the master communications command is executed, and the MRQ flag is set.
3. If the arbitration is won, the master address, slave address, and control field will be transmitted. An error generated before the control field transmission will be handled as a transmission error. In this case, the TxE flag is set and the error contents will be reflected in IETEF.
4. The message length field is received from the slave unit. If no parity error is detected and reception is performed correctly, the receive start detection flag (RxS) is set to 1. If a parity error occurs, it is handled as a receive error. A receive start detection (RxS) interrupt (receive status interrupt (IERSI)) occurs and the DTC initialization described in (2) is performed. After DTC initialization, the RxS flag is cleared to 0.
5. When the first data is received, the RxRDY flag is set to 1. A DTC transfer request by IERxI occurs and the DTC loads data from the IEBus receive buffer register (IERBR) and clears the RxRDY flag.
6. Similarly, the above data field receive and load operations are repeated.
7. When the last data is received, the DTC completes the data transfer for the specified number of bytes after loading the receive data to the RAM. In this case, the DTC does not clear the RxRDY flag. It, however, clears the DTC enable register G (DTCEG). Accordingly, hereafter, no transfer request will be issued to the DTC.
8. When the DTC transfer has been completed, an RxRDY interrupt (IERxI) is issued to the CPU. In this interrupt handling routine, the RxRDY flag is cleared.
9. When the last data is received, a receive normal completion (RxF) interrupt (IERSI) occurs. In this case, the CPU clears the RxF flag to complete the receive normal completion interrupt. The MRQ flag is cleared to 0.

Notes: 1. As a receive status interrupt (IERSI), an receive error completion (RxE) interrupt as well as the receive start detection (RxS) and receive normal completion (RxF) interrupts must be enabled. If a receive error completion interrupt is disabled, no interrupt is generated even if the reception is terminated by an error.

2. The interrupt occurs after the DTC transfer has been completed. Accordingly, the interrupt described in item 8 actually occurs after item 9 above.

- (c) Setting the IEBus Transmit Message Length Register (IETBFL)
Specify the message length bits.
- (d) Setting the IEBus Transmit/Runaway Interrupt Enable Register (IEIET)
Enable the TxRDY (IETxI), TxS, and TxE (IETSI) interrupts.

The above registers can be specified in any order. (The register specification order does not affect the IEB operation.)

(2) DTC Initialization

1. Set the start address of the RAM which stores the register information necessary for the DTC transfer in the vector address (H'000004D4) to be accessed a DTC transfer request is generated.
2. Set the following data from the start address of the RAM.
 - Transfer source address (SAR): Start address of the RAM which stores data to be transmitted from the data field.
 - Transfer destination address (DAR): Address (H'FFF808) of the IEBus transmit buffer register (IETBR)
 - Transfer count (CRA): The same value as IETBFL
3. Set bit DTCEG5 in the DTC enabler register G (DTCERG), and enable the TxRDY interrupt (IETxI).

Because the TxRDY flag is retained after reset, the DTC transfer is executed when the IETxI is enabled and the first data field data is written to IETBR. The DTC negates the TxRDY flag and the DTC transfer of the first byte is completed.

(3) Slave Transmission Flow

Figure 14.12 shows the slave transmission flow. Numbers in the following description correspond to the numbers in Figure 14.12.

1. After the IEB and DTC have been initialized, a slave communications request command is issued from IECMR. During slave reception, the command execution status flag (CMX) in IEFLG is set and the slave communications request will not be issued.
2. The CMX flag is cleared when the slave reception is completed, the slave communications command is executed, and the SRQ flag is set.
3. If data up to the control field has been received correctly and if the contents of the control bits is H'3 or H'7, the transmit start detection flag (TxS) in IETSR register is set to 1. In this case, the TxS flag is cleared in the TxS interrupt handling routine.
4. The slave then transmits the message length field, and the IEB loads the transmit data in the data field from IETBR when the ACK is received. Then the TxRDY flag is set to 1. A DTC

transfer request by IETxI is generated and the second byte data is written to the transmit buffer.

5. Similarly, the above data field load and transmission operations are repeated.
6. The DTC completes the data transfer for the number of specified bytes when data to be transmitted in the last byte is written to. At this time, the DTC does not clear the TxRDY flag. It, however, clears bit DTCEG5 in the DTC enable register G (DTCERG) not to generate more DTC transfer request.
7. A TxRDY interrupt (IETxI) is issued to the CPU when the DTC transfer is completed. In this interrupt handling routine, the TxRDY flag can be cleared. However, since the TxRDY interrupt will be generated again after the last byte transfer, the TxRDY flag remains set. (Note that the LUEE bit should be cleared to 0 because an underrun error occurs to terminate the transfer if the LUEE bit in IECTR is set to 1.) Note, however, that the TxRDY interrupt should be disabled because the TxRDY interrupt is always generated.
8. After the last data transfer has been completed, a transmit normal completion (TxF) interrupt occurs. In this case, the CPU clears the TxF flag and completes the normal completion interrupt and clears the SRQ flag to 0.

- Notes:
1. As a transmit status interrupt (IETSI), a transmit error termination (TxE) interrupt as well as the transmit start detection (TxS) and transmit normal completion (TxF) interrupts must be enabled. If a transmit error completion interrupt is disabled, no interrupt is generated even if the transfer is terminated by an error.
 2. If the control bits sent from the master unit is H'0, H'4, H'5, or H'6 in slave transmission, the IEB automatically performs processing and the TxS and TxF flags are not set.

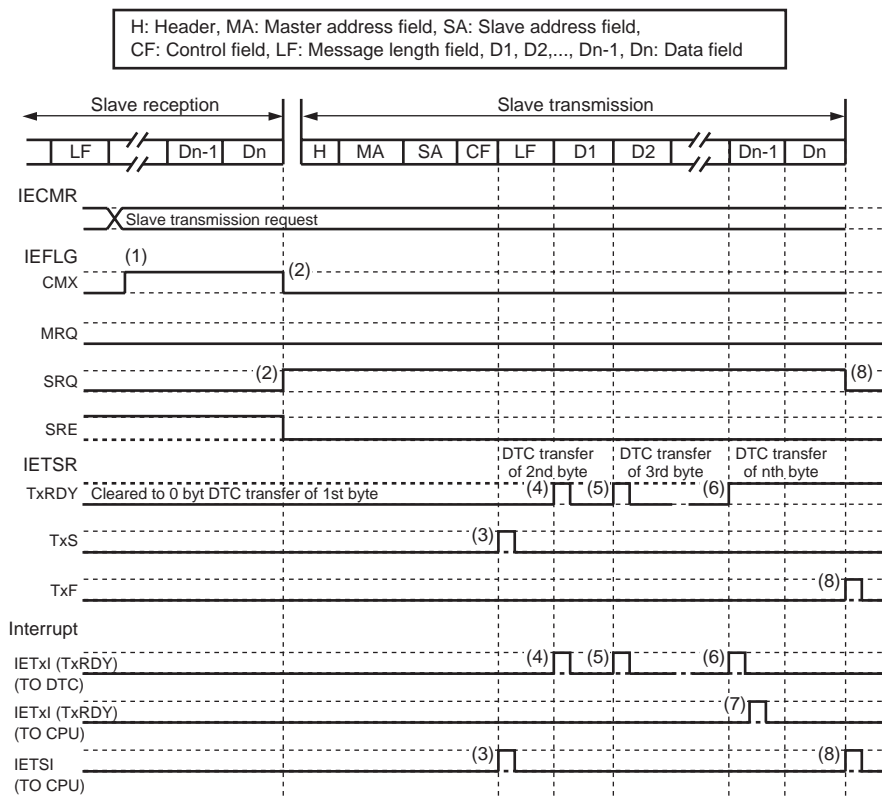


Figure 14.12 Slave Transmit Operation Timing

14.5 Interrupt Sources

Figures 14.13 and 14.14 show the transmit and receive interrupt sources, respectively.

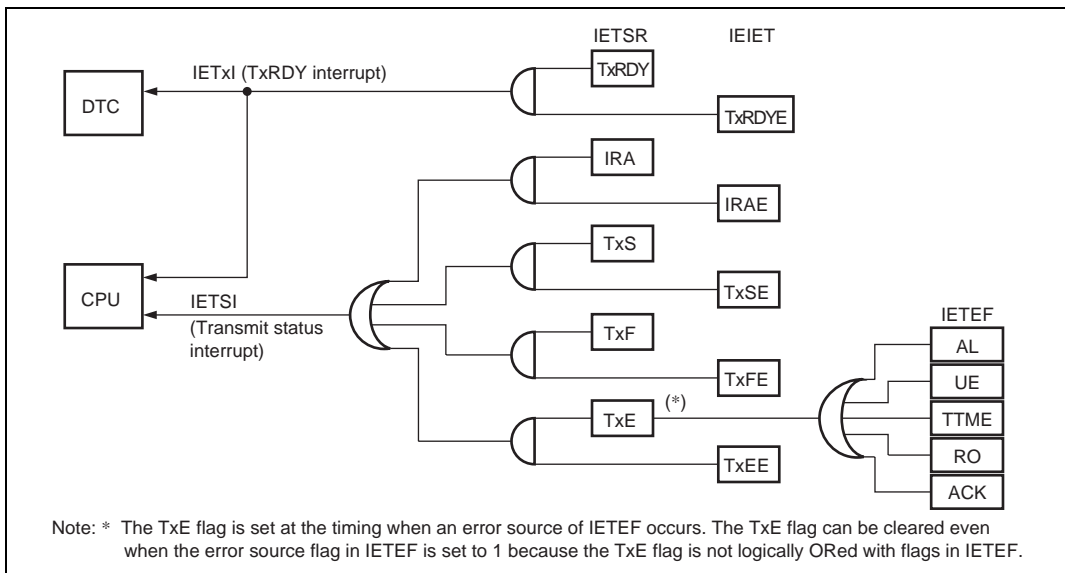


Figure 14.13 Relationships among Transfer Interrupt Sources

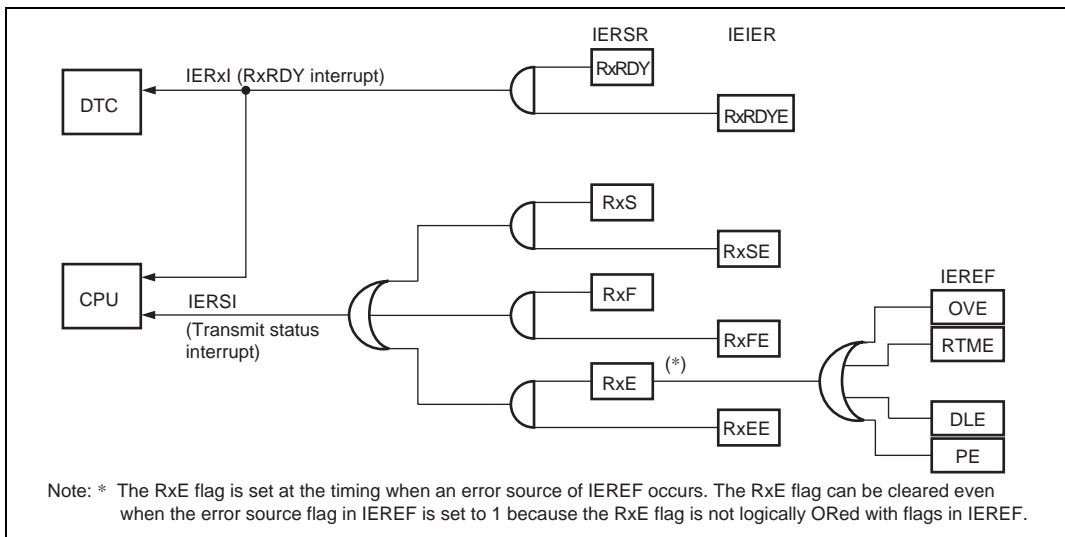


Figure 14.14 Relationships among Receive Interrupt Sources

14.6 Usage Notes

14.6.1 Setting Module Stop Mode

The IEB is enabled or disabled by setting the module stop control register. In the initial state, the IEB is disabled. After the module stop mode is canceled, registers can be accessed. For details, see section 24, Power-Down Modes.

14.6.2 TxRDY Flag and Underrun Error

1. The TxRDY flag indicates that IETBR is empty. Writing to IETBR by the DTC clears the TxRDY flag. Meanwhile, the TxRDY flag must be cleared by software since writing to IETBR by the CPU does not clear the TxRDY flag.
2. If the CPU fails to write to IETBR by the timing of the frame transmission or if the number of transfer words is less than the length specified by the message length bits, an underrun error occurs.
3. The IEB decides that an underrun error occurred when the data is loaded from IETBR to the transmit shift register while the TxRDY flag is set to 1. In this case, the IEB sets the TxE flag in IETSR and enters the wait state. The UE flag in IETEF is also set to 1.
4. On the receive side, the unit decides that a timing error has occurred because the communications are terminated.
5. In data transfer using the DTC, the TxRDY flag in IETSR is not cleared after the last byte data is transferred to IETBR and a CPU interrupt caused by the DTC interrupt will occur.
If the TxRDY flag is not cleared in this CPU interrupt handling routine, an underrun error will occur when the last byte data is loaded from IETBR to the transmit shift register. In this case, if the LUEE bit is cleared to 0 (initial value), no underrun error occurs and the last byte of the data field is transmitted correctly. (If the LUEE bit is set to 1, an underrun error occurs.)
6. Although the DTC is used as described in item 5, if the number of DTC transfer words is less than the length specified by the message length bits, the LUEE bit setting is invalid. (The LUEE bit is valid only when data is transmitted for the number of bytes specified by the message length bits has been transmitted.) In this case, an underrun error occurs, data is transmitted for one byte less than the DTC transfer words, and the transfer is terminated by a transmit error.

14.6.3 RxRDY Flag and Overrun Error

1. The RxRDY flag indicates that IERBR stores data. Reading from IERBR by the DTC clears the RxRDY flag. Meanwhile, the RxRDY flag must be cleared by software since reading from IERBR by the CPU does not clear the RxRDY flag.
2. If the CPU fails to read from IERBR by the timing of the frame reception or if the number of transfer words is less than the length specified by the message length bits, an overrun error occurs.
3. The IEB receives data while the RxRDY flag is set and sets the OVE flag when the parity bit reception starts. If the OVE flag is set when the acknowledge bit is transmitted, the IEB assumes that an overrun error has occurred, returns a NAK, and discards the data in the receive shift register.
4. On the transmit side, the unit continues retransfer until an ACK is received because it receives a NAK.
5. If the OVE flag is cleared without loading the receive data from IERBR in the RxE interrupt handling routine caused when the OVE flag is set to 1, the IEB decides that the overrun error has been cleared and sends an ACK to other units. In this case, the transmit unit completes the communications correctly. However, no receive data is loaded from the IERBR and the receive unit continues reception. Accordingly, in an interrupt handling routine caused by the OVE flag, receive data must be loaded from IERBR, the RxRDY flag must be cleared. The DTC, thus, should be ready to receive the next byte, and then the OVE flag must be cleared.
6. Item 5 above will not occur when the DTC transfer words is specified as the IERBFL value.

14.6.4 Error Flag s in the IETEF

(1) AL Flag

The AL Flag is set to 1 when arbitration is lost even if retransfer is performed for the number of times specified by IEMCR after arbitration has been lost. The AL flag is not set when arbitration is won during retransfer. If the AL flag is set to 1, the TxE flag is set and the wait state is entered.

(2) UE Flag

If the UE flag is set to 1, the TxE flag is set and the wait state is entered. For details, see section 14.6.2, TxRDY Flag and Underrun Error.

(3) TTME Flag

If a timing error occurs during data transfer, the TTME and TxE flags are set, and the wait state is entered.

(4) RO Flag

When retransfer is performed up to the maximum number of transfer bytes defined by the protocol because of reception of a NAK from the receive side during data field transmission, the number of transferred bytes may be less than that of bytes specified by the message length. At this time the RO flag is set. Moreover, when the value of the message length bits is greater than the maximum number of transfer bytes, the RO flag is also set. The RO flag is not set if the maximum number of transfer bytes defined by the protocol is specified (for example, 32-byte message length is specified in mode 1) and the transfer is performed correctly.

If the RO flag is set to 1, the TxE flag is set to 1 and the wait state is entered.

(5) ACK Flag

- If a NAK is received in an acknowledge bit before the message length field transmission, the ACK flag is set, the TxE flag is set, and then the wait state is entered.
- If a NAK is received in an acknowledge bit of the data field, data is automatically retransmitted up to the maximum number of transfer bytes defined by the protocol. If an ACK is received in an acknowledge bit during retransfer and the following data is transmitted correctly, the ACK flag is not set. If a NAK is received in the last data transfer during the retransfer for the maximum number of transfer bytes, the ACK flag is set to 1 and the wait state is entered.

Note: Even if a NAK is received from the receive side during the data field transmission, retransfer is performed up to the maximum number of transfer bytes defined by the protocol, and the number of transferred bytes is less than that of bytes specified by the message length bits, an ACK may be received in the acknowledge bit in the last data transfer. In this case, the ACK flag is not set although the RO flag is set.

14.6.5 Error Flags in IEREF

(1) OVE Flag

When the OVE flag is set, the RxE flag is also set. If an overrun error is cleared and the OVE flag is also cleared, the IEBus receive operation is continued. For details, see section 14.6.3, RxRDY Flag and Overrun Error.

(2) RTME Flag

If a timing error occurs during data reception after reception starts (the RxS flag is set to 1), the RTME flag is set to 1, RxE flag is set to 1, and the wait state is entered. When a timing error occurs before reception starts, this flag is not set and the reception frame is discarded.

(3) DLE Flag

When retransfer is performed up to the maximum number of transfer bytes defined by the protocol because of reception of a NAK caused by a parity or an overrun error during data field reception, the number of transferred bytes may be greater than that of bytes specified by the message length. At this time the DLE flag is set. Moreover, when the value of the message length bits is greater than the maximum number of transfer bytes, the DLE flag is also set. The DLE flag is not set if the maximum number of transfer bytes defined by the protocol is specified and the transfer is performed correctly.

If the DLE flag is set to 1, the RxE flag is set to 1 and the wait state is entered.

(4) PE Flag

If a parity error occurs after reception starts (the RxS flag is set to 1), a NAK is sent to perform re-reception.

If a parity error is not cleared when the maximum number of transfer bytes specified by the protocol is received, the PE flag is set to 1, the RxE flag is set to 1 and the wait state is entered. If a parity error is cleared during the rereception and if the following data is received correctly, the PE flag is not set.

- Notes:
1. If the reception is performed up to the maximum number of transfer bytes defined by the protocol because of a parity or an overrun error during data field reception, the number of receive bytes is less than that of bytes specified by the message length bits, no parity error or overrun error may occur at the last byte reception. In this case, the DLE flag is set. However, the OVE and PE flags are not set.
 2. The flags in IEREF are set after reception starts. Accordingly, the RxE flag is valid and set after the RxS flag has been set. If an error occurs before reception starts, the frame is discarded and no interrupt occurs.

14.6.6 Notes on Slave Transmission

When the slave unit transmits the slave status and upper and lower locked addresses, a parity or an overrun error occurs in the master reception side and the data cannot be received. Accordingly, even if a NAK is returned, the slave unit is not capable of retransfer.

In this case, the master unit must discard the frame in which an error occurred and request the above operation in the master reception to receive the correct frame.

14.6.7 Notes on DTC Specification

When transmit or receive data is transferred by the DTC, bit 5 (for transmission) or bit 6 (for reception) in DTCERG must be set by the bit manipulation instruction (such as BSET or BCLR). In this case, other bits (bits 7 and 4 to 0) in DTCERG must not be set to 1.

14.6.8 Error Handling in Transmission

Figure 14.15 shows the operation when a timing error occurs.

When a timing error occurs in data transmission (1), there is a possibility that the next data is already transferred to the transmit buffer by the DTC and the TxRDY flag that is the DTC initiation source is already cleared to 0 (2).

In this case, if retransfer is performed, data remained in the transmit buffer (previous frame data) is transmitted as the first byte data of the data field (3).

To avoid this error, in master transmission, the first byte data in the data field should be written to the transmit buffer by software instead of using the DTC. After that, data can be transferred by the DTC. In this case, the SAR (transfer source address) and CRA (transfer counter) should be specified as follows.

- An address of the on-chip memory that stores the second byte data → SAR
- The number of bytes specified by message length – 1 → CRA

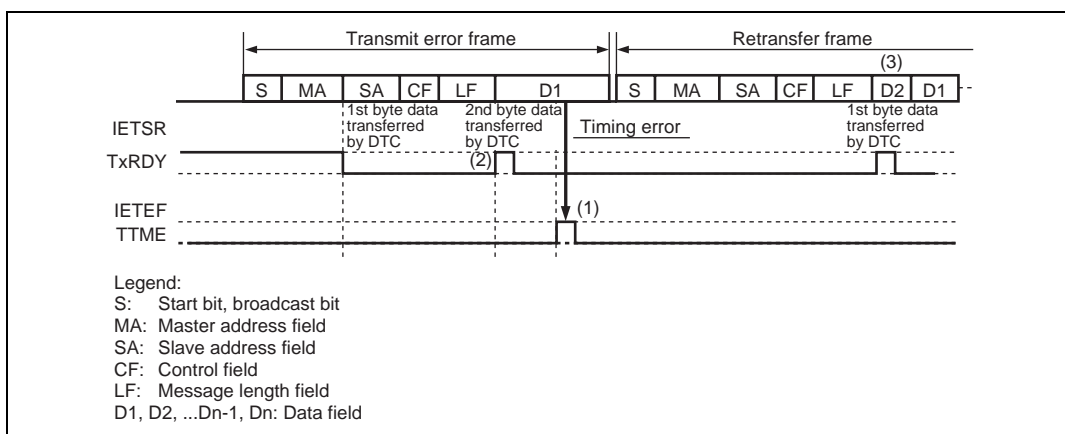


Figure 14.15 Error Processing in Transfer

14.6.9 Power-Down Mode Operation

The IEB stops operation and is initialized in power-down modes such as module stop, watch, software standby and hardware standby modes.

To initialize the IEB, the module stop mode must be specified. To reduce power consumption during IEB operation, the sleep mode must be used.

14.6.10 Notes on Middle-Speed Mode

In middle-speed mode, the IEB registers must not be read from or written to.

14.6.11 Notes on Register Access

The IEB registers can be accessed in bytes. The IEB registers must not be accessed in words or longwords.

Section 15 Serial Communication Interface (SCI)

This LSI has independent serial communication interfaces (SCIs). The SCI can handle both asynchronous and clocked synchronous serial communication. Serial data communication can be carried out using standard asynchronous communication chips such as a Universal Asynchronous Receiver/Transmitter (UART) or an Asynchronous Communication Interface Adapter (ACIA). A function is also provided for serial communication between processors (multiprocessor communication function). The SCI also supports an IC card (Smart Card) interface conforming to ISO/IEC 7816-3 (Identification Card) as a serial communication interface extended function.

15.1 Features

- The number of on-chip channels
H8S/2258 Group, H8S/2239 Group, H8S/2238 Group, and H8S/2237 Group: Four channels (channels 0, 1, 2, and 3)
H8S/2227 Group: Three channels (channels 0, 1, and 3)
- Choice of asynchronous or clocked synchronous serial communication mode
- Full-duplex communication capability
The transmitter and receiver are mutually independent, enabling transmission and reception to be executed simultaneously.
Double-buffering is used in both the transmitter and the receiver, enabling continuous transmission and continuous reception of serial data.
- On-chip baud rate generator allows any bit rate to be selected
External clock can be selected as a transfer clock source (except for in Smart Card interface mode).
- Choice of LSB-first or MSB-first transfer (except in the case of asynchronous mode 7-bit data)
- Four interrupt sources
Transmit-end, transmit-data-empty, receive-data-full, and receive error — that can issue requests.
The transmit-data-empty interrupt and receive data full interrupts can be used to activate the data transfer controller (DTC) or the direct memory access controller (DMAC) (H8S/2239 Group only).
- Module stop mode can be set

Asynchronous mode

- Data length: 7 or 8 bits
- Stop bit length: 1 or 2 bits

- Parity: Even, odd, or none
- Receive error detection: Parity, overrun, and framing errors
- Break detection: Break can be detected by reading the RxD pin level directly in the case of a framing error
- Average transfer rate generator (SCI_0): 720 kbps, 460.784 kbps, or 115.192 kbps can be selected at 16-MHz operation (H8S/2239 Group only).
- Transfer rate clock can be input from the TPU (SCI_0) (H8S/2239 Group only).
- Communications between multi-processors are possible.

Clocked Synchronous mode

- Data length: 8 bits
- Receive error detection: Overrun errors detected
- SCI selection (SCI_0) : When $\overline{\text{IRQ7}} = 1$, fixed input of TxD0 = Hi-Z and SCK0 = High can be selected. (H8S/2239 Group only)

Smart Card Interface

- Automatic transmission of error signal (parity error) in receive mode
- Error signal detection and automatic data retransmission in transmit mode
- Direct convention and inverse convention both supported



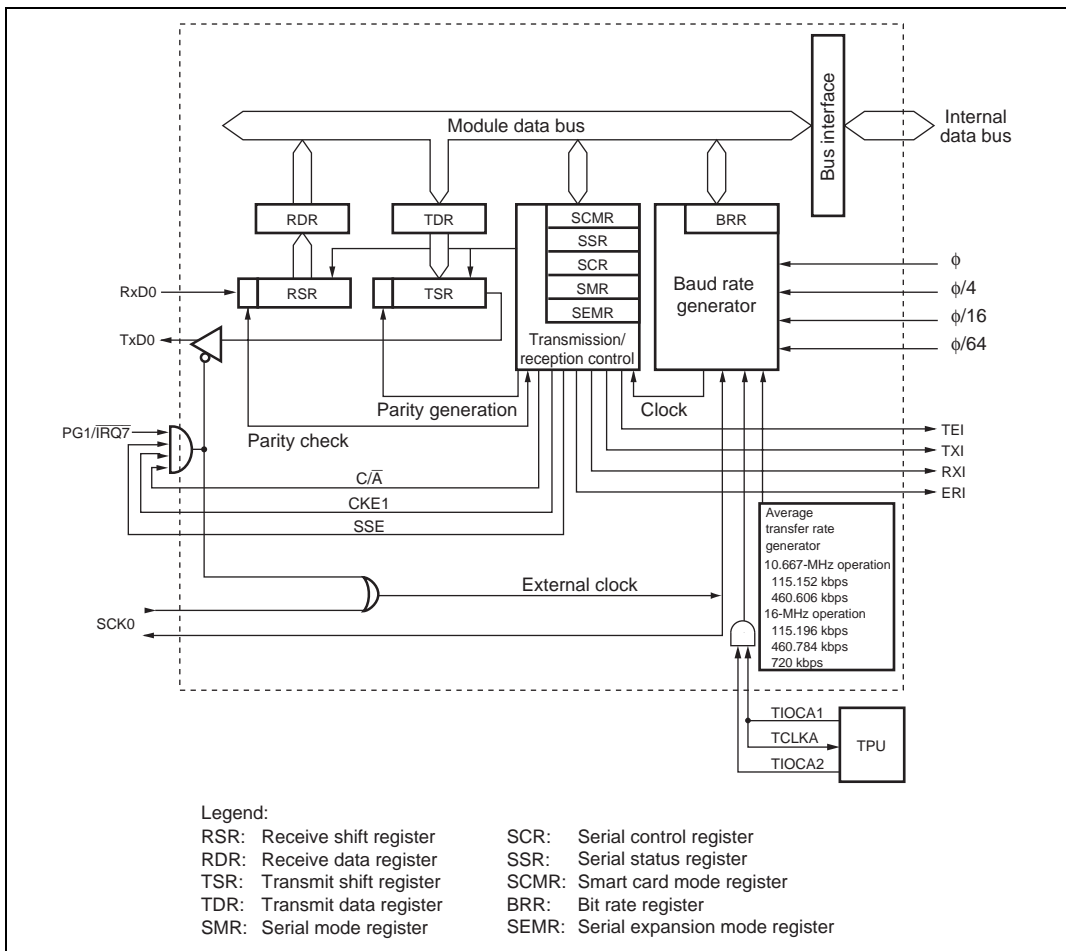


Figure 15.2 Block Diagram of SCI_0 of H8S/2239 Group

15.2 Input/Output Pins

Table 15.1 shows the pin configuration for each SCI channel.

Table 15.1 Pin Configuration

| Channel | Pin Name ^{*1} | I/O | Function |
|-----------------|------------------------|--------|---------------------------|
| 0 | SCK0 | I/O | SCI0 clock input/output |
| | RxD0 | Input | SCI0 receive data input |
| | TxD0 | Output | SCI0 transmit data output |
| 1 | SCK1 | I/O | SCI1 clock input/output |
| | RxD1 | Input | SCI1 receive data input |
| | TxD1 | Output | SCI1 transmit data output |
| 2 ^{*2} | SCK2 | I/O | SCI2 clock input/output |
| | RxD2 | Input | SCI2 receive data input |
| | TxD2 | Output | SCI2 transmit data output |
| 3 | SCK3 | I/O | SCI3 clock input/output |
| | RxD3 | Input | SCI3 receive data input |
| | TxD3 | Output | SCI3 transmit data output |

Notes: 1. Pin names SCK, RxD, and TxD are used in the text for all channels, omitting the channel designation.

2. The channel is not provided for the H8S/2227 Group.

15.3 Register Descriptions

The SCI has the following registers for each channel. For details on register addresses and register states during each process, refer to appendix A, Internal I/O Register. The serial mode register (SMR), serial status register (SSR), and serial control register (SCR) are described separately for normal serial communication interface mode and Smart Card interface mode because their bit functions differ in part.

- Receive shift register (RSR)
- Receive data register (RDR)
- Transmit data register (TDR)
- Transmit shift register (TSR)
- Serial mode register (SMR)
- Serial control register (SCR)

- Serial status register (SSR)
- Smart card mode register (SCMR)
- Bit rate register (BRR)
- Serial expansion mode register (SEMR0)*

Note: * This register is in the channel 0 of the H8S/2239 Group only.

15.3.1 Receive Shift Register (RSR)

RSR is a shift register that is used to receive serial data input to the RxD pin and convert it into parallel data. When one byte of data has been received, it is transferred to RDR automatically.

RSR cannot be directly accessed by the CPU.

15.3.2 Receive Data Register (RDR)

RDR is an 8-bit register that stores received data. When the SCI has received one byte of serial data, it transfers the received serial data from RSR to RDR, where it is stored. After this, RSR is receive-enabled. As RSR and RDR function as a double buffer in this way, continuous receive operations are possible. After confirming that the RDRF bit in SSR is set to 1, read RDR only once.

RDR cannot be written to by the CPU.

RDR is initialized to H'00 by a reset, in standby mode, watch mode, subactive mode, subsleep mode, or module stop mode.

15.3.3 Transmit Data Register (TDR)

TDR is an 8-bit register that stores data for transmission. When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission. The double-buffered structure of TDR and TSR enables continuous serial transmission. If the next transmit data has already been written to TDR during serial transmission, the SCI transfers the written data to TSR to continue transmission. Although TDR can be read or written to by the CPU at all times, to achieve reliable serial transmission, write transmit data to TDR only once after confirming that the TDRE bit in SSR is set to 1.

TDR is initialized to H'FF by a reset, in standby mode, watch mode, subactive mode, subsleep mode or module stop mode.

15.3.4 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data. To perform serial data transmission, the SCI first transfers transmit data from TDR to TSR, then sends the data to the TxD pin. TSR cannot be directly accessed by the CPU.

15.3.5 Serial Mode Register (SMR)

SMR is used to set the SCI's serial transfer format and select the baud rate generator clock source.

Some bit functions of SMR differ between normal serial communication interface mode and Smart Card interface mode.

- Normal Serial Communication Interface Mode (When SMIF in SCMR is 0)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|--------------|---------------|-----|---|
| 7 | C/ \bar{A} | 0 | R/W | Communication Mode 0: Asynchronous mode 1: Clocked synchronous mode |
| 6 | CHR | 0 | R/W | Character Length (enabled only in asynchronous mode) 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length. LSB-first is fixed and the MSB (bit 7) of TDR is not transmitted in transmission. In clocked synchronous mode, a fixed data length of 8 bits is used. |
| 5 | PE | 0 | R/W | Parity Enable (enabled only in asynchronous mode) When this bit is set to 1, the parity bit is added to transmit data before transmission, and the parity bit is checked in reception. For a multiprocessor format, parity bit addition and checking are not performed regardless of the PE bit setting. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|--------------------------|---------------|-----|---|
| 4 | O/ $\overline{\text{E}}$ | 0 | R/W | <p>Parity Mode (enabled only when the PE bit is 1 in asynchronous mode)</p> <p>0: Selects even parity.</p> <p>When even parity is set, parity bit addition is performed in transmission so that the total number of 1 bits in the transmit character plus the parity bit is even. In reception, a check is performed to see if the total number of 1 bits in the receive character plus parity bit is even.</p> <p>1: Selects odd parity.</p> <p>When odd parity is set, parity bit addition is performed in transmission so that the total number of 1 bits in the transmit character plus the parity bit is odd. In reception, a check is performed to see if the total number of 1 bits in the receive character plus the parity bit is odd.</p> |
| 3 | STOP | 0 | R/W | <p>Stop Bit Length (enabled only in asynchronous mode)</p> <p>Selects the stop bit length in transmission.</p> <p>0: 1 stop bit</p> <p>1: 2 stop bits</p> <p>In reception, only the first stop bit is checked. If the second stop bit is 0, it is treated as the start bit of the next transmit character.</p> |
| 2 | MP | 0 | R/W | <p>Multiprocessor Mode (enabled only in asynchronous mode)</p> <p>When this bit is set to 1, the multiprocessor communication function is enabled. The PE bit and O/$\overline{\text{E}}$ bit settings are invalid in multiprocessor mode.</p> <p>For details, see section 15.5, Multiprocessor Communication Function.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | CKS1 | 0 | R/W | Clock Select 0 and 1 |
| 0 | CKS0 | 0 | R/W | <p>These bits select the clock source for the baud rate generator.</p> <p>00: ϕ clock ($n = 0$)</p> <p>01: $\phi/4$ clock ($n = 1$)</p> <p>10: $\phi/16$ clock ($n = 2$)</p> <p>11: $\phi/64$ clock ($n = 3$)</p> <p>For the relationship between the bit rate register setting and the baud rate, see section 15.3.9, Bit Rate Register (BRR). n is the decimal representation of the value of n in BRR (see section 15.3.9, Bit Rate Register (BRR)).</p> |

- Smart Card Interface Mode (When SMIF in SCMR is 1)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | GM | 0 | R/W | <p>GSM Mode</p> <p>When this bit is set to 1, the SCI operates in GSM mode. In GSM mode, the timing of the TEND setting is advanced by 11.0 etu (Elementary Time Unit: the time for transfer of 1 bit), and clock output control mode addition is performed. For details, refer to section 15.7.8, Clock Output Control.</p> <p>0: Normal smart card interface mode operation (initial value)</p> <ul style="list-style-type: none"> The TEND flag is generated 12.5 etu (11.5 etu in the block transfer mode) after the beginning of the start bit. Clock output on/off control only <p>1: GSM mode operation in smart card interface mode</p> <ul style="list-style-type: none"> The TEND flag is generated 11.0 etu after the beginning of the start bit. In addition to clock output on/off control, high/low fixed control is supported (set using SCR). |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|-------------------|---------------|-----|--|
| 6 | BLK | 0 | R/W | <p>When this bit is set to 1, the SCI operates in block transfer mode. For details on block transfer mode, refer to section 15.7.3, Block Transfer Mode.</p> <p>0: Normal smart card interface mode operation (initial value)</p> <ul style="list-style-type: none"> Error signal transmission, detection, and automatic data retransmission are performed. The TXI interrupt is generated by the TEND flag. The TEND flag is set 12.5 etu (11.0 etu in the GSM mode) after transmission starts. <p>1: Operation in block transfer mode</p> <ul style="list-style-type: none"> Error signal transmission, detection, and automatic data retransmission are not performed. The TXI interrupt is generated by the TDRE flag. The TEND flag is set 11.5 etu (11.0 etu in the GSM mode) after transmission starts. |
| 5 | PE | 0 | R/W | <p>Parity Enable (enabled only in asynchronous mode)</p> <p>When this bit is set to 1, the parity bit is added to transmit data in transmission, and the parity bit is checked in reception. In Smart Card interface mode, this bit must be set to 1.</p> |
| 4 | O/ \overline{E} | 0 | R/W | <p>Parity Mode (enabled only when the PE bit is 1 in asynchronous mode)</p> <p>0: Selects even parity.</p> <p>1: Selects odd parity.</p> <p>For details on setting this bit in Smart Card interface mode, refer to section 15.7.2, Data Format (Except for Block Transfer Mode).</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | BCP1 | 0 | R/W | Base Clock Pulse 0 and 1 |
| 2 | BCP0 | 0 | R/W | <p>These bits specify the number of base clock periods in a 1-bit transfer interval on the Smart Card interface.</p> <p>00: 32 clock (S = 32)</p> <p>01: 64 clock (S = 64)</p> <p>10: 372 clock (S = 372)</p> <p>11: 256 clock (S = 256)</p> <p>For details, refer to section 15.7.4, Receive Data Sampling Timing and Reception Margin. S stands for the value of S in BRR (see section 15.3.9, Bit Rate Register (BRR)).</p> |
| 1 | CKS1 | 0 | R/W | Clock Select 0 and 1 |
| 0 | CKS0 | 0 | R/W | <p>These bits select the clock source for the baud rate generator.</p> <p>00: ϕ clock (n = 0)</p> <p>01: $\phi/4$ clock (n = 1)</p> <p>10: $\phi/16$ clock (n = 2)</p> <p>11: $\phi/64$ clock (n = 3)</p> <p>For the relationship between the bit rate register setting and the baud rate, see section 15.3.9, Bit Rate Register (BRR). n is the decimal representation of the value of n in BRR (see section 15.3.9, Bit Rate Register (BRR)).</p> |

Note: etu (Elementary Time Unit): Time for transfer of 1 bit

15.3.6 Serial Control Register (SCR)

SCR is a register that enables or disables SCI transfer operations and interrupt requests, and is also used to selection of the transfer clock source. For details on interrupt requests, refer to section 15.9, Interrupt Sources. Some bit functions of SCR differ between normal serial communication interface mode and Smart Card interface mode.

- Normal Serial Communication Interface Mode (When SMIF in SCMR is 0)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | TIE | 0 | R/W | <p>Transmit Interrupt Enable</p> <p>When this bit is set to 1, the TXI interrupt request is enabled.</p> <p>TXI interrupt request cancellation can be performed by reading 1 from the TDRE flag in SSR, then clearing it to 0, or clearing the TIE bit to 0.</p> |
| 6 | RIE | 0 | R/W | <p>Receive Interrupt Enable</p> <p>When this bit is set to 1, RXI and ERI interrupt requests are enabled.</p> <p>RXI and ERI interrupt request cancellation can be performed by reading 1 from the RDRF, FER, PER, or ORER flag in SSR, then clearing the flag to 0, or clearing the RIE bit to 0.</p> |
| 5 | TE | 0 | R/W | <p>Transmit Enable</p> <p>When this bit is set to 1, transmission is enabled.</p> <p>In this state, serial transmission is started when transmit data is written to TDR and the TDRE flag in SSR is cleared to 0.</p> <p>SMR setting must be performed to decide the transfer format before setting the TE bit to 1. When this bit is cleared to 0, the transmission operation is disabled, and the TDRE flag is fixed at 1.</p> |
| 4 | RE | 0 | R/W | <p>Receive Enable</p> <p>When this bit is set to 1, reception is enabled.</p> <p>Serial reception is started in this state when a start bit is detected in asynchronous mode or serial clock input is detected in clocked synchronous mode.</p> <p>SMR setting must be performed to decide the reception format before setting the RE bit to 1.</p> <p>Clearing the RE bit to 0 does not affect the RDRF, FER, and ORER flags, which retain their states.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | MPIE | 0 | R/W | <p>Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 in asynchronous mode)</p> <p>When this bit is set to 1, receive data in which the multiprocessor bit is 0 is skipped, and setting of the RDRF, FER, and ORER status flags in SSR is prohibited. On receiving data in which the multiprocessor bit is 1, this bit is automatically cleared and normal reception is resumed. For details, refer to section 15.5, Multiprocessor Communication Function.</p> <p>When receive data including MPB = 0 is received, receive data transfer from RSR to RDR, receive error detection, and setting of the RERF, FER, and ORER flags in SSR, are not performed.</p> <p>When receive data including MPB = 1 is received, the MPB bit in SSR is set to 1, the MPIE bit is cleared to 0 automatically, and generation of RXI and ERI interrupts (when the TIE and RIE bits in SCR are set to 1) and FER and ORER flag setting are enabled.</p> |
| 2 | TEIE | 0 | R/W | <p>Transmit End Interrupt Enable</p> <p>This bit is set to 1, TEI interrupt request is enabled.</p> <p>TEI cancellation can be performed by reading 1 from the DRE flag in SSR, then clearing it to 0 and clearing the TEND flag to 0, or clearing the TEIE bit to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | CKE1 | 0 | R/W | Clock Enable 0 and 1 |
| 0 | CKE0 | 0 | R/W | Selects the clock source and SCK pin function. Asynchronous mode 00: On-chip baud rate generator SCK pin functions as I/O port 01: On-chip baud rate generator Outputs a clock of the same frequency as the bit rate from the SCK pin. 1x: External clock Inputs a clock with a frequency 16 times the bit rate from the SCK pin. Clocked synchronous mode 0x: Internal clock (SCK pin functions as clock output) 1x: External clock (SCK pin functions as clock input) |

Legend:

×: Don't care

- Smart Card Interface Mode (When SMIF in SCMR is 1)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | TIE | 0 | R/W | <p>Transmit Interrupt Enable</p> <p>When this bit is set to 1, TXI interrupt request is enabled.</p> <p>TXI interrupt request cancellation can be performed by reading 1 from the TDRE flag in SSR, then clearing it to 0, or clearing the TIE bit to 0.</p> |
| 6 | RIE | 0 | R/W | <p>Receive Interrupt Enable</p> <p>When this bit is set to 1, RXI and ERI interrupt requests are enabled.</p> <p>RXI and ERI interrupt request cancellation can be performed by reading 1 from the RDRF, FER, PER, or ORER flag in SSR, then clearing the flag to 0, or clearing the RIE bit to 0.</p> |
| 5 | TE | 0 | R/W | <p>Transmit Enable</p> <p>When this bit is set to 1, transmission is enabled.</p> <p>In this state, serial transmission is started when transmit data is written to TDR and the TDRE flag in SSR is cleared to 0.</p> <p>SMR setting must be performed to decide the transfer format before setting the TE bit to 1. When this bit is cleared to 0, the transmission operation is disabled, and the TDRE flag is fixed at 1.</p> |
| 4 | RE | 0 | R/W | <p>Receive Enable</p> <p>When this bit is set to 1, reception is enabled.</p> <p>Serial reception is started in this state when a start bit is detected in asynchronous mode or serial clock input is detected in clocked synchronous mode.</p> <p>SMR setting must be performed to decide the reception format before setting the RE bit to 1.</p> <p>Clearing the RE bit to 0 does not affect the RDRF, FER, and ORER flags, which retain their states.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 3 | MPIE | 0 | R/W | <p>Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 in asynchronous mode)</p> <p>Write 0 to this bit in Smart Card interface mode.</p> <p>When receive data including MPB = 0 is received, receive data transfer from RSR to RDR, receive error detection, and setting of the RERF, FER, and ORER flags in SSR, are not performed.</p> <p>When receive data including MPB = 1 is received, the MPB bit in SSR is set to 1, the MPIE bit is cleared to 0 automatically, and generation of RXI and ERI interrupts (when the TIE and RIE bits in SCR are set to 1) and FER and ORER flag setting are enabled.</p> |
| 2 | TEIE | 0 | R/W | <p>Transmit End Interrupt Enable</p> <p>Write 0 to this bit in Smart Card interface mode.</p> <p>TEI cancellation can be performed by reading 1 from the TDRE flag in SSR, then clearing it to 0 and clearing the TEND flag to 0, or clearing the TEIE bit to 0.</p> |
| 1 | CKE1 | 0 | R/W | Clock Enable 0 and 1 |
| 0 | CKE0 | 0 | | <p>Enables or disables clock output from the SCK pin. The clock output can be dynamically switched in GSM mode. For details, refer to section 15.7.8, Clock Output Control.</p> <p>When the GM bit in SMR is 0:</p> <p>00: Output disabled (SCK pin can be used as an I/O port pin)</p> <p>01: Clock output</p> <p>1x: Reserved</p> <p>When the GM bit in SMR is 1:</p> <p>00: Output fixed low</p> <p>01: Clock output</p> <p>10: Output fixed high</p> <p>11: Clock output</p> |

Legend:

x: Don't care

15.3.7 Serial Status Register (SSR)

SSR is a register containing status flags of the SCI and multiprocessor bits for transfer. 1 cannot be written to flags TDRE, RDRF, ORER, PER, and FER; they can only be cleared. Some bit functions of SSR differ between normal serial communication interface mode and Smart Card interface mode.

- Normal Serial Communication Interface Mode (When SMIF in SCMR is 0)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|---|
| 7 | TDRE | 1 | R/(W) ^{*1} | <p>Transmit Data Register Empty</p> <p>Displays whether TDR contains transmit data.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • When the TE bit in SCR is 0 • When data is transferred from TDR to TSR and data can be written to TDR <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC^{*2} or the DTC^{*3} is activated by a TXI interrupt request and writes data to TDR |
| 6 | RDRF | 0 | R/(W) ^{*1} | <p>Receive Data Register Full</p> <p>Indicates that the received data is stored in RDR.</p> <p>[Setting condition]</p> <p>When serial reception ends normally and receive data is transferred from RSR to RDR</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written to RDRF after reading RDRF = 1 • When the DMAC^{*2} or the DTC^{*3} is activated by an RXI interrupt and transferred data from RDR <p>The RDRF flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.</p> <p>If reception of the next data is completed while the RDRF flag is still set to 1, an overrun error will occur and the receive data will be lost.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------|---|
| 5 | ORER | 0 | R/(W)*1 | <p>Overrun Error</p> <p>Indicates that an overrun error occurred during reception, causing abnormal termination.</p> <p>[Setting condition]</p> <p>When the next serial reception is completed while RDRF = 1</p> <p>The receive data prior to the overrun error is retained in RDR, and the data received subsequently is lost. Also, subsequent serial reception cannot be continued while the ORER flag is set to 1. In clocked synchronous mode, serial transmission cannot be continued either.</p> <p>[Clearing condition]</p> <p>When 0 is written to ORER after reading ORER = 1</p> <p>The ORER flag is not affected and retains its previous state when the RE bit in SCR is cleared to 0.</p> |
| 4 | FER | 0 | R/(W)*1 | <p>Framing Error</p> <p>Indicates that a framing error occurred during reception in asynchronous mode, causing abnormal termination.</p> <p>[Setting condition]</p> <p>When the stop bit is 0</p> <p>In 2 stop bit mode, only the first stop bit is checked for a value to 1; the second stop bit is not checked. If a framing error occurs, the receive data is transferred to RDR but the RDRF flag is not set. Also, subsequent serial reception cannot be continued while the FER flag is set to 1. In clocked synchronous mode, serial transmission cannot be continued, either.</p> <p>[Clearing condition]</p> <p>When 0 is written to FER after reading FER = 1</p> <p>In 2-stop-bit mode, only the first stop bit is checked.</p> <p>The FER flag is not affected and retains its previous state when the RE bit in SCR is cleared to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|--|
| 3 | PER | 0 | R/(W) ^{*1} | <p>Parity Error</p> <p>Indicates that a parity error occurred during reception using parity addition in asynchronous mode, causing abnormal termination.</p> <p>[Setting condition]</p> <p>When a parity error is detected during reception</p> <p>If a parity error occurs, the receive data is transferred to RDR but the RDRF flag is not set. Also, subsequent serial reception cannot be continued while the PER flag is set to 1. In clocked synchronous mode, serial transmission cannot be continued, either.</p> <p>[Clearing condition]</p> <p>When 0 is written to PER after reading PER = 1</p> <p>The PER flag is not affected and retains its previous state when the RE bit in SCR is cleared to 0.</p> |
| 2 | TEND | 1 | R | <p>Transmit End</p> <p>Indicates that transmission has been ended.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • When the TE bit in SCR is 0 • When TDRE = 1 at transmission of the last bit of a 1-byte serial transmit character <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC^{*2} or the DTC^{*3} is activated by a TXI interrupt request and transfer transmission data to TDR |
| 1 | MPB | 0 | R | <p>Multiprocessor Bit</p> <p>MPB stores the multiprocessor bit in the receive data. When the RE bit in SCR is cleared to 0 its previous state is retained.</p> |
| 0 | MPBT | 0 | R/W | <p>Multiprocessor Bit Transfer</p> <p>MPBT stores the multiprocessor bit to be added to the transmit data.</p> |

- Notes:
1. Only a 0 can be written to this bit, to clear the flag.
 2. Supported only by the H8S/2239 Group.
 3. DTC can clear this bit only when DISEL is 0 with the transfer counter not being 0.

- Smart Card Interface Mode (When SMIF in SCMR is 1)

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|--|
| 7 | TDRE | 1 | R/(W) ^{*1} | <p>Transmit Data Register Empty</p> <p>Indicates whether TDR contains transmit data.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • When the TE bit in SCR is 0 • When data is transferred from TDR to TSR and data can be written to TDR <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written to TDRE after reading TDRE = 1 • When the DMAC^{*2} or the DTC^{*3} is activated by a TXI interrupt request and writes data to TDR |
| 6 | RDRF | 0 | R/(W) ^{*1} | <p>Receive Data Register Full</p> <p>Indicates that the received data is stored in RDR.</p> <p>[Setting condition]</p> <p>When serial reception ends normally and receive data is transferred from RSR to RDR</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written to RDRF after reading RDRF = 1 • When the DTC^{*3} is activated by an RXI interrupt and transferred data from RDR <p>The RDRF flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.</p> <p>If reception of the next data is completed while the RDRF flag is still set to 1, an overrun error will occur and the receive data will be lost.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------|---|
| 5 | ORER | 0 | R/(W)*1 | <p>Overrun Error</p> <p>Indicates that an overrun error occurred during reception, causing abnormal termination.</p> <p>[Setting condition]</p> <p>When the next serial reception is completed while RDRF = 1</p> <p>The receive data prior to the overrun error is retained in RDR, and the data received subsequently is lost. Also, subsequent serial cannot be continued while the ORER flag is set to 1. In clocked synchronous mode, serial transmission cannot be continued, either.</p> <p>[Clearing condition]</p> <p>When 0 is written to ORER after reading ORER = 1</p> <p>The ORER flag is not affected and retains its previous state when the RE bit in SCR is cleared to 0.</p> |
| 4 | ERS | 0 | R/(W)*1 | <p>Error Signal Status</p> <p>Indicates that the status of an error signal returned from the receiving end at reception</p> <p>[Setting condition]</p> <p>When the low level of the error signal is sampled</p> <p>[Clearing condition]</p> <p>When 0 is written to ERS after reading ERS = 1</p> <p>The ERS flag is not affected and retains its previous state when the TE bit in SCR is cleared to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|---------------------|--|
| 3 | PER | 0 | R/(W) ^{*1} | <p>Parity Error</p> <p>Indicates that a parity error occurred during reception using parity addition in asynchronous mode, causing abnormal termination.</p> <p>[Setting condition]</p> <p>When a parity error is detected during reception</p> <p>If a parity error occurs, the receive data is transferred to RDR but the RDRF flag is not set. Also, subsequent serial reception cannot be continued while the PER flag is set to 1. In clocked synchronous mode, serial transmission cannot be continued, either.</p> <p>[Clearing condition]</p> <p>When 0 is written to PER after reading PER = 1</p> <p>The PER flag is not affected and retains its previous state when the RE bit in SCR is cleared to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | TEND | 1 | R | <p>Transmit End</p> <p>This bit is set to 1 when no error signal has been sent back from the receiving end and the next transmit data is ready to be transferred to TDR.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> When the TE bit in SCR is 0 and the ERS bit is also 0 When the ERS bit is 0 and the TDRE bit is 1 after the specified interval following transmission of 1-byte data. The timing of bit setting differs according to the register setting as follows: When GM = 0 and BLK = 0, 12.5 etu after transmission starts When GM = 0 and BLK = 1, 11.5 etu after transmission starts When GM = 1 and BLK = 0, 11.0 etu after transmission starts When GM = 1 and BLK = 1, 11.0 etu after transmission starts <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is written to TDRE after reading TDRE = 1 When the DMAC^{*2} or the DTC^{*3} is activated by a TXI interrupt and transfers transmission data to TDR |
| 1 | MPB | 0 | R | <p>Multiprocessor Bit</p> <p>This bit is not used in Smart Card interface mode.</p> |
| 0 | MPBT | 0 | R/W | <p>Multiprocessor Bit Transfer</p> <p>Write 0 to this bit in Smart Card interface mode.</p> |

Notes: 1. Only 0 can be written to this bit, to clear the flag.

2. Supported only by the H8S/2239 Group.

3. DTC can clear this bit only when DISEL is 0 with the transfer counter not being 0.

15.3.8 Smart Card Mode Register (SCMR)

SCMR is a register that selects Smart Card interface mode and its transfer format.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 to 4 | — | All 1 | — | Reserved These bits are always read as 1, and cannot be modified. |
| 3 | SDIR | 0 | R/W | Smart Card Data Transfer Direction Selects the serial/parallel conversion format. 0: LSB-first in transfer 1: MSB-first in transfer The bit setting is valid only when the transfer data format is 8 bits. Except in the case of 7-bit data in asynchronous mode, either LSB-first or MSB-first may be selected regardless of the serial communication mode. For 7-bit data, set this bit to 0 to select LSB-first in transfer. |
| 2 | SINV | 0 | R/W | Smart Card Data Invert Specifies inversion of the data logic level. The SINV bit does not affect the logic level of the parity bit. To invert the parity bit, invert the O/E bit in SMR. 0: TDR contents are transmitted as they are. Receive data is stored as it is in RDR 1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR |
| 1 | — | 1 | — | Reserved This bit is always read as 1, and cannot be modified. |
| 0 | SMIF | 0 | R/W | Smart Card Interface Mode Select This bit is set to 1 to make the SCI operate in Smart Card interface mode. 0: Normal asynchronous mode or clocked synchronous mode 1: Smart card interface mode |

15.3.9 Bit Rate Register (BRR)

BRR is an 8-bit register that adjusts the bit rate. As the SCI performs baud rate generator control independently for each channel, different bit rates can be set for each channel. Table 15.2 shows the relationships between the N setting in BRR and bit rate B for normal asynchronous mode, clocked synchronous mode, and Smart Card interface mode. The initial value of BRR is H'FF, and it can be read or written to by the CPU at all times.

Table 15.2 The Relationships between the N Setting in BRR and Bit Rate B

| Communication Mode | ABCS bit* | Bit Rate | Error |
|---------------------------|-----------|--|---|
| Asynchronous Mode | 0 | $B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (N+1)}$ | $\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$ |
| | 1 | $B = \frac{\phi \times 10^6}{32 \times 2^{2n-1} \times (N+1)}$ | $\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$ |
| Clocked Synchronous Mode | — | $B = \frac{\phi \times 10^6}{8 \times 2^{2n-1} \times (N+1)}$ | — |
| Smart Card Interface Mode | — | $B = \frac{\phi \times 10^6}{S \times 2^{2n+1} \times (N+1)}$ | $\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1 \right\} \times 100$ |

Legend:

B: Bit rate (bit/s)

N: BRR setting for baud rate generator ($0 \leq N \leq 255$)

ϕ : Operating frequency (MHz)

n and S: Determined by the SMR settings shown in the following tables.

Note: * If the ABCS bit is set to 1, SCI_0 on the H8S/2239 Group only valid bit rate.

| SMR Setting | | Clock Source | n |
|-------------|------|--------------|---|
| CKS1 | CKS0 | | |
| 0 | 0 | ϕ | 0 |
| 0 | 1 | $\phi/4$ | 1 |
| 1 | 0 | $\phi/16$ | 2 |
| 1 | 1 | $\phi/64$ | 3 |

| SMR Setting | | |
|-------------|------|-----|
| BGP1 | BGP0 | S |
| 0 | 0 | 32 |
| 0 | 1 | 64 |
| 1 | 0 | 372 |
| 1 | 1 | 256 |

Table 15.3 shows sample N settings in BRR in normal asynchronous mode. Table 15.4 shows the maximum bit rate for each frequency in normal asynchronous mode. Table 15.6 shows sample N settings in BRR in clocked synchronous mode. Table 15.8 shows sample N settings in BRR in Smart Card interface mode. In Smart Card interface mode, S (the number of base clock periods in a 1-bit transfer interval) can be selected. For details, refer to section 15.7.4, Receive Data Sampling Timing and Reception Margin. Tables 15.5 and 15.7 show the maximum bit rates with external clock input.

When the ABCS bit in SEMR_0 of SCI_0 is set to 1 in asynchronous mode, the maximum bit rate is twice the value shown in tables 15.4 and 15.5 (valid for H8S/2239 Group only).

Table 15.3 BRR Settings for Various Bit Rates (Asynchronous Mode)

| Bit Rate (bps) ^{*1} | Operating Frequency ϕ (MHz) | | | | | | | | | | | |
|---------------------------------|----------------------------------|-----|-----------|------------------------|-----|-----------|----------------------|-----|-----------|-----------------|-----|-----------|
| | 2 ^{*3} | | | 2.097152 ^{*3} | | | 2.4576 ^{*3} | | | 3 ^{*3} | | |
| | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 1 | 141 | 0.03 | 1 | 148 | -0.04 | 1 | 174 | -0.26 | 1 | 212 | 0.03 |
| 150 | 1 | 103 | 0.16 | 1 | 108 | 0.21 | 1 | 127 | 0.00 | 1 | 155 | 0.16 |
| 300 | 0 | 207 | 0.16 | 0 | 217 | 0.21 | 0 | 255 | 0.00 | 1 | 77 | 0.16 |
| 600 | 0 | 103 | 0.16 | 0 | 108 | 0.21 | 0 | 127 | 0.00 | 0 | 155 | 0.16 |
| 1200 | 0 | 51 | 0.16 | 0 | 54 | -0.70 | 0 | 63 | 0.00 | 0 | 77 | 0.16 |
| 2400 | 0 | 25 | 0.16 | 0 | 26 | 1.14 | 0 | 31 | 0.00 | 0 | 38 | 0.16 |
| 4800 | 0 | 12 | 0.16 | 0 | 13 | -2.48 | 0 | 15 | 0.00 | 0 | 19 | -2.34 |
| 9600 | — | — | — | 0 | 6 | -2.48 | 0 | 7 | 0.00 | 0 | 9 | -2.34 |
| 19200 | — | — | — | — | — | — | 0 | 3 | 0.00 | 0 | 4 | -2.34 |
| 31250 | 0 | 1 | 0.00 | — | — | — | — | — | — | 0 | 2 | 0.00 |
| 38400 | — | — | — | — | — | — | 0 | 1 | 0.00 | — | — | — |

| Bit Rate (bps) ^{*1} | Operating Frequency ϕ (MHz) | | | | | | | | | | | |
|---------------------------------|----------------------------------|-----|-----------|-----------------|-----|-----------|----------------------|-----|-----------|-----------------|-----|-----------|
| | 3.6864 ^{*3} | | | 4 ^{*3} | | | 4.9152 ^{*3} | | | 5 ^{*3} | | |
| | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 2 | 64 | 0.70 | 2 | 70 | 0.03 | 2 | 86 | 0.31 | 2 | 88 | −0.25 |
| 150 | 1 | 191 | 0.00 | 1 | 207 | 0.16 | 1 | 255 | 0.00 | 2 | 64 | 0.16 |
| 300 | 1 | 95 | 0.00 | 1 | 103 | 0.16 | 1 | 127 | 0.00 | 1 | 129 | 0.16 |
| 600 | 0 | 191 | 0.00 | 0 | 207 | 0.16 | 0 | 255 | 0.00 | 1 | 64 | 0.16 |
| 1200 | 0 | 95 | 0.00 | 0 | 103 | 0.16 | 0 | 127 | 0.00 | 0 | 129 | 0.16 |
| 2400 | 0 | 47 | 0.00 | 0 | 51 | 0.16 | 0 | 63 | 0.00 | 0 | 64 | 0.16 |
| 4800 | 0 | 23 | 0.00 | 0 | 25 | 0.16 | 0 | 31 | 0.00 | 0 | 32 | −1.36 |
| 9600 | 0 | 11 | 0.00 | 0 | 12 | 0.16 | 0 | 15 | 0.00 | 0 | 15 | 1.73 |
| 19200 | 0 | 5 | 0.00 | — | — | — | 0 | 7 | 0.00 | 0 | 7 | 1.73 |
| 31250 | — | — | — | 0 | 3 | 0.00 | 0 | 4 | −1.70 | 0 | 4 | 0.00 |
| 38400 | 0 | 2 | 0.00 | — | — | — | 0 | 3 | 0.00 | 0 | 3 | 1.73 |

| Bit Rate (bps) ^{*1} | Operating Frequency ϕ (MHz) | | | | | | | | | | | |
|---------------------------------|----------------------------------|-----|-----------|---------------------|-----|-----------|----------------------|-----|-----------|-----------------|-----|-----------|
| | 6 ^{*3} | | | 6.144 ^{*3} | | | 7.3728 ^{*3} | | | 8 ^{*3} | | |
| | N | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 2 | 106 | −0.44 | 2 | 108 | 0.08 | 2 | 130 | −0.07 | 2 | 141 | 0.03 |
| 150 | 2 | 77 | 0.16 | 2 | 79 | 0.00 | 2 | 95 | 0.00 | 2 | 103 | 0.16 |
| 300 | 1 | 155 | 0.16 | 1 | 159 | 0.00 | 1 | 191 | 0.00 | 1 | 207 | 0.16 |
| 600 | 1 | 77 | 0.16 | 1 | 79 | 0.00 | 1 | 95 | 0.00 | 1 | 103 | 0.16 |
| 1200 | 0 | 155 | 0.16 | 0 | 159 | 0.00 | 0 | 191 | 0.00 | 0 | 207 | 0.16 |
| 2400 | 0 | 77 | 0.16 | 0 | 79 | 0.00 | 0 | 95 | 0.00 | 0 | 103 | 0.16 |
| 4800 | 0 | 38 | 0.16 | 0 | 39 | 0.00 | 0 | 47 | 0.00 | 0 | 51 | 0.16 |
| 9600 | 0 | 19 | −2.34 | 0 | 19 | 0.00 | 0 | 23 | 0.00 | 0 | 25 | 0.16 |
| 19200 | 0 | 9 | −2.34 | 0 | 9 | 0.00 | 0 | 11 | 0.00 | 0 | 12 | 0.16 |
| 31250 | 0 | 5 | 0.00 | 0 | 5 | 2.40 | — | — | — | 0 | 7 | 0.00 |
| 38400 | 0 | 4 | −2.34 | 0 | 4 | 0.00 | 0 | 5 | 0.00 | — | — | — |

| Bit Rate (bps) ^{*1} | Operating Frequency ϕ (MHz) | | | | | | | | | | | |
|---------------------------------|----------------------------------|-----|-----------|----|-----|-----------|----|-----|-----------|--------|-----|-----------|
| | 9.8304 ^{*3} | | | 10 | | | 12 | | | 12.288 | | |
| | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 2 | 174 | −0.26 | 2 | 177 | −0.25 | 2 | 212 | 0.03 | 2 | 217 | 0.08 |
| 150 | 2 | 127 | 0.00 | 2 | 129 | 0.16 | 2 | 155 | 0.16 | 2 | 159 | 0.00 |
| 300 | 1 | 255 | 0.00 | 2 | 64 | 0.16 | 2 | 77 | 0.16 | 2 | 79 | 0.00 |
| 600 | 1 | 127 | 0.00 | 1 | 129 | 0.16 | 1 | 155 | 0.16 | 1 | 159 | 0.00 |
| 1200 | 0 | 255 | 0.00 | 1 | 64 | 0.16 | 1 | 77 | 0.16 | 1 | 79 | 0.00 |
| 2400 | 0 | 127 | 0.00 | 0 | 129 | 0.16 | 0 | 155 | 0.16 | 0 | 159 | 0.00 |
| 4800 | 0 | 63 | 0.00 | 0 | 64 | 0.16 | 0 | 77 | 0.16 | 0 | 79 | 0.00 |
| 9600 | 0 | 31 | 0.00 | 0 | 32 | −1.36 | 0 | 38 | 0.16 | 0 | 39 | 0.00 |
| 19200 | 0 | 15 | 0.00 | 0 | 15 | 1.73 | 0 | 19 | −2.34 | 0 | 19 | 0.00 |
| 31250 | 0 | 9 | −1.70 | 0 | 9 | 0.00 | 0 | 11 | 0.00 | 0 | 11 | 2.40 |
| 38400 | 0 | 7 | 0.00 | 0 | 7 | 1.73 | 0 | 9 | −2.34 | 0 | 9 | 0.00 |

| Bit Rate (bps) ^{*1} | Operating Frequency ϕ (MHz) | | | | | | | | | | | |
|---------------------------------|----------------------------------|-----|-----------|-----------------------|-----|-----------|------------------|-----|-----------|-----------------------|-----|-----------|
| | 14 ^{*2} | | | 14.7456 ^{*2} | | | 16 ^{*2} | | | 17.2032 ^{*2} | | |
| | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 2 | 248 | −0.17 | 3 | 64 | 0.70 | 3 | 70 | 0.03 | 3 | 75 | 0.48 |
| 150 | 2 | 181 | 0.16 | 2 | 191 | 0.00 | 2 | 207 | 0.16 | 2 | 223 | 0.00 |
| 300 | 2 | 90 | 0.16 | 2 | 95 | 0.00 | 2 | 103 | 0.16 | 2 | 111 | 0.00 |
| 600 | 1 | 181 | 0.16 | 1 | 191 | 0.00 | 1 | 207 | 0.16 | 1 | 223 | 0.00 |
| 1200 | 1 | 90 | 0.16 | 1 | 95 | 0.00 | 1 | 103 | 0.16 | 1 | 111 | 0.00 |
| 2400 | 0 | 181 | 0.16 | 0 | 191 | 0.00 | 0 | 207 | 0.16 | 0 | 223 | 0.00 |
| 4800 | 0 | 90 | 0.16 | 0 | 95 | 0.00 | 0 | 103 | 0.16 | 0 | 111 | 0.00 |
| 9600 | 0 | 45 | −0.93 | 0 | 47 | 0.00 | 0 | 51 | 0.16 | 0 | 55 | 0.00 |
| 19200 | 0 | 22 | −0.93 | 0 | 23 | 0.00 | 0 | 25 | 0.16 | 0 | 27 | 0.00 |
| 31250 | 0 | 13 | 0.00 | 0 | 14 | −1.70 | 0 | 15 | 0.00 | 0 | 16 | 1.20 |
| 38400 | — | — | — | 0 | 11 | 0.00 | 0 | 12 | 0.16 | 0 | 13 | 0.00 |

| Operating Frequency ϕ (MHz) | | | | | | | | | |
|----------------------------------|------------------|-----|-----------|-----------------------|-----|-----------|------------------|-----|-----------|
| Bit Rate (bps) ^{*1} | 18 ^{*2} | | | 19.6608 ^{*2} | | | 20 ^{*2} | | |
| | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 3 | 79 | -0.12 | 3 | 86 | 0.31 | 3 | 88 | -0.25 |
| 150 | 2 | 233 | 0.16 | 2 | 255 | 0.00 | 3 | 64 | 0.16 |
| 300 | 2 | 116 | 0.16 | 2 | 127 | 0.00 | 2 | 129 | 0.16 |
| 600 | 1 | 233 | 0.16 | 1 | 255 | 0.00 | 2 | 64 | 0.16 |
| 1200 | 1 | 116 | 0.16 | 1 | 127 | 0.00 | 1 | 129 | 0.16 |
| 2400 | 0 | 233 | 0.16 | 0 | 255 | 0.00 | 1 | 64 | 0.16 |
| 4800 | 0 | 116 | 0.16 | 0 | 127 | 0.00 | 0 | 129 | 0.16 |
| 9600 | 0 | 58 | -0.69 | 0 | 63 | 0.00 | 0 | 64 | 0.16 |
| 19200 | 0 | 28 | 1.02 | 0 | 31 | 0.00 | 0 | 32 | -1.36 |
| 31250 | 0 | 17 | 0.00 | 0 | 19 | -1.70 | 0 | 19 | 0.00 |
| 38400 | 0 | 14 | -2.34 | 0 | 15 | 0.00 | 0 | 15 | 1.73 |

Notes: 1. Example when the SEMR0 register ABCS bit is 0. The bit rate is doubled when ABCS is set to 1.

2. Supported only by the H8S/2239 Group.

3. The H8S/2258 Group is out of operation.

Table 15.4 Maximum Bit Rate for Each Frequency (Asynchronous Mode)

| ϕ (MHz) | Maximum Bit Rate (kbps) | n | N | ϕ (MHz) | Maximum Bit Rate (kbps) | n | N |
|------------------------|-------------------------|---|---|-----------------------|-------------------------|---|---|
| 2 ^{*2} | 62.5 | 0 | 0 | 9.8304 ^{*2} | 307.2 | 0 | 0 |
| 2.097152 ^{*2} | 65.536 | 0 | 0 | 10 | 312.5 | 0 | 0 |
| 2.4576 ^{*2} | 76.8 | 0 | 0 | 12 | 375.0 | 0 | 0 |
| 3 ^{*2} | 93.75 | 0 | 0 | 12.288 | 384.0 | 0 | 0 |
| 3.6864 ^{*2} | 115.2 | 0 | 0 | 14 ^{*1} | 437.5 | 0 | 0 |
| 4 ^{*2} | 125.0 | 0 | 0 | 14.7456 ^{*1} | 460.8 | 0 | 0 |
| 4.9152 ^{*2} | 153.6 | 0 | 0 | 16 ^{*1} | 500.0 | 0 | 0 |
| 5 ^{*2} | 156.25 | 0 | 0 | 17.2032 ^{*1} | 537.6 | 0 | 0 |
| 6 ^{*2} | 187.5 | 0 | 0 | 18 ^{*1} | 562.5 | 0 | 0 |
| 6.144 ^{*2} | 192.0 | 0 | 0 | 19.6608 ^{*1} | 614.4 | 0 | 0 |
| 7.3728 ^{*2} | 230.4 | 0 | 0 | 20 ^{*1} | 625.0 | 0 | 0 |
| 8 ^{*2} | 250.0 | 0 | 0 | | | | |

Notes: 1. Supported only by the H8S/2239 Group.
 2. The H8S/2258 Group is out of operation.

Table 15.5 Maximum Bit Rate with External Clock Input (Asynchronous Mode)

| ϕ (MHz) | External Input Clock (MHz) | Maximum Bit Rate (kbps) | ϕ (MHz) | External Input Clock (MHz) | Maximum Bit Rate (kbps) |
|------------------------|----------------------------|-------------------------|-----------------------|----------------------------|-------------------------|
| 2 ^{*2} | 0.5000 | 31.25 | 9.8304 ^{*2} | 2.4576 | 153.6 |
| 2.097152 ^{*2} | 0.5243 | 32.768 | 10 | 2.5000 | 156.25 |
| 2.4576 ^{*2} | 0.6144 | 38.4 | 12 | 3.0000 | 187.5 |
| 3 ^{*2} | 0.7500 | 46.875 | 12.288 | 3.0720 | 192.0 |
| 3.6864 ^{*2} | 0.9216 | 57.6 | 14 ^{*1} | 3.5000 | 218.75 |
| 4 ^{*2} | 1.0000 | 62.5 | 14.7456 ^{*1} | 3.6864 | 230.4 |
| 4.9152 ^{*2} | 1.2288 | 76.8 | 16 ^{*1} | 4.0000 | 250.0 |
| 5 ^{*2} | 1.2500 | 78.125 | 17.2032 ^{*1} | 4.3008 | 268.8 |
| 6 ^{*2} | 1.5000 | 93.75 | 18 ^{*1} | 4.5000 | 281.3 |
| 6.144 ^{*2} | 1.5360 | 96.0 | 19.6608 ^{*1} | 4.9152 | 307.2 |
| 7.3728 ^{*2} | 1.8432 | 115.2 | 20 ^{*1} | 5.0000 | 312.5 |
| 8 ^{*2} | 2.0000 | 125.0 | | | |

Notes: 1. Supported only by the H8S/2239 Group.
 2. The H8S/2258 Group is out of operation.

Table 15.6 BRR Settings for Various Bit Rates (Clocked Synchronous Mode)

| Bit Rate (bps) | Operating Frequency ϕ (MHz) | | | | | | | |
|-------------------|----------------------------------|-----|----------|-----|----------|-----|----------|-----|
| | 2^{*2} | | 4^{*2} | | 6^{*2} | | 8^{*2} | |
| | n | N | n | N | n | N | n | N |
| 110 | 3 | 70 | — | — | | | | |
| 250 | 2 | 124 | 2 | 249 | | | 3 | 124 |
| 500 | 1 | 249 | 2 | 124 | | | 2 | 249 |
| 1 k | 1 | 124 | 1 | 249 | | | 2 | 124 |
| 2.5 k | 0 | 199 | 1 | 99 | 1 | 149 | 1 | 199 |
| 5 k | 0 | 99 | 0 | 199 | 1 | 74 | 1 | 99 |
| 10 k | 0 | 49 | 0 | 99 | 0 | 149 | 0 | 199 |
| 25 k | 0 | 19 | 0 | 39 | 0 | 59 | 0 | 79 |
| 50 k | 0 | 9 | 0 | 19 | 0 | 29 | 0 | 39 |
| 100 k | 0 | 4 | 0 | 9 | 0 | 14 | 0 | 19 |
| 250 k | 0 | 1 | 0 | 3 | 0 | 5 | 0 | 7 |
| 500 k | 0 | 0* | 0 | 1 | 0 | 2 | 0 | 3 |
| 1 M | | | 0 | 0* | | | 0 | 1 |
| 2.5 M | | | | | | | | |
| 5 M | | | | | | | | |

| Bit Rate (bps) | Operating Frequency ϕ (MHz) | | | | | |
|-------------------|----------------------------------|-----|------------------|-----|------------------|-----|
| | 10 | | 16 ^{*1} | | 20 ^{*1} | |
| | n | N | n | N | n | N |
| 110 | | | | | | |
| 250 | — | — | 3 | 249 | | |
| 500 | — | — | 3 | 124 | — | — |
| 1 k | — | — | 2 | 249 | — | — |
| 2.5 k | 1 | 249 | 2 | 99 | 2 | 124 |
| 5 k | 1 | 124 | 1 | 199 | 1 | 249 |
| 10 k | 0 | 249 | 1 | 99 | 1 | 124 |
| 25 k | 0 | 99 | 0 | 159 | 0 | 199 |
| 50 k | 0 | 49 | 0 | 79 | 0 | 99 |
| 100 k | 0 | 24 | 0 | 39 | 0 | 49 |
| 250 k | 0 | 9 | 0 | 15 | 0 | 19 |
| 500 k | 0 | 4 | 0 | 7 | 0 | 9 |
| 1 M | | | 0 | 3 | 0 | 4 |
| 2.5 M | 0 | 0* | | | 0 | 1 |
| 5 M | | | | | 0 | 0* |

Legend:

Blank: Cannot be set.

—: Can be set, but there will be a degree of error.

*: Continuous transfer is not possible.

Notes: 1. Supported only by the H8S/2239 Group.

2. The H8S/2258 Group is out of operation.

Table 15.7 Maximum Bit Rate with External Clock Input (Clocked Synchronous Mode)

| ϕ (MHz) | External Input Clock (MHz) | Maximum Bit Rate (bps) | ϕ (MHz) | External Input Clock (MHz) | Maximum Bit Rate (bps) |
|-----------------|----------------------------|------------------------|------------------|----------------------------|------------------------|
| 2 ^{*2} | 0.3333 | 0.333 | 12 | 2.0000 | 2.000 |
| 4 ^{*2} | 0.6667 | 0.667 | 14 ^{*1} | 2.3333 | 2.333 |
| 6 ^{*2} | 1.0000 | 1.000 | 16 ^{*1} | 2.6667 | 3.667 |
| 8 ^{*2} | 1.3333 | 1.333 | 18 ^{*1} | 3.0000 | 3.000 |
| 10 | 1.6667 | 1.667 | 20 ^{*1} | 3.3333 | 3.333 |

Notes 1. Supported only by the H8S/2239 Group.

2. The H8S/2258 Group is out of operation.

Table 15.8 Examples of Bit Rate for Various BRR Settings (Smart Card Interface Mode)
(When $n = 0$ and $S = 372$)

| Bit Rate (bps) | Operating Frequency ϕ (MHz) | | | | | | | | | |
|-------------------|----------------------------------|-----------|--------------------|-----------|----------------------|-----------|-------|-----------|---------|-----------|
| | 5.00 ^{*2} | | 7.00 ^{*2} | | 7.1424 ^{*2} | | 10.00 | | 10.7136 | |
| | N | Error (%) | N | Error (%) | N | Error (%) | N | Error (%) | N | Error (%) |
| 6720 | 0 | 0.01 | 1 | 30.00 | 1 | 28.57 | 1 | 0.01 | 1 | 7.14 |
| 9600 | 0 | 30.00 | 0 | 1.99 | 0 | 0.00 | 1 | 30.00 | 1 | 25.00 |

| Bit Rate (bps) | Operating Frequency ϕ (MHz) | | | | | | | | | |
|-------------------|----------------------------------|-----------|-----------------------|-----------|---------------------|-----------|---------------------|-----------|---------------------|-----------|
| | 13.00 | | 14.2848 ^{*1} | | 16.00 ^{*1} | | 18.00 ^{*1} | | 20.00 ^{*1} | |
| | N | Error (%) | N | Error (%) | N | Error (%) | N | Error (%) | N | Error (%) |
| 6720 | 2 | 13.33 | 2 | 4.76 | 2 | 6.67 | 3 | 9.99 | 3 | 0.01 |
| 9600 | 1 | 8.99 | 1 | 0.00 | 1 | 12.01 | 2 | 15.99 | 2 | 6.66 |

Notes: 1. Supported only by the H8S/2239 Group.

2. The H8S/2258 Group is out of operation.

Table 15.9 Maximum Bit Rate at Various Frequencies (Smart Card Interface Mode)
(When $S = 372$)

| ϕ (MHz) | Maximum Bit Rate (bps) | n | N |
|-----------------------|------------------------|---|---|
| 5.00 ^{*2} | 6720 | 0 | 0 |
| 7.00 ^{*2} | 9409 | 0 | 0 |
| 7.1424 ^{*2} | 9600 | 0 | 0 |
| 10.00 | 13441 | 0 | 0 |
| 10.7136 | 14400 | 0 | 0 |
| 13.00 | 17473 | 0 | 0 |
| 14.2848 ^{*1} | 19200 | 0 | 0 |
| 16.00 ^{*1} | 21505 | 0 | 0 |
| 18.00 ^{*1} | 24194 | 0 | 0 |
| 20.00 ^{*1} | 26882 | 0 | 0 |

Notes: 1. Supported only by the H8S/2239 Group.

2. The H8S/2258 Group is out of operation.

15.3.10 Serial Expansion Mode Register (SEMR_0)

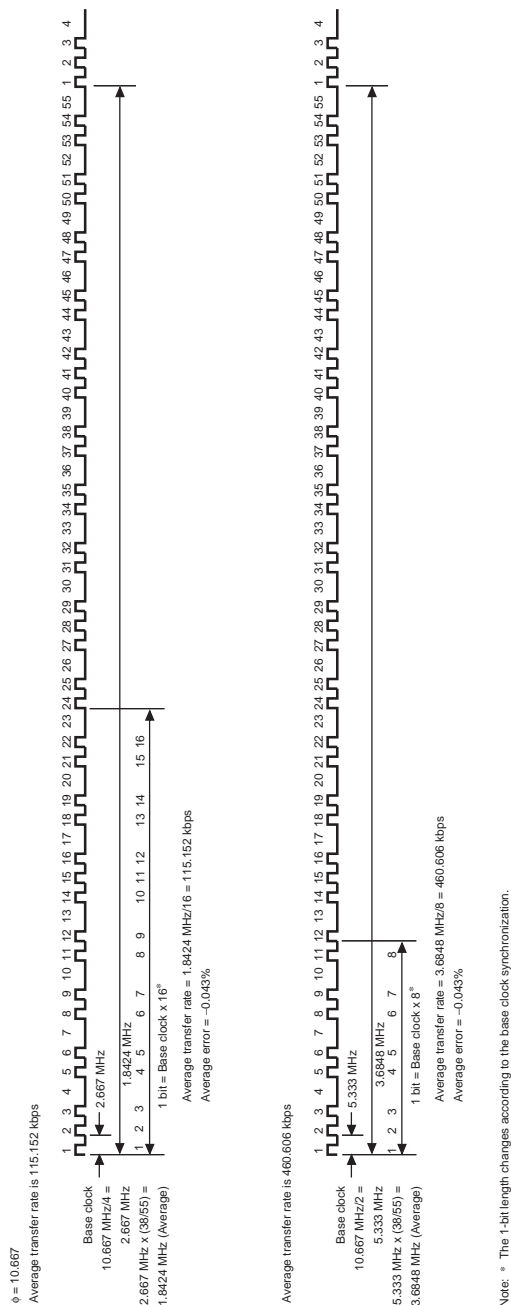
SEMR_0 is an 8-bit register that expands SCI_0 functions; such as setting of the base clock, selecting of the clock source, and automatic setting of the transfer rate.

Note: Supported only by the H8S/2239 Group only.

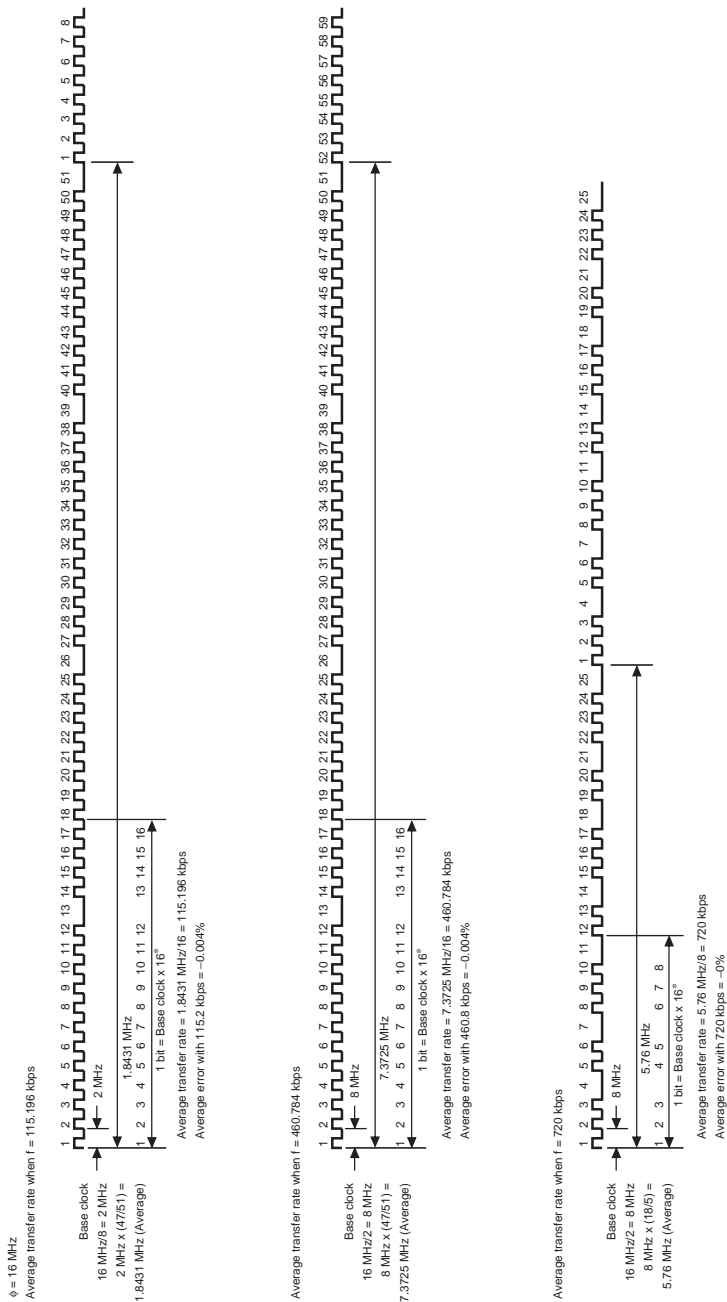
| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | SSE | 0 | R/W | <p>SCI_0 Select Enable</p> <p>This bit enables or disables the SCI_0 select function when an external clock is input in clocked synchronous mode. When 1 is set to the PG1/IRQ7 pin, while the SCI_0 select function is enabled, the TxD0 output becomes Hi-Z and the SCK0 input in this LSI is fixed high making the SCI_0 data transfer terminated. The SSE setting is valid when the external clock input is selected (CKE in SCR = 0) in clocked synchronous mode (C/\bar{A} in SMR = 1).</p> <p>0: SCI_0 select is disabled.</p> <p>1: SCI_0 select is enabled.</p> <p>When then PG1/$\overline{IRQ7}$ pin = 1, the TxD0 output becomes Hi-Z and the SCK0 clock input is fixed high.</p> |
| 6 to 4 | — | Undefined | — | <p>Reserved</p> <p>These bits are always read as 0, and cannot be modified.</p> |
| 3 | ABCS | 0 | R/W | <p>Asynchronous Base Clock Select</p> <p>Selects the 1-bit-interval base clock in asynchronous mode.</p> <p>The ABCS setting is valid in asynchronous mode (C/\bar{A} in SMR = 0).</p> <p>0: Operates on a base clock with a frequency of 16 times the transfer rate.</p> <p>1: Operates on a base clock with a frequency of 8 times the transfer rate.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | ACS2 | 0 | R/W | Asynchronous Clock Source Select |
| 1 | ACS1 | 0 | R/W | When an average transfer rate is selected, the base clock is set automatically regardless of the ABCS value. Note that average transfer rates are not supported for operating frequencies other than 10.667 MHz and 16 MHz. |
| 0 | ACS0 | 0 | R/W | <p>The ACS0 to ACS0 settings are valid when the external clock input is selected (CKE in SCR = 0) in asynchronous mode (C/\bar{A} in SMR = 0).</p> <p>000: External clock input</p> <p>001: Selects the average transfer rate 115.152 kbps only for ϕ = 10.667 MHz (operates on a base clock with a frequency of 16 times the transfer rate).</p> <p>010: Selects the average transfer rate 460.606 kbps only for ϕ = 10.667 MHz (operates on a base clock with a frequency of 8 times the transfer rate).</p> <p>011: Reserved</p> <p>100: TPU clock input (logical AND of TIOCA1 and TIOCA2)</p> <p>101: Selects the average transfer rate 115.196 kbps only for ϕ = 16 MHz (operates on a base clock with a frequency of 16 times the transfer rate).</p> <p>110: Selects the average transfer rate 460.784 kbps only for ϕ = 16 MHz (operates on a base clock with a frequency of 16 times the transfer rate).</p> <p>111: Selects the average transfer rate 720 kbps only for ϕ = 16 MHz (operates on a base clock with a frequency of 8 times the transfer rate).</p> |

Figures 15.3 and 15.4 show an example of the internal base clock when the average transfer rate is selected.



**Figure 15.3 Example of the Internal Base Clock
When the Average Transfer Rate Is Selected (1)**



Note: * The 1-bit length changes according to the base clock synchronization.

**Figure 15.4 Example of the Internal Base Clock
When the Average Transfer Rate Is Selected (2)**

15.4 Operation in Asynchronous Mode

Figure 15.5 shows the general format for asynchronous serial communication. One frame consists of a start bit (low level), followed by data, a parity bit, and finally stop bits (high level). In asynchronous serial communication, the transmission line is usually held in the mark state (high level). The SCI monitors the transmission line, and when it goes to the space state (low level), recognizes a start bit and starts serial communication. Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communication. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transfer. In asynchronous mode, the SCI performs synchronization at the falling edge of the start bit in reception. The SCI samples the data on the 8th pulse of a clock with a frequency of 16 times the length of one bit, so that the transfer data is latched at the center of each bit.

The SCI_0 samples the data on the 4th pulse of a clock with a frequency of 8 times the length of one bit when the ABCS bit in SEMR_0 is 1 (H8S/2239 Group only).

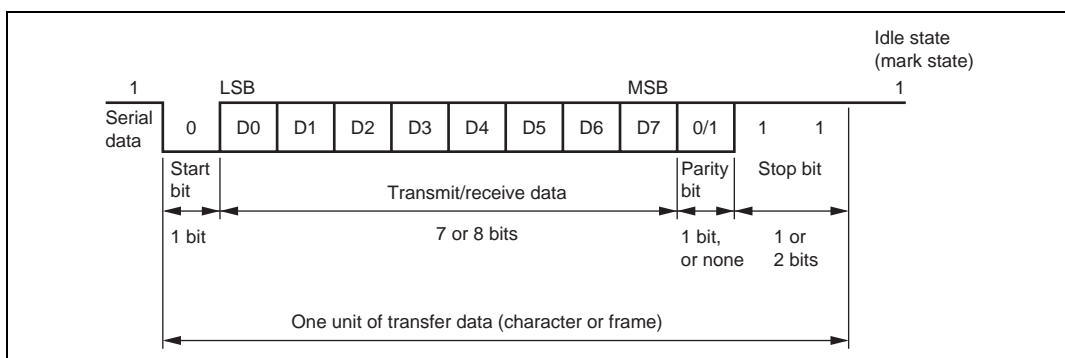


Figure 15.5 Data Format in Asynchronous Communication
(Example with 8-Bit Data, Parity, Two Stop Bits)

15.4.1 Data Transfer Format

Table 15.10 shows the data transfer formats that can be used in asynchronous mode. Any of 12 transfer formats can be selected according to the SMR setting. For details on the multiprocessor bit, refer to section 15.5, Multiprocessor Communication Function.

Table 15.10 Serial Transfer Formats (Asynchronous Mode)

| SMR Settings | | | | Serial Transfer Format and Frame Length | | | | | | | | | | | |
|--------------|----|----|------|---|------------|---|---|---|---|---|---|------|------|------|------|
| CHR | PE | MP | STOP | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
| 0 | 0 | 0 | 0 | S | 8-bit data | | | | | | | | STOP | | |
| 0 | 0 | 0 | 1 | S | 8-bit data | | | | | | | | STOP | STOP | |
| 0 | 1 | 0 | 0 | S | 8-bit data | | | | | | | | P | STOP | |
| 0 | 1 | 0 | 1 | S | 8-bit data | | | | | | | | P | STOP | STOP |
| 1 | 0 | 0 | 0 | S | 7-bit data | | | | | | | STOP | | | |
| 1 | 0 | 0 | 1 | S | 7-bit data | | | | | | | STOP | STOP | | |
| 1 | 1 | 0 | 0 | S | 7-bit data | | | | | | | P | STOP | | |
| 1 | 1 | 0 | 1 | S | 7-bit data | | | | | | | P | STOP | STOP | |
| 0 | — | 1 | 0 | S | 8-bit data | | | | | | | | MPB | STOP | |
| 0 | — | 1 | 1 | S | 8-bit data | | | | | | | | MPB | STOP | STOP |
| 1 | — | 1 | 0 | S | 7-bit data | | | | | | | MPB | STOP | | |
| 1 | — | 1 | 1 | S | 7-bit data | | | | | | | MPB | STOP | STOP | |

Legend:

S: Start bit

STOP: Stop bit

P: Parity bit

MPB: Multiprocessor bit

15.4.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times the transfer rate. In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. Receive data is latched internally at the rising edge of the 8th pulse of the base clock as shown in figure 15.6. Thus, the reception margin in asynchronous mode is given by formula (1) below.

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100 [\%] \quad \dots \text{Formula (1)}$$

Where M: Reception margin (%)

N: Bit rate ratio relative to clock (N = 16, but in the H8S/2239 Group N = 8 if ABCS in SEMR_0 is set to 1.)

D: Clock duty (D = 0 to 1.0)

L: Frame length (L = 9 to 12)

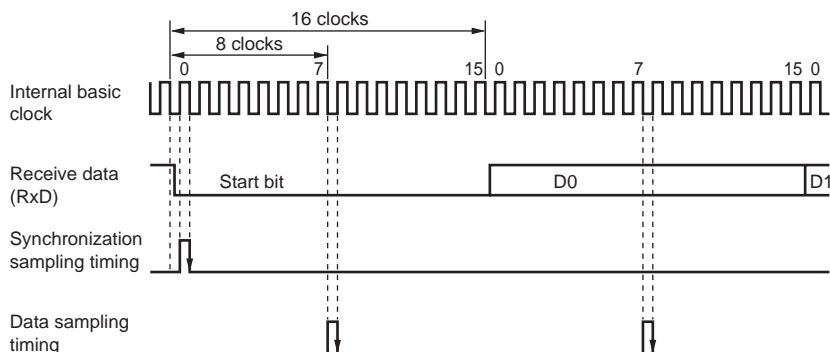
F: Clock frequency deviation absolute value

Assuming values of F (absolute value of clock rate deviation) = 0, D (clock duty) = 0.5, and N (ratio of bit rate to clock) = 16 in formula (1), the reception margin can be given by the formula.

$$M = \{ 0.5 - 1/(2 \times 16) \} \times 100 [\%] = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed for in system design.

Note: Example for H8S/2239 Group with the ABCS bit in SEMR_0 set to a value other than 1. When ABCS is set to 1, the clock frequency is 8 times the bit rate and sampling of received data takes place at the fourth rising edge of the basic clock.



Note: Example for H8S/2239 Group with the ABCS bit in SEMR_0 set to a value other than 1.
When ABCS is set to 1, the clock frequency is 8 times the bit rate and sampling of received data takes place at the fourth rising edge of the basic clock.

Figure 15.6 Receive Data Sampling Timing in Asynchronous Mode

15.4.3 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 SCI's serial clock, according to the setting of the C/\overline{A} bit in SMR and the CKE0 and CKE1 bits in SCR. When an external clock is input at the SCK pin, the clock frequency should be 16 times the bit rate used.

When the SCI is operated on an internal clock, the clock can be output from the SCK pin when setting CKE1 = 0 and CKE0 = 1. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in figure 15.7.

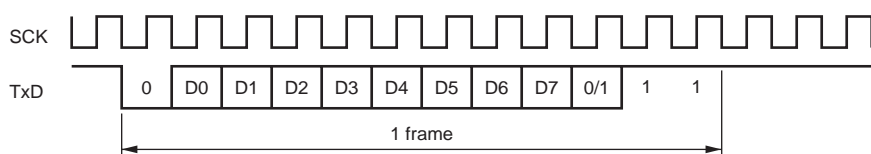


Figure 15.7 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode)

15.4.4 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, you should first clear the TE and RE bits in SCR to 0, then initialize the SCI as described in figure 15.8. When the operating mode, or transfer format, is changed for example, 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, the TDRE flag is set to 1. Note that clearing the RE bit to 0 does not initialize the contents of the RDRF, PER, FER, and ORER flags, or the contents of RDR. When the external clock is used in asynchronous mode, the clock must be supplied even during initialization.

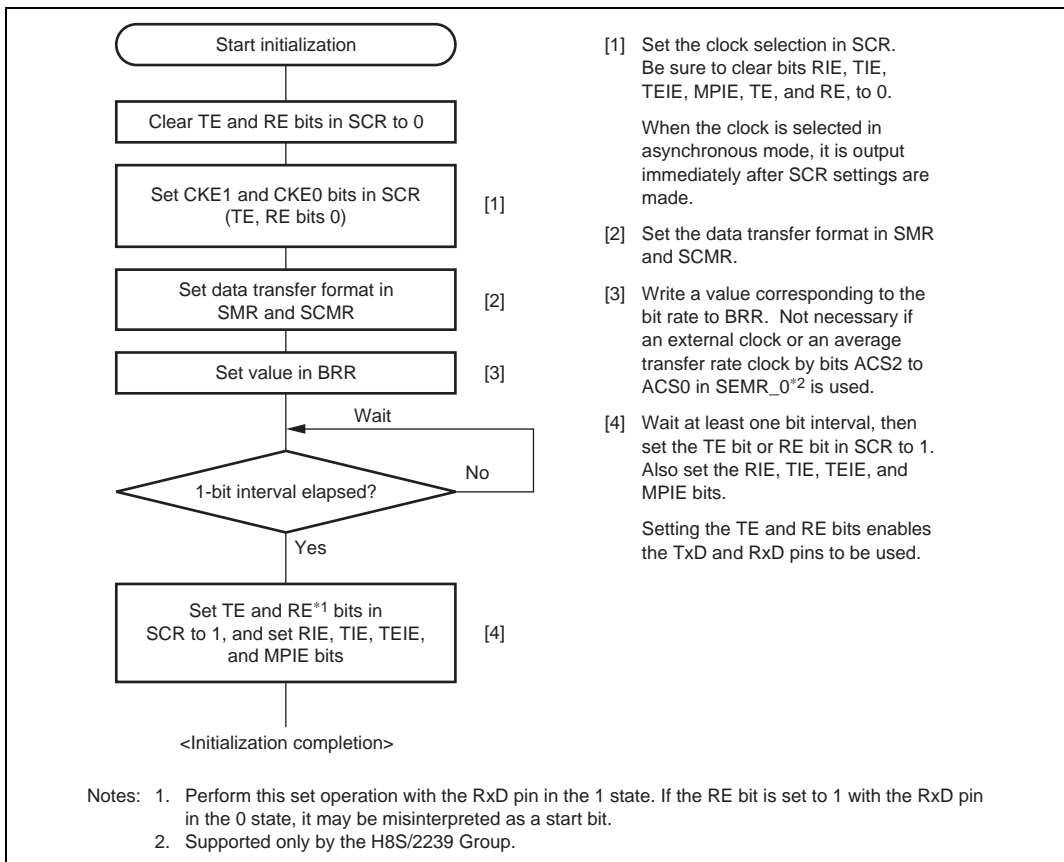
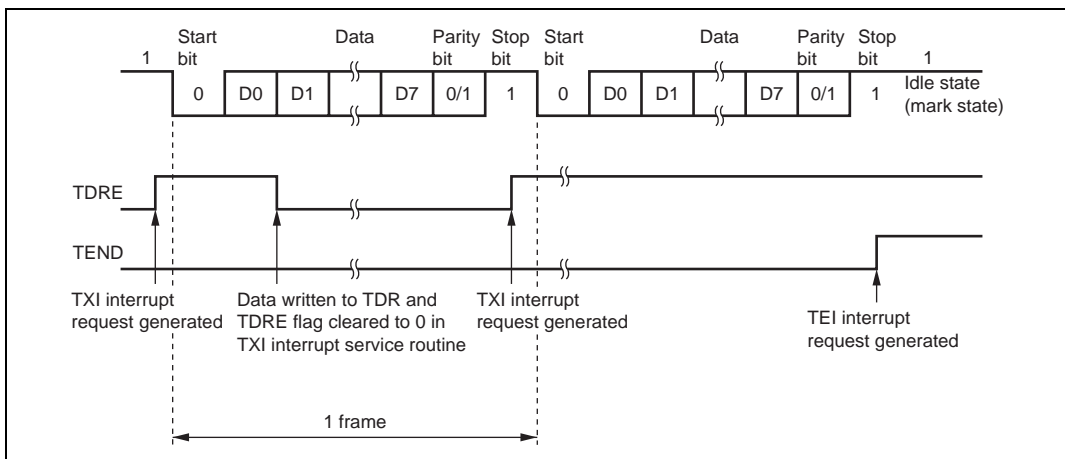


Figure 15.8 Sample SCI Initialization Flowchart

15.4.5 Serial Data Transmission (Asynchronous Mode)

Figure 15.9 shows an example of operation for transmission in asynchronous mode. In transmission, the SCI operates as described below.

1. The SCI monitors the TDRE flag in SSR. If the flag is cleared to 0, the SCI recognizes that data has been written to TDR, and transfers the data from TDR to TSR.
2. After transferring data from TDR to TSR, the SCI sets the TDRE flag to 1 and starts transmission. If the TIE bit is set to 1 at this time, a transmit data empty interrupt request (TXI) is generated. Continuous transmission is possible because the TXI interrupt routine writes next transmit data to TDR before transmission of the current transmit data has been completed.
3. Data is sent from the TxD pin in the following order: start bit, transmit data, parity bit or multiprocessor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks the TDRE flag at the timing for sending the stop bit.
5. If the TDRE flag is 0, the data is transferred from TDR to TSR, the stop bit is sent, and then serial transmission of the next frame is started.
6. If the TDRE flag is 1, the TEND flag in SSR is set to 1, the stop bit is sent, and then the “mark state” is entered, in which 1 is output. If the TEIE bit in SCR is set to 1 at this time, a TEI interrupt request is generated.



**Figure 15.9 Example of Operation in Transmission in Asynchronous Mode
(Example with 8-Bit Data, Parity, One Stop Bit)**

Figure 15.10 shows a sample flowchart for data transmission.

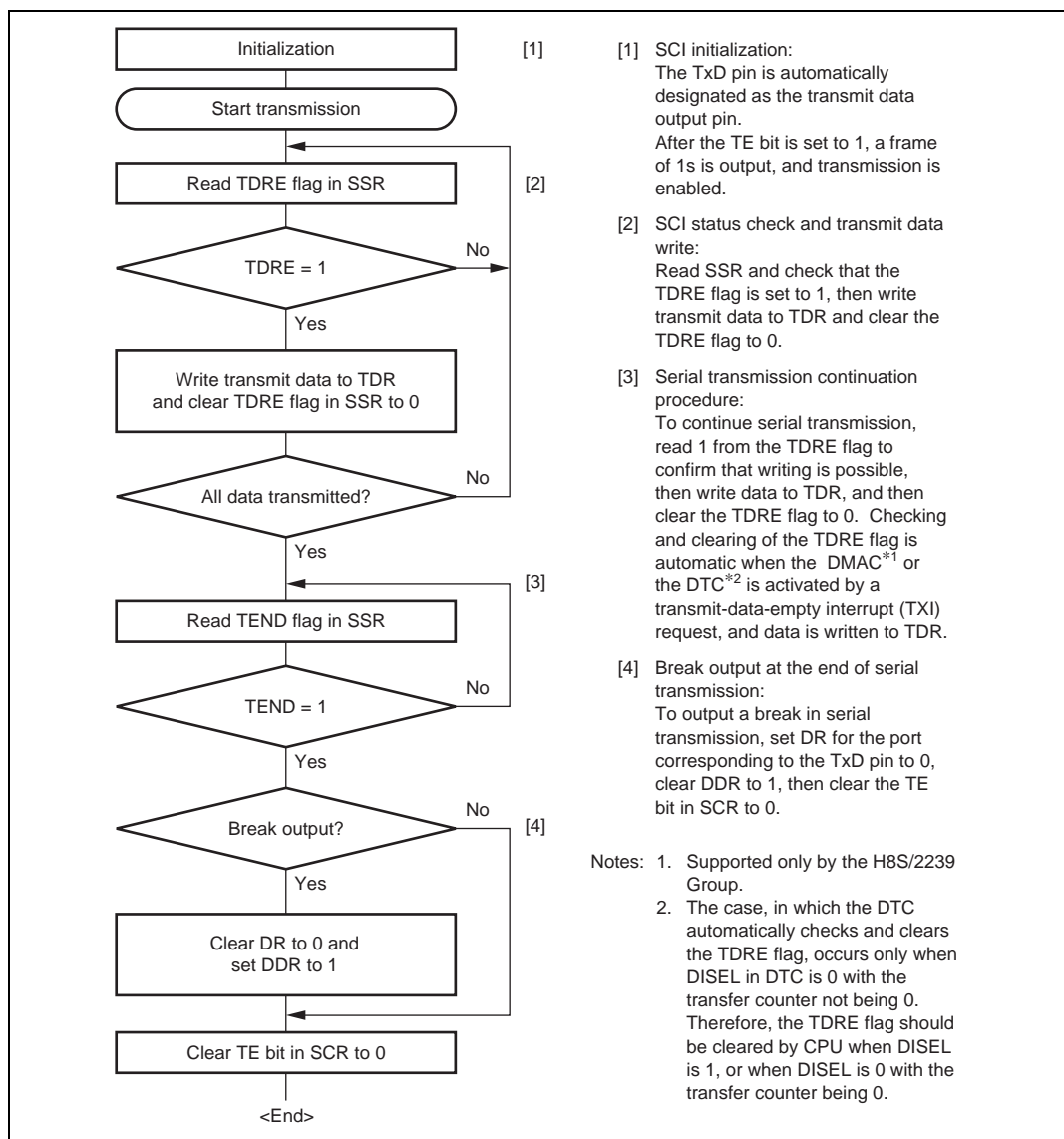
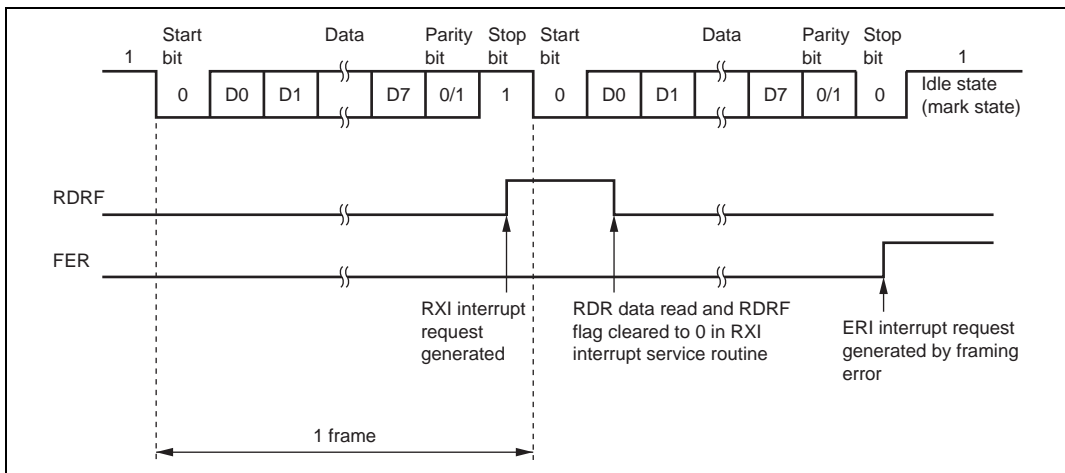


Figure 15.10 Sample Serial Transmission Flowchart

15.4.6 Serial Data Reception (Asynchronous Mode)

Figure 15.11 shows an example of operation for reception in asynchronous mode. In serial reception, the SCI operates as described below.

1. The SCI monitors the communication line. If a start bit is detected, the SCI performs internal synchronization, receives receive data in RSR, and checks the parity bit and stop bit.
2. If an overrun error occurs (when reception of the next data is completed while the RDRF flag is still set to 1), the ORER bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR. The RDRF flag remains to be set to 1.
3. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated.
4. If a framing error is detected (when the stop bit is 0), the FER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated.
5. If reception is completed successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is possible because the RXI interrupt routine reads the receive data transferred to RDR before reception of the next receive data has been completed.



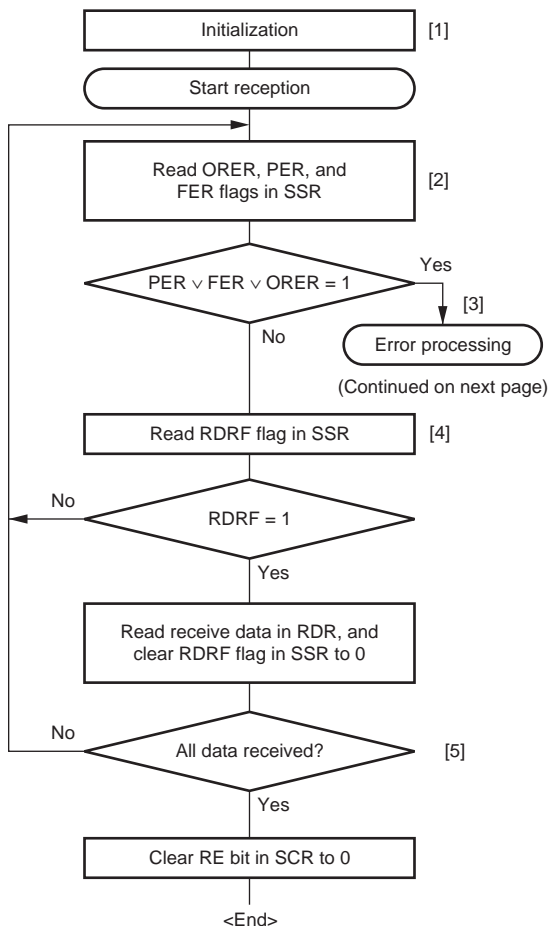
**Figure 15.11 Example of SCI Operation in Reception
(Example with 8-Bit Data, Parity, One Stop Bit)**

Table 15.11 shows the states of the SSR status flags and receive data handling when a receive error is detected. If a receive error is detected, the RDRF flag retains its state before receiving data. Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF bits to 0 before resuming reception. Figure 15.12 shows a sample flow chart for serial data reception.

Table 15.11 SSR Status Flags and Receive Data Handling

| SSR Status Flag | | | | Receive Data | Receive Error Type |
|-----------------|------|-----|-----|--------------------|--|
| RDRF* | ORER | FER | PER | | |
| 1 | 1 | 0 | 0 | Lost | Overrun error |
| 0 | 0 | 1 | 0 | Transferred to RDR | Framing error |
| 0 | 0 | 0 | 1 | Transferred to RDR | Parity error |
| 1 | 1 | 1 | 0 | Lost | Overrun error + framing error |
| 1 | 1 | 0 | 1 | Lost | Overrun error + parity error |
| 0 | 0 | 1 | 1 | Transferred to RDR | Framing error + parity error |
| 1 | 1 | 1 | 1 | Lost | Overrun error + framing error + parity error |

Note: * The RDRF flag retains the state it had before data reception.



- [1] SCI initialization:
The RxD pin is automatically designated as the receive data input pin.
- [2] [3] Receive error processing and break detection:
If a receive error occurs, read the ORER, PER, and FER flags in SSR to identify the error. After performing the appropriate error processing, ensure that the ORER, PER, and FER flags are all cleared to 0. Reception cannot be resumed if any of these flags are set to 1. In the case of a framing error, a break can be detected by reading the value of the input port corresponding to the RxD pin.
- [4] SCI status check and receive data read:
Read SSR and check that RDRF = 1, then read the receive data in RDR and clear the RDRF flag to 0. Transition of the RDRF flag from 0 to 1 can also be identified by an RXI interrupt.
- [5] Serial reception continuation procedure:
To continue serial reception, before the stop bit for the current frame is received, read the RDRF flag, read RDR, and clear the RDRF flag to 0. The RDRF flag is cleared automatically when the DMAC*1 or the DTC*2 is activated by an RXI interrupt and the RDR value is read.

- Notes: 1. Supported only by the H8S/2239 Group.
2. The case, in which the DTC automatically clears the RDRF flag, occurs only when DISEL in DTC is 0 with the transfer counter not being 0. Therefore, the RDRF flag should be cleared by CPU when DISEL is 1, or when DISEL is 0 with the transfer counter being 0.

Figure 15.12 Sample Serial Reception Data Flowchart (1)

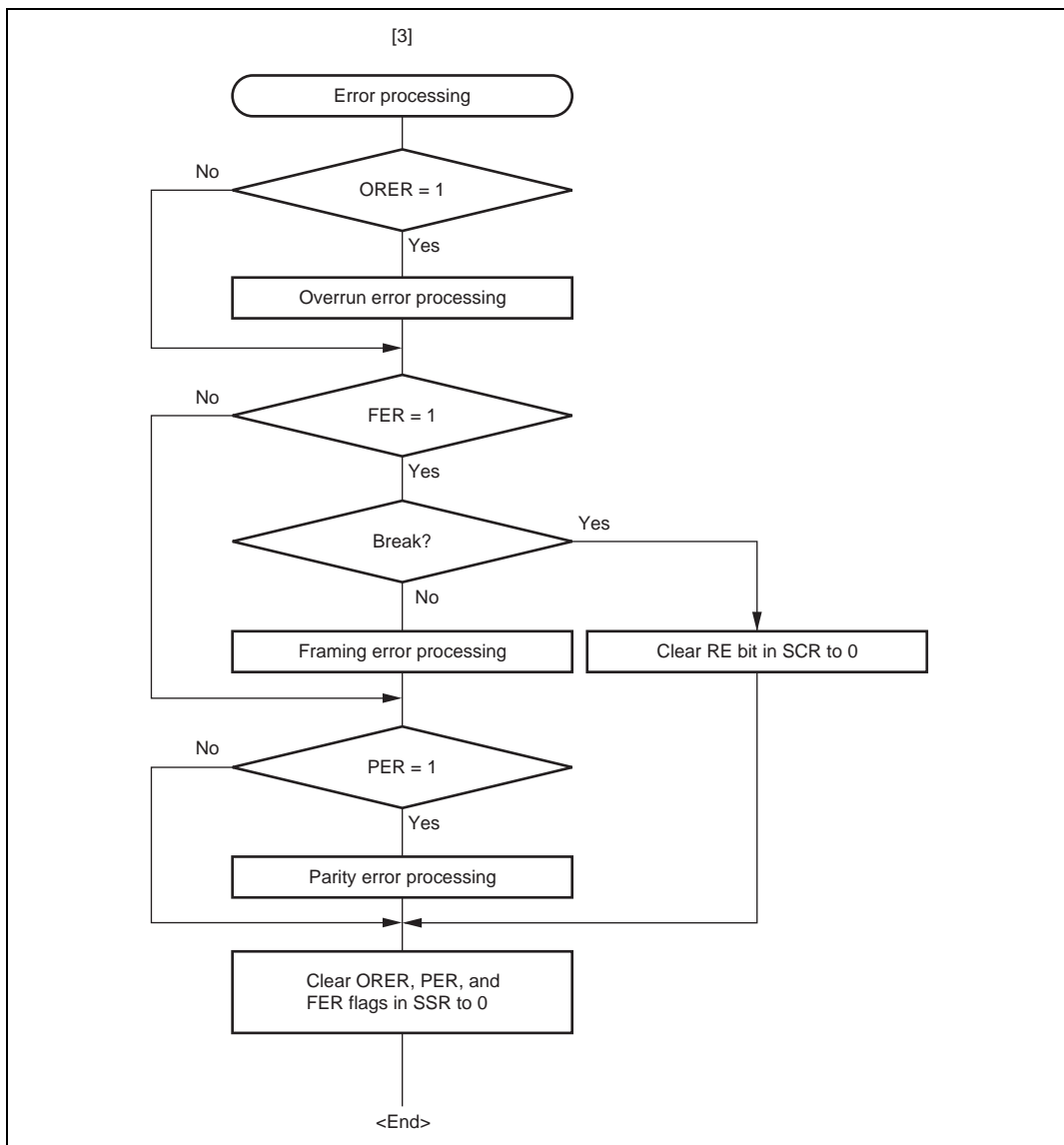


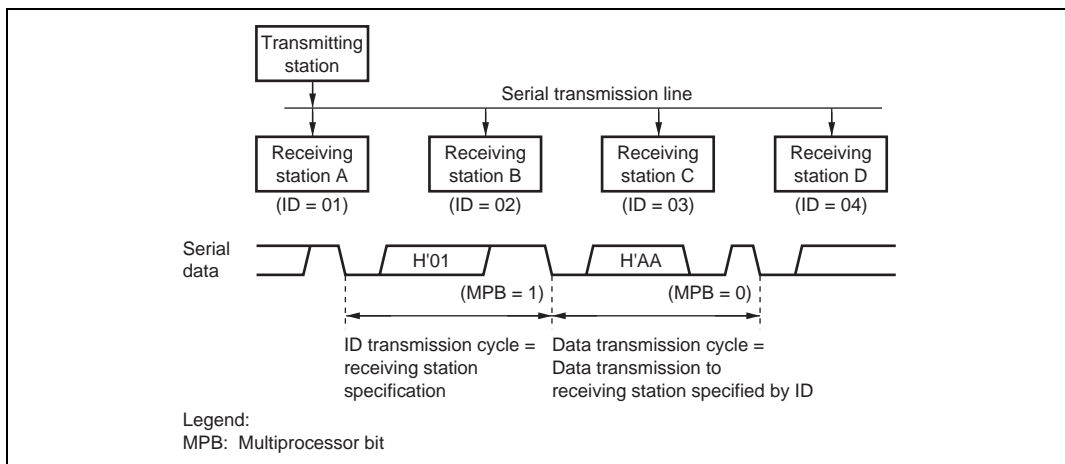
Figure 15.12 Sample Serial Reception Data Flowchart (2)

15.5 Multiprocessor Communication Function

Use of the multiprocessor communication function enables data transfer between a number of processors sharing communication lines by asynchronous serial communication using the multiprocessor format, in which a multiprocessor bit is added to the transfer data. When multiprocessor communication is performed, each receiving station is addressed by a unique ID code. The serial communication cycle consists of two component cycles; an ID transmission cycle that specifies the receiving station, and a data transmission cycle. The multiprocessor bit is used to differentiate between the ID transmission cycle and the data transmission cycle. If the multiprocessor bit is 1, the cycle is an ID transmission cycle; if the multiprocessor bit is 0, the cycle is a data transmission cycle. Figure 15.13 shows an example of inter-processor communication using the multiprocessor format. The transmitting station first sends the ID code of the receiving station with which it wants to perform serial communication as data with a 1 multiprocessor bit added. It then sends transmit data as data with a 0 multiprocessor bit added. When data with a 1 multiprocessor bit is received, the receiving station compares that data with its own ID. The station whose ID matches then receives the data sent next. Stations whose IDs do not match continue to skip data until data with a 1 multiprocessor bit is again received.

The SCI uses the MPIE bit in SCR to implement this function. When the MPIE bit is set to 1, transfer of receive data from RSR to RDR, error flag detection, and setting the SSR status flags, RDRF, FER, and ORER to 1, are inhibited until data with a 1 multiprocessor bit is received. On reception of a receive character with a 1 multiprocessor bit, the MPB bit in SSR is set to 1 and the MPIE bit is automatically cleared, thus normal reception is resumed. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt is generated.

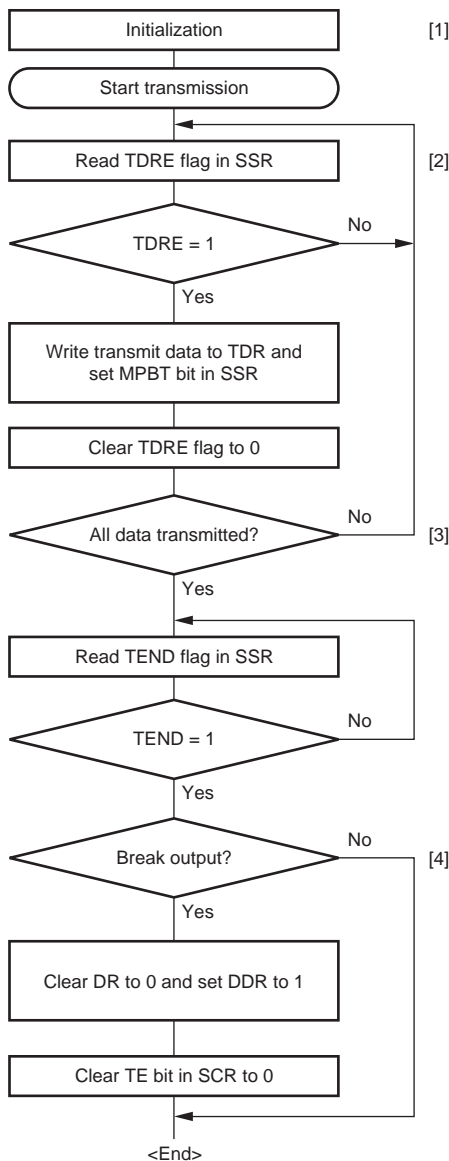
When the multiprocessor format is selected, the parity bit setting is rendered invalid. All other bit settings are the same as those in normal asynchronous mode. The clock used for multiprocessor communication is the same as that in normal asynchronous mode.



**Figure 15.13 Example of Communication Using Multiprocessor Format
(Transmission of Data H'AA to Receiving Station A)**

15.5.1 Multiprocessor Serial Data Transmission

Figure 15.14 shows a sample flowchart for multiprocessor serial data transmission. For an ID transmission cycle, set the MPBT bit in SSR to 1 before transmission. For a data transmission cycle, clear the MPBT bit in SSR to 0 before transmission. All other SCI operations are the same as those in asynchronous mode.



[1] SCI initialization:
The TxD pin is automatically designated as the transmit data output pin.
After the TE bit is set to 1, a frame of 1s is output, and transmission is enabled.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR. Set the MPBT bit in SSR to 0 or 1.
Finally, clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:
To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear the TDRE flag to 0. Checking and clearing of the TDRE flag is automatic when the DMAC*¹ or the DTC*² is activated by a transmit-data-empty interrupt (TXI) request, and data is written to TDR.

[4] Break output at the end of serial transmission:
To output a break in serial transmission, set the port DR to 0, clear DDR to 1, then clear the TE bit in SCR to 0.

- Notes:
1. Supported only by the H8S/2239 Group.
 2. The case, in which the DTC automatically clears the TDRE flag, occurs only when DISEL in DTC is 0 with the transfer counter not being 0. Therefore, the TDRE flag should be cleared by CPU when DISEL is 1, or when DISEL is 0 with the transfer counter being 0.

Figure 15.14 Sample Multiprocessor Serial Transmission Flowchart

15.5.2 Multiprocessor Serial Data Reception

Figure 15.16 shows a sample flowchart for multiprocessor serial data reception. If the MPIE bit in SCR is set to 1, data is skipped until data with a 1 multiprocessor bit is sent. On receiving data with a 1 multiprocessor bit, the receive data is transferred to RDR. An RXI interrupt request is generated at this time. All other SCI operations are the same as in asynchronous mode. Figure 15.15 shows an example of SCI operation for multiprocessor format reception.

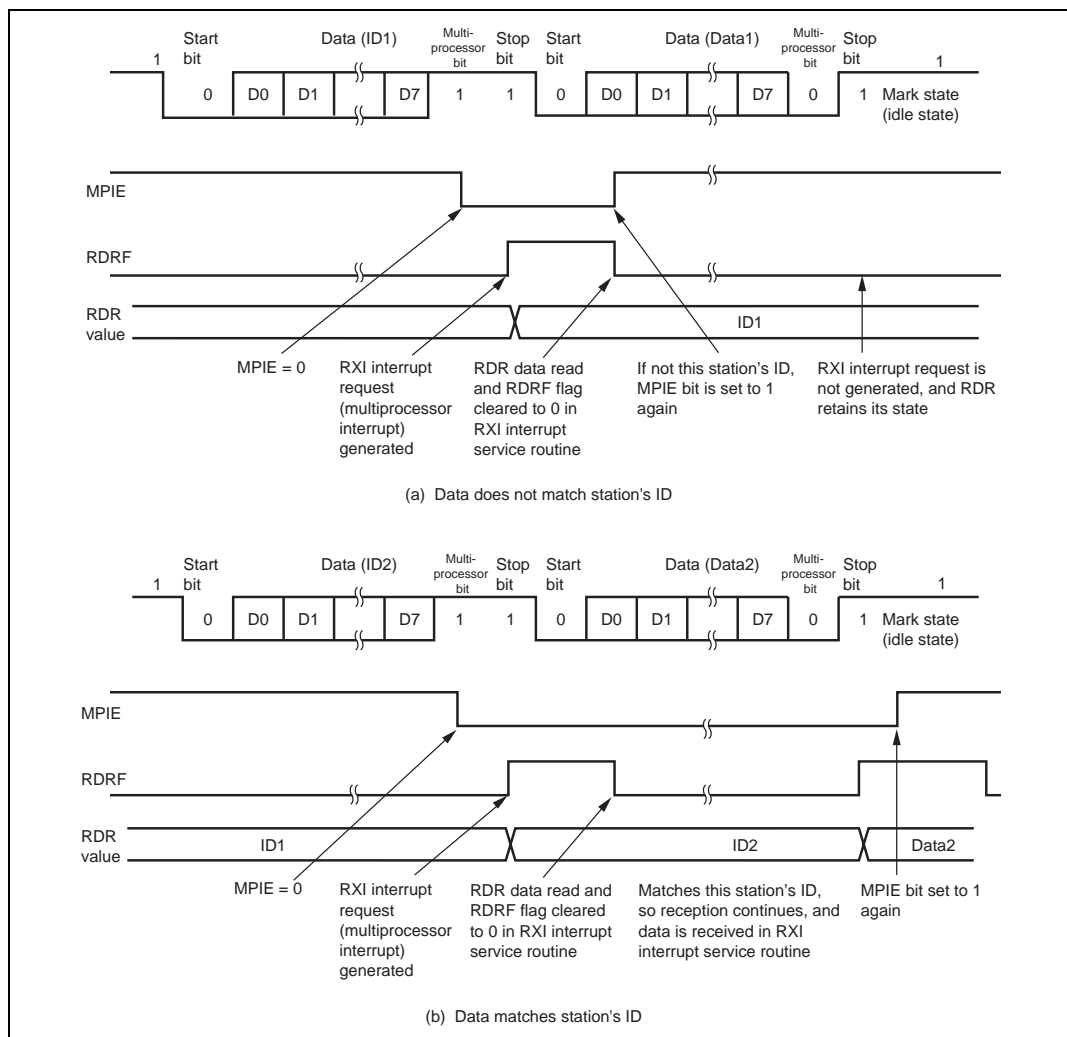


Figure 15.15 Example of SCI Operation in Reception
(Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)

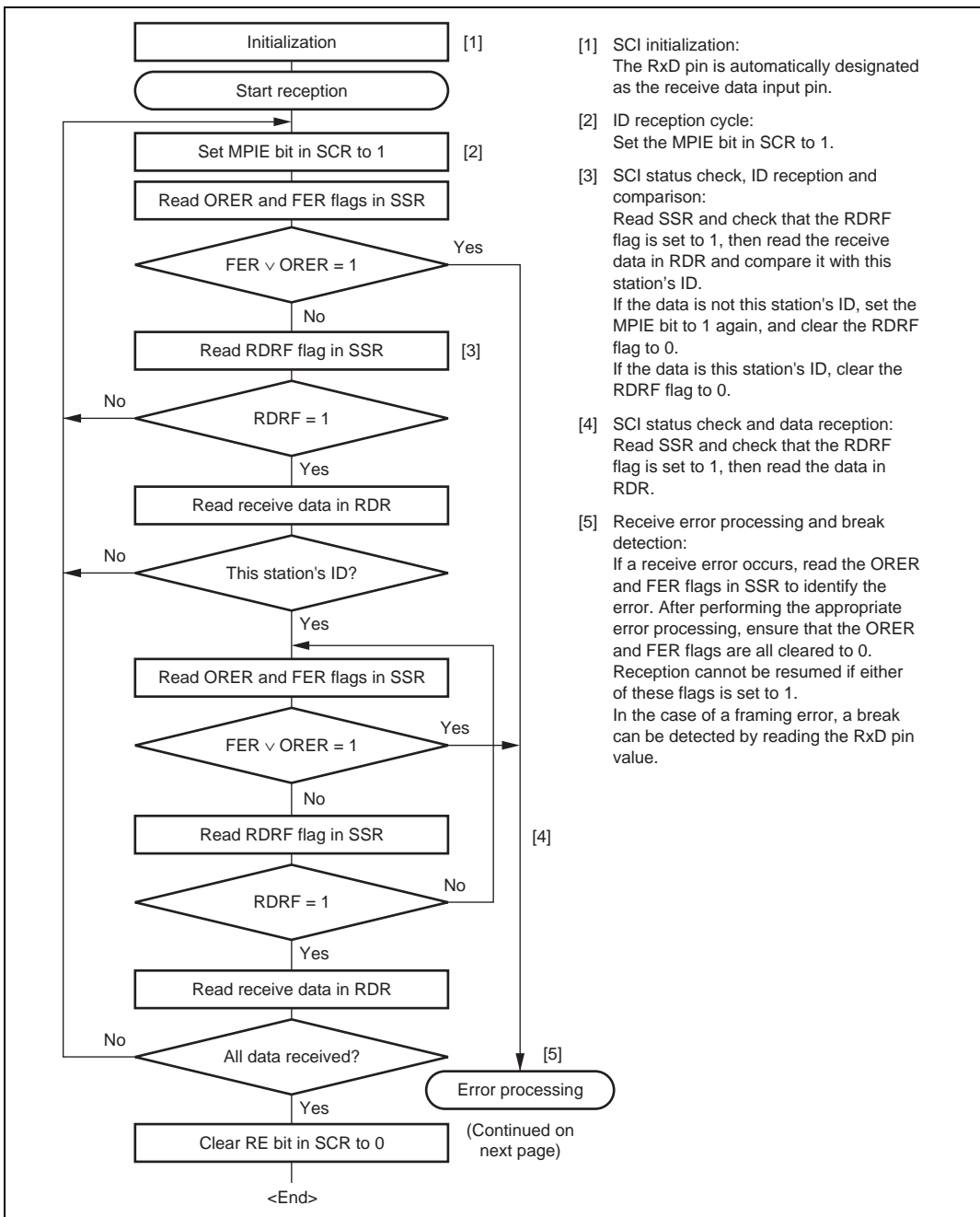
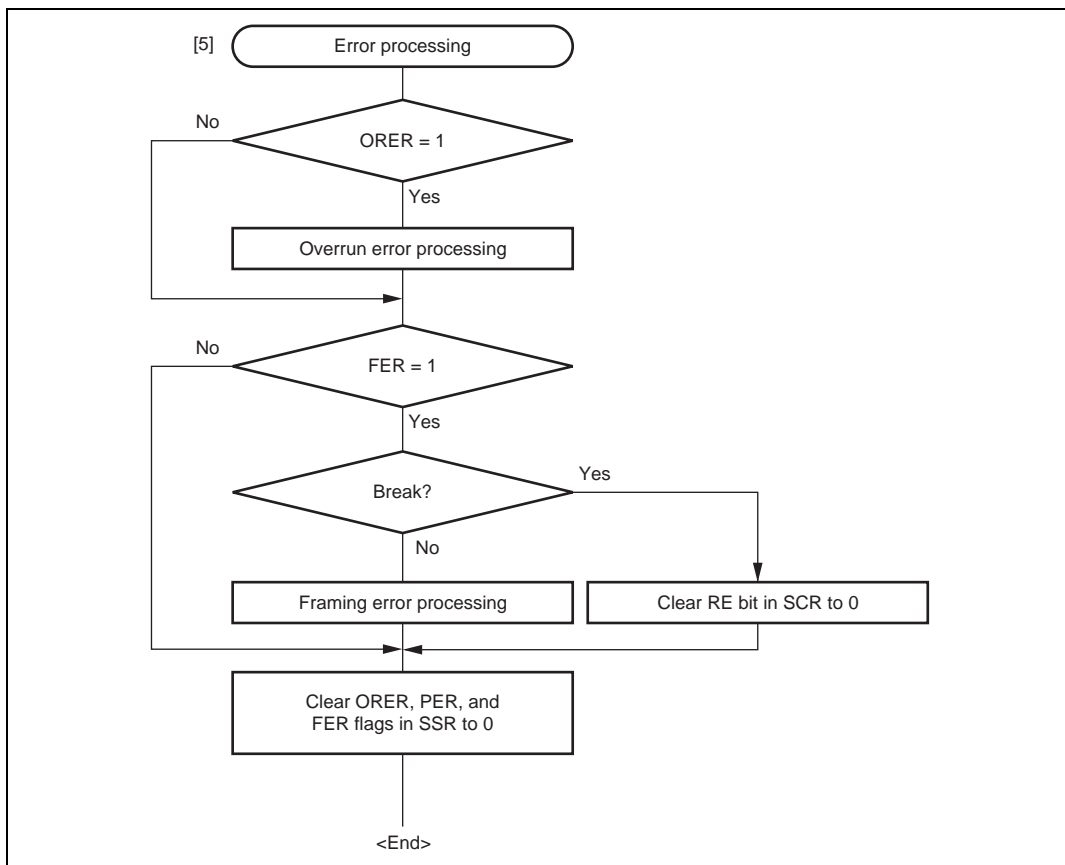


Figure 15.16 Sample Multiprocessor Serial Reception Flowchart (1)

**Figure 15.16 Sample Multiprocessor Serial Reception Flowchart (2)**

15.6 Operation in Clocked Synchronous Mode

Figure 15.17 shows the general format for clocked synchronous communication. In clocked synchronous mode, data is transmitted or received synchronous with clock pulses. In clocked synchronous serial communication, data on the transmission line is output from one falling edge of the serial clock to the next. In clocked synchronous mode, the SCI receives data in synchronous with the rising edge of the serial clock. After 8-bit data is output, the transmission line holds the MSB state. In clocked synchronous mode, no parity or multiprocessor bit is added. Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communication through the use of a common clock. Both the transmitter and the receiver also have a double-buffered structure, so data can be read or written during transmission or reception, enabling continuous data transfer.

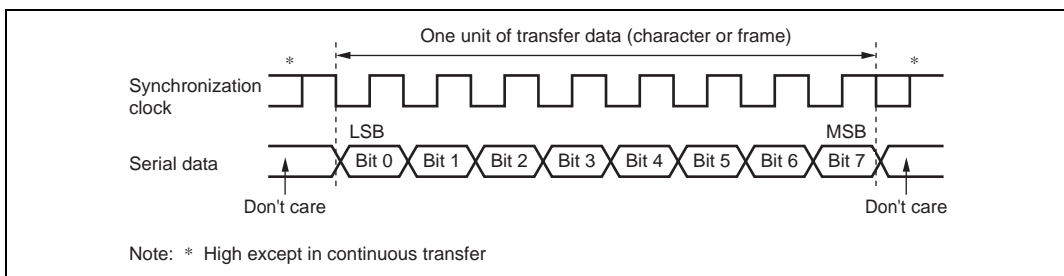


Figure 15.17 Data Format in Synchronous Communication (For LSB-First)

15.6.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCK pin can be selected, according to the setting of CKE0 and CKE1 bits in SCR. When the SCI is operated on an internal clock, the serial clock is output from the SCK pin. Eight serial clock pulses are output in the transfer of one character, and when no transfer is performed the clock is fixed high.

15.6.2 SCI Initialization (Clocked Synchronous Mode)

Before transmitting and receiving data, the TE and RE bits in SCR should be cleared to 0, then the SCI should be initialized as described in a sample flowchart in figure 15.18. When the operating mode, or transfer format, is changed for example, 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, the TDRE flag is set to 1. Note that clearing the RE bit to 0 does not change the contents of the RDRF, PER, FER, and ORER flags, or the contents of RDR.

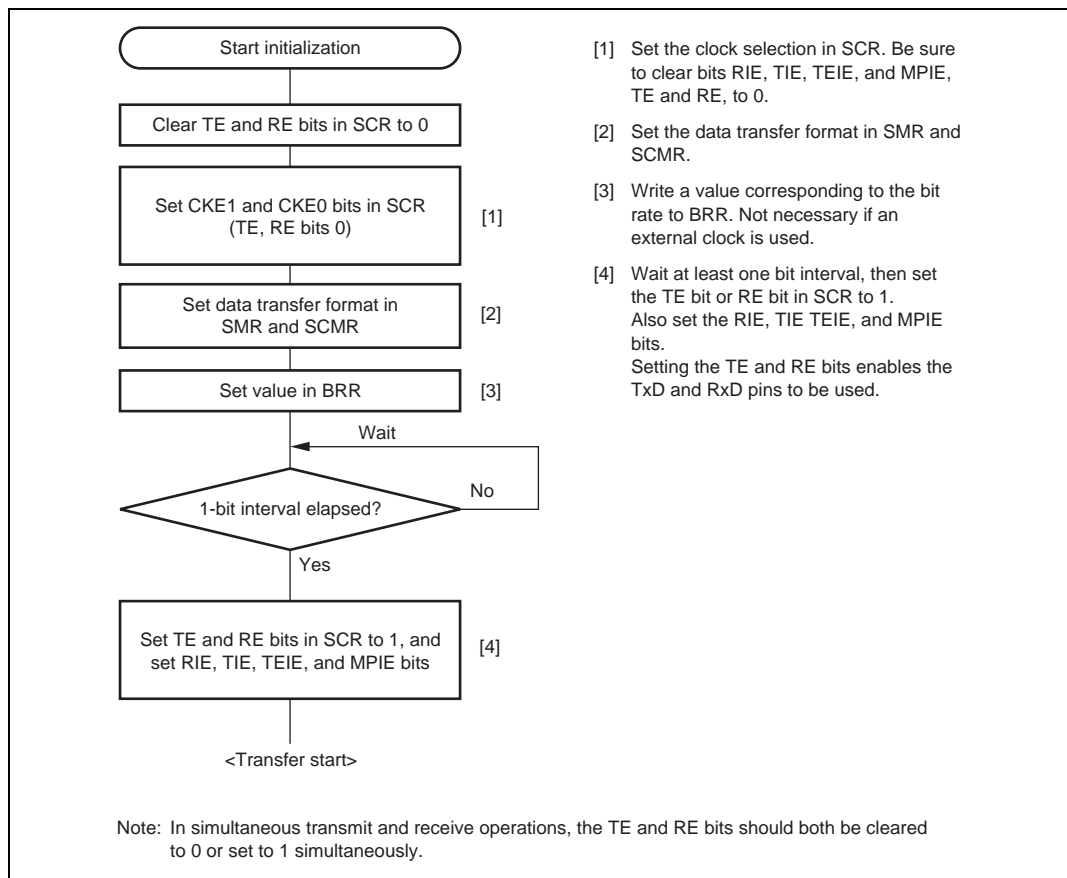


Figure 15.18 Sample SCI Initialization Flowchart

15.6.3 Serial Data Transmission (Clocked Synchronous Mode)

Figure 15.19 shows an example of SCI operation for transmission in clocked synchronous mode. In serial transmission, the SCI operates as described below.

1. The SCI monitors the TDRE flag in SSR, and if the flag is 0, the SCI recognizes that data has been written to TDR, and transfers the data from TDR to TSR.
2. After transferring data from TDR to TSR, the SCI sets the TDRE flag to 1 and starts transmission. If the TIE bit in SCR is set to 1 at this time, a transmit data empty interrupt (TXI) is generated. Continuous transmission is possible because the TXI interrupt routine writes the next transmit data to TDR before transmission of the current transmit data has been completed.

3. 8-bit data is sent from the TxD pin synchronized with the output clock when output clock mode has been specified, and synchronized with the input clock when use of an external clock has been specified.
4. The SCI checks the TDRE flag at the timing for sending the MSB (bit 7).
5. If the TDRE flag is cleared to 0, data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If the TDRE flag is set to 1, the TEND flag in SSR is set to 1, and the TDRE flag maintains the output state of the last bit. If the TEIE bit in SCR is set to 1 at this time, a TEI interrupt request is generated. The SCK pin is fixed high.

Figure 15.20 shows a sample flow chart for serial data transmission. Even if the TDRE flag is cleared to 0, transmission will not start while a receive error flag (ORER, FER, or PER) is set to 1. Make sure that the receive error flags are cleared to 0 before starting transmission. Note that clearing the RE bit to 0 does not clear the receive error flags.

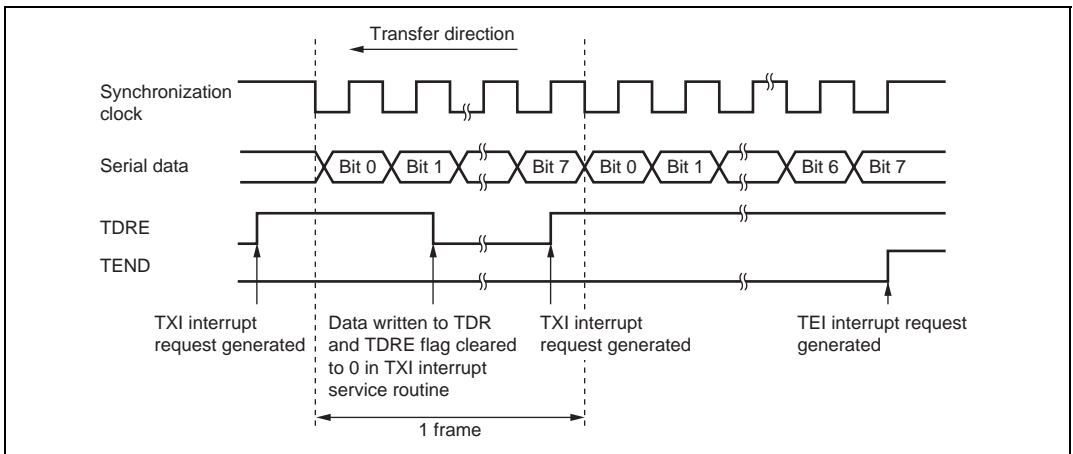
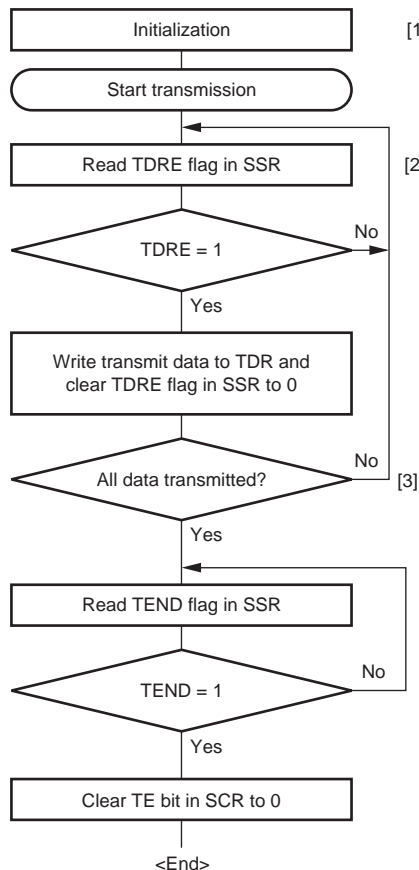


Figure 15.19 Sample SCI Transmission Operation in Clocked Synchronous Mode



[1] SCI initialization:
The TxD pin is automatically designated as the transmit data output pin.

[2] SCI status check and transmit data write:
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.

[3] Serial transmission continuation procedure:
To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR, and then clear the TDRE flag to 0.
Checking and clearing of the TDRE flag is automatic when the DMAC^{*1} or the DTC^{*2} is activated by a transmit data empty interrupt (TXI) request and data is written to TDR.

Notes: 1. Supported only by the H8S/2239 Group.
2. The case, in which the DTC automatically clears the TDRE flag, occurs only when DISEL in DTC is 0 with the transfer counter not being 0. Therefore, the TDRE flag should be cleared by CPU when DISEL is 1, or when DISEL is 0 with the transfer counter being 0.

Figure 15.20 Sample Serial Transmission Flowchart

15.6.4 Serial Data Reception (Clocked Synchronous Mode)

Figure 15.21 shows an example of SCI operation for reception in clocked synchronous mode. In serial reception, the SCI operates as described below.

1. The SCI performs internal initialization synchronous with a synchronous clock input or output, starts receiving data, and stores the received data in RSR.
2. If an overrun error occurs (when reception of the next data is completed while the RDRF flag in SSR is still set to 1), the ORER bit in SSR is set to 1. If the RIE bit in SCR is set to 1 at this time, an ERI interrupt request is generated, receive data is not transferred to RDR, and the RDRF flag remains to be set to 1.
3. If reception is completed successfully, the RDRF bit in SSR is set to 1, and receive data is transferred to RDR. If the RIE bit in SCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is possible because the RXI interrupt routine reads the receive data transferred to RDR before reception of the next receive data has finished.

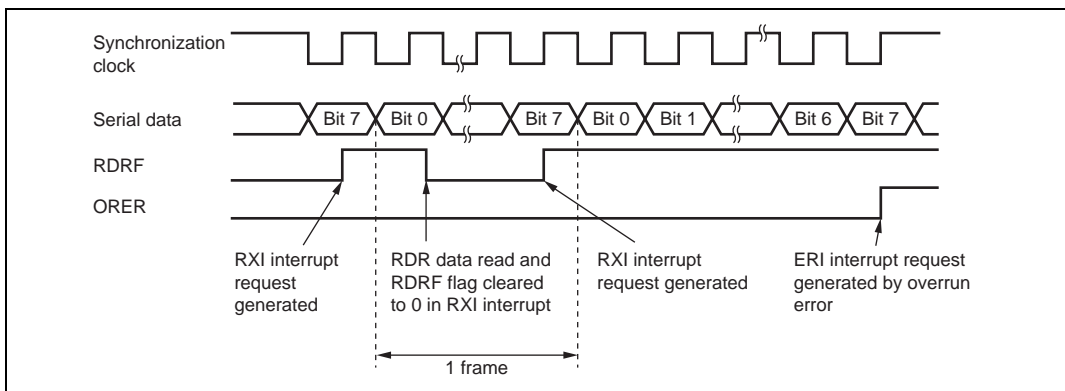


Figure 15.21 Example of SCI Operation in Reception

Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF bits to 0 before resuming reception. Figure 15.22 shows a sample flow chart for serial data reception.

An overrun error occurs or synchronous clocks are output until the RE bit is cleared to 0 when an internal clock is selected and only receive operation is possible. When a transmission and reception will be carried out in a unit of one frame, be sure to carry out a dummy transmission with only one frame by the simultaneous transmit and receive operations at the same time.

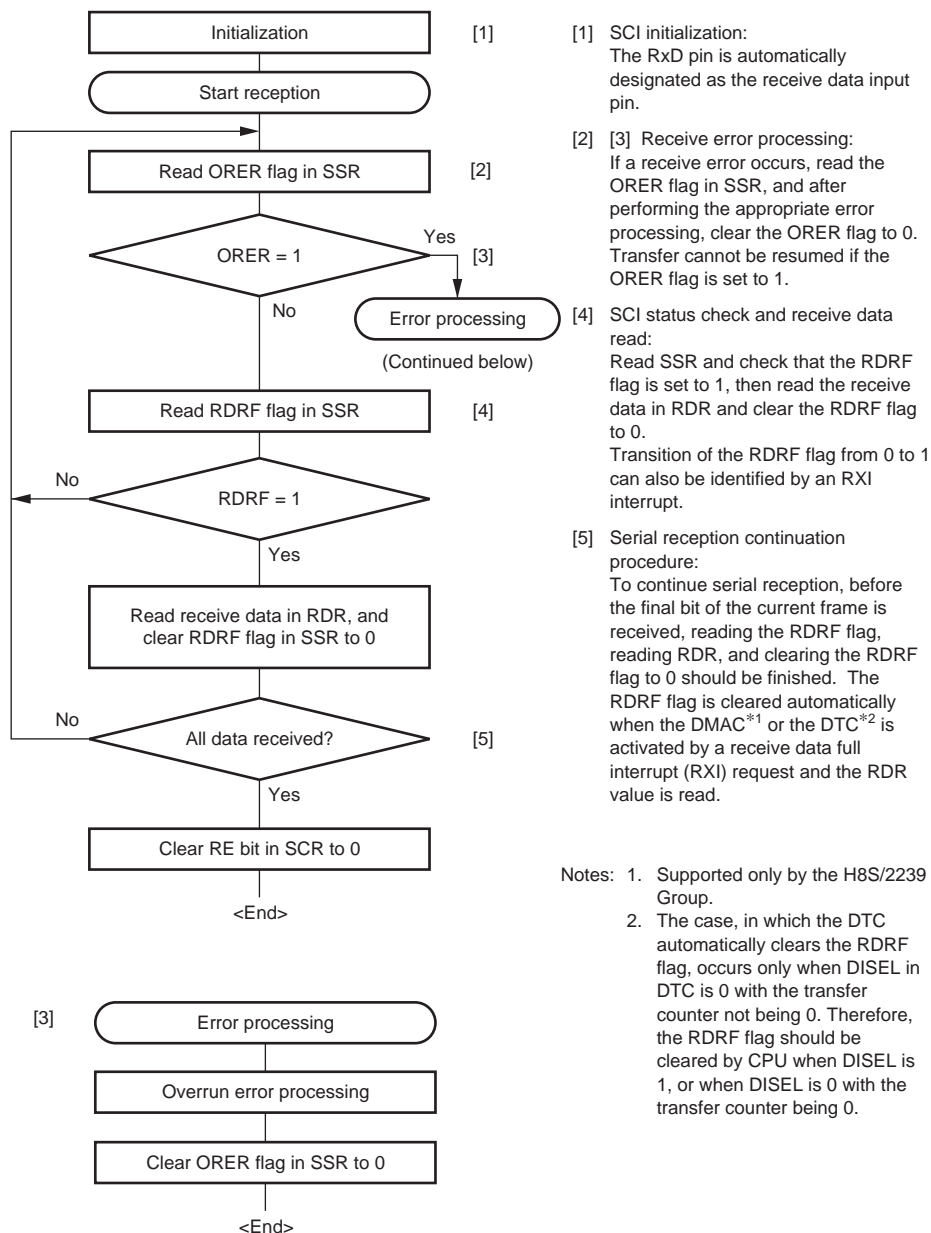
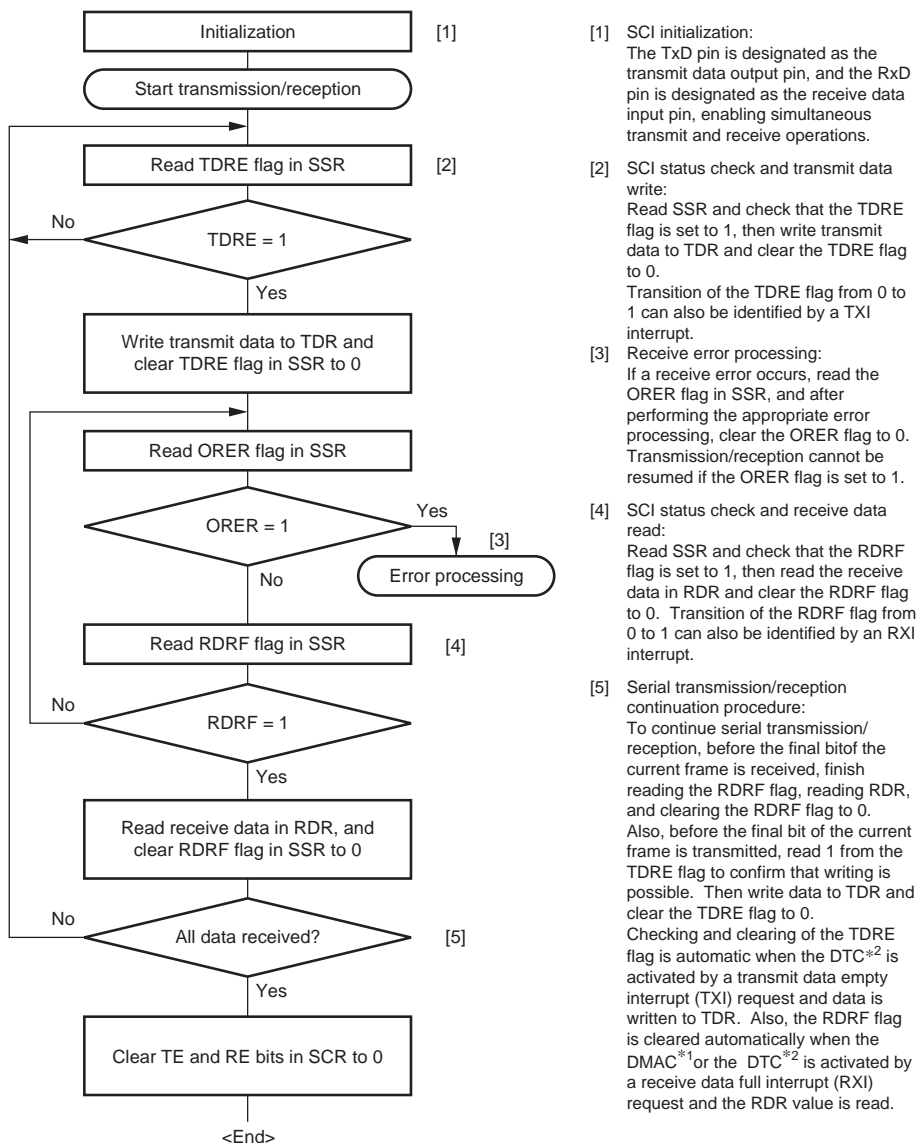


Figure 15.22 Sample Serial Reception Flowchart

15.6.5 Simultaneous Serial Data Transmission and Reception (Clocked Synchronous Mode)

Figure 15.23 shows a sample flowchart for simultaneous serial transmit and receive operations. The following procedure should be used for simultaneous serial data transmit and receive operations. To switch from transmit mode to simultaneous transmit and receive mode, after checking that the SCI has finished transmission and the TDRE and TEND flags are set to 1, clear TE to 0. Then simultaneously set TE and RE to 1 with a single instruction. To switch from receive mode to simultaneous transmit and receive mode, after checking that the SCI has finished reception, clear RE to 0. Then after checking that the RDRF and receive error flags (ORER, FER, and PER) are cleared to 0, simultaneously set TE and RE to 1 with a single instruction.



Notes: When switching from transmit or receive operation to simultaneous transmit and receive operations, first clear the TE bit and RE bit to 0, then set both these bits to 1 by one instruction simultaneously.

1. Supported only by the H8S/2239 Group.
2. The case, in which the DTC automatically clears the TDRE flag or RDRF flag, occurs only when DISEL in the corresponding DTC transfer is 0 with the transfer counter not being 0. Therefore, the corresponding flag should be cleared by CPU when DISEL in the corresponding DTC transfer is 1, or when DISEL is 0 with the transfer counter being 0.

Figure 15.23 Sample Flowchart of Simultaneous Serial Transmit and Receive Operations

15.7 Operation in Smart Card Interface

The SCI supports an IC card (Smart Card) interface that conforms to ISO/IEC 7816-3 (Identification Card) as a serial communication interface extension function. Switching between the normal serial communication interface and the Smart Card interface mode is carried out by means of a register setting.

15.7.1 Pin Connection Example

Figure 15.24 shows an example of connection with the Smart Card. In communication with an IC card, as both transmission and reception are carried out on a single data transmission line, the TxD pin and RxD pin should be connected to the LSI pin. The data transmission line should be pulled up to the V_{cc} power supply with a resistor. If an IC card is not connected, and the TE and RE bits are both set to 1, closed transmission/reception is possible, enabling self-diagnosis to be carried out. When the clock generated on the Smart Card interface is used by an IC card, the SCK pin output is input to the CLK pin of the IC card. This LSI port output is used as the reset signal.

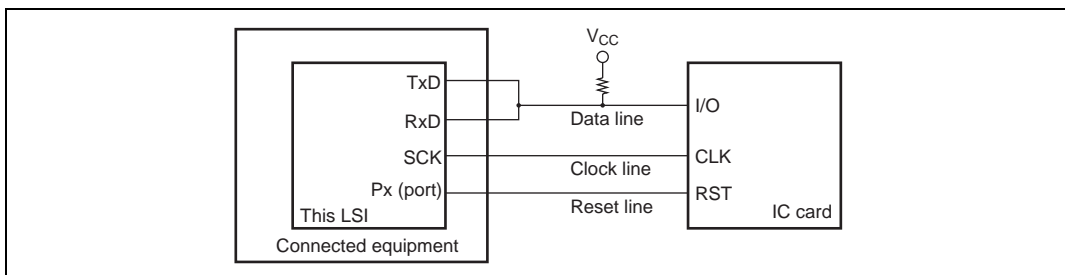


Figure 15.24 Schematic Diagram of Smart Card Interface Pin Connections

15.7.2 Data Format (Except for Block Transfer Mode)

Figure 15.25 shows the transfer data format in Smart Card interface mode.

- One frame consists of 8-bit data plus a parity bit in asynchronous mode.
- In transmission, a guard time of at least 2 etu (Elementary Time Unit: the time for transfer of 1 bit) is left between the end of the parity bit and the start of the next frame.
- If a parity error is detected during reception, a low error signal level is output for one etu period, 10.5 etu after the start bit.
- If an error signal is sampled during transmission, the same data is retransmitted automatically after a delay of 2 etu or longer.

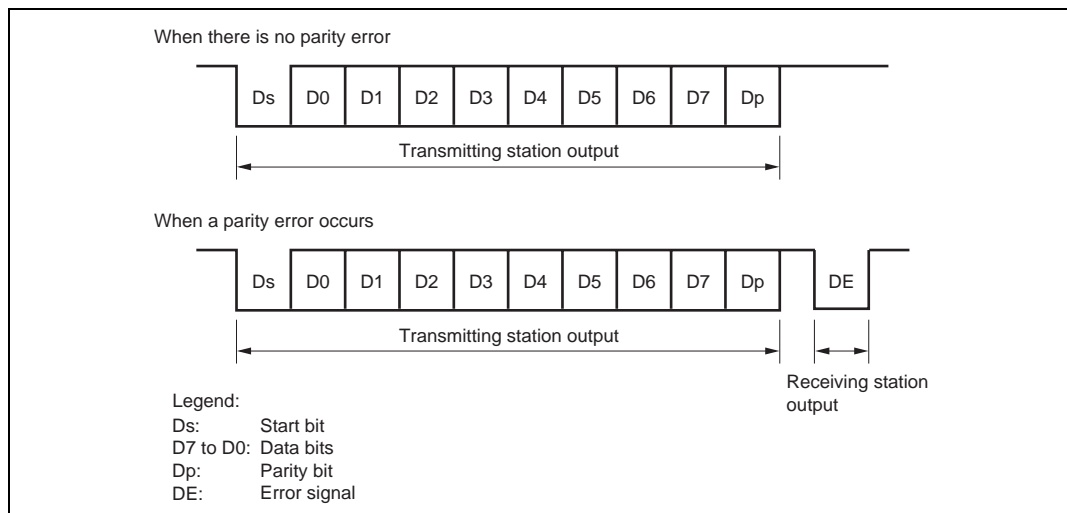


Figure 15.25 Normal Smart Card Interface Data Format

Data transfer with other types of IC cards (direct convention and inverse convention) are performed as described in the following.

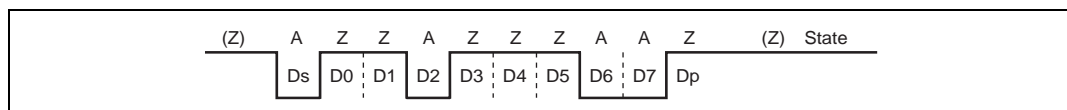


Figure 15.26 Direct Convention ($SDIR = SINV = \overline{O/E} = 0$)

With the direction convention type IC and the above sample start character, the logic 1 level corresponds to state Z and the logic 0 level to state A, and transfer is performed in LSB-first order. The start character data above is H'3B. For the direct convention type, clear the SDIR and SINV bits in SCMR to 0. According to Smart Card regulations, clear the $\overline{O/E}$ bit in SMR to 0 to select even parity mode.

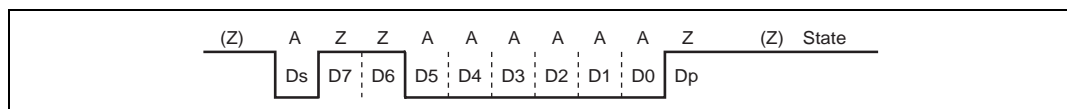


Figure 15.27 Inverse Convention ($SDIR = SINV = \overline{O/E} = 1$)

With the inverse convention type, the logic 1 level corresponds to state A and the logic 0 level to state Z, and transfer is performed in MSB-first order. The start character data for the above is H'3F. For the inverse convention type, set the SDIR and SINV bits in SCMR to 1. According to Smart Card regulations, even parity mode is the logic 0 level of the parity bit, and corresponds to

state Z. In this LSI, the SINV bit inverts only data bits D0 to D7. Therefore, set the O \overline{E} bit in SMR to 1 to invert the parity bit for both transmission and reception.

15.7.3 Block Transfer Mode

Operation in block transfer mode is the same as that in the normal Smart Card interface mode, except for the following points.

- In reception, though the parity check is performed, no error signal is output even if an error is detected. However, the PER bit in SSR is set to 1 and must be cleared before receiving the parity bit of the next frame.
- In transmission, a guard time of at least 1 etu is left between the end of the parity bit and the start of the next frame.
- In transmission, because retransmission is not performed, the TEND flag is set to 1, 11.5 etu after transmission start.
- As with the normal Smart Card interface, the ERS flag indicates the error signal status, but since error signal transfer is not performed, this flag is always cleared to 0.

15.7.4 Receive Data Sampling Timing and Reception Margin

In Smart Card interface mode an internal clock generated by the on-chip baud rate generator can only be used as a transmission/reception clock. In this mode, the SCI operates on a base clock with a frequency of 32, 64, 372, or 256 times the transfer rate (fixed to 16 times in normal asynchronous mode) as determined by bits BCP1 and BCP0. In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. As shown in figure 15.28, by sampling receive data at the rising-edge of the 16th, 32nd, 186th, or 128th pulse of the base clock, data can be latched at the middle of the bit. The reception margin is given by the following formula.

$$M = \left| \left(0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

Where M: Reception margin (%)

N: Ratio of bit rate to clock (N = 32, 64, 372, and 256)

D: Clock duty (D = 0 to 1.0)

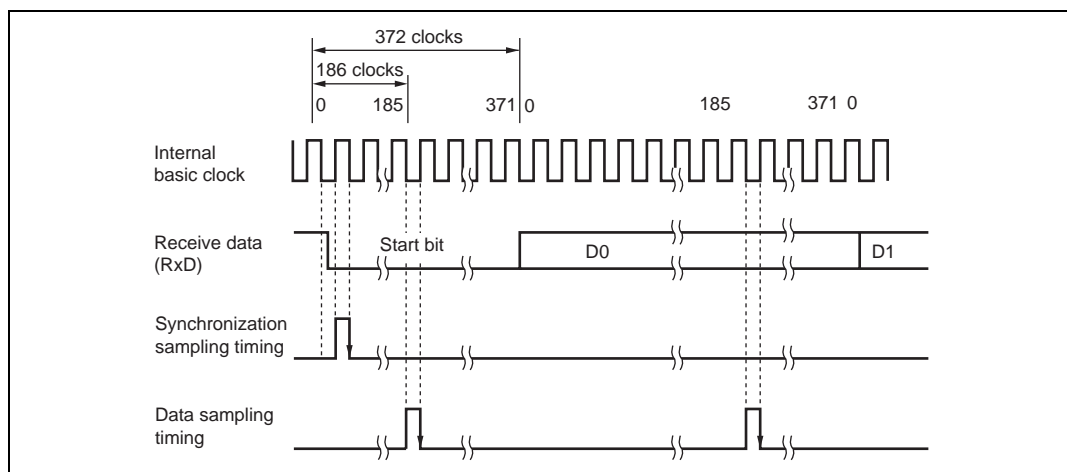
L: Frame length (L = 10)

F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5 and N = 372 in the above formula, the reception margin formula is as follows.

$$M = (0.5 - 1/2 \times 372) \times 100\%$$

$$= 49.866\%$$



**Figure 15.28 Receive Data Sampling Timing in Smart Card Mode
(Using Clock of 372 Times the Transfer Rate)**

15.7.5 Initialization

Before transmitting and receiving data, initialize the SCI as described below. Initialization is also necessary when switching from transmit mode to receive mode, or vice versa.

1. Clear the TE and RE bits in SCR to 0.
2. Clear the error flags ERS, PER, and ORER in SSR to 0.
3. Set the GM, BLK, O/ \bar{E} , BCP0, BCP1, CKS0, CKS1 bits in SMR. Set the PE bit to 1.
4. Set the SMIF, SDIR, and SIN V bits in SCMR.

When the SMIF bit is set to 1, the TxD and RxD pins are both switched from ports to SCI pins, and are placed in the high-impedance state.

5. Set the value corresponding to the bit rate in BRR.
6. Set the CKE0 and CKE1 bits in SCR. Clear the TIE, RIE, TE, RE, MPIE, and TEIE bits to 0.
If the CKE0 bit is set to 1, the clock is output from the SCK pin.
7. Wait at least one bit interval, then set the TIE, RIE, TE, and RE bits in SCR. Do not set the TE bit and RE bit at the same time, except for self-diagnosis.

To switch from receive mode to transmit mode, after checking that the SCI has finished reception, initialize the SCI, and set RE to 0 and TE to 1. Whether SCI has finished reception or not can be checked with the RDRF, PER, or ORER flags. To switch from transmit mode to receive mode,

after checking that the SCI has finished transmission, initialize the SCI, and set TE to 0 and RE to 1. Whether SCI has finished transmission or not can be checked with the TEND flag.

15.7.6 Serial Data Transmission (Except for Block Transfer Mode)

As data transmission in Smart Card interface mode involves error signal sampling and retransmission processing, the operations are different from those in normal serial communication interface mode (except for block transfer mode). Figure 15.29 illustrates the retransfer operation when the SCI is in transmit mode.

1. If an error signal is sent back from the receiving end after transmission of one frame is complete, the ERS bit in SSR is set to 1. If the RIE bit in SCR is enabled at this time, an ERI interrupt request is generated. The ERS bit in SSR should be cleared to 0 by the time the next parity bit is sampled.
2. The TEND bit in SSR is not set for a frame in which an error signal indicating an abnormality is received. Data is retransferred from TDR to TSR, and retransmitted automatically.
3. If an error signal is not sent back from the receiving end, the ERS bit in SSR is not set. Transmission of one frame, including a retransfer, is judged to have been completed, and the TEND bit in SSR is set to 1. If the TIE bit in SCR is enabled at this time, a TXI interrupt request is generated. Writing transmit data to TDR transfers the next transmit data.

Figure 15.31 shows a flowchart for transmission. A sequence of transmit operations can be performed automatically by specifying the DTC to be activated with a TXI interrupt source. In a transmit operation, the TDRE flag is set to 1 at the same time as the TEND flag in SSR is set, and a TXI interrupt will be generated if the TIE bit in SCR has been set to 1. If the TXI request is designated beforehand as a DTC activation source, the DTC will be activated by the TXI request, and transfer of the transmit data will be carried out. At this moment, if DISEL in DTC is 0 with the transfer counter not being 0, the TDRE and TEND flags are automatically cleared to 0 when data is transferred by the DTC. When DISEL is 1, or DISEL is 0 with the transfer counter being 0, the DTC writes the transfer data to the TDR but does not clear the flag. Therefore, the flag should be cleared by CPU. In addition, in the event of the error, the SCI retransmits the same data automatically. During this period, the TEND flag remains cleared to 0 and the DTC is not activated. Therefore, the SCI and DTC will automatically transmit the specified number of bytes in the event of an error, including retransmission. However, the ERS flag is not cleared automatically when an error occurs, and so the RIE bit should be set to 1 beforehand so that an ERI request will be generated in the event of an error, and the ERS flag will be cleared.

When performing transfer using the DTC, it is essential to set and enable the DTC before carrying out SCI setting. For details of the DTC setting procedures, refer to section 9, Data Transfer Controller (DTC).

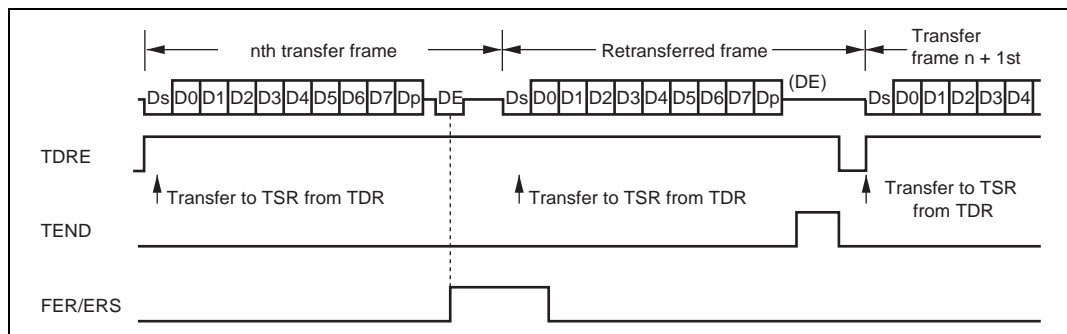


Figure 15.29 Retransfer Operation in SCI Transmit Mode

The timing for setting the TEND flag depends on the value of the GM bit in SMR. The TEND flag set timing is shown in figure 15.30.

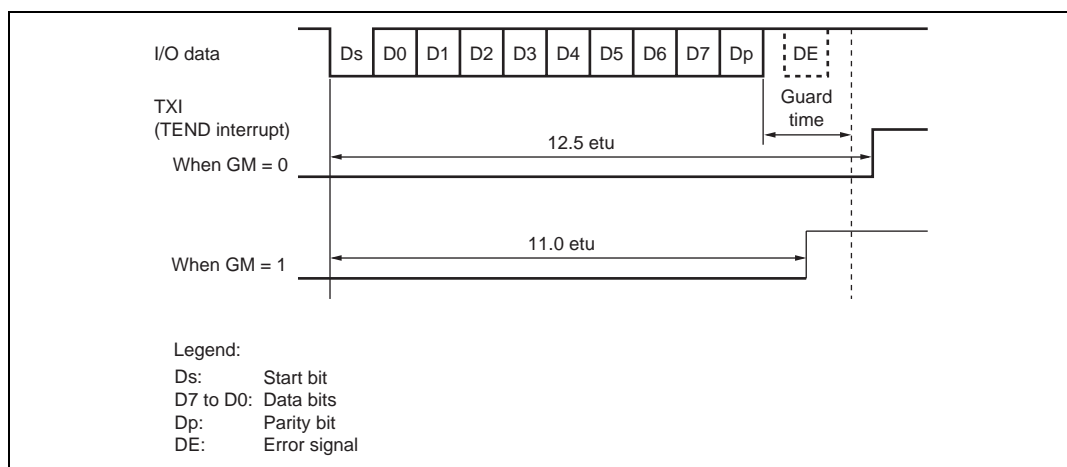


Figure 15.30 TEND Flag Generation Timing in Transmission Operation

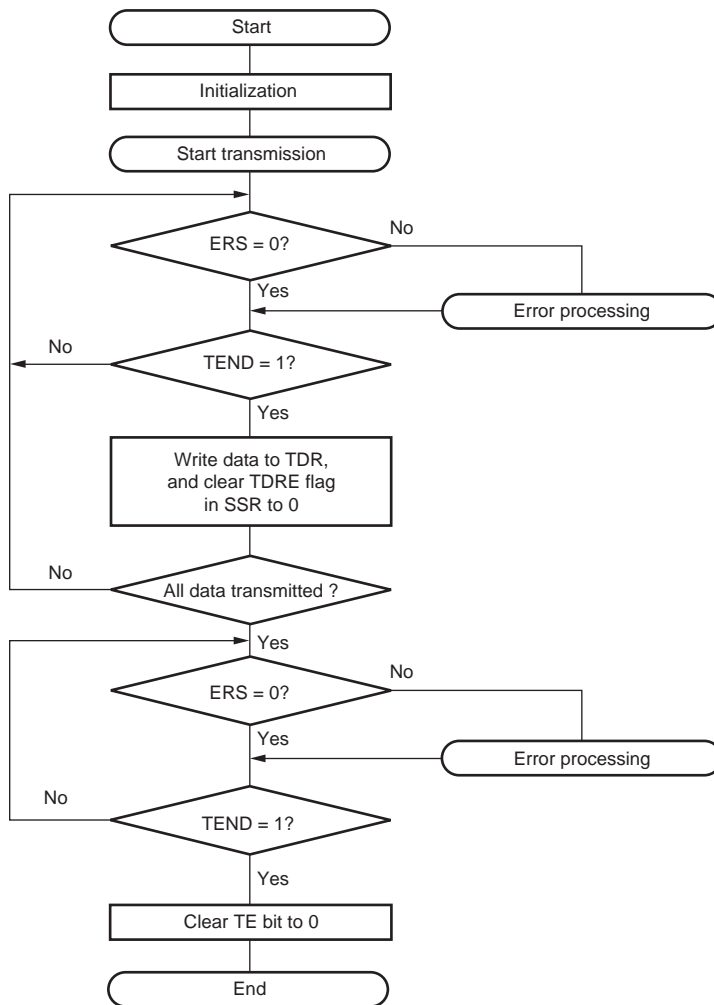


Figure 15.31 Example of Transmission Processing Flow

15.7.7 Serial Data Reception (Except for Block Transfer Mode)

Data reception in Smart Card interface mode uses the same operation procedure as for normal serial communication interface mode. Figure 15.32 illustrates the retransfer operation when the SCI is in receive mode.

1. If an error is found when the received parity bit is checked, the PER bit in SSR is automatically set to 1. If the RIE bit in SCR is set at this time, an ERI interrupt request is generated. The PER bit in SSR should be kept cleared to 0 until the next parity bit is sampled.
2. The RDRF bit in SSR is not set for a frame in which an error has occurred.
3. If no error is found when the received parity bit is checked, the PER bit in SSR is not set to 1, the receive operation is judged to have been completed normally, and the RDRF flag in SSR is automatically set to 1. If the RIE bit in SCR is enabled at this time, an RXI interrupt request is generated.

Figure 15.33 shows a flowchart for reception. A sequence of receive operations can be performed automatically by specifying the DTC to be activated using an RXI interrupt source. In a receive operation, an RXI interrupt request is generated when the RDRF flag in SSR is set to 1. If the RXI request is designated beforehand as a DTC activation source, the DTC will be activated by the RXI request, and the receive data will be transferred. At this moment, if DISEL in DTC is 0 with the transfer counter not being 0, the RDRF flag is automatically cleared. When DISEL is 1, or DISEL is 0 with the transfer counter being 0, the DTC transfers receive data but does not clear the flag. Therefore, the flag should be cleared by CPU. If an error occurs in receive mode and the ORER or PER flag is set to 1, a transfer error interrupt (ERI) request will be generated. Hence, so the error flag must be cleared to 0. In the event of an error, the DTC is not activated and receive data is skipped. Therefore, receive data is transferred for only the specified number of bytes in the event of an error. Even when a parity error occurs in receive mode and the PER flag is set to 1, the data that has been received is transferred to RDR and can be read from there.

Note: For details on receive operations in block transfer mode, refer to section 15.4, Operation in Asynchronous Mode.

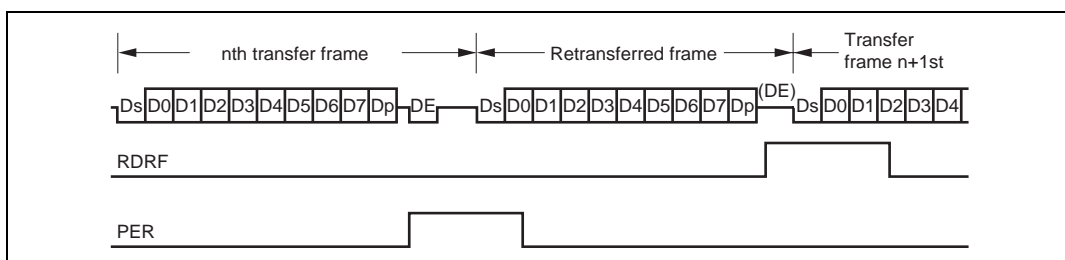


Figure 15.32 Retransfer Operation in SCI Receive Mode

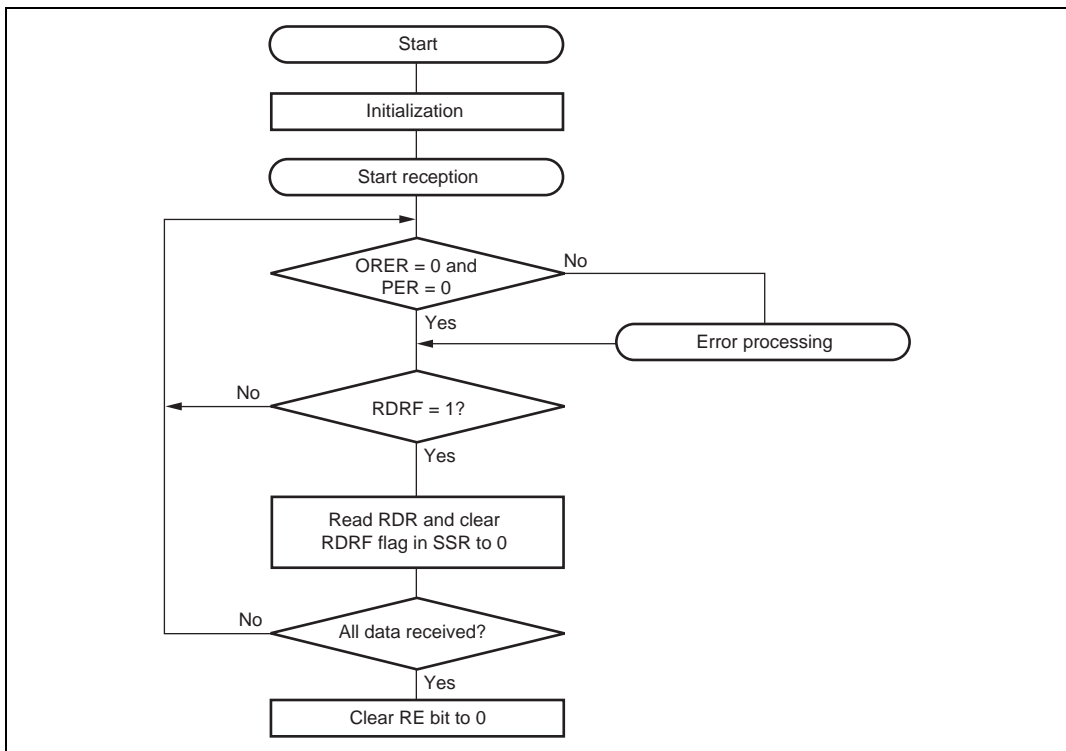


Figure 15.33 Example of Reception Processing Flow

15.7.8 Clock Output Control

When the GM bit in SMR is set to 1, the clock output level can be fixed with bits CKE0 and CKE1 in SCR. At this time, the minimum clock pulse width can be made the specified width. Figure 15.34 shows the timing for fixing the clock output level. In this example, GM is set to 1, CKE1 is cleared to 0, and the CKE0 bit is controlled.

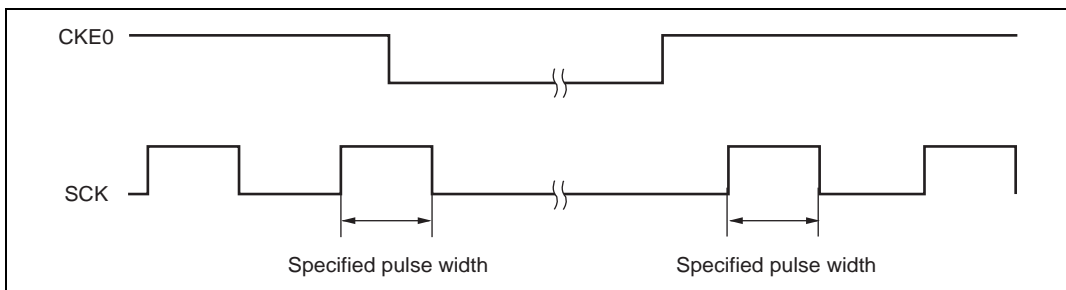


Figure 15.34 Timing for Fixing Clock Output Level

When turning on the power or switching between Smart Card interface mode and software standby mode, the following procedures should be followed in order to maintain the clock duty.

Powering On: To secure clock duty from power-on, the following switching procedure should be followed.

1. The initial state is port input and high impedance. Use a pull-up resistor or pull-down resistor to fix the potential.
2. Fix the SCK pin to the specified output level with the CKE1 bit in SCR.
3. Set SMR and SCMR, and switch to smart card mode operation.
4. Set the CKE0 bit in SCR to 1 to start clock output.

When changing from smart card interface mode to software standby mode:

1. Set the data register (DR) and data direction register (DDR) corresponding to the SCK pin to the value for the fixed output state in software standby mode.
2. Write 0 to the TE bit and RE bit in the serial control register (SCR) to halt transmit/receive operation. At the same time, set the CKE1 bit to the value for the fixed output state in software standby mode.
3. Write 0 to the CKE0 bit in SCR to halt the clock.
4. Wait for one serial clock period.

During this interval, clock output is fixed at the specified level, with the duty preserved.

5. Make the transition to the software standby state.

When returning to smart card interface mode from software standby mode:

1. Exit the software standby state.
2. Write 1 to the CKE0 bit in SCR and output the clock. Signal generation is started with the normal duty.

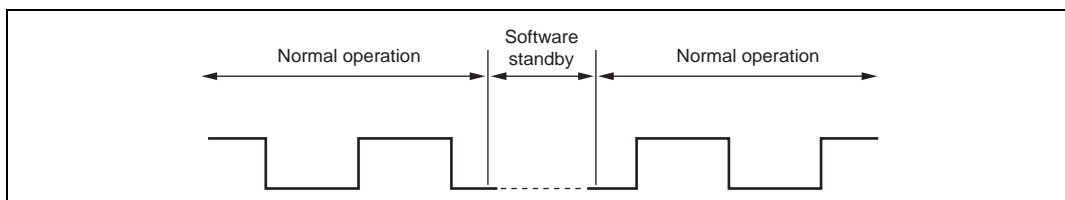


Figure 15.35 Clock Halt and Restart Procedure

15.8 SCI Select Function (H8S/2239 Group Only)

SCI_0 provides the SCI select function that enables one-to-one clocked synchronous communication between a master LSI and multiple slave LSIs (these LSIs). Figure 15.36 shows an example of communication using the SCI select function and figure 15.37 shows the summary of its operation.

The master LSI enables to communicate with the slave LSI_A by setting the $\overline{\text{SEL_A}}$ signal to low and the $\overline{\text{SEL_B}}$ signal to high. In this case, the TxD0_B pin of the slave LSI_B becomes Hi-Z and that fixes the on-chip SCK0_B signal high, causing the communication terminated. To communicate with the slave LSI_B, set the $\overline{\text{SEL_A}}$ signal to high and the $\overline{\text{SEL_B}}$ signal to low.*

The slave LSI detects its being selected by the low input interrupt of $\overline{\text{IRQ7}}$ and handles data transferring smoothly.

Note: * Change the select signal of the master LSI ($\overline{\text{SEL_A}}$ or $\overline{\text{SEL_B}}$) while the serial clock (M_SCK) is high after the last bit of the transmit data has been output. In addition, set only one select signal to low at a time.

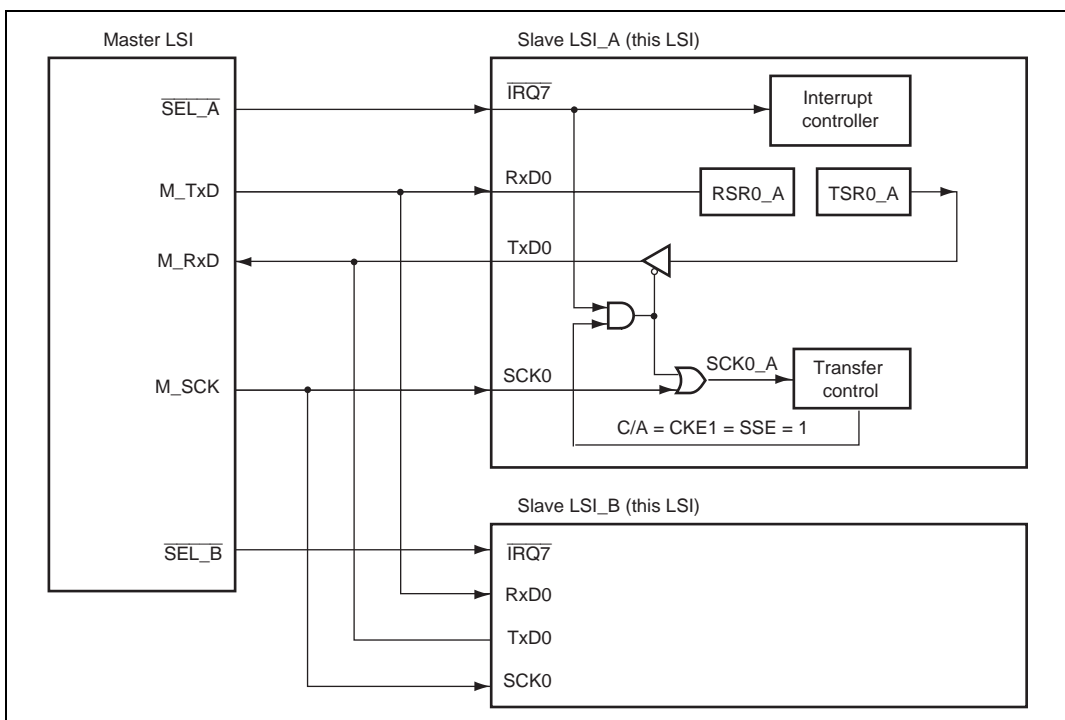


Figure 15.36 Example of Communication Using SCI Select Function

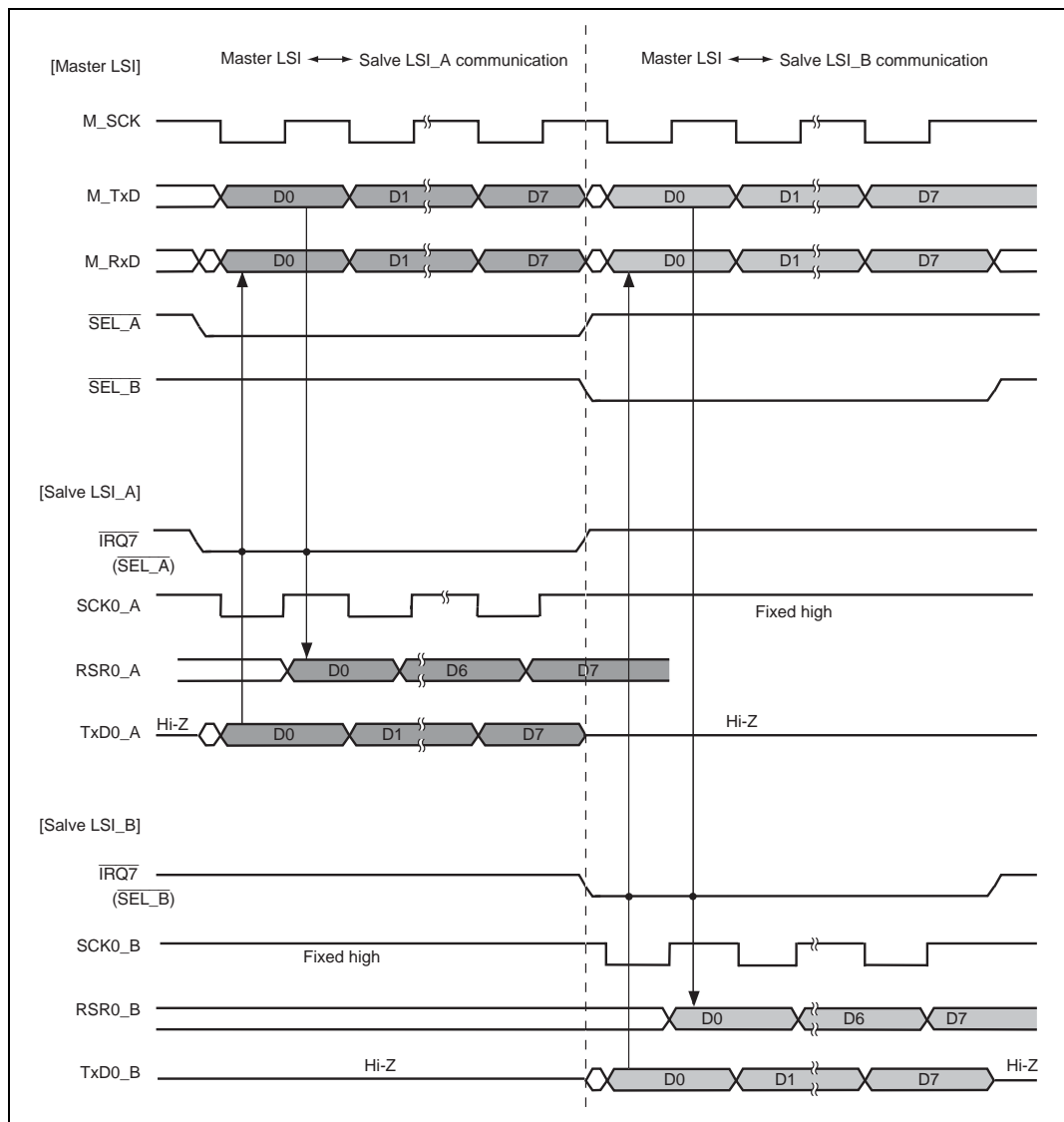


Figure 15.37 Summary of SCI Select Function Operation

15.9 Interrupt Sources

15.9.1 Interrupts in Normal Serial Communication Interface Mode

Table 15.12 shows the interrupt sources in normal serial communication interface mode. A different interrupt vector is assigned to each interrupt source, and individual interrupt sources can be enabled or disabled using the enable bits in SCR.

When the TDRE flag in SSR is set to 1, a TXI interrupt request is generated. When the TEND flag in SSR is set to 1, a TEI interrupt request is generated.

A TXI interrupt can activate the DTC to perform data transfer. The TDRE flag is cleared to 0 automatically when data is transferred by the DMAC^{*1} or the DTC^{*2}.

When the RDRF flag in SSR is set to 1, an RXI interrupt request is generated. When the ORER, PER, or FER flag in SSR is set to 1, an ERI interrupt request is generated.

An RXI interrupt request can activate the DMAC^{*1} or the DTC^{*2} to transfer data. The RDRF flag is cleared to 0 automatically when data is transferred by the DMAC^{*1} or the DTC^{*2}.

A TEI interrupt is requested when the TEND flag is set to 1 and the TEIE bit is set to 1. If a TEI interrupt and a TXI interrupt are requested simultaneously, the TXI interrupt has priority for acceptance. However, if the TDRE and TEND flags are cleared simultaneously by the TXI interrupt routine, the SCI cannot branch to the TEI interrupt routine later.

- Notes:
1. Supported only by the H8S/2239 Group.
 2. The flag is cleared only when DISEL in DTC is 0 with the transfer counter not being 0.

Table 15.12 Interrupt Sources of Serial Communication Interface Mode

| Channel | Name | Interrupt Source | Interrupt Flag | DTC Activation | DMAC Activation ^{*2} | Priority ^{*1} |
|-----------------|------|---------------------|----------------|----------------|-------------------------------|------------------------|
| 0 | ERI0 | Receive Error | ORER, FER, PER | Not possible | Not possible | High |
| | RXI0 | Receive Data Full | RDRF | Possible | Possible | |
| | TXI0 | Transmit Data Empty | TDRE | Possible | Possible | |
| | TEI0 | Transmission End | TEND | Not possible | Not possible | |
| 1 | ERI1 | Receive Error | ORER, FER, PER | Not possible | Not possible | |
| | RXI1 | Receive Data Full | RDRF | Possible | Possible | |
| | TXI1 | Transmit Data Empty | TDRE | Possible | Possible | |
| | TEI1 | Transmission End | TEND | Not possible | Not possible | |
| 2 ^{*3} | ERI2 | Receive Error | ORER, FER, PER | Not possible | Not possible | |
| | RXI2 | Receive Data Full | RDRF | Possible | Not possible | |
| | TXI2 | Transmit Data Empty | TDRE | Possible | Not possible | |
| | TEI2 | Transmission End | TEND | Not possible | Not possible | |
| 3 | ERI3 | Receive Error | ORER, FER, PER | Not possible | Not possible | Low |
| | RXI3 | Receive Data Full | RDRF | Possible | Not possible | |
| | TXI3 | Transmit Data Empty | TDRE | Possible | Not possible | |
| | TEI3 | Transmission End | TEND | Not possible | Not possible | |

Notes: 1. Indicates the initial state immediately after a reset. Priorities in channels can be changed by the interrupt controller.

2. Supported only by the H8S/2239 Group.

3. Not available in the H8S/2227 Group.

15.9.2 Interrupts in Smart Card Interface Mode

Table 15.13 shows the interrupt sources in Smart Card interface mode. The transmit end interrupt (TEI) request cannot be used in this mode.

Note: In case of block transfer mode, see section 15.9.1, Interrupts in Normal Serial Communication Interface Mode.

Table 15.13 Interrupt Sources in Smart Card Interface Mode

| Channel | Name | Interrupt Source | Interrupt Flag | DTC Activation | DMAC Activation ^{*2} | Priority ^{*1} |
|-----------------|------|--------------------------|----------------|----------------|-------------------------------|---|
| 0 | ERI0 | Receive Error, detection | ORER, PER, ERS | Not possible | Not possible | <div>High</div> <div>↑</div> <div>Low</div> |
| | RXI0 | Receive Data Full | RDRF | Possible | Possible | |
| | TXI0 | Transmit Data Empty | TEND | Possible | Possible | |
| 1 | ERI1 | Receive Error, detection | ORER, PER, ERS | Not possible | Not possible | |
| | RXI1 | Receive Data Full | RDRF | Possible | Possible | |
| | TXI1 | Transmit Data Empty | TEND | Possible | Possible | |
| 2 ^{*3} | ERI2 | Receive Error, detection | ORER, PER, ERS | Not possible | Not possible | |
| | RXI2 | Receive Data Full | RDRF | Possible | Not possible | |
| | TXI2 | Transmit Data Empty | TEND | Possible | Not possible | |
| 3 | ERI3 | Receive Error, detection | ORER, PER, ERS | Not possible | Not possible | |
| | RXI3 | Receive Data Full | RDRF | Possible | Not possible | |
| | TXI3 | Transmit Data Empty | TEND | Possible | Not possible | |

Notes: 1. Indicates the initial state immediately after a reset. Priorities in channels can be changed by the interrupt controller.

2. Supported only by the H8S/2239 Group.

3. Not available in the H8S/2227 Group.

15.10 Usage Notes

15.10.1 Module Stop Mode Setting

SCI operation can be disabled or enabled using the module stop control register. The initial setting is for SCI operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 24, Power-Down Modes.

15.10.2 Break Detection and Processing (Asynchronous Mode Only)

When framing error (FER) detection is performed, a break can be detected by reading the RxD pin value directly. In a break, the input from the RxD pin becomes all 0s, setting the FER flag, and possibly the PER flag. Note that as the SCI continues the receive operation after receiving a break, even if the FER flag is cleared to 0, it will be set to 1 again.

15.10.3 Mark State and Break Detection (Asynchronous Mode Only)

When TE is 0, the TxD pin is used as an I/O port whose direction (input or output) and level are determined by DDR. This can be used to set the TxD pin to mark state (high level) or send a break during serial data transmission. To maintain the communication line at mark state until TE is set to 1, set both DDR and DR to 1. As TE is cleared to 0 at this point, the TxD pin becomes an I/O port, and 1 is output from the TxD pin. To send a break during serial transmission, first set PDR to 1 and DR to 0, and then clear TE to 0. When TE is cleared to 0, the transmitter is initialized regardless of the current transmission state, the TxD pin becomes an I/O port, and 0 is output from the TxD pin.

15.10.4 Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only)

Transmission cannot be started when a receive error flag (ORER, PER, or FER) is set to 1, even if the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before starting transmission. Note also that receive error flags cannot be cleared to 0 even if the RE bit is cleared to 0.

15.10.5 Restrictions on Use of DMAC* or DTC

- When an external clock source is used as the serial clock, the transmit clock should not be input until at least 5 ϕ clock cycles after the TDR is updated by the DMAC* or the DTC. Misoperation may occur if the transmit clock is input within 4 ϕ clocks after TDR is updated (figure 15.38).
- When RDR is read by the DMAC* or the DTC, be sure to set the activation source to the relevant SCI reception data full interrupt (RXI).
- The flag is cleared only when DISEL in DTC is 0 with the transfer counter not being 0. When DISEL is 1, or DISEL is 0 with the transfer counter being 0, the flag should be cleared by CPU. Note that transmitting, in particular, may not successfully be executed unless the TDRE flag is cleared by CPU.

Note: * Supported only by the H8S/2239 Group.

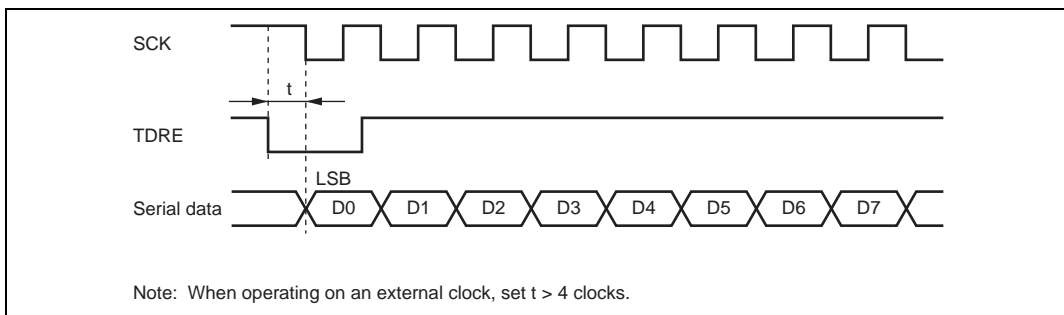


Figure 15.38 Example of Clocked Synchronous Transmission by DMAC* or DTC

Note: * Supported only by the H8S/2239 Group.

15.10.6 Operation in Case of Mode Transition

- Transmission

Operation should be stopped (by clearing TE, TIE, and TEIE to 0) before making a module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode transition. TSR, TDR, and SSR are reset. The output pin states in module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode depend on the port settings, and becomes high-level output after the relevant mode is cleared. If a transition is made during transmission, the data being transmitted will be undefined. When transmitting without changing the transmit mode after the relevant mode is cleared, transmission can be started by setting TE to 1 again, and performing the following sequence: SSR read → TDR write →

TDRE clearance. To transmit with a different transmit mode after clearing the relevant mode, the procedure must be started again from initialization. Figure 15.39 shows a sample flowchart for mode transition during transmission. Port pin states are shown in figures 15.40 and 15.41.

Operation should also be stopped (by clearing TE, TIE, and TEIE to 0) before making a transition from transmission by DTC transfer to module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode transition. To perform transmission with the DTC after the relevant mode is cleared, setting TE and TIE to 1 will set the TXI flag and start DTC transmission.

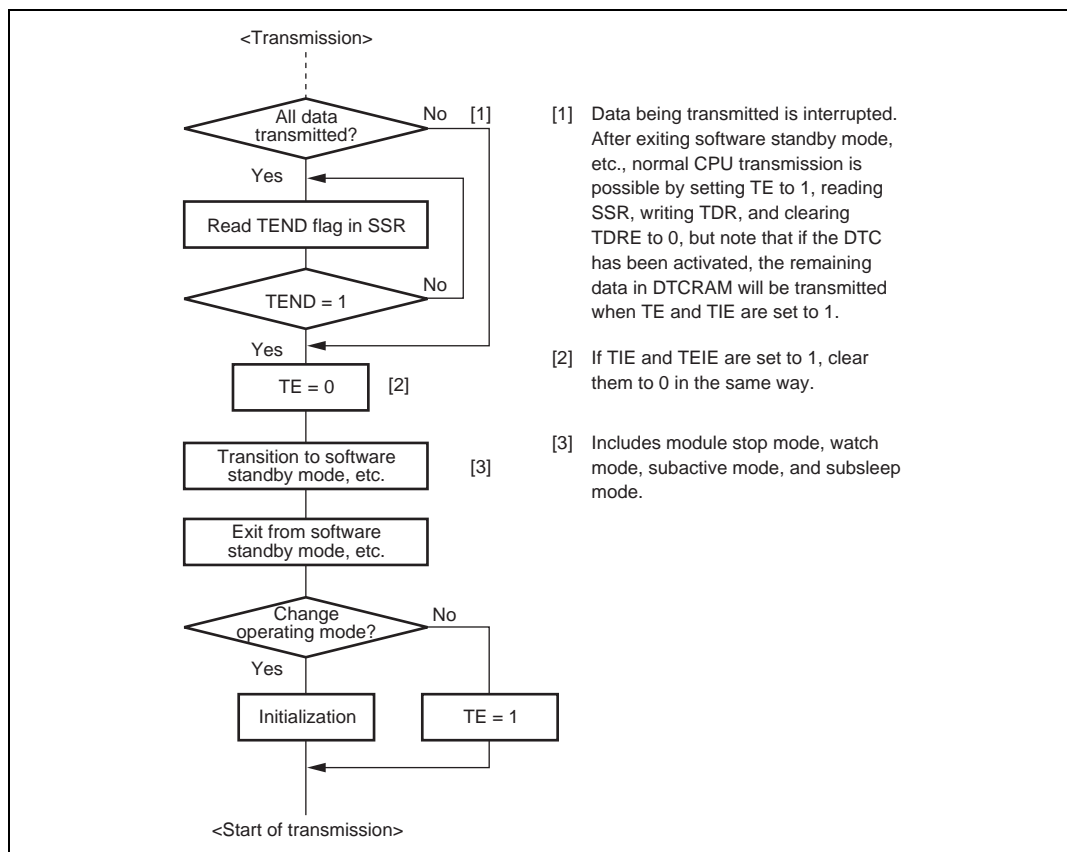


Figure 15.39 Sample Flowchart for Mode Transition during Transmission

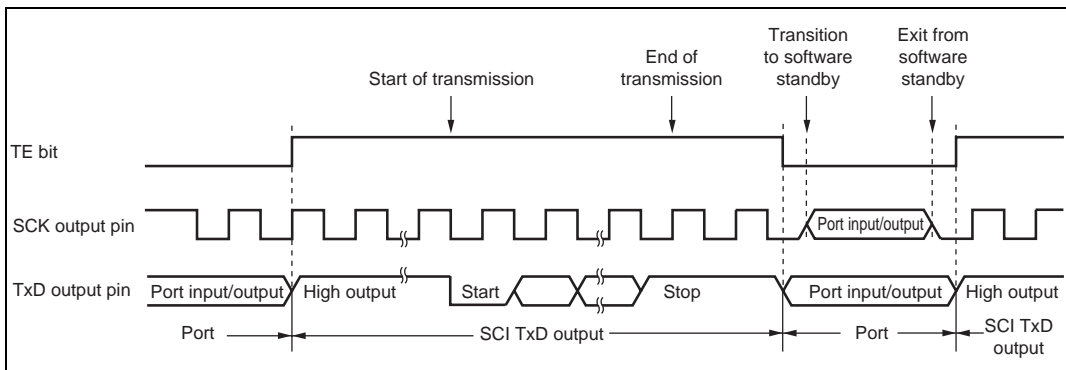


Figure 15.40 Asynchronous Transmission Using Internal Clock

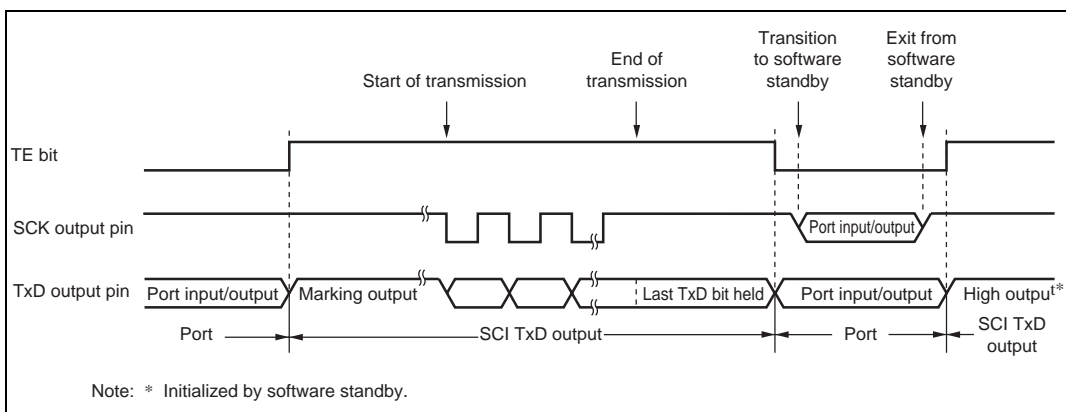


Figure 15.41 Synchronous Transmission Using Internal Clock

- Reception

Receive operation should be stopped (by clearing RE to 0) before making a module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode transition. RSR, RDR, and SSR are reset. If a transition is made without stopping operation, the data being received will be invalid.

To continue receiving without changing the reception mode after the relevant mode is cleared, set RE to 1 before starting reception. To receive with a different receive mode, the procedure must be started again from initialization.

Figure 15.42 shows a sample flowchart for mode transition during reception.

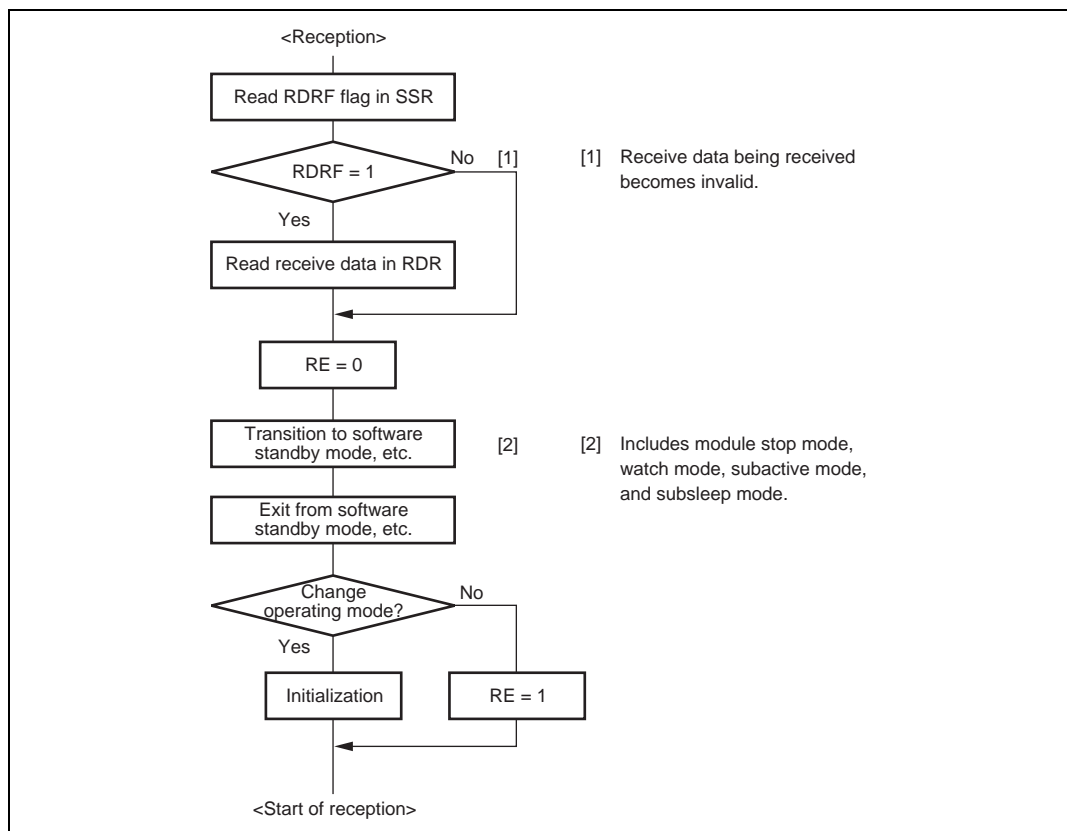


Figure 15.42 Sample Flowchart for Mode Transition during Reception

15.10.7 Switching from SCK Pin Function to Port Pin Function

- Problem in Operation

When switching the SCK pin function to the output port function (high-level output) by making the following settings while $\text{DDR} = 1$, $\text{DR} = 1$, $\text{C}/\overline{\text{A}} = 1$, $\text{CKE1} = 0$, $\text{CKE0} = 0$, and $\text{TE} = 1$ (synchronous mode), low-level output occurs for one half-cycle.

1. End of serial data transmission
2. $\text{TE} \text{ bit} = 0$
3. $\text{C}/\overline{\text{A}} \text{ bit} = 0 \dots$ Switchover to port output
4. Occurrence of low-level output

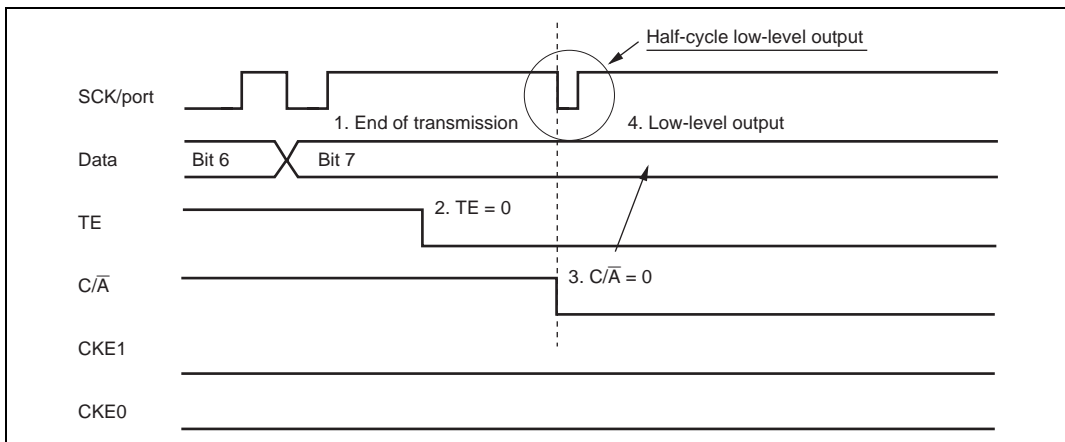


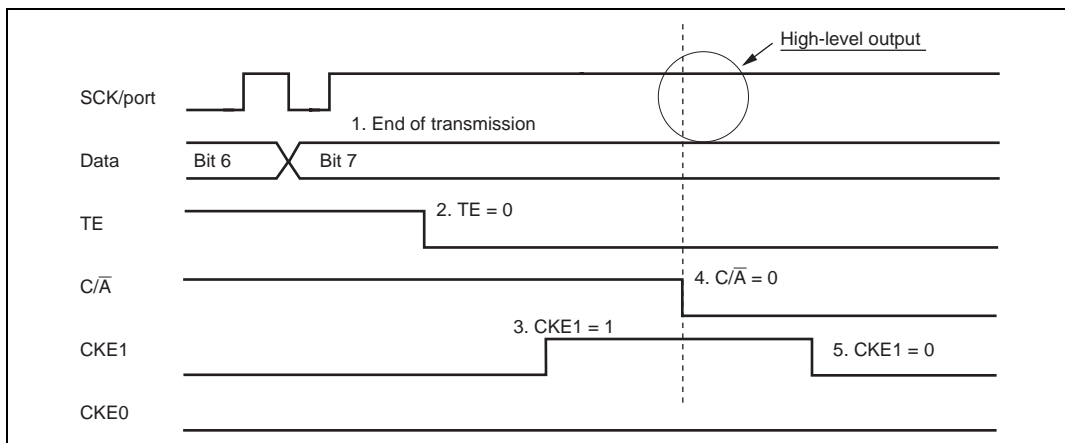
Figure 15.43 Operation when Switching from SCK Pin Function to Port Pin Function

- Sample Procedure for Avoiding Low-Level Output

As this sample procedure temporarily places the SCK pin in the input state, the SCK/port pin should be pulled up beforehand with an external circuit.

With $\text{DDR} = 1$, $\text{DR} = 1$, $\text{C}/\overline{\text{A}} = 1$, $\text{CKE1} = 0$, $\text{CKE0} = 0$, and $\text{TE} = 1$, make the following settings in the order shown.

1. End of serial data transmission
2. $\text{TE} \text{ bit} = 0$
3. $\text{CKE1} \text{ bit} = 1$
4. $\text{C}/\overline{\text{A}} \text{ bit} = 0 \dots$ Switchover to port output
5. $\text{CKE1} \text{ bit} = 0$



**Figure 15.44 Operation when Switching from SCK Pin Function to Port Pin Function
(Example of Preventing Low-Level Output)**

15.10.8 Assignment and Selection of Registers

Some serial communication interface registers are assigned to the same address as other registers. Register selection is performed by means of the IICE bit in the serial control register (SCRX). For details on register addresses, see section 26.1, Register Addresses (In Address Order).

Section 16 I²C Bus Interface (IIC) (Option)

An I²C bus interface is available as an option. Observe the following notes when using this option.

1. For masked ROM versions, a W is added to the part number in products in which this optional function is used.

Examples: HD6432239WTE

The H8S/2258 Group, H8S/2239 Group, and H8S/2238 Group have an internal I²C bus interface of two channels.

The I²C bus interface conforms to and provides a subset of the Philips I²C bus (inter-IC bus) interface functions. The register configuration that controls the I²C bus differs partly from the Philips configuration, however.

The I²C bus interface data transfer is performed using a data line (SDA) and a clock line (SCL) for each channel, which allows efficient use of connectors and the area of the PCB.

- Notes:
1. An I²C bus interface is not available in the H8S/2237 Group and H8S/2227 Group.
 2. When the power supply voltage ranges from 2.2 V to 2.7 V, the I²C bus interface is not available.

16.1 Features

- Selection of I²C bus format or clocked synchronous serial format
 - I²C bus format: addressing format with acknowledge bit, for master/slave operation
 - Clocked synchronous serial format: non-addressing format without acknowledge bit, for master operation only

I²C bus format

- Two ways of setting slave address
- Start and stop conditions generated automatically in master mode
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Wait function in master mode

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.

- Wait function in slave mode

A wait request can be generated by driving the SCL pin low after data transfer, excluding acknowledgement. The wait request is cleared when the next transfer becomes possible.
- Interrupt sources
 - Data transfer end (including transmission mode transition with I²C bus format and address reception after loss of master arbitration)
 - Address match: when any slave address matches or the general call address is received in slave receive mode
 - Start condition detection (in master mode)
 - Stop condition detection (in slave mode)
- Selection of 16 internal clocks (in master mode)
- Direct bus drive
 - Two pins, P35/SCL0 and P34/SDA0, function as NMOS open-drain outputs when the bus drive function is selected.
 - Two pins, P33/SCL1 and P32/SDA1, function as NMOS-only outputs when the bus drive function is selected.

Figure 16.1 shows a block diagram of the I²C bus interface. Figure 16.2 shows an example of I/O pin connections to external circuits. Channel I/O pins are NMOS open drains, and it is possible to apply voltages in excess of the power supply (V_{cc}) voltage for this LSI. Set the upper limit of voltage applied to the power supply (V_{cc}) power supply range +0.3 V. Channel 1 I/O pins are driven solely by NMOS, so in terms of appearance they carry out the same operations as an NMOS open drain. However, the voltage which can be applied to the I/O pins depends on the voltage of the power supply (V_{cc}) of this LSI.

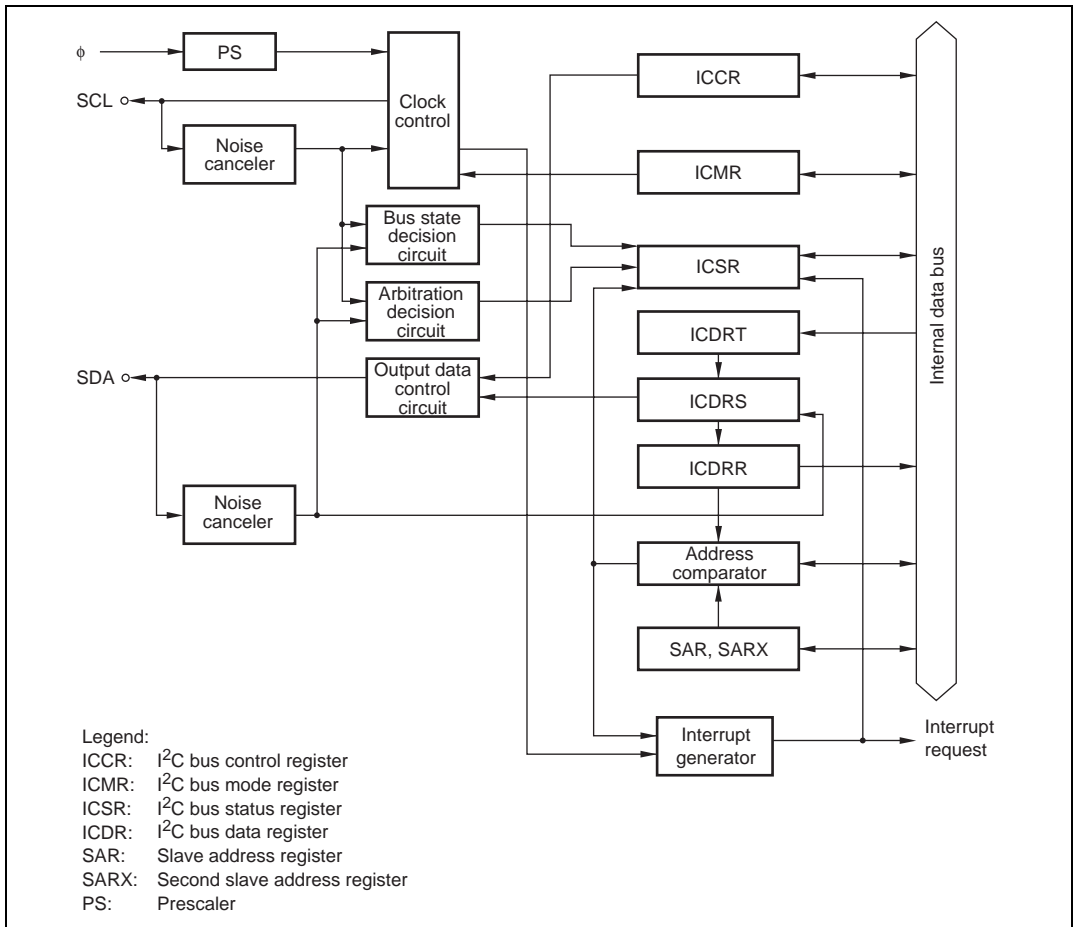


Figure 16.1 Block Diagram of I²C Bus Interface

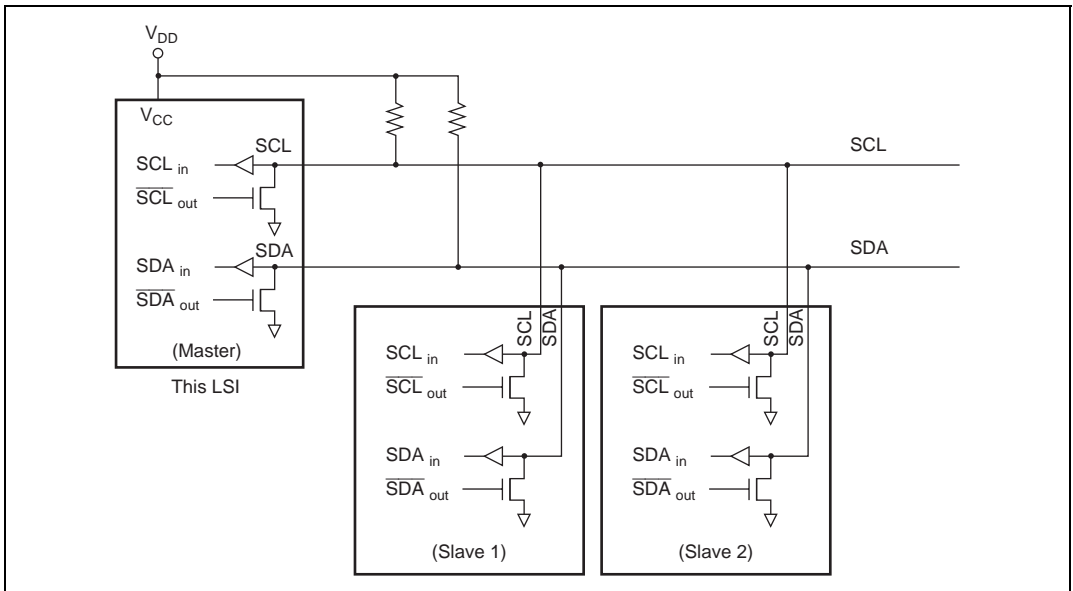


Figure 16.2 I²C Bus Interface Connections (Example: This LSI as Master)

16.2 Input/Output Pins

Table 16.1 shows the pin configuration for the I²C bus interface.

Table 16.1 Pin Configuration

| Name | Abbreviation* | I/O | Function |
|--------------|---------------|-----|---------------------------------|
| Serial clock | SCL0 | I/O | IIC_0 serial clock input/output |
| Serial data | SDA0 | I/O | IIC_0 serial data input/output |
| Serial clock | SCL1 | I/O | IIC_1 serial clock input/output |
| Serial data | SDA1 | I/O | IIC_1 serial data input/output |

Note: * Pin names SCL and SDA are used in the text for all channels, omitting the channel designation.

16.3 Register Descriptions

The I²C bus interface has the following registers. Registers ICDR and SARX and registers ICMR and SAR are allocated to the same addresses. Accessible addresses differ depending on the ICE bit in ICCR. SAR and SARX are accessed when ICE is 0, and ICMR and ICDR are accessed when

ICE is 1. For details on the module stop control register, refer to section 24.1.2, Module Stop Control Registers A to C (MSTPCRA to MSTPCRC).

- I²C bus data register_0 (ICDR_0)*
- Slave address register_0 (SAR_0)*
- Second slave address register_0 (SARX_0)*
- I²C bus mode register_0 (ICMR_0)*
- I²C bus control register_0 (ICCR_0)*
- I²C bus status register_0 (ICSR_0)*
- I²C bus data register_1 (ICDR_1)*
- Slave address register_1 (SAR_1)*
- Second slave address register_1 (SARX_1)*
- I²C bus mode register_1 (ICMR_1)*
- I²C bus control register_1 (ICCR_1)*
- I²C bus status register_1 (ICSR_1)*
- DDC switch register (DDCSWR)
- Serial control register X (SCRX)

Note: * Some of the registers in the I²C bus interface are allocated to the same addresses of other registers. The IICE bit in serial control register X (SCRX) selects each register.

16.3.1 I²C Bus Data Register (ICDR)

ICDR is an 8-bit readable/writable register that is used as a transmit data register when transmitting and a receive data register when receiving. ICDR is divided internally into a shift register (ICDRS), receive buffer (ICDRR), and transmit buffer (ICDRT). Data transfers among the three registers are performed automatically in coordination with changes in the bus state, and affect the status of internal flags such as TDRE and RDRF. When TDRE is 1 and the transmit buffer is empty, TDRE shows that the next transmit data can be written from the CPU. When RDRF is 1, it shows that the valid receive data is stored in the receive buffer.

If I²C is in transmit mode and the next data is in ICDRT (the TDRE flag is 0) following transmission/reception of one frame of data using ICDRS, data is transferred automatically from ICDRT to ICDRS. If I²C is in receive mode and no previous data remains in ICDRR (the RDRF flag is 0) following transmission/reception of one frame of data using ICDRS, data is transferred automatically from ICDRS to ICDRR.

If the number of bits in a frame, excluding the acknowledge bit, is less than 8, transmit data and receive data are stored differently. Transmit data should be written justified toward the MSB side

when $MLS = 0$, and toward the LSB side when $MLS = 1$. Receive data bits read from the LSB side should be treated as valid when $MLS = 0$, and bits read from the MSB side when $MLS = 1$.

ICDR can be written and read only when the ICE bit is set to 1 in ICCR. The value of ICDR is undefined after a reset.

The TDRE and RDRF flags are set and cleared under the conditions shown below. Setting the TDRE and RDRF flags affects the status of the interrupt flags.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| — | TDRE | — | — | Transmit Data Register Empty [Setting conditions] <ul style="list-style-type: none"> • In transmit mode, when a start condition is detected in the bus line state after a start condition is issued in master mode with the I²C bus format or serial format selected • When data is transferred from ICDRT to ICDRS • When a switch is made from receive mode to transmit mode after detection of a start condition [Clearing conditions] <ul style="list-style-type: none"> • When transmit data is written in ICDR in transmit mode • When a stop condition is detected in the bus line state after a stop condition is issued with the I²C bus format or serial format selected • When a stop condition is detected with the I²C bus format selected • In receive mode |
| — | RDRF | — | — | Receive Data Register Full [Setting condition] When data is transferred from ICDRS to ICDRR [Clearing condition] When ICDR (ICDRR) receive data is read in receive mode |

16.3.2 Slave Address Register (SAR)

SAR selects the slave address and selects the transfer format. SAR can be written and read only when the ICE bit is cleared to 0 in ICCR.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | SVA6 | 0 | R/W | Slave Address 6 to 0 |
| 6 | SVA5 | 0 | R/W | Sets a slave address. |
| 5 | SVA4 | 0 | R/W | |
| 4 | SVA3 | 0 | R/W | |
| 3 | SVA2 | 0 | R/W | |
| 2 | SVA1 | 0 | R/W | |
| 1 | SVA0 | 0 | R/W | |
| 0 | FS | 0 | R/W | Selects the transfer format together with the FSX bit in SARX. Refer to table 16.2. |

16.3.3 Second Slave Address Register (SARX)

SARX stores the second slave address and selects the transfer format. SARX can be written and read only when the ICE bit is cleared to 0 in ICCR.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | SVAX6 | 0 | R/W | Second Slave Address 6 to 0 |
| 6 | SVAX5 | 0 | R/W | Sets the second slave address. |
| 5 | SVAX4 | 0 | R/W | |
| 4 | SVAX3 | 0 | R/W | |
| 3 | SVAX2 | 0 | R/W | |
| 2 | SVAX1 | 0 | R/W | |
| 1 | SVAX0 | 0 | R/W | |
| 0 | FSX | 1 | R/W | Selects the transfer format together with the FS bit in SAR. Refer to table 16.2. |

Table 16.2 Transfer Format

| SAR | SARX | I ² C Transfer Format |
|-----|------|--|
| FS | FSX | |
| 0 | 0 | SAR and SARX are used as the slave addresses with the I ² C bus format. |
| 0 | 1 | Only SAR is used as the slave address with the I ² C bus format. |
| 1 | 0 | Only SARX is used as the slave address with the I ² C bus format. |
| 1 | 1 | Clock synchronous serial format (SAR and SARX are invalid) |

16.3.4 I²C Bus Mode Register (ICMR)

ICMR sets the transfer format and transfer rate. It can only be accessed when the ICE bit in ICCR is 1.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | MLS | 0 | R/W | MSB-First/LSB-First Select 0: MSB-first 1: LSB-first Set this bit to 0 when the I ² C bus format is used. |
| 6 | WAIT | 0 | R/W | Wait Insertion Bit This bit is valid only in master mode with the I ² C bus format. When WAIT is set to 1, after the fall of the clock for the final data bit, the IRIC flag is set to 1 in ICCR, and a wait state begins (with SCL at the low level). When the IRIC flag is cleared to 0 in ICCR, the wait ends and the acknowledge bit is transferred. If WAIT is cleared to 0, data and acknowledge bits are transferred consecutively with no wait inserted. The IRIC flag in ICCR is set to 1 on completion of the acknowledge bit transfer, regardless of the WAIT setting. |
| 5 | CKS2 | 0 | R/W | Serial Clock Select 2 to 0 |
| 4 | CKS1 | 0 | R/W | This bit is valid only in master mode. |
| 3 | CKS0 | 0 | R/W | These bits select the required transfer rate, together with the IICX 1 and IICX0 bit in SCRX. Refer to table 16.3. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 2 | BC2 | 0 | R/W | Bit Counter 2 to 0 |
| 1 | BC1 | 0 | R/W | These bits specify the number of bits to be transferred next. With the I ² C bus format, the data is transferred with one addition acknowledge bit. Bit BC2 to BC0 settings should be made during an interval between transfer frames. If bits BC2 to BC0 are set to a value other than 000, the setting should be made while the SCL line is low. The value returns to 000 at the end of a data transfer, including the acknowledge bit. I ² C Bus Format Clocked Synchronous Mode 000: 9 bits 000: 8 bits 001: 2 bits 001: 1 bit 010: 3 bits 010: 2 bits 011: 4 bits 011: 3 bits 100: 5 bits 100: 4 bits 101: 6 bits 101: 5 bits 110: 7 bits 110: 6 bits 111: 8 bits 111: 7 bits |
| 0 | BC0 | 0 | R/W | |

Table 16.3 I²C Transfer Rate

| SCRX | | ICMR | | | Transfer Rate | | | | |
|-------------|-------|-------|-------|------------|-----------------------------|-----------------------------|-------------------------|------------------------------|------------------------------|
| Bit 5 and 6 | Bit 5 | Bit 4 | Bit 3 | | | | | | |
| IICX | CKS2 | CKS1 | CKS0 | Clock | $\phi = 5 \text{ MHz}^{*3}$ | $\phi = 8 \text{ MHz}^{*3}$ | $\phi = 10 \text{ MHz}$ | $\phi = 16 \text{ MHz}^{*2}$ | $\phi = 20 \text{ MHz}^{*2}$ |
| 0 | 0 | 0 | 0 | $\phi/28$ | 179 MHz | 286 kHz | 357 kHz | 571 kHz ^{*1} | 714 kHz ^{*1} |
| 0 | 0 | 0 | 1 | $\phi/40$ | 125 kHz | 200 kHz | 250 kHz | 400 kHz | 500 kHz ^{*1} |
| 0 | 0 | 1 | 0 | $\phi/48$ | 104 kHz | 167 kHz | 208 kHz | 333 kHz | 417 kHz ^{*1} |
| 0 | 0 | 1 | 1 | $\phi/64$ | 78.1 kHz | 125 kHz | 156 kHz | 250 kHz | 313 kHz |
| 0 | 1 | 0 | 0 | $\phi/80$ | 62.5 kHz | 100 kHz | 125 kHz | 200 kHz | 250 kHz |
| 0 | 1 | 0 | 1 | $\phi/100$ | 50.0 kHz | 80.0 kHz | 100 kHz | 160 kHz | 200 kHz |
| 0 | 1 | 1 | 0 | $\phi/112$ | 44.6 kHz | 71.4 kHz | 89.3 kHz | 143 kHz | 179 kHz |
| 0 | 1 | 1 | 1 | $\phi/128$ | 39.1 kHz | 62.5 kHz | 78.1 kHz | 125 kHz | 156 kHz |
| 1 | 0 | 0 | 0 | $\phi/56$ | 89.3 kHz | 143 kHz | 179 kHz | 286 kHz | 357 kHz |
| 1 | 0 | 0 | 1 | $\phi/80$ | 62.5 kHz | 100 kHz | 125 kHz | 200 kHz | 250 kHz |
| 1 | 0 | 1 | 0 | $\phi/96$ | 52.1 kHz | 83.3 kHz | 104 kHz | 167 kHz | 208 kHz |
| 1 | 0 | 1 | 1 | $\phi/128$ | 39.1 kHz | 62.5 kHz | 78.1 kHz | 125 kHz | 156 kHz |
| 1 | 1 | 0 | 0 | $\phi/160$ | 31.3 kHz | 50.0 kHz | 62.5 kHz | 100 kHz | 125 kHz |
| 1 | 1 | 0 | 1 | $\phi/200$ | 25.0 kHz | 40.0 kHz | 50.0 kHz | 80.0 kHz | 100 kHz |
| 1 | 1 | 1 | 0 | $\phi/224$ | 22.3 kHz | 35.7 kHz | 44.6 kHz | 71.4 kHz | 89.3 kHz |
| 1 | 1 | 1 | 1 | $\phi/256$ | 19.5 kHz | 31.3 kHz | 39.1 kHz | 62.5 kHz | 78.1 kHz |

Notes: 1. Out of the range of the I²C bus interface specification (normal mode: 100 kHz in max, and high-speed mode: 400 kHz in max).

2. Supported only by the H8S/2239 Group.

3. The H8S/2258 Group is out of operation.

16.3.5 Serial Control Register X (SCRX)

SCRX controls the IIC operating modes.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | — | 0 | R/W | Reserved The initial value should not be changed. |
| 6 | IICX1 | 0 | R/W | I ² C Transfer Rate Select 1 and 0 |
| 5 | IICX0 | 0 | R/W | Selects the transfer rate in master mode, together with bits CKS2 to CKS0 in ICMR. Refer to table 16.3. IICX1 controls IIC_1 and IICX0 controls IIC_0. |
| 4 | IICE | 0 | R/W | I ² C Master Enable Controls CPU access to the IIC data register and control registers (ICCR, ICSR, ICDR/SARX, and ICMR/SAR). 0: CPU access to the IIC data register and control registers is disabled. 1: CPU access to the IIC data register and control registers is enabled. |
| 3 | FLSHE | 0 | R/W | For details on this bit, refer to section 20.5.7, Serial Control Register X (SCRX). |
| 2 to 0 | — | All 0 | R/W | Reserved The initial value should not be changed. |

16.3.6 I²C Bus Control Register (ICCR)

I²C bus control register (ICCR) consists of the control bits and interrupt request flags of I²C bus interface.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | ICE | 0 | R/W | <p>I²C Bus Interface Enable</p> <p>When this bit is set to 1, the I²C bus interface module is enabled to send/receive data and drive the bus since it is connected to the SCL and SDA pins. ICMR and ICDR can be accessed.</p> <p>SCL and SDA output is disabled (and input to SCL and SDA is enabled) when this bit is cleared to 0. SAR and SARX can be accessed.</p> |
| 6 | IEIC | 0 | R/W | <p>I²C Bus Interface Interrupt Enable</p> <p>When this bit is 1, interrupts are enabled by IRIC.</p> |
| 5 | MST | 0 | R/W | Master/Slave Select |
| 4 | TRS | 0 | | <p>Transmit/Receive Select</p> <p>00: Slave receive mode</p> <p>01: Slave transmit mode</p> <p>10: Master receive mode</p> <p>11: Master transmit mode</p> <p>Both these bits will be cleared by hardware when they lose in a bus contention in master mode of the I²C bus format. In slave receive mode, the R/\overline{W} bit in the first frame immediately after the start automatically sets these bits in receive mode or transmit mode by using hardware. The settings can be made again for the bits that were set/cleared by hardware, by reading these bits. When the TRS bit is intended to change during a transfer, the bit will not be switched until the frame transfer is completed, including acknowledgement.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 3 | ACKE | 0 | R/W | <p>Acknowledge Bit Judgement Selection</p> <p>0: The value of the acknowledge bit is ignored, and continuous transfer is performed. The value of the received acknowledge bit is not indicated by the ACKB bit, which is always 0.</p> <p>1: If the acknowledge bit is 1, continuous transfer is interrupted.</p> <p>In this LSI, the DTC can be used to perform continuous transfer. The DTC is activated when the IRTR interrupt flag is set to 1 (IRTR is one of two interrupt flags, the other being IRIC). When the ACKE bit is 0, the TDRE, IRIC, and IRTR flags are set on completion of data transmission, regardless of the acknowledge bit. When the ACKE bit is 1, the TDRE, IRIC, and IRTR flags are set on completion of data transmission when the acknowledge bit is 0, and the IRIC flag alone is set on completion of data transmission when the acknowledge bit is 1.</p> <p>When the DTC is activated, the TDRE, IRIC, and IRTR flags are cleared to 0 after the specified number of data transfers have been executed. Consequently, interrupts are not generated during continuous data transfer, but if data transmission is completed with a 1 acknowledge bit when the ACKE bit is set to 1, the DTC is not activated and an interrupt is generated, if enabled.</p> <p>Depending on the receiving device, the acknowledge bit may be significant, in indicating completion of processing of the received data, for instance, or may be fixed at 1 and have no significance.</p> |
| 2 | BBSY | 0 | R/W | <p>Bus Busy</p> <p>In slave mode, reading the BBSY flag enables to confirm whether the I²C bus is occupied or released. The BBSY flag is set to 0 when the SDA level changes from high to low under the condition of SCL = high, assuming that the start condition has been issued. The BBSY flag is cleared to 0 when the SDA level changes from low to high under the condition of SCL = high, assuming that the start condition has been issued. Writing to the BBSY flag in slave mode is disabled.</p> <p>In master mode, the BBSY flag is used to issue start and stop conditions. Write 1 to BBSY and 0 to SCP to issue a start condition. Follow this procedure when also re-transmitting a start condition. To issue a start/stop condition, use the MOV instruction. The I²C bus interface must be set in master transmit mode before the issue of a start condition.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 1 | IRIC | 0 | R/W | <p>I²C Bus Interface Interrupt Request Flag Also see table 16.4.</p> <p>[Setting conditions]</p> <p>In I²C bus format master mode</p> <ul style="list-style-type: none"> When a start condition is detected in the bus line state after a start condition is issued (when the TDRE flag is set to 1 because of first frame transmission) When a wait is inserted between the data and acknowledge bit when WAIT = 1 At the end of data transfer (when the TDRE or RDRF flag is set to 1) When a slave address is received after bus arbitration is lost (when the AL flag is set to 1) When 1 is received as the acknowledge bit when the ACKE bit is 1 (when the ACKB bit is set to 1) <p>In I²C bus format slave mode</p> <ul style="list-style-type: none"> When the slave address (SVA, SVAX) matches (when the AAS and AASX flags are set to 1) and at the end of data transfer up to the subsequent retransmission start condition or stop condition detection (when the TDRE or RDRF flag is set to 1) When the general call address (one frame including a R/W bit is H'00) is detected (when the ADZ flag is set to 1) and at the end of data transfer up to the subsequent retransmission start condition or stop condition detection (when the TDRE or RDRF flag is set to 1) When 1 is received as the acknowledge bit when the ACKE bit is 1 (when the ACKB bit is set to 1) When a stop condition is detected (when the STOP or ESTP flag is set to 1) <p>With clocked synchronous serial format</p> <ul style="list-style-type: none"> At the end of data transfer (when the TDRE or RDRF flag is set to 1) When a start condition is detected with serial format selected <p>When a condition occurs in which internal flag of TDRE and RDRF is set to 1 except for the above</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is written in IRIC after reading IRIC = 1 When ICDR is read/written by DTC <p>(When TDRE or RDRF flag is cleared to 0) (AS it might not be a condition to clear, for details, see section 16.4.8, Operation Using the DTC).</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 0 | SCP | 1 | W | <p>Start Condition/Stop Condition Prohibit bit</p> <p>The SCP bit controls the issue of start/stop conditions in master mode.</p> <p>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.</p> |

When, with the I²C bus format selected, IRIC is set to 1 and an interrupt is generated, other flags must be checked in order to identify the source that set IRIC to 1. Although each source has a corresponding flag, caution is needed at the end of a transfer.

When the TDRE or RDRF internal flag is set, the readable IRTR flag may or may not be set. Even when data transfer is complete, the DTC activation request flag, IRTR, is not set until a retransmission start condition or stop condition is detected after a slave address (SVA) or general call address matched in the I²C bus format slave mode.

Even when the IRIC flag and IRTR flag are set, the TDRE or RDRF internal flag may not be set. For a continuous transfer using the DTC, the IRIC or IRTR flag is not cleared at the completion of the specified number of times of transfers. On the other hand, the TDRE and RDRF flags are cleared because the specified number of times of read/write operations have been complete.

Table 16.4 shows the relationship between the flags and the transfer states.

Table 16.4 Flags and Transfer States

| MST | TRS | BBSY | ESTP | STOP | IRTR | AASX | AL | AAS | ADZ | ACKB | State |
|-----|-----|------|------|------|------|------|----|-----|-----|------|---|
| 1/0 | 1/0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Idle state (flag clearing required) |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | Start condition issuance |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | Start condition established |
| 1 | 1/0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0/1 | Master mode wait |
| 1 | 1/0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0/1 | Master mode transmit/receive end |
| 0 | 0 | 1 | 0 | 0 | 0 | 1/0 | 1 | 1/0 | 1/0 | 0 | Arbitration lost |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | SAR match by first frame in slave mode |
| 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | General call address match |
| 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | SARX match |
| 0 | 1/0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0/1 | Slave mode transmit/receive end (except after SARX match) |
| 0 | 1/0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | Slave mode transmit/receive end (after SARX match) |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | |
| 0 | 1/0 | 0 | 1/0 | 1/0 | 0 | 0 | 0 | 0 | 0 | 0/1 | Stop condition detected |

16.3.7 I²C Bus Status Register (ICSR)

ICSR consists of status flags.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|--------|--|
| 7 | ESTP | 0 | R/(W)* | <p>Error Stop Condition Detection Flag</p> <p>This bit is valid in I²C bus format slave mode.</p> <p>[Setting condition]</p> <p>When a stop condition is detected during frame transfer</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is written in ESTP after reading the state of 1 When the IRIC flag is cleared to 0 |
| 6 | STOP | 0 | R/(W)* | <p>Normal Stop Condition Detection Flag</p> <p>This bit is valid in I²C bus format slave mode.</p> <p>[Setting condition]</p> <p>When a stop condition is detected during frame transfer</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is written in STOP after reading STOP = 1 When the IRIC flag is cleared to 0 |
| 5 | IRTR | 0 | R/(W)* | <p>I²C Bus Interface Continuous Transmission/Reception Interrupt Request Flag</p> <p>[Setting conditions]</p> <p>In I²C bus interface slave mode</p> <ul style="list-style-type: none"> When the TDRE or RDRF flag is set to 1 when AASX = 1 <p>In I²C bus interface other modes</p> <ul style="list-style-type: none"> When the TDRE or RDRF flag is set to 1 <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is written in IRTR after reading IRTR = 1 When the IRIC flag is cleared to 0 while ICE is 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|--------|---|
| 4 | AASX | 0 | R/(W)* | <p>Second Slave Address Recognition Flag</p> <p>[Setting condition]</p> <p>When the second slave address is detected in slave receive mode and FSX = 0</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written in AASX after reading AASX = 1 • When a start condition is detected • In master mode |
| 3 | AL | 0 | R/(W)* | <p>Arbitration Lost Flag</p> <p>Indicates that bus arbitration was lost in master mode.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • When the internal SDA and SDA pin do not match at the rise of SCL • When the internal SCL is high at the fall of SCL <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written in AL after reading AL = 1 • When ICDR data is written (transmit mode) or read (receive mode) |
| 2 | AAS | 0 | R/(W)* | <p>Slave Address Recognition Flag</p> <p>[Setting condition]</p> <p>When the slave address or general call address (one frame including a R/W bit is H'00) is detected in slave receive mode and FS = 0</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When ICDR data is written (transmit mode) or read (receive mode) • When 0 is written in AAS after reading AAS = 1 • In master mode |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|--------|---|
| 1 | ADZ | 0 | R/(W)* | <p>General Call Address Recognition Flag</p> <p>In I²C bus format slave receive mode, this flag is set to 1 if the first frame following a start condition is the general call address (H'00).</p> <p>[Setting condition]</p> <p>When the general call address (one frame including a R/\overline{W} bit is H'00) is detected in slave receive mode and (FS = 0 or FSX = 0)</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When ICDR is written to (transmit mode) or read from (receive mode) • When 0 is written in ADZ after reading ADZ = 1 • In master mode <p>If a general call address is detected while FS = 1 and FSX = 0, the ADZ flag is set to 1; however, the general call address is not recognized (AAS flag is not set to 1).</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 0 | ACKB | 0 | R/W | <p>Acknowledge Bit</p> <p>Stores acknowledge data.</p> <p>Transmit mode:</p> <p>[Setting condition]</p> <p>When 1 is received as the acknowledge bit when ACKE = 1 in transmit mode</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> When 0 is received as the acknowledge bit when ACKE = 1 in transmit mode When 0 is written to the ACKE bit <p>Receive mode:</p> <p>0: Returns 0 as acknowledge data after data reception</p> <p>1: Returns 1 as acknowledge data after data reception</p> <p>When this bit is read, the value loaded from the bus line (returned by the receiving device) is read in transmission (when TRS = 1). In reception (when TRS = 0), the value set by internal software is read.</p> <p>When this bit is written, acknowledge data that is returned after receiving is rewritten regardless of the TRS value. If bit in ICSR is written using bit-manipulation instructions, the acknowledge data should be re-set since the acknowledge data setting is rewritten by the ACKB bit reading value.</p> <p>Write the ACKE bit to 0 to clear the ACKB flag to 0, before transmission is ended and a stop condition is issued in master mode, or before transmission is ended and SDA is released to issue a stop condition by a master device.</p> |

Note: * Only a 0 can be written to this bit, to clear the flag.

16.3.8 DDC Switch Register (DDCSWR)

DDCSWR controls the I²C bus interface format automatic switching function and internal latch clear.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|--------|---|
| 7 to 4 | — | All 0 | R/(W)* | Reserved The write value should always be 0. |
| 3 | CLR3 | 1 | W | I ² C Bus Interface Clear 3 to 0 |
| 2 | CLR2 | 1 | W | When bits CLR3 to CLR0 are set, a clear signal is generated for the I ² C bus interface internal latch circuit, and the internal state is initialized. The write data for these bits is not retained. To perform I ² C clearance, bits CLR3 to CLR0 must be written to simultaneously using an MOV instruction. Do not use a bit manipulation instruction such as BCLR. 00xx: Setting prohibited 0100: Setting prohibited 0101: IIC_0 internal latch cleared 0110: IIC_1 internal latch cleared 0111: IIC_0, IIC_1 internal latch cleared 1xxx: Invalid setting |
| 1 | CLR1 | 1 | W | |
| 0 | CLR0 | 1 | W | |
| | | | | |

Legend:

×: Don't care

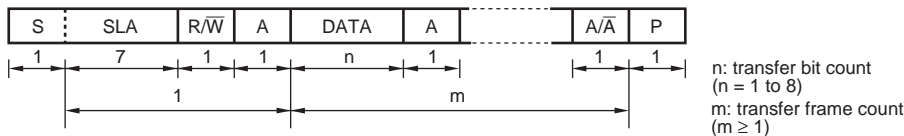
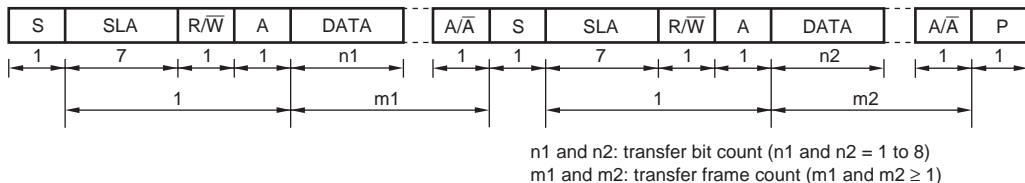
Note: * Only 0 can be written to these bits, to clear the flag.

16.4 Operation

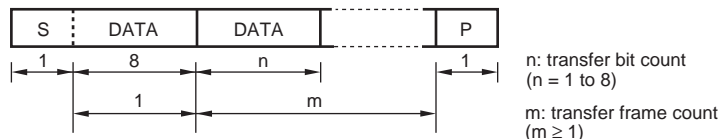
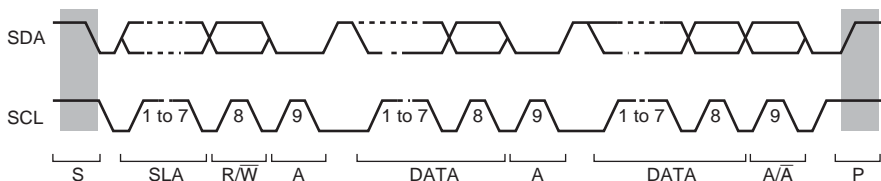
The I²C bus interface has serial and I²C bus formats.

16.4.1 I²C Bus Data Format

The I²C bus formats are addressing formats and an acknowledge bit is inserted. The first frame following a start condition always consists of 8 bits. The I²C bus format is shown in figure 16.3. The clocked synchronous serial format is a non-addressing format with no acknowledge bit. This is shown in figure 16.4. Figure 16.5 shows the I²C bus timing.

(a) I²C bus format (FS = 0 or FSX = 0)

(b) I²C bus format (start condition retransmission, FS = 0 or FSX = 0)

Figure 16.3 I²C Bus Data Formats (I²C Bus Formats)

FS = 1 and FSX = 1


Figure 16.4 I²C Bus Data Format (Serial Format)

Figure 16.5 I²C Bus Timing

Legend:

- S: Start condition. The master device drives SDA from high to low while SCL is high
- SLA: Slave address
- R/W: Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0
- A: Acknowledge. The receiving device drives SDA
- DATA: Transferred data
- P: Stop condition. The master device drives SDA from low to high while SCL is high

16.4.2 Initial Setting

At startup the following procedure is used to initialize the IIC.

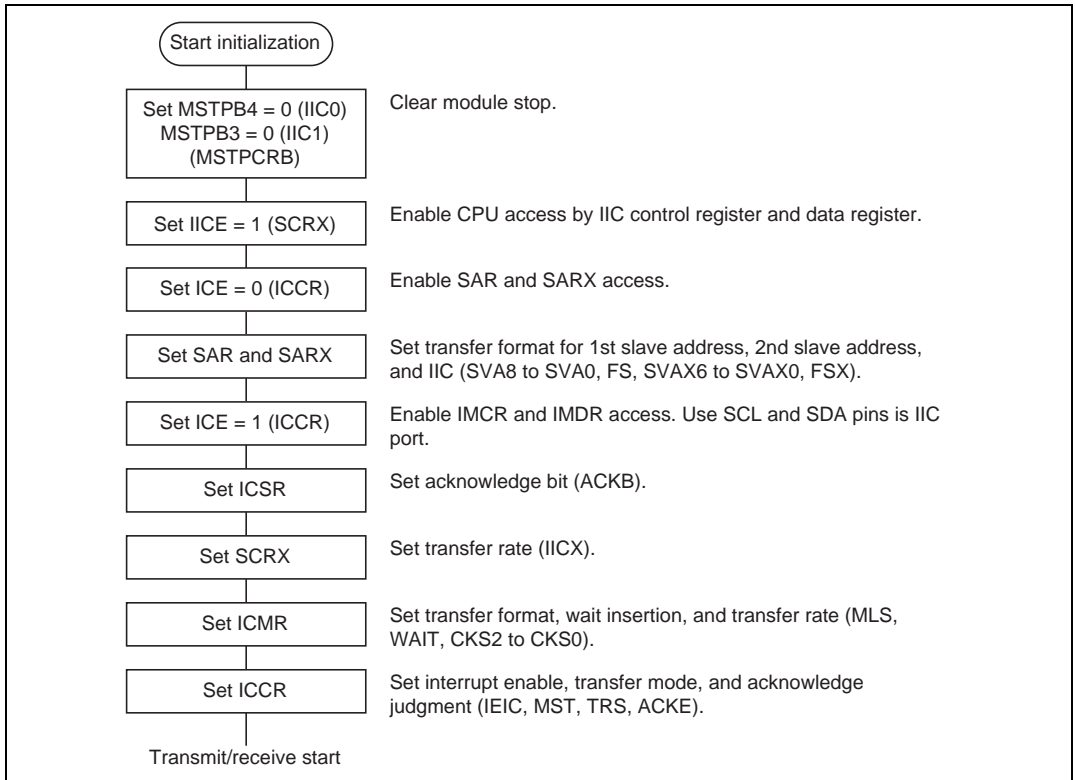


Figure 16.6 Flowchart for IIC Initialization (Example)

Note: The ICMR register should be written to only after transmit or receive operations have completed.

Writing to the ICMR register while a transmit or receive operation is in progress could cause an erroneous value to be written to bit counter bits BC2 to BC0. This could result in improper operation.

16.4.3 Master Transmit Operation

In I²C bus format master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal.

Figure 16.7 is a flowchart showing an example of the master transmit mode.

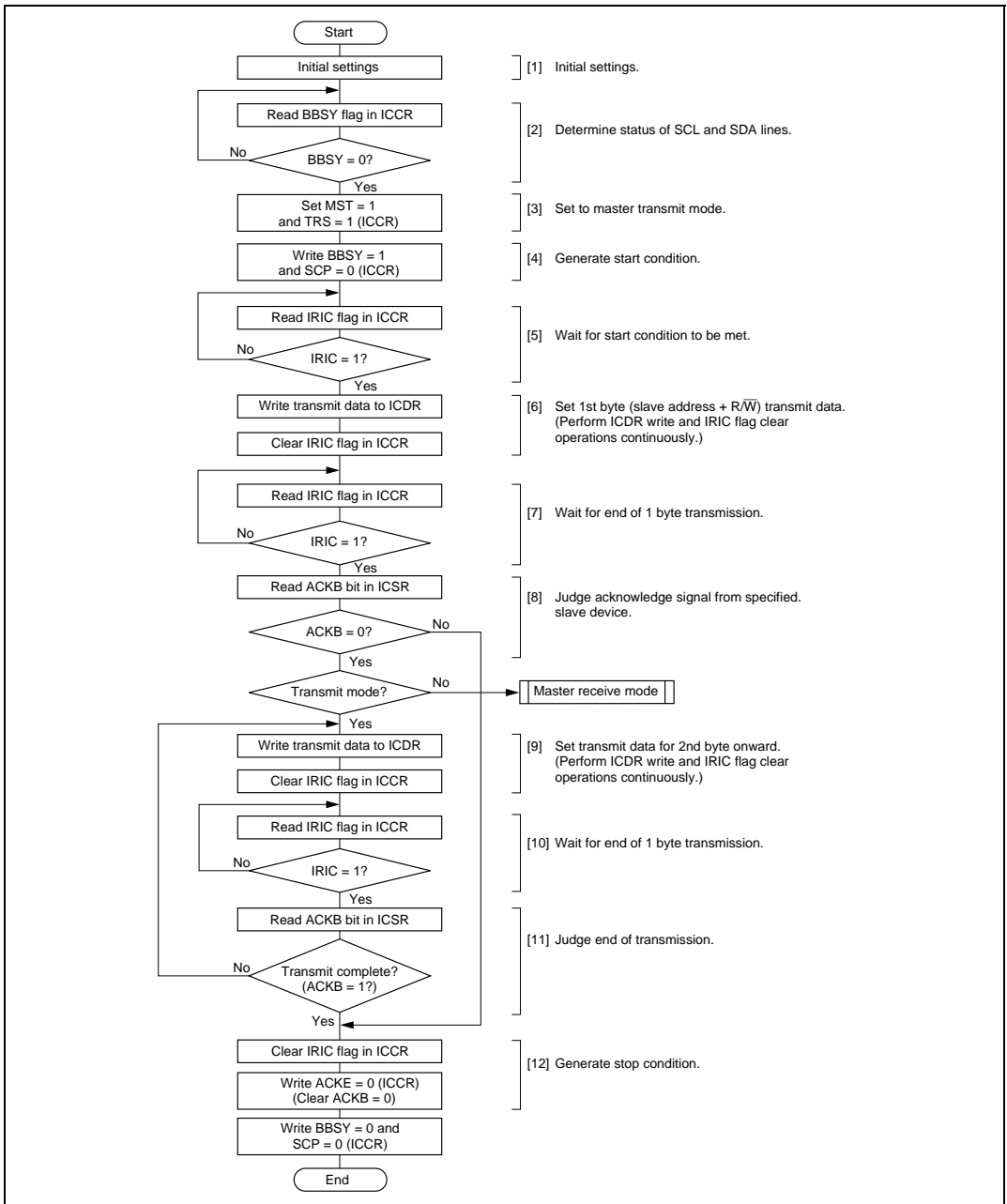


Figure 16.7 Flowchart for Master Transmit Mode (Example)

The procedure for transmitting data sequentially, synchronized with ICDR (ICDRT) write operations, is described below.

- [1] Perform initial settings as described in section 16.4.2, Initial Setting.
- [2] Read the BBSY flag in ICCR to confirm that the bus is free.
- [3] Set bits MST and TSR in ICCR to 1 to switch to the master transmit mode.
- [4] Write 1 to BBSY and 0 to SCP in ICCR. This changes SDA from high to low when SCL is high, and generates the start condition.
- [5] The IRIC and IRTR flags are set to 1 when the start condition is generated. If the IEIC bit in ICCR has been set to 1, an interrupt request is sent to the CPU.
- [6] After the start condition is detected, write the data (slave address + R/\overline{W}) to ICDR. With the I²C bus format (when the FS bit in SAR or the FSX bit in SARX is 0), the first frame data following the start condition indicates the 7-bit slave address and transmit/receive direction (R/\overline{W}). Next, clear the IRIC flag to 0 to indicate the end of the transfer. Continue successively writing to ICDR and clearing the IRIC flag to ensure that processing of other interrupts does not intervene. If the time required to transmit one byte of data elapses by the time the IRIC flag is cleared, it will not be possible to determine the end of the transmission. The master device sequentially sends the transmit clock and the data written to ICDR using the timing shown in figure 16.8. The selected slave device (i.e., the slave device with the matching slave address) drives SDA low at the 9th transmit clock pulse and returns an acknowledge signal.
- [7] When one frame of data has been transmitted, the IRIC flag is set to 1 at the rise of the 9th transmit clock pulse. After one frame has been transmitted, SCL is automatically fixed low in synchronization with the internal clock until the next transmit data is written.
- [8] Read the ACKB bit in ICSR to confirm that its value is 0. If the slave device has not returned an acknowledge signal and the value of ACKB is 1, perform the transmit end processing described in step [12] and then recommence the transmit operation from the beginning.
- [9] Write the transmit data to ICDR. Next, clear the IRIC flag to 0 to indicate the end of the transfer. Then continue successively writing to ICDR and clearing the IRIC flag as described in step [6]. Transmission of the next frame is synchronized with the internal clock.
- [10] When one frame of data has been transmitted, the IRIC flag is set to 1 at the rise of the 9th transmit clock pulse. After one frame has been transmitted, SCL is automatically fixed low in synchronization with the internal clock until the next transmit data is written.
- [11] Read the ACKB bit in ICSR to confirm that the slave device has returned an acknowledge signal and the value of ACKB is 0. If the slave device has not returned an acknowledge signal and the value of ACKB is 1, perform the transmit end processing described in step [12].
- [12] Clear the IRIC flag to 0. Write 0 to BBSY and SCP in ICCR. This changes SDA from low to high when SCL is high, and generates the stop condition.

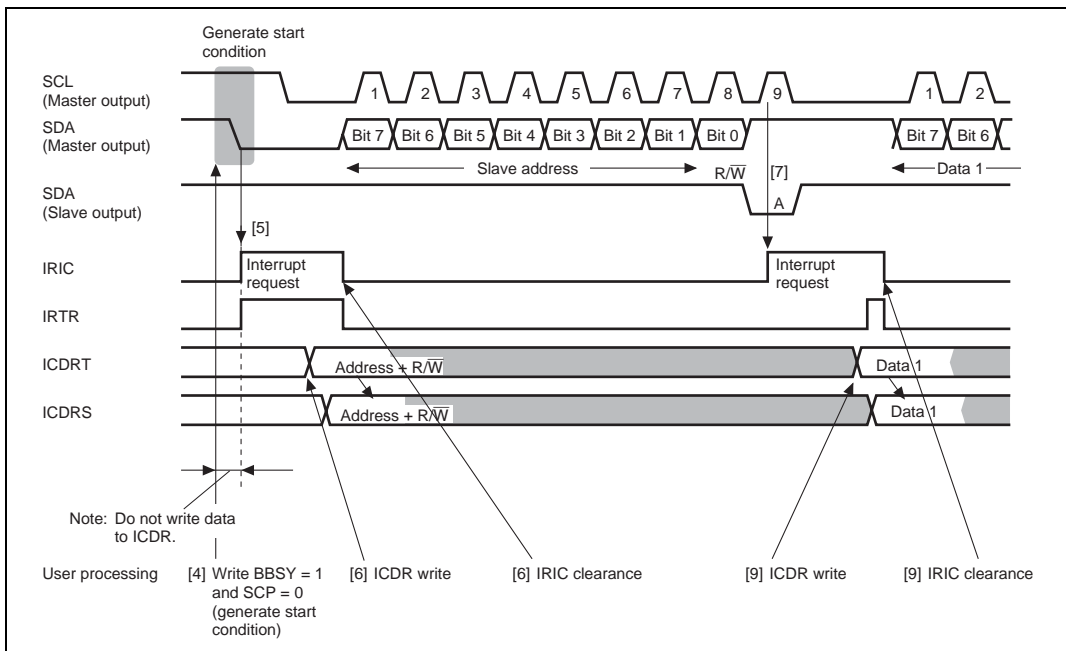


Figure 16.8 Example of Master Transmit Mode Operation Timing (MLS = WAIT = 0)

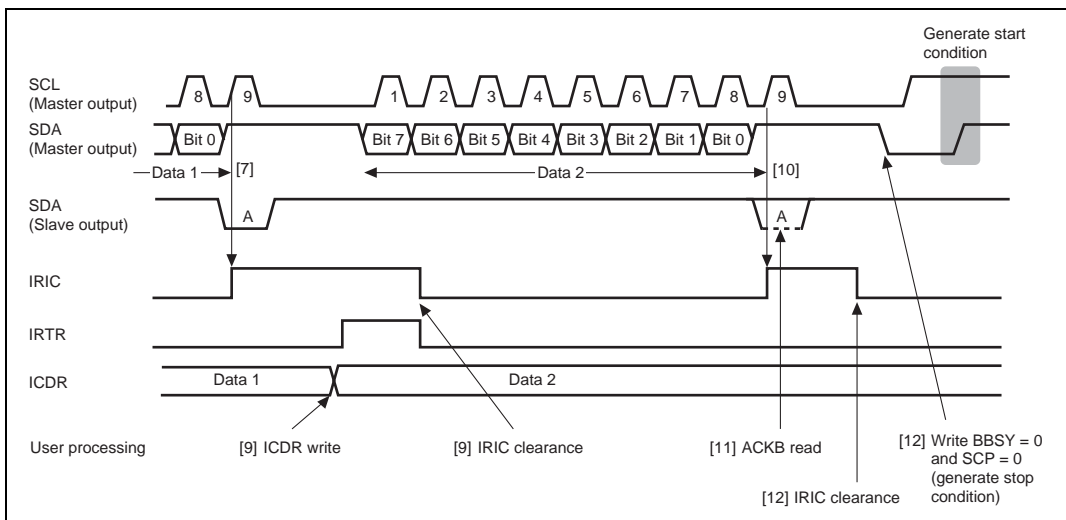


Figure 16.9 Example of Master Transmit Mode Stop Condition Generation Timing (MLS = WAIT = 0)

16.4.4 Master Receive Operation

In I²C bus format master receive mode, the master device outputs the receive clock, receives data, and returns an acknowledge signal. The slave device transmits data.

The master device transmits the data containing the slave address + $\overline{R/\overline{W}}$ (0: read) in the 1st frame after a start condition is generated in the master transmit mode. After the slave device is selected the switch to receive operation takes place.

(1) Receive Operation Using Wait States

Figures 16.10 and 16.11 are flowcharts showing examples of the master receive mode (WAIT = 1).

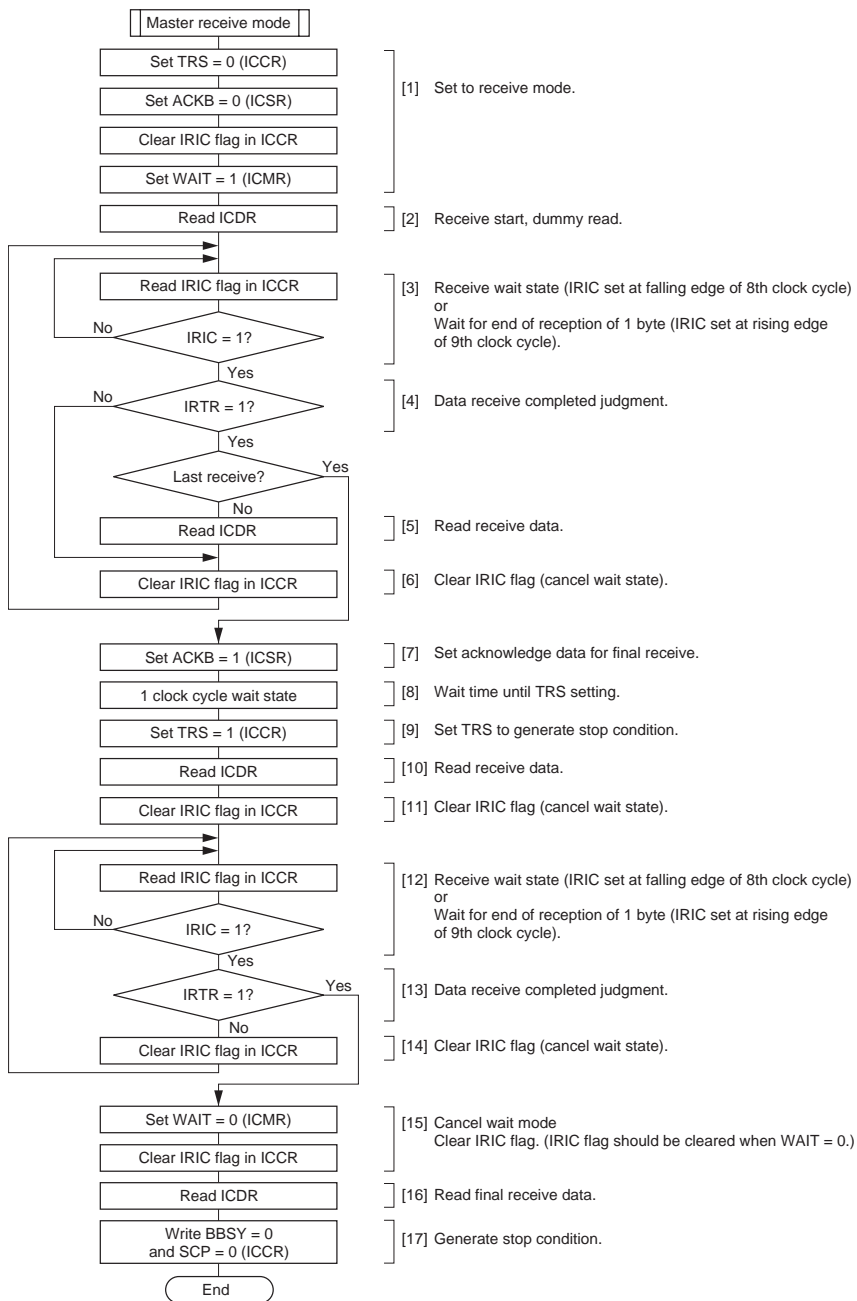


Figure 16.10 Flowchart for Master Receive Mode (Receiving Multiple Bytes) (WAIT = 1) (Example)

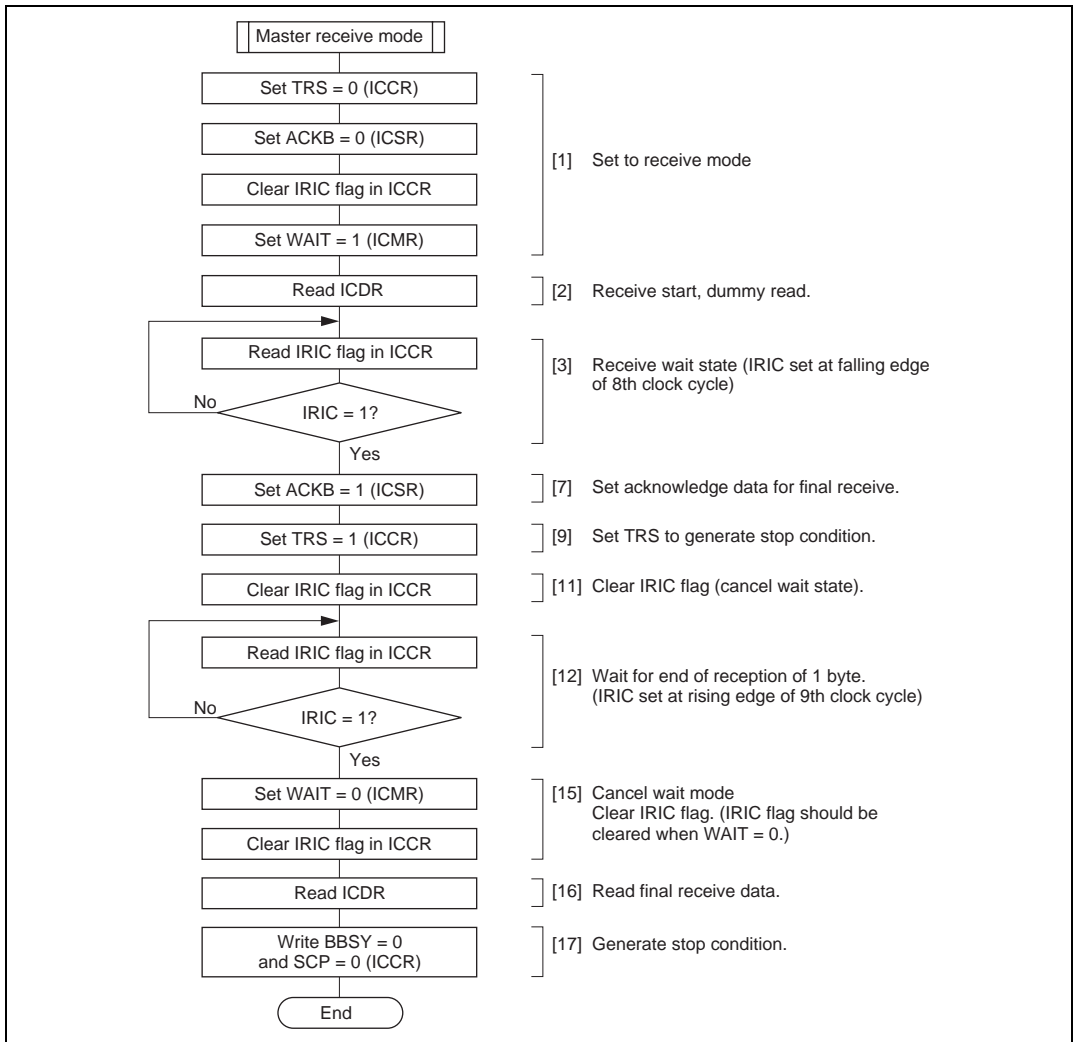


Figure 16.11 Flowchart for Master Receive Mode (Receiving 1 Byte) (WAIT = 1) (Example)

The procedure for receiving data sequentially, using the wait states (WAIT bit) for synchronization with ICDR (ICDRR) read operations, is described below.

The procedure below describes the operation for receiving multiple bytes. Note that some of the steps are omitted when receiving only 1 byte. Refer to figure 16.11 for details.

- [1] Clear the TRS bit in ICCR to 0 to switch from transmit mode to receive mode. Clear the ACKB bit in ICSR to 0 (acknowledge data setting). Clear the IRIC flag to 0, then set the WAIT bit in ICMR to 1.
- [2] When ICDR is read (dummy data read), reception is started, and the receive clock is output, and data received, in synchronization with the internal clock.
- [3] The IRIC flag is set to 1 by the following two conditions. At that point, an interrupt request is issued to the CPU if the IEIC bit in ICCR is set to 1.
 - (1) The flag is set at the falling edge of the 8th clock cycle of the receive clock for 1 frame. SCL is automatically held low, in synchronization with the internal clock, until the IRIC flag is cleared.
 - (2) The flag is set at the rising edge of the 9th clock cycle of the receive clock for 1 frame. The IRTR flag is set to 1, indicating that reception of 1 frame of data has ended. The master device continues to output the receive clock for the receive data.
- [4] Read the IRTR flag in ICSR. If the IRTR flag value is 0, the wait state is cancelled by clearing the IRIC flag as described in step [6] below. If the IRTR flag value is 1 and the next receive data is the final receive data, perform the end processing described in step [7] below.
- [5] If the IRTR flag value is 1, read the ICDR receive data.
- [6] Clear the IRIC flag to 0. The reading of the ICDR flag described in step [5] and the clearing of the IRIC flag to 0 should be performed consecutively, with no interrupt processing occurring between them. During wait operation, clear the IRIC flag to 0 when the value of counter BC2 to BC0 is 2 or greater. If the IRIC flag is cleared to 0 when the value of counter BC2 to BC0 is 1 or 0, it will not be possible to determine when the transfer has completed. If condition [3]-1 is true, the master device drives SDA to low level and returns an acknowledge signal when the receive clock outputs the 9th clock cycle. Further data can be received by repeating steps [3] through [6].
- [7] Set the ACKB bit in ICSR to 1 to set the acknowledge data for the final receive.
- [8] Wait for at least 1 clock cycle after the IRIC flag is set to 1 and then wait for the rising edge of the 1st clock cycle of the next receive data.
- [9] Set the TSR bit in ICCR to 1 to switch from the receive mode to the transmit mode. The TSR bit setting value at this point becomes valid when the rising edge of the next 9th clock cycle is input.
- [10] Read the ICDR receive data.
- [11] Clear the IRIC flag to 0. As in step [6], read the ICDR flag and clear the IRIC flag to 0 consecutively, with no interrupt processing occurring between them. During wait operation, clear the IRIC flag to 0 when the value of counter BC2 to BC0 is 2 or greater.
- [12] The IRIC flag is set to 1 by the following two conditions.

- (1) The flag is set at the falling edge of the 8th clock cycle of the receive clock for 1 frame.
SCL is automatically held low, in synchronization with the internal clock, until the IRIC flag is cleared.
 - (2) The flag is set at the rising edge of the 9th clock cycle of the receive clock for 1 frame.
The IRTR flag is set to 1, indicating that reception of 1 frame of data has ended. The master device continues to output the receive clock for the receive data.
- [13] Read the IRTR flag in ICSR. If the IRTR flag value is 0, the wait state is cancelled by clearing the IRIC flag as described in step [14] below. If the IRTR flag value is 1 and the receive operation has finished, perform the issue stop condition processing described in step [15] below.
 - [14] If the IRTR flag value is 0, clear the IRIC flag to 0 to cancel the wait state. Return to reading the IRIC flag, as described in step [12], to detect the end of the receive operation.
 - [15] Clear the WAIT bit in ICMR to 0 to cancel the wait mode. Then clear the IRIC flag to 0. The IRIC flag should be cleared when the value of WAIT is 0. (The stop condition may not be output properly when the issue stop condition instruction is executed if the WAIT bit was cleared to 0 after the IRIC flag is cleared to 0.)
 - [16] Read the final receive data in ICDR.
 - [17] Write 0 to BBSY and SCP in ICCR. This changes SDA from low to high when SCL is high, and generates the stop condition.

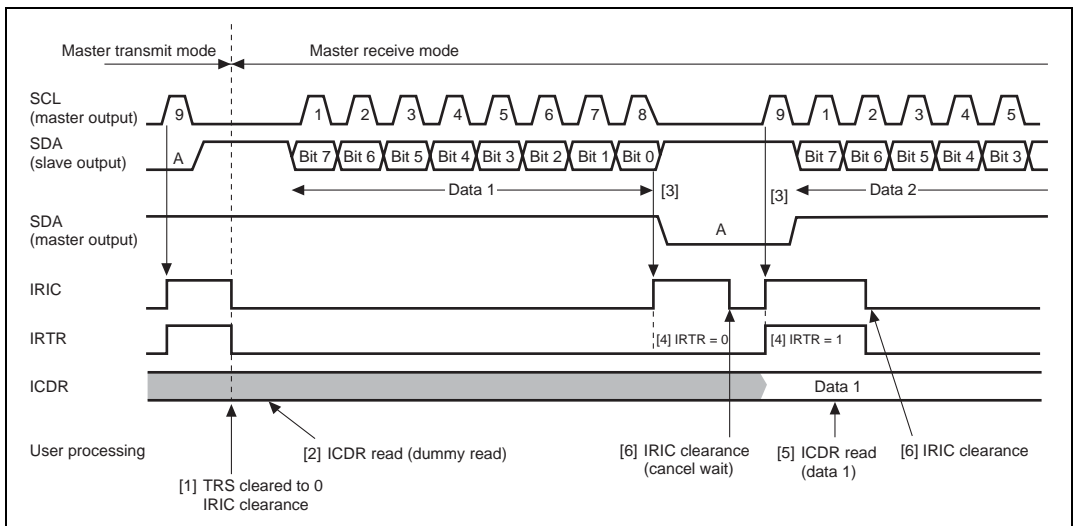


Figure 16.12 Example of Master Receive Mode Operation Timing
(MLS = ACKB = 0, WAIT = 1)

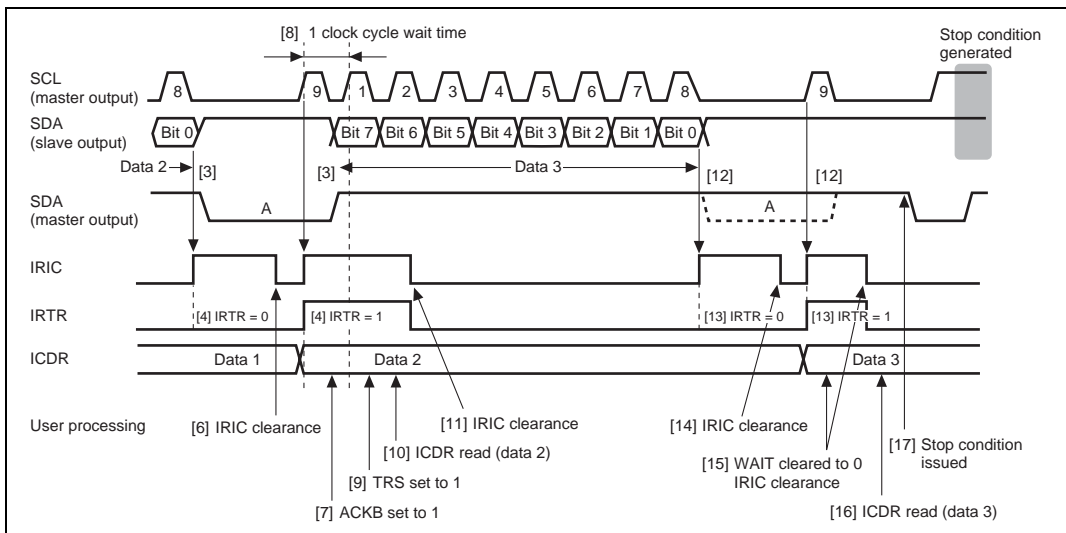


Figure 16.13 Example of Master Receive Mode Stop Condition Generation Timing
(MLS = ACKB = 0, WAIT = 1)

16.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal.

The slave device compares its own address with the slave address in the first frame following the establishment of the start condition issued by the master device. If the addresses match, the slave device operates as the slave device designated by the master device.

Figure 16.14 is a flowchart showing an example of slave receive mode operation.

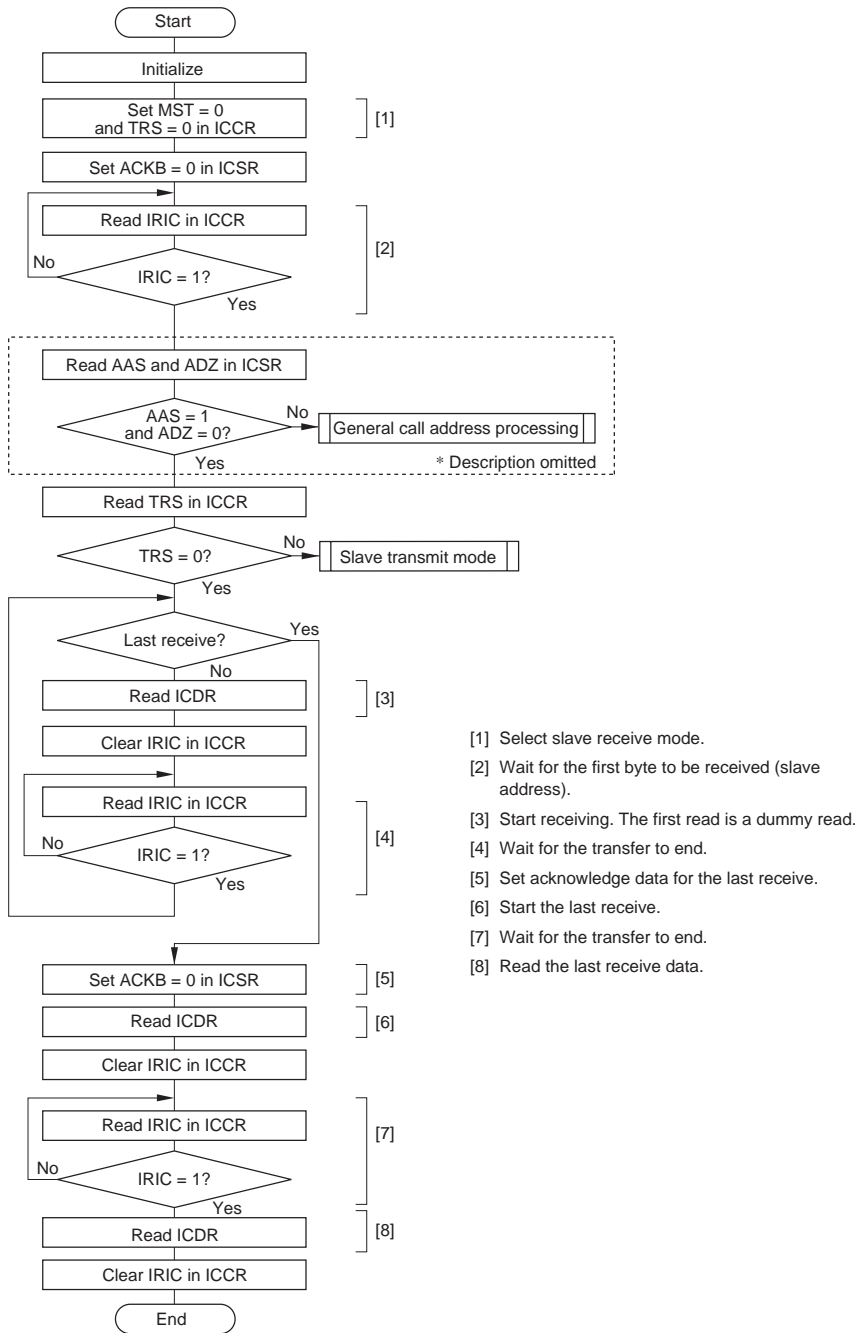


Figure 16.14 Flowchart for Slave Transmit Mode (Example)

The reception procedure and operations in slave receive mode are described below.

- [1] Set the ICE bit in ICCR to 1. Set the MLS bit in ICMR and the MST and TRS bits in ICCR according to the operating mode.
- [2] When the start condition output by the master device is detected, the BBSY flag in ICCR is set to 1.
- [3] When the slave address matches in the first frame following the start condition, the device operates as the slave device specified by the master device. If the 8th data bit (R/ \overline{W}) is 0, the TRS bit in ICCR remains cleared to 0, and slave receive operation is performed.
- [4] At the 9th clock pulse of the receive frame, the slave device drives SDA low and returns an acknowledge signal. At the same time, the IRIC flag in ICCR is set to 1. If the IEIC bit in ICCR has been set to 1, an interrupt request is sent to the CPU. If the RDRF internal flag has been cleared to 0, it is set to 1, and the receive operation continues. If the RDRF internal flag has been set to 1, the slave device drives SCL low from the fall of the receive clock until data is read into ICDR.
- [5] Read ICDR and clear the IRIC flag in ICCR to 0. The RDRF flag is cleared to 0. Read the IRDR flag and clear the IRIC flag to 0 consecutively, with no interrupt processing occurring between them. If the time needed to transmit one byte of data elapses before the IRIC flag is cleared, it will not be possible to determine when the transfer has completed.

Receive operations can be performed continuously by repeating steps [4] and [5]. When SDA is changed from low to high when SCL is high, and the stop condition is detected, the BBSY flag in ICCR is cleared to 0.

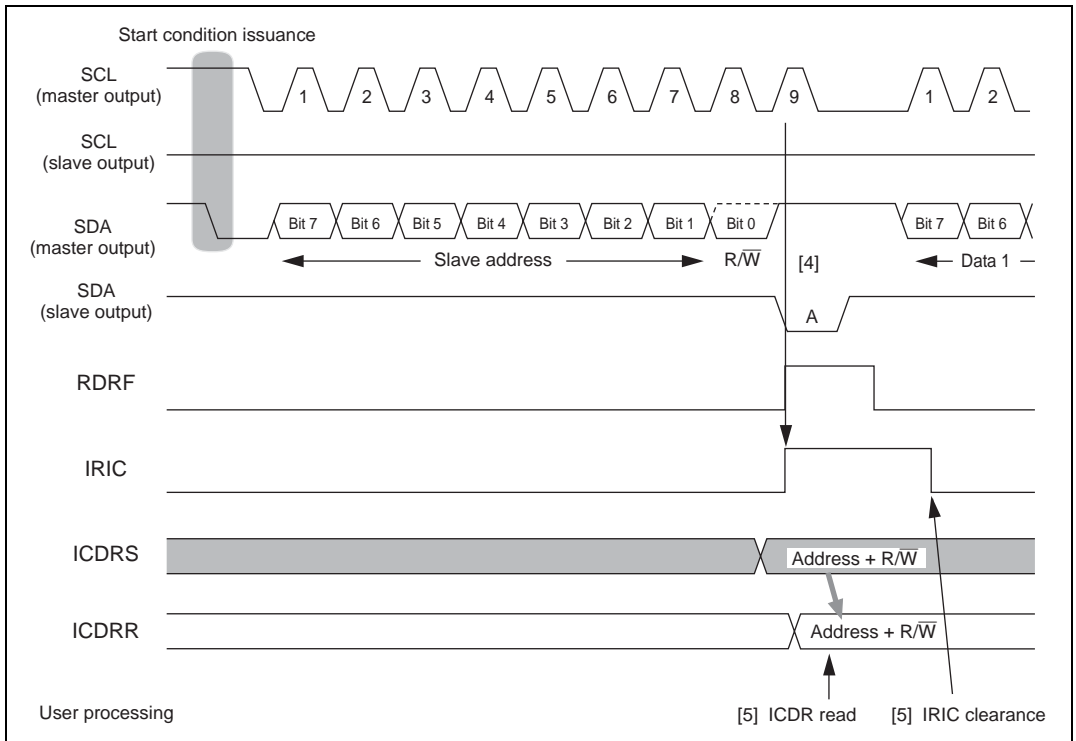


Figure 16.15 Example of Slave Receive Mode Operation Timing (1)
($MLS = ACKB = 0$)

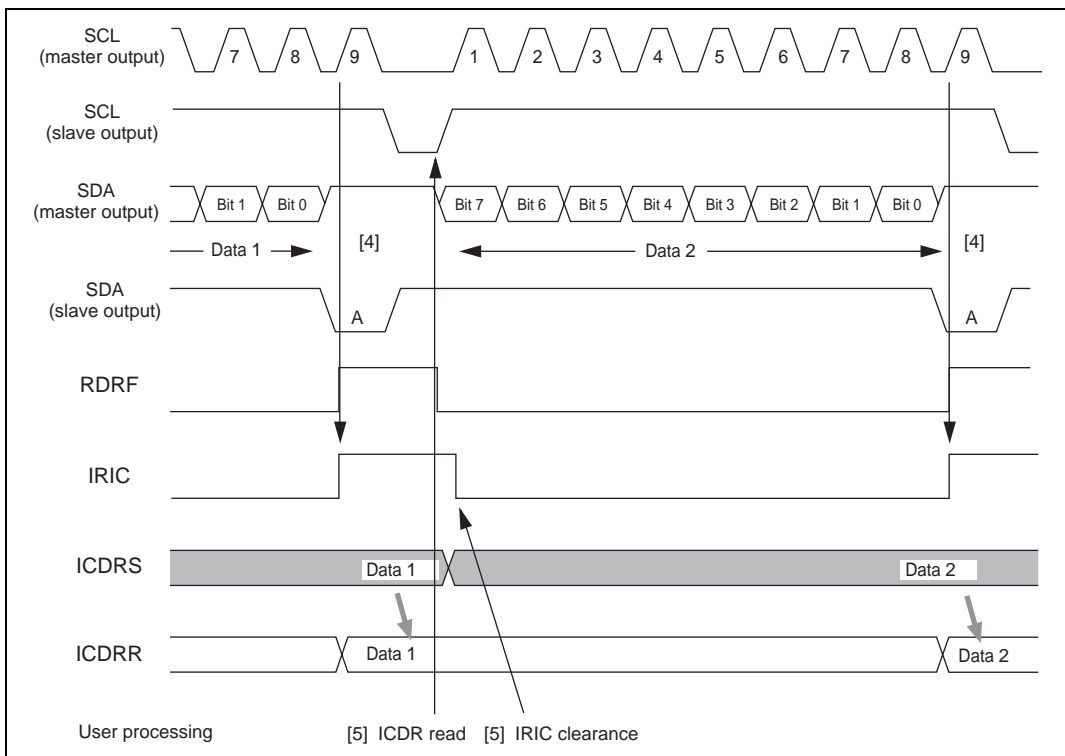


Figure 16.16 Example of Slave Receive Mode Operation Timing (2)
($MLS = ACKB = 0$)

16.4.6 Slave Transmit Operation

If the slave address matches to the address in the first frame (address reception frame) following the start condition detection when the 8th bit data (R/W) is 1 (read), the TRS bit in ICCR is automatically set to 1 and the mode changes to slave transmit mode.

Figure 16.17 shows the sample flowchart for the operations in slave transmit mode.

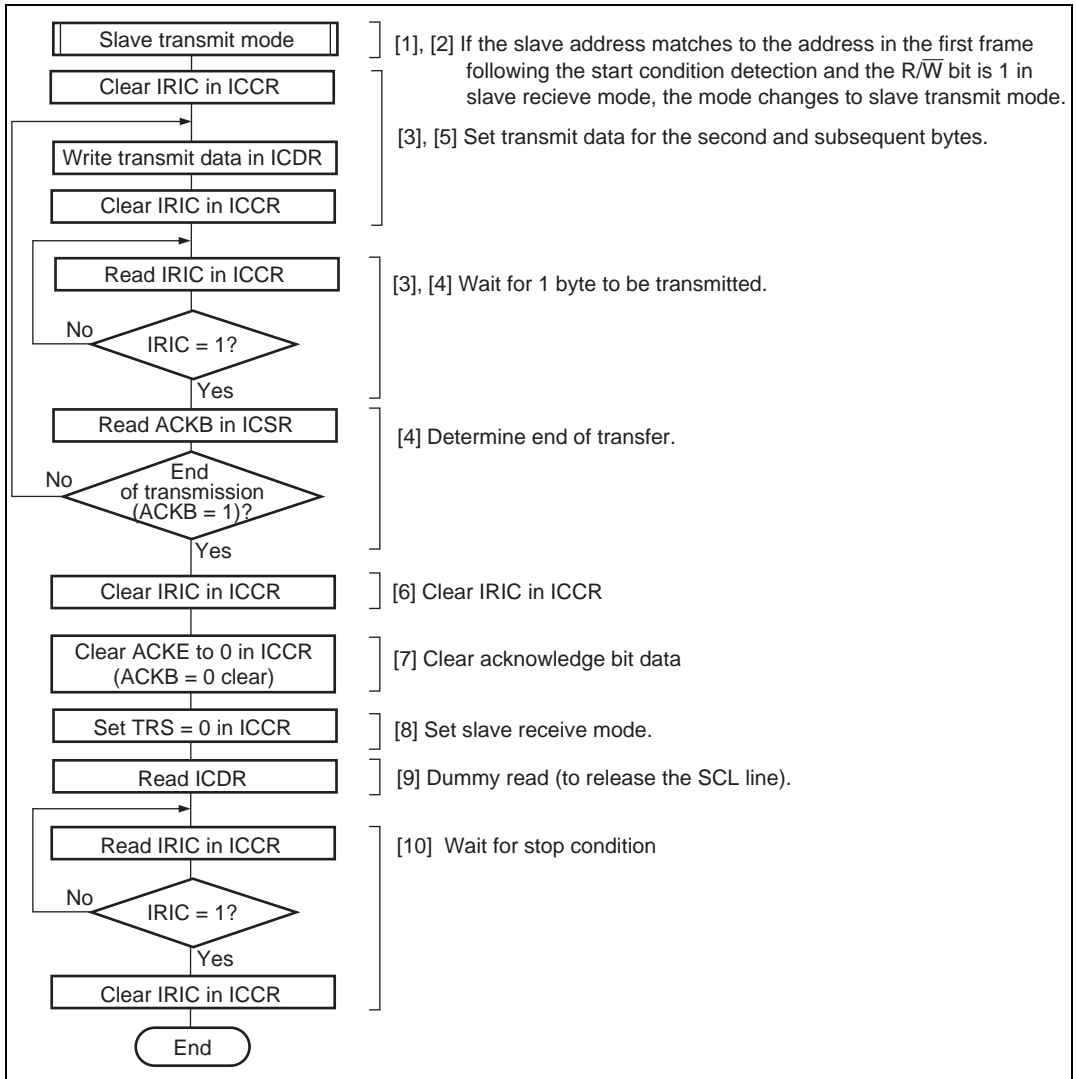


Figure 16.17 Sample Flowchart for Slave Transmit Mode

In slave transmit mode, the slave device outputs the transmit data, while the master device outputs the receive clock and returns an acknowledge signal. The transmission procedure and operations in slave transmit mode are described below.

1. Initialize slave receive mode and wait for slave address reception.
When making initial settings for slave receive mode, set the ACKE bit in ICCR to 1. This is necessary in order to enable reception of the acknowledge bit after entering slave transmit mode.
2. When the slave address matches in the first frame following detection of the start condition, the slave device drives SDA low at the 9th clock pulse and returns an acknowledge signal. If the 8th data bit (R/W) is 1, the TRS bit in ICCR is set to 1, and the mode changes to slave transmit mode automatically. The IRIC flag is set to 1 at the rise of the 9th clock. If the IEIC bit in ICCR has been set to 1, an interrupt request is sent to the CPU. At the same time, the TDRE internal flag is set to 1. The slave device drives SCL low from the fall of the transmit 9th clock until ICDR data is written, to disable the master device to output the next transfer clock.
3. After clearing the IRIC flag to 0, write data to ICDR. At this time, the TDRE internal flag is cleared to 0. The written data is transferred to ICDRS, and the TDRE internal flag and IRIC flag are set to 1 again. The slave device sequentially sends the data written into ICDRS in accordance with the clock output by the master device.
The IRIC flag is cleared to 0 to detect the end of transmission. Processing from the ICDR register writing to the IRIC flag clearing should be performed continuously. Prevent any processing that includes interrupt processing during this period. If a duration sufficient for one byte of data to be transferred elapses before the IRIC flag is cleared, it will not be possible to determine that the transfer has completed.
4. The master device drives SDA low at the 9th clock pulse, and returns an acknowledge signal. When the value of the ACKE bit in ICSR is 1, the acknowledge signal state is stored in the ACKB bit, so the ACKB bit can be used to determine whether the transfer operation was performed successfully. When one frame of data has been transmitted, the IRIC flag in ICCR is set to 1 at the rise of the 9th transmit clock pulse. When the TDRE internal flag is 0, the data written into ICDR is transferred to ICDRS, transmission starts, and the TDRE internal flag and IRIC flag are set to 1 again. If the TDRE internal flag has been set to 1, this slave device drives SCL low from the fall of the 9th transmit clock until data is written to ICDR.
5. To continue transmission, write the next data to be transmitted into ICDR. The TDRE internal flag is cleared to 0. The IRIC flag is cleared to 0 to detect the end of transmission. Processing from the ICDR writing to the IRIC flag clearing should be performed continuously. Prevent any processing that includes interrupt processing during this period.

Transmit operations can be performed continuously by repeating steps [4] and [5].

6. Clear the IRIC flag to 0.
7. To end transmission, clear the ACKE bit in ICCR to 0, to clear the acknowledge bit stored in the ACKB bit to 0.
8. Clear the TRS bit to 0 for the next address reception, to set slave receive mode.
9. Dummy-read ICDR to release SCL on the slave side.
10. When the stop condition is detected, that is, when SDA is changed from low to high when SCL is high, the BBSY flag in ICCR is cleared to 0 and the STOP flag in ICSR is set to 1. At the same time, the IRIC flag is set to 1. If the IRIC flag has been set, it is cleared to 0.
To restart slave transmit mode operation, make the initial settings once again.

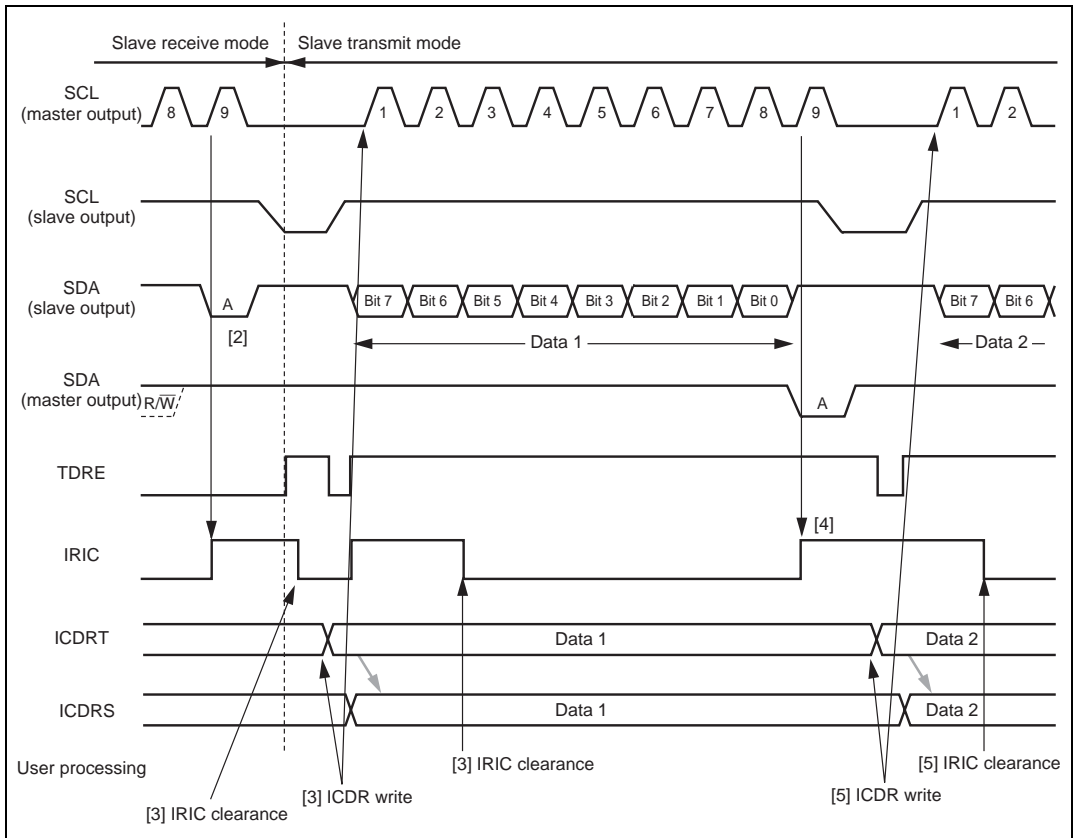


Figure 16.18 Example of Slave Transmit Mode Operation Timing (MLS = 0)

16.4.7 IRIC Setting Timing and SCL Control

The interrupt request flag (IRIC) is set at different times depending on the WAIT bit in ICMR, the FS bit in SAR, and the FSX bit in SARX. If the TDRE or RDRF internal flag is set to 1, SCL is automatically held low after one frame has been transferred; this timing is synchronized with the internal clock. Figure 16.19 shows the IRIC set timing and SCL control.

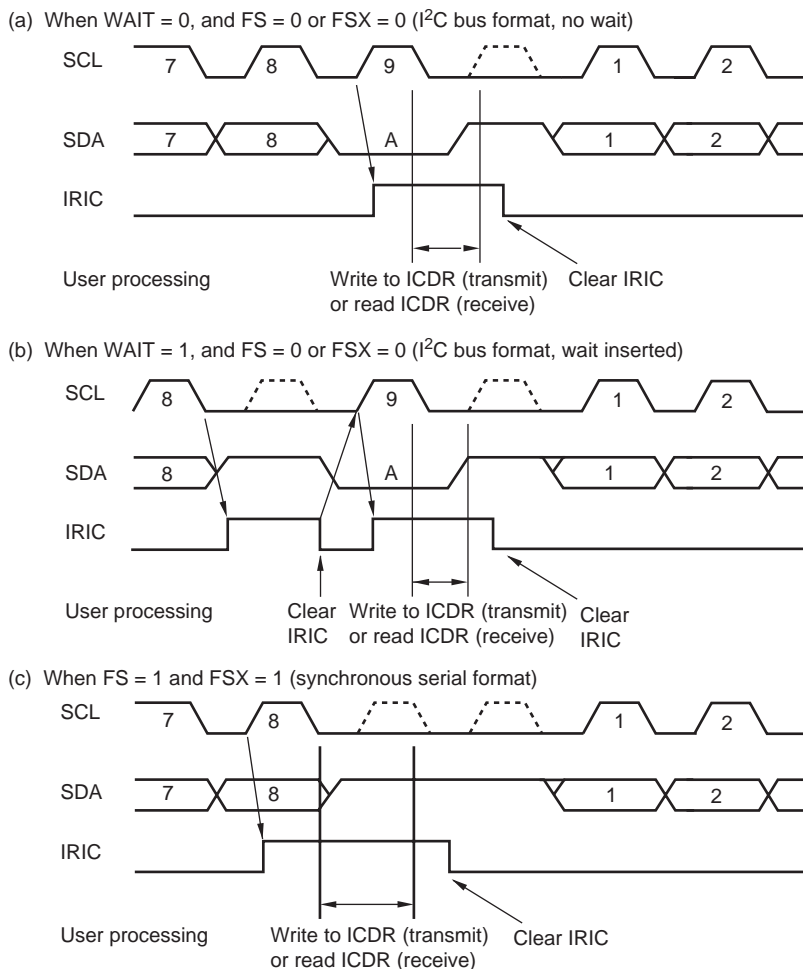


Figure 16.19 IRIC Setting Timing and SCL Control

16.4.8 Operation Using the DTC

The I²C bus format provides for selection of the slave device and transfer direction by means of the slave address and the R/W bit, confirmation of reception with acknowledge bit, indication of the last frame, and so on. Therefore, continuous data transfer using the DTC must be carried out in conjunction CPU processing by means of interrupts.

Table 16.5 shows some example of processing using the DTC. These examples assume that the number of transfer data bytes is known in slave mode.

Table 16.5 Flags and Transfer States

| Item | Master Transmit Mode | Master Receive Mode | Slave Transmit Mode | Slave Receive Mode |
|---|--|----------------------------------|---|------------------------------|
| Slave address + R/W bit Transmission/ reception | Transmission by DTC (ICDR write) | Transmission by CPU (ICDR write) | Reception by CPU (ICDR read) | Reception by CPU (ICDR read) |
| Dummy data read | — | Processing by CPU (ICDR read) | — | — |
| Actual data transmission | Transmission by DTC (ICDR write) | Reception by DTC (ICDR read) | Transmission by DTC (ICDR write) | Reception by DTC (ICDR read) |
| Dummy data (H'FF) write | — | — | Processing by DTC (ICDR write) | — |
| Last frame processing | Not necessary | Reception by CPU (ICDR read) | Not necessary | Reception by CPU (ICDR read) |
| Transfer request processing after last frame processing | 1st time: Clearing by CPU 2nd time: End condition issuance by CPU | Not necessary | Automatic clearing on detection of end condition during transmission of dummy data (H'FF) | Not necessary |
| Setting of number of DTC transfer data frames | Transmission: Actual data count + 1 (+ 1 equivalent to slave address + R/W bits) | Reception: Actual data count | Transmission: Actual data count + 1 (+ 1 equivalent to dummy data (H'FF)) | Reception: Actual data count |

16.4.9 Noise Canceler

The logic levels at the SCL and SDA pins are routed through noise cancelers before being latched internally. Figure 16.20 shows a block diagram of the noise canceler circuit.

The noise canceler consists of two cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the system clock, 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.

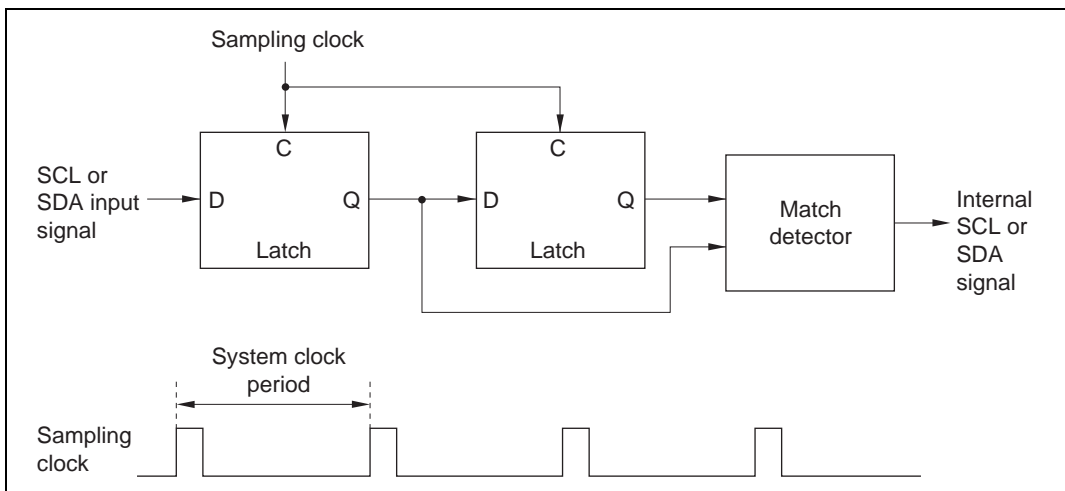


Figure 16.20 Block Diagram of Noise Canceler

16.4.10 Initialization of Internal State

The IIC has a function for forcible initialization of its internal state if a deadlock occurs during communication.

Initialization is executed by (1) setting bits CLR3 to CLR0 in the DDCSWR register or (2) clearing the ICE bit. For details of settings for bits CLR3 to CLR0, see section 16.3.8, DDC Switch Register (DDCSWR).

Scope of Initialization: The initialization executed by this function covers the following items:

- TDRE and RDRF internal flags
- Transmit/receive sequencer and internal operating clock counter
- Internal latches for retaining the output state of the SCL and SDA pins (wait, clock, data output, etc.)

The following items are not initialized:

- Actual register values (ICDR, SAR, SARX, ICMR, ICCR, ICSR, DDCCSWR, and STCR)
- Internal latches used to retain register read information for setting/clearing flags in the ICMR, ICCR, ICSR, and DDCCSWR registers
- The value of the ICMR register bit counter (BC2 to BC0)
- Generated interrupt sources (interrupt sources transferred to the interrupt controller)

Notes on Initialization:

- Interrupt flags and interrupt sources are not cleared, and so flag clearing measures must be taken as necessary.
- Basically, other register flags are not cleared either, and so flag clearing measures must be taken as necessary.
- When initialization is performed by means of the DDCCSWR register, the write data for bits CLR3 to CLR0 is not retained. To perform IIC clearance, bits CLR3 to CLR0 must be written to simultaneously using an MOV instruction. Do not use a bit manipulation instruction such as BCLR. Similarly, when clearing is required again, all the bits must be written to simultaneously in accordance with the setting.
- If a flag clearing setting is made during transmission/reception, the IIC module will stop transmitting/receiving at that point and the SCL and SDA pins will be released. When transmission/reception is started again, register initialization, etc., must be carried out as necessary to enable correct communication as a system.

The value of the BBSY bit cannot be modified directly by this module clear function, but since the stop condition pin waveform is generated according to the state and release timing of the SCL and SDA pins, the BBSY bit may be cleared as a result. Similarly, state switching of other bits and flags may also have an effect.

To prevent problems caused by these factors, the following procedure should be used when initializing the IIC state.

1. Execute initialization of the internal state according to the setting of bits CLR3 to CLR0, or according to the ICE bit.
2. Execute a stop condition issuance instruction (write 0 to BBSY and SCP) to clear the BBSY bit to 0, and wait for two transfer rate clock cycles.
3. Re-execute initialization of the internal state according to the setting of bits CLR3 to CLR0, or according to the ICE bit.
4. Initialize (re-set) the IIC registers.

16.5 Interrupt Source

IICI is the interrupt source of IIC. Table 16.6 shows each interrupt source and its priority. The ICCR interrupt enable bit sets each interrupt and the setting is independently sent to the interrupt controller.

Table 16.6 IIC Interrupt Source

| Channel | Name | Enable Bit | Interrupt Source | Interrupt Flag | Interrupt Priority |
|---------|-------|------------|--|----------------|--------------------|
| 0 | IICI0 | IEIC | I ² C bus interface interrupt request | IRIC | High |
| 1 | IICI1 | IEIC | I ² C bus interface interrupt request | IRIC | Low |

16.6 Usage Notes

1. In master mode, if an instruction to generate a start condition is issued and then an instruction to generate a stop condition is issued before the start condition is output to the I²C bus, neither condition will be output correctly. To output the start condition followed by the stop condition, after issuing the instruction that generates the start condition, read PORT in each I²C bus output pin, and check that SCL and SDA are both low. Even if the ICE bit is set to 1, it is possible to monitor the pin state by reading the PORT register so long as the DDR I/O port register corresponding to the pin has been cleared to 0. Then issue the instruction that generates the stop condition. Note that SCL may not yet have gone low when BBSY is cleared to 0.
2. Either of the following two conditions will start the next transfer. Pay attention to these conditions when reading or writing to ICDR.
 - Write access to ICDR when ICE = 1 and TRS = 1 (including automatic transfer from ICDRT to ICDRS)
 - Read access to ICDR when ICE = 1 and TRS = 0 (including automatic transfer from ICDRS to ICDRR)
3. Table 16.7 shows the timing of SCL and SDA output in synchronization with the internal clock. Timings on the bus are determined by the rise and fall times of signals affected by the bus load capacitance, series resistance, and parallel resistance.

Table 16.7 I²C Bus Timing (SCL and SDA Output)

| Item | Symbol | Output Timing | Unit | Notes |
|--|--------------------|--|------|--------------|
| SCL output cycle time | t_{SCLO} | 28 t_{cyc} to 256 t_{cyc} | ns | Figure 27.34 |
| SCL output high pulse width | t_{SCLHO} | 0.5 t_{SCLO} | ns | |
| SCL output low pulse width | t_{SCLLO} | 0.5 t_{SCLO} | ns | |
| SDA output bus free time | t_{BUFO} | 0.5 t_{SCLO} – 1 t_{cyc} | ns | |
| Start condition output hold time | t_{STAH0} | 0.5 t_{SCLO} – 1 t_{cyc} | ns | |
| Retransmission start condition output setup time | t_{STAS0} | 1 t_{SCLO} | ns | |
| Stop condition output setup time | t_{STOS0} | 0.5 t_{SCLO} + 2 t_{cyc} | ns | |
| Data output setup time (master) | t_{SDAS0} | 1 t_{SCLLO} – 3 t_{cyc} | ns | |
| Data output setup time (slave)* ¹ | | 1 t_{SCLL} – 3 t_{cyc} | ns | |
| Data output setup time (slave)* ² | | 1 t_{SCLL} – (6 t_{cyc} or 12 t_{cyc})* ³ | ns | |
| Data output hold time | t_{SDAHO} | 3 t_{cyc} | ns | |

Notes: 1. Not supported by the H8S/2258 Group.

2. Supported only by the H8S/2258 Group.

3. 6 t_{cyc} when IICX is 0, 12 t_{cyc} when IICX is 1.

4. SCL and SDA inputs are sampled in synchronization with the internal clock. The AC timing therefore depends on the system clock cycle t_{cyc} , as shown in table 27.22 (H8S/2239 Group) and table 27.34 (H8S/2238B and H8S/2236B). Note that the I²C bus interface AC timing specifications will not be met with a system clock frequency of less than 5 MHz.
5. The I²C bus interface specification for the SCL rise time t_{sr} is under 1000 ns (300 ns for high-speed mode). In master mode, the I²C bus interface monitors the SCL line and synchronizes one bit at a time during communication. If t_{sr} (the time for SCL to go from low to V_{IH}) exceeds the time determined by the input clock of the I²C bus interface, the high period of SCL is extended. The SCL rise time is determined by the pull-up resistance and load capacitance of the SCL line. To insure proper operation at the set transfer rate, adjust the pull-up resistance and load capacitance so that the SCL rise time does not exceed the values given in the table in table 16.8.

Table 16.8 Permissible SCL Rise Time (t_{sr}) Values

| | | | Time Indication | | | | | |
|------|-------------------------|-----------------|--|---------------------------------|---------------------------------|--------------------|----------------------------------|----------------------------------|
| IICX | t_{cyc} Indication | | I ² C Bus Specification (Max) | $\phi =$ 5 MHz ^{*2} | $\phi =$ 8 MHz ^{*2} | $\phi =$ 10 MHz | $\phi =$ 16 MHz ^{*1} | $\phi =$ 20 MHz ^{*1} |
| 0 | 7.5 t_{cyc} | Normal mode | 1000 ns | 1000 ns | 937 ns | 750 ns | 468 ns | 375 ns |
| | | High-speed mode | 300 ns | 300 ns | 300 ns | 300 ns | 300 ns | 300 ns |
| 1 | 17.5 t_{cyc} | Normal mode | 1000 ns | 1000 ns | 1000 ns | 1000 ns | 1000 ns | 875 ns |
| | | High-speed mode | 300 ns | 300 ns | 300 ns | 300 ns | 300 ns | 300 ns |

Notes: 1. Supported only by the H8S/2239 Group.

2. The H8S/2258 Group is out of operation.

6. The I²C bus interface specifications for the SCL and SDA rise and fall times are under 1000 ns and 300 ns. The I²C bus interface SCL and SDA output timing is prescribed by t_{cyc} , as shown in table 16.7. However, because of the rise and fall times, the I²C bus interface specifications may not be satisfied at the maximum transfer rate. Table 16.9 shows output timing calculations for different operating frequencies, including the worst-case influence of rise and fall times. The values in the above table will vary depending on the settings of the IICX bit and bits CKS0 to CKS2. Depending on the frequency it may not be possible to achieve the maximum transfer rate; therefore, whether or not the I²C bus interface specifications are met must be determined in accordance with the actual setting conditions.

t_{BUFO} fails to meet the I²C bus interface specifications at any frequency. The solution is either (a) to provide coding to secure the necessary interval (approximately 1 μ s) between issuance of a stop condition and issuance of a start condition, or (b) to select devices whose input timing permits this output timing for use as slave devices connected to the I²C bus.

t_{SCLLO} in high-speed mode and t_{STASO} in standard mode fail to satisfy the I²C bus interface specifications for worst-case calculations of t_{sr}/t_{sr} . Possible solutions that should be investigated include (a) adjusting the rise and fall times by means of a pull-up resistor and capacitive load, (b) reducing the transfer rate to meet the specifications, or (c) selecting devices whose input timing permits this output timing for use as slave devices connected to the I²C bus.

Table 16.9 I²C Bus Timing (with Maximum Influence of t_{Sr}/t_{SF})

| Item | t_{cyc} Indication | | Time Indication (at Maximum Transfer Rate) [ns] | | | | | | |
|--------------------------------------|--|-----------------|---|---|---------------------------------|---------------------------------|--------------------|----------------------------------|----------------------------------|
| | | | t_{Sr}/t_{SF} Influence (Max) | I ² C Bus Specifi- cation (Min) | $\phi =$ 5 MHz ^{*7} | $\phi =$ 8 MHz ^{*7} | $\phi =$ 10 MHz | $\phi =$ 16 MHz ^{*3} | $\phi =$ 20 MHz ^{*3} |
| t_{SCLHO} | $0.5t_{SCLO} (-t_{Sr})$ | Standard mode | -1000 | 4000 | 4000 | 4000 | 4000 | 4000 | 4000 |
| | | High-speed mode | -300 | 600 | 950 | 950 | 950 | 950 | 950 |
| t_{SCLLO} | $0.5t_{SCLO} (-t_{Sr})$ | Standard mode | -250 | 4700 | 4750 | 4750 | 4750 | 4750 | 4750 |
| | | High-speed mode | -250 | 1300 | 1000 ^{*1} | 1000 ^{*1} | 1000 ^{*1} | 1000 ^{*1} | 1000 ^{*1} |
| t_{BUFO} | $0.5t_{SCLO} - 1t_{cyc}$ ($-t_{Sr}$) | Standard mode | -1000 | 4700 | 3800 ^{*1} | 3875 ^{*1} | 3900 ^{*1} | 3938 ^{*1} | 3950 ^{*1} |
| | | High-speed mode | -300 | 1300 | 750 ^{*1} | 825 ^{*1} | 850 ^{*1} | 888 ^{*1} | 900 ^{*1} |
| t_{STAHO} | $0.5t_{SCLO} - 1t_{cyc}$ ($-t_{Sr}$) | Standard mode | -250 | 4000 | 4550 | 4625 | 4650 | 4688 | 4700 |
| | | High-speed mode | -250 | 600 | 800 | 875 | 900 | 938 | 950 |
| t_{STASO} | $1t_{SCLO} (-t_{Sr})$ | Standard mode | -1000 | 4700 | 9000 | 9000 | 9000 | 9000 | 9000 |
| | | High-speed mode | -300 | 600 | 2200 | 2200 | 2200 | 2200 | 2200 |
| t_{STOSO} | $0.5t_{SCLO} + 2t_{cyc}$ ($-t_{Sr}$) | Standard mode | -1000 | 4000 | 4400 | 4250 | 4200 | 4125 | 4100 |
| | | High-speed mode | -300 | 600 | 1350 | 1200 | 1150 | 1075 | 1050 |
| t_{SDASO} (master) | $1t_{SCLLO}^{*2} - 3t_{cyc}$ ($-t_{Sr}$) | Standard mode | -1000 | 250 | 3100 | 3325 | 3400 | 3513 | 3550 |
| | | High-speed mode | -300 | 100 | 400 | 625 | 700 | 813 | 850 |
| t_{SDASO} (slave) ^{*4} | $1t_{SCLL}^{*2} - 3t_{cyc}$ ($-t_{Sr}$) | Standard mode | -1000 | 250 | 3100 | 3325 | 3400 | 3513 | 3550 |
| | | High-speed mode | -300 | 100 | 400 | 625 | 700 | 813 | 850 |
| t_{SDASO} (slave) ^{*5} | $1t_{SCLL}^{*2} - 12t_{cyc}^{*6}$ ($-t_{Sr}$) | Standard mode | -1000 | 250 | — | — | 2500 | — | — |
| | | High-speed mode | -300 | 100 | — | — | -200 ^{*1} | — | — |
| t_{SDAHO} | $3t_{cyc}$ | Standard mode | 0 | 0 | 600 | 375 | 300 | 188 | 150 |
| | | High-speed mode | 0 | 0 | 600 | 375 | 300 | 188 | 150 |

Notes: 1. Does not meet the I²C bus interface specification. Remedial action such as the following is necessary: (a) secure a start/stop condition issuance interval; (b) adjust the rise and fall times by means of a pull-up resistor and capacitive load; (c) reduce the transfer rate; (d) select slave devices whose input timing permits this output timing.

The values in the above table will vary depending on the settings of the IICX bit and bits CKS0 to CKS2. Depending on the frequency it may not be possible to achieve the maximum transfer rate; therefore, whether or not the I²C bus interface specifications are met must be determined in accordance with the actual setting conditions.

2. Calculated using the I²C bus specification values (standard mode: 4700 ns min; high-speed mode: 1300 ns min).
3. Supported only by the H8S/2239 Group.
4. Not supported by the H8S/2258 Group.
5. Supported only by the H8S/2258 Group.

6. Value when the IICX bit is set to 1. When the IICX bit is cleared to 0, the value is $6t_{\text{cyc}}$.
7. The H8S/2258 Group is out of operation.

7. Note on ICDR Read at End of Master Reception

To halt reception after completion of a receive operation in master receive mode, set the TRS bit to 1 and write 0 to BBSY and SCP in ICCR. This changes the SDA pin from low to high when the SCL pin is high, and generates the stop condition. After this, receive data can be read by means of an ICDR read, but if data remains in the buffer the ICDRS receive data will not be transferred to ICDR, and so it will not be possible to read the second byte of data. If it is necessary to read the second byte of data, issue the stop condition in master receive mode (i.e. with the TRS bit cleared to 0). When reading the receive data, first confirm that the BBSY bit in ICCR is cleared to 0, the stop condition has been generated, and the bus has been released, then read ICDR with TRS cleared to 0. Note that if the receive data (ICDR data) is read in the interval between execution of the instruction for issuance of the stop condition (writing of 0 to BBSY and SCP in ICCR) and the actual generation of the stop condition, the clock may not be output correctly in subsequent master transmission.

Clearing of the MST bit after completion of master transmission/reception, or other modifications of IIC control bits to change the transmit/receive operating mode or settings, must be carried out during interval (a) in figure 16.21 (after confirming that the BBSY bit has been cleared to 0 in the ICCR register).

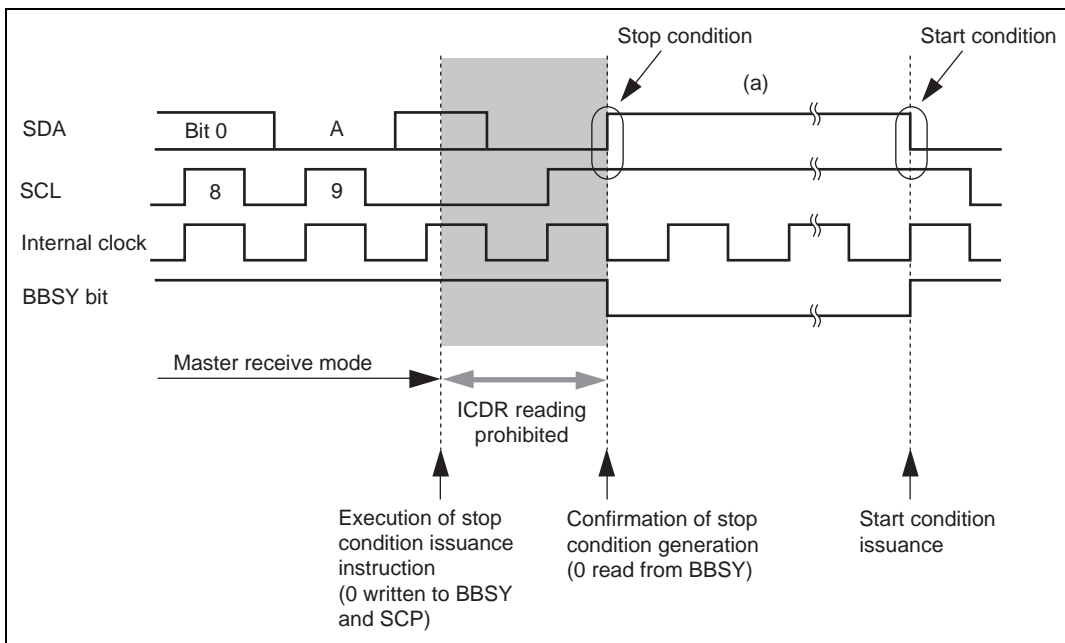


Figure 16.21 Points for Attention Concerning Reading of Master Receive Data

8. Notes on Start Condition Issuance for Retransmission

Depending on the timing combination with the start condition issuance and the subsequently writing data to ICDR, it may not be possible to issue the retransmission and the data transmission after retransmission condition issuance.

After start condition issuance is done and determined the start condition, write the transmit data to ICDR, as shown below. Figure 16.22 shows the timing of start condition issuance for retransmission, and the timing for subsequently writing data to ICDR, together with the corresponding flowchart.

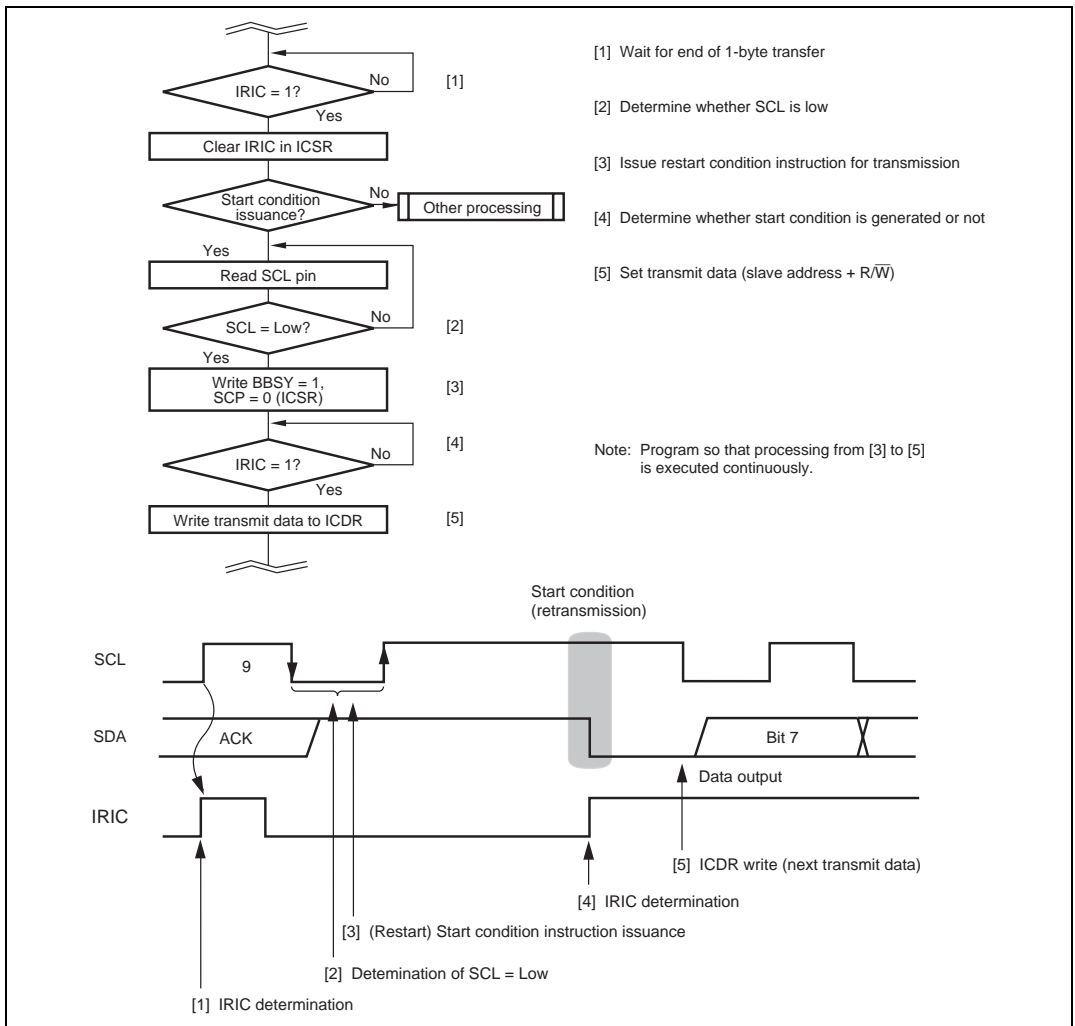


Figure 16.22 Flowchart and Timing of Start Condition Instruction Issuance for Retransmission

9. Notes on I²C Bus Interface Stop Condition Instruction Issuance

If the rise time of the 9th SCL acknowledge exceeds the specification because the bus load capacitance is large, or if there is a slave device of the type that drives SCL low to effect a wait, issue the stop condition instruction after reading SCL and determining it to be low, as shown below.

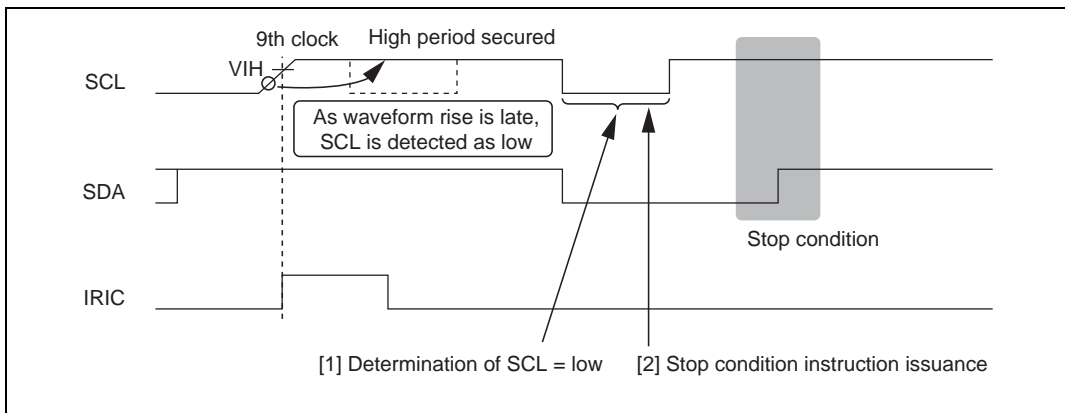


Figure 16.23 Timing of Stop Condition Issuance

10. Notes on IRIC Flag Clearance when Using Wait Function

If the SCL rise time exceeds the designated duration or if the slave device is of the type that keeps SCL low and applies a wait state when the wait function is used in the master mode of the I²C bus interface, read SCL and clear the IRIC flag after determining that SCL has gone low, as shown below.

Clearing the IRIC flag to 0 when WAIT is set to 1 and SCL is being held at high level can cause the SDA value to change before SCL goes low, resulting in a start condition or stop condition being generated erroneously.

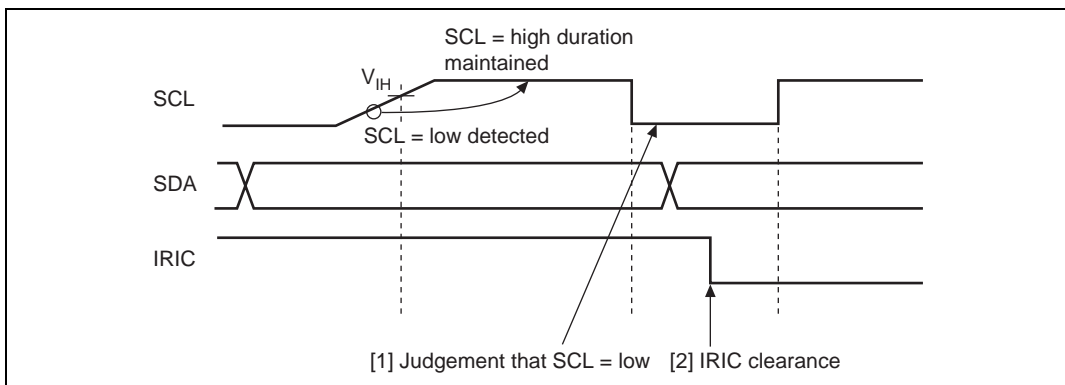


Figure 16.24 IRIC Flag Clearance in WAIT = 1 Status

11. Notes on ICDR Reads and ICCR Access in Slave Transmit Mode

In a transmit operation in the slave mode of the I²C bus interface, do not read the ICDR register or read or write to the ICCR register during the period indicated by the shaded portion in figure 16.25.

Normally, when interrupt processing is triggered in synchronization with the rising edge of the 9th clock cycle, the period in question has already elapsed when the transition to interrupt processing takes place, so there is no problem with reading the ICDR register or reading or writing to the ICCR register.

To ensure that the interrupt processing is performed properly, one of the following two conditions should be applied.

- (1) Make sure that reading received data from the ICDR register, or reading or writing to the ICCR register, is completed before the next slave address receive operation starts.
- (2) Monitor the BC2 to BC0 counter in the ICMR register and, when the value of BC2 to BC0 is 000 (8th or 9th clock cycle), allow a waiting time of at least 2 transfer clock cycles in order to involve the problem period in question before reading from the ICDR register, or reading or writing to the ICCR register.

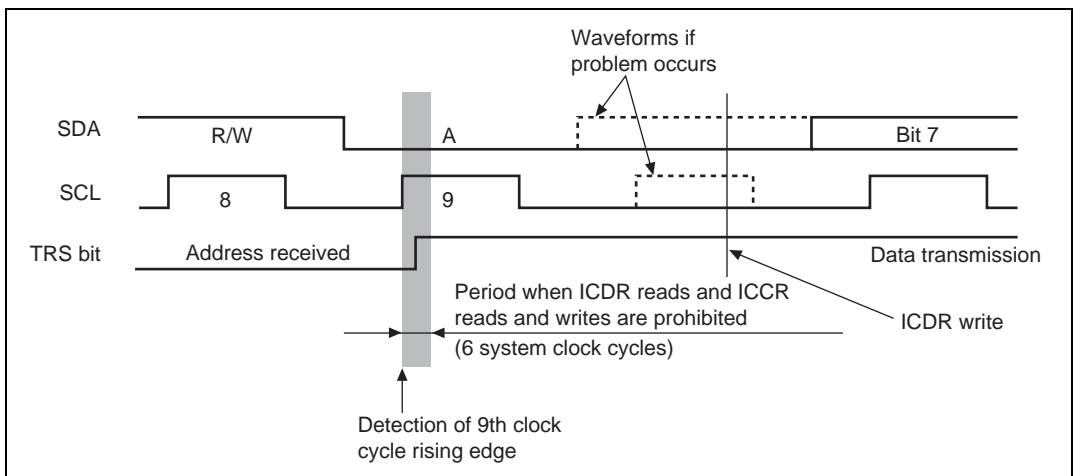


Figure 16.25 ICDR Read and ICCR Access Timing in Slave Transmit Mode

12. Notes on TRS Bit Setting in Slave Mode

From the detection of the rising edge of the 9th clock cycle or of a stop condition to when the rising edge of the next SCL pin signal is detected (the period indicated as (a) in figure 16.26) in the slave mode of the I²C bus interface, the value set in the TRS bit in the ICCR register is effective immediately.

However, at other times (indicated as (b) in figure 16.26) the value set in the TRS bit is put on hold until the next rising edge of the 9th clock cycle or stop condition is detected, rather than taking effect immediately.

This results in the actual internal value of the TRS bit remaining 1 (transmit mode) and no acknowledge bit being sent at the 9th clock cycle address receive completion in the case of an address receive operation following a restart condition input with no stop condition intervening.

When receiving an address in the slave mode, clear the TRS bit to 0 during the period indicated as (a) in figure 16.26.

To cancel the holding of the SCL bit low by the wait function in the slave mode, clear the TRS bit to 0 and then perform a dummy read of the ICDR register.

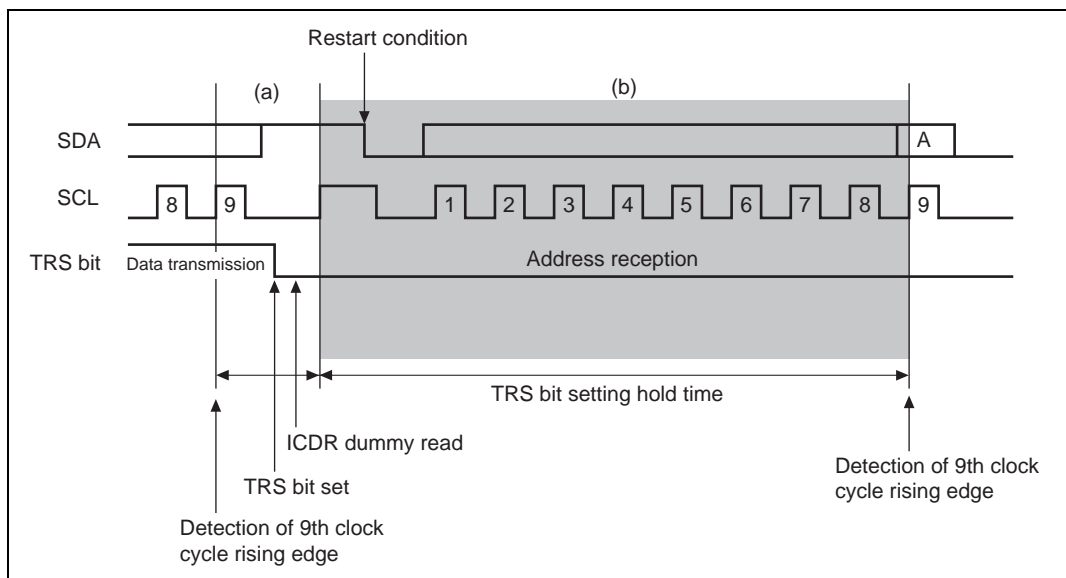


Figure 16.26 TRS Bit Setting Timing in Slave Mode

13. Notes on ICDR Reads in Transmit Mode and ICDR Writes in Receive Mode

When attempting to read ICDR in the transmit mode (TRS = 1) or write to ICDR in the receive mode (TRS = 0) under certain conditions, the SCL pin may not be held low after the completion of the transmit or receive operation and a clock may not be output to the SCL bus line before the ICDR register access operation can take place properly.

When accessing ICDR, always change the setting to the transmit mode before performing a read operation, and always change the setting to the receive mode before performing a write operation.

14. Notes on ACKE Bit and TRS Bit in Slave Mode

When using the I²C bus interface, if an address is received in the slave mode immediately after 1 is received as an acknowledge bit (ACKB = 1) in the transmit mode (TRS = 1), an interrupt may be generated at the rising edge of the 9th clock cycle if the address does not match.

When performing slave mode operations using the IIC bus interface module, make sure to do the following.

- (1) When a 1 is received as an acknowledge bit for the final transmit data after completing a series of transmit operations, clear the ACKE bit in the ICCR register to 0 to initialize the ACKB bit to 0.
- (2) In the slave mode, change the setting to the receive mode (TRS = 0) before the start condition is input. To ensure that the switch from the slave transmit mode to the slave receive mode is accomplished properly, end the transmission as described in figure 16.17.

15. Notes on Arbitration Lost in Master Mode

The I²C bus interface recognizes the data in transmit/receive frame as an address when arbitration is lost in master mode and a transition to slave receive mode is automatically carried out.

When arbitration is lost not in the first frame but in the second frame or subsequent frame, transmit/receive data that is not an address is compared with the value set in the SAR or SARX register as an address. If the receive data matches with the address in the SAR or SARX register, the I²C bus interface erroneously recognizes that the address call has occurred. (See figure 16.27.)

In multi-master mode, a bus conflict could happen. When The I²C bus interface is operated in master mode, check the state of the AL bit in the ICSR register every time after one frame of data has been transmitted or received.

When arbitration is lost during transmitting the second frame or subsequent frame, take avoidance measures.

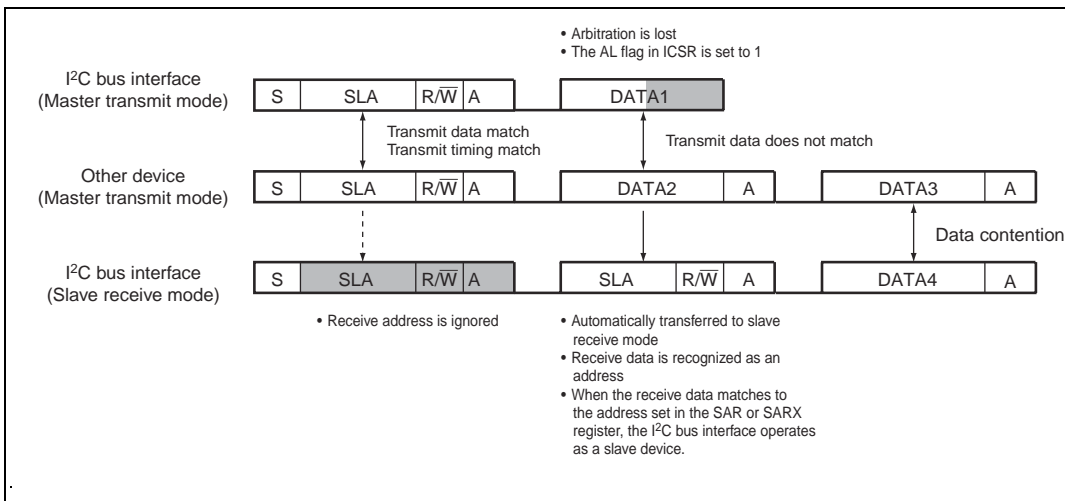


Figure 16.27 Diagram of Erroneous Operation When Arbitration Is Lost

Though it is prohibited in the normal I²C protocol, the same problem may occur when the MST bit is erroneously set to 1 and a transition to master mode is occurred during data transmission or reception in slave mode. In multi-master mode, pay attention to the setting of the MST bit when a bus conflict may occur. In this case, the MST bit in the ICCR register should be set to 1 according to the order below.

- (1) Make sure that the BBSY flag in the ICCR register is 0 and the bus is free before setting the MST bit.
- (2) Set the MST bit to 1.
- (3) To confirm that the bus was not entered to the busy state while the MST bit is being set, check that the BBSY flag in the ICCR register is 0 immediately after the MST bit has been set.

16. Note on Wait Operation in Master Mode

When the interrupt request flag (IRIC) is cleared from 1 to 0 between the falling edge of the 7th clock and the falling edge of the 8th clock in master mode using the wait function, a wait may not be inserted after the falling edge of the 8th clock and 9th clock pulse may be output continuously.

When using the wait operation, note the following to clear the IRIC flag.

After the IRIC flag is set to 1 at the rising edge of the 9th clock, clear the IRIC flag before the rising edge of the 7th clock (when the value of the BC2 to BC0 counter is 2 or more).

If the clearing of the IRIC flag is delayed due to interrupt handling etc. and the value of the BC counter reaches 1 or 0, confirm that the SCL pin is low and then clear the IRIC flag after the BC2 to BC0 counter reaches 0 (see figure 16.28).

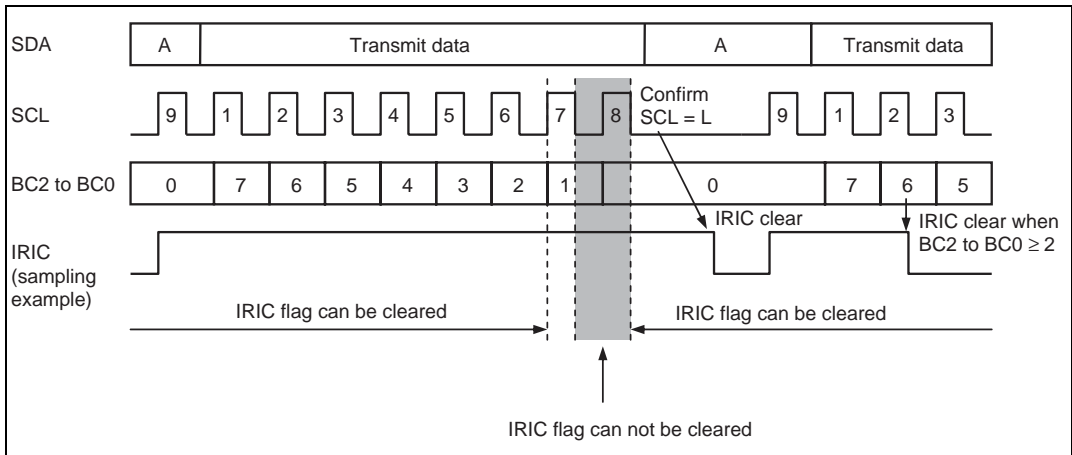


Figure 16.28 IRIC Flag Clearing Timing in Wait Operation

17. Interrupt during Module Stop Mode

When the module is stopped in the state that an interrupt is requested, the interrupt source of the CPU or activation source of the DTC is not cleared. Be sure to enter module stop mode by disabling the interrupt beforehand.

16.6.1 Module Stop Mode Setting

IIC operation can be disabled or enabled using the module stop control register. The initial setting is for IIC operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 24, Power-Down Modes.

Section 17 A/D Converter

This LSI includes a successive approximation type 10-bit A/D converter that allows up to eight analog input channels to be selected. A block diagram of the A/D converter is shown in figure 17.1.

17.1 Features

- 10-bit resolution
- Eight input channels
- Conversion time: 9.6 μ s per channel (at 13.5 MHz operation)
- Two operating modes
 - Single mode: Single-channel A/D conversion
 - Scan mode: Continuous A/D conversion on 1 to 4 channels
- Four data registers
 - Conversion results are held in a 16-bit data register for each channel
- Sample and hold function
- Three methods conversion start
 - Software
 - Timer (TPU or 8-bit timer) conversion start trigger
 - External trigger signal
- Interrupt request
 - An A/D conversion end interrupt request (ADI) can be generated
- Module stop mode can be set
- Selectable range voltages of analog inputs
 - The range of voltages of analog inputs to be converted can be specified using the V_{ref} signal as the analog reference voltage.

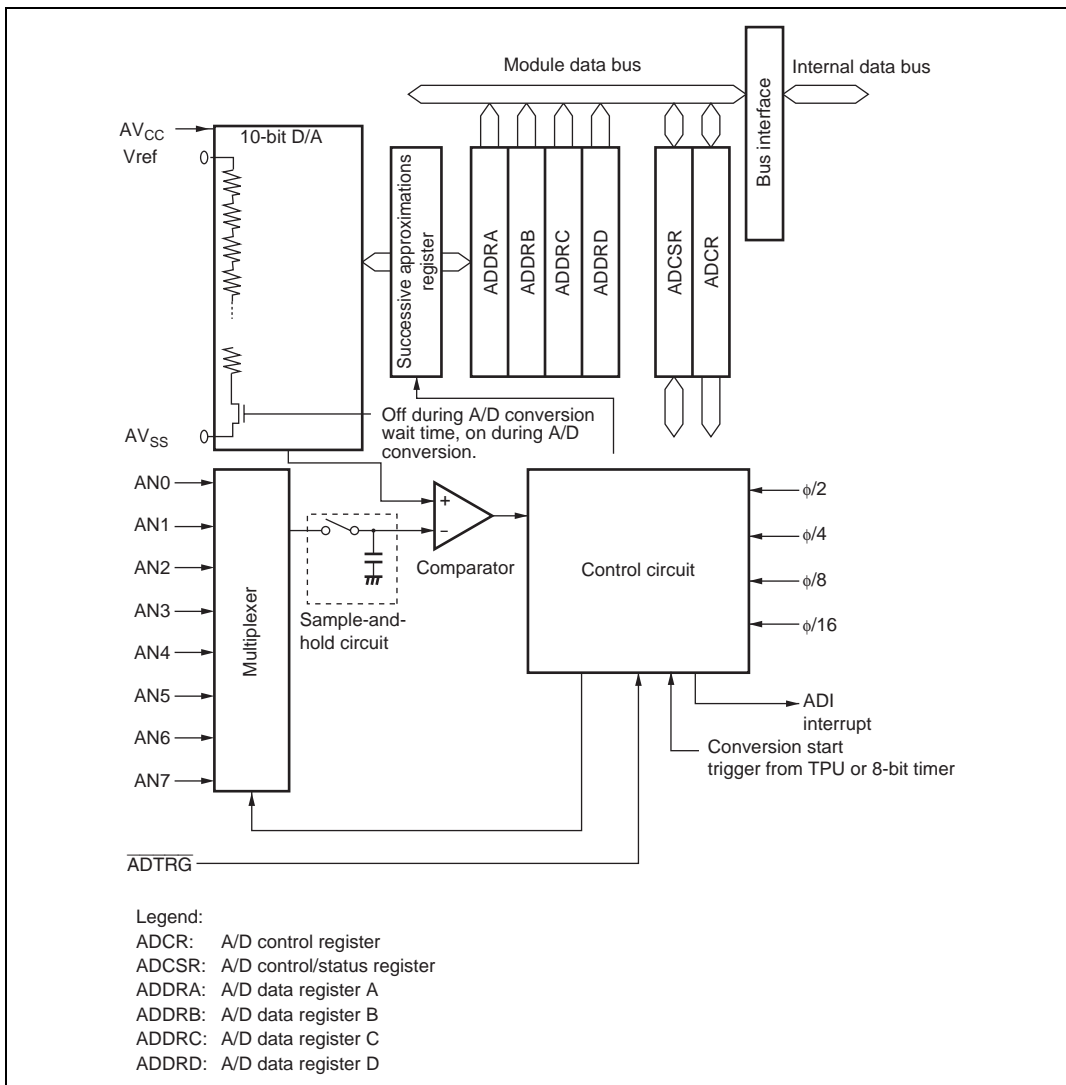


Figure 17.1 Block Diagram of A/D Converter

17.2 Input/Output Pins

Table 17.1 summarizes the input pins used by the A/D converter. The eight analog input pins are divided into two groups each of which consists of four channels; analog input pins 0 to 3 (AN0 to AN3) comprising group 0 and analog input pins 4 to 7 (AN4 to AN7) comprising group 1. The AVcc and AVss pins are the power supply pins for the analog block in the A/D converter. The Vref pin is the A/D conversion reference voltage pin.

Table 17.1 Pin Configuration

| Pin Name | Symbol | I/O | Function |
|--------------------------------|------------------|-------|--|
| Analog power supply pin | AV _{cc} | Input | Analog block power supply and reference voltage |
| Analog ground pin | AV _{ss} | Input | Analog block ground and reference voltage |
| Reference voltage pin | Vref | Input | Reference voltage for A/D conversion |
| Analog input pin 0 | AN0* | Input | Group 0 analog input pins |
| Analog input pin 1 | AN1* | Input | |
| Analog input pin 2 | AN2 | Input | |
| Analog input pin 3 | AN3 | Input | |
| Analog input pin 4 | AN4 | Input | Group 1 analog input pins |
| Analog input pin 5 | AN5 | Input | |
| Analog input pin 6 | AN6 | Input | |
| Analog input pin 7 | AN7 | Input | |
| A/D external trigger input pin | ADTRG | Input | External trigger input pin for starting A/D conversion |

Note: * In the case of the H8S/2239 Group, H8S/2227 Group, H8S/2238R, and H8S/2236R, AN0 and AN1 may be used only when Vcc = AVcc.

17.3 Register Descriptions

The A/D converter has the following registers. For details on the module stop control register, refer to section 24.1.2, Module Stop Control Registers A to C (MSTPCRA to MSTPCRC).

- A/D data register A (ADDRA)
- A/D data register B (ADDRB)
- A/D data register C (ADDRC)
- A/D data register D (ADDRD)
- A/D control/status register (ADCSR)
- A/D control register (ADCR)

17.3.1 A/D Data Registers A to D (ADDRA to ADDRD)

There are four 16-bit read-only ADDR registers; ADDRA to ADDRD, used to store the results of A/D conversion. The ADDR registers, which store a conversion result for each channel, are shown in table 17.2.

The converted 10-bit data is stored in bits 6 to 15. The lower 6 bits are always read as 0.

The data bus between the CPU and the A/D converter is 8 bits wide. The upper byte can be read directly from the CPU, however the lower byte should be read via a temporary register. Therefore, when reading the ADDR, read only the upper byte, or read in word unit.

Table 17.2 Analog Input Channels and Corresponding ADDR Registers

| Analog Input Channel | | A/D Data Register to be Stored the Results of A/D Conversion |
|----------------------|-------------------|--|
| Group 0 (CH2 = 0) | Group 1 (CH2 = 1) | |
| AN0 | AN4 | ADDRA |
| AN1 | AN5 | ADDRB |
| AN2 | AN6 | ADDRC |
| AN3 | AN7 | ADDRD |

17.3.2 A/D Control/Status Register (ADCSR)

ADCSR controls A/D conversion operations.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|--------|--|
| 7 | ADF | 0 | R/(W)* | <p>A/D End Flag</p> <p>A status flag that indicates the end of A/D conversion.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> • When A/D conversion ends in single mode • When A/D conversion ends on all specified channels in scan mode <p>[Clearing conditions]</p> <ul style="list-style-type: none"> • When 0 is written after reading ADF = 1 • When the data transfer controller (DTC) is activated by an ADI interrupt and DISEL in DTC is 0 with the transfer counter not being 0 |
| 6 | ADIE | 0 | R/W | <p>A/D Interrupt Enable</p> <p>A/D conversion end interrupt (ADI) request enabled when 1 is set</p> |
| 5 | ADST | 0 | R/W | <p>A/D Start</p> <p>Clearing this bit to 0 stops A/D conversion, and the A/D converter enters the wait state.</p> <p>Setting this bit to 1 starts A/D conversion. In single mode, this bit is cleared to 0 automatically when conversion on the specified channel is complete. In scan mode, conversion continues sequentially on the specified channels until this bit is cleared to 0 by software, a reset, software standby mode, hardware standby mode, or module stop mode.</p> <p>The ADST bit can be set to 1 by software, a timer conversion start trigger, or the A/D external trigger input pin (ADTRG).</p> |

| Bit | Bit Name | Initial Value | R/W | Description | | | | | | | | | | | | | | | | | | |
|---------------|------------------|---------------|-----|--|---------------|---------------|----------|----------|----------|------------------|----------|-----------------|----------|-----------------|----------|----------|----------|------------------|----------|-----------------|----------|-----------------|
| 4 | SCAN | 0 | R/W | Scan Mode Selects single mode or scan mode as the A/D conversion operating mode. Only set the SCAN bit while conversion is stopped (ADST = 0). 0: Single mode 1: Scan mode | | | | | | | | | | | | | | | | | | |
| 3 | — | 0 | R/W | Reserved This bit is always read as 0. The write value should always be 0. | | | | | | | | | | | | | | | | | | |
| 2 | CH2 | 0 | R/W | Channel Select 2 to 0 | | | | | | | | | | | | | | | | | | |
| 1 | CH1 | 0 | R/W | Select analog input channels. | | | | | | | | | | | | | | | | | | |
| 0 | CH0 | 0 | R/W | <table><tr><th>When SCAN = 0</th><th>When SCAN = 1</th></tr><tr><td>000: AN0</td><td>000: AN0</td></tr><tr><td>001: AN1</td><td>001: AN0 and AN1</td></tr><tr><td>010: AN2</td><td>010: AN0 to AN2</td></tr><tr><td>011: AN3</td><td>011: AN0 to AN3</td></tr><tr><td>100: AN4</td><td>100: AN4</td></tr><tr><td>101: AN5</td><td>101: AN4 and AN5</td></tr><tr><td>110: AN6</td><td>110: AN4 to AN6</td></tr><tr><td>111: AN7</td><td>111: AN4 to AN7</td></tr></table> | When SCAN = 0 | When SCAN = 1 | 000: AN0 | 000: AN0 | 001: AN1 | 001: AN0 and AN1 | 010: AN2 | 010: AN0 to AN2 | 011: AN3 | 011: AN0 to AN3 | 100: AN4 | 100: AN4 | 101: AN5 | 101: AN4 and AN5 | 110: AN6 | 110: AN4 to AN6 | 111: AN7 | 111: AN4 to AN7 |
| When SCAN = 0 | When SCAN = 1 | | | | | | | | | | | | | | | | | | | | | |
| 000: AN0 | 000: AN0 | | | | | | | | | | | | | | | | | | | | | |
| 001: AN1 | 001: AN0 and AN1 | | | | | | | | | | | | | | | | | | | | | |
| 010: AN2 | 010: AN0 to AN2 | | | | | | | | | | | | | | | | | | | | | |
| 011: AN3 | 011: AN0 to AN3 | | | | | | | | | | | | | | | | | | | | | |
| 100: AN4 | 100: AN4 | | | | | | | | | | | | | | | | | | | | | |
| 101: AN5 | 101: AN4 and AN5 | | | | | | | | | | | | | | | | | | | | | |
| 110: AN6 | 110: AN4 to AN6 | | | | | | | | | | | | | | | | | | | | | |
| 111: AN7 | 111: AN4 to AN7 | | | | | | | | | | | | | | | | | | | | | |

Note: * Only 0 can be written to clear this bit.

17.3.3 A/D Control Register (ADCR)

The ADCR enables A/D conversion started by an external trigger signal.

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|--|
| 7 | TRGS1 | 0 | R/W | Timer Trigger Select 1 and 0 |
| 6 | TRGS0 | 0 | R/W | <p>Enables the start of A/D conversion by a trigger signal. Only set bits TRGS0 and TRGS1 while conversion is stopped (ADST = 0).</p> <p>00: A/D conversion start by software is enabled</p> <p>01: A/D conversion start by TPU conversion start trigger is enabled</p> <p>10: A/D conversion start by 8-bit timer conversion start trigger is enabled</p> <p>11: A/D conversion start by external trigger pin (ADTRG) is enabled</p> |
| 5, 4 | — | All 1 | — | <p>Reserved</p> <p>These bits are always read as 1 and cannot be modified.</p> |
| 3 | CKS1 | 0 | R/W | Clock Select 1 and 0 |
| 2 | CKS0 | 0 | R/W | <p>These bits specify the A/D conversion time. The conversion time should be changed only when ADST = 0. Specify a setting that gives a value within the range shown in table 27.10 (H8S/2258 Group), table 27.23 (H8S/2239 Group), table 27.35 (H8S/2238B and H8S/ 2236B), table 27.47 (H8S/2238R and H8S/ 2236R), or table 27.57 (H8S/2237 Group and H8S/2227 Group).</p> <p>00: Conversion time = 530 states (max)</p> <p>01: Conversion time = 266 states (max)</p> <p>10: Conversion time = 134 states (max)</p> <p>11: Conversion time = 68 states (max)</p> |
| 1, 0 | — | All 1 | — | <p>Reserved</p> <p>These bits are always read as 1 and cannot be modified.</p> |

17.4 Interface to Bus Master

ADDRA to ADDR D are 16-bit registers. As the data bus to the bus master is 8 bits wide, the bus master accesses to the upper byte of the registers directly while to the lower byte of the registers via the temporary register (TEMP).

Data in ADDR is read in the following way: When the upper-byte data is read, the upper-byte data will be transferred to the CPU and the lower-byte data will be transferred to TEMP. Then, when the lower-byte data is read, the lower-byte data will be transferred to the CPU.

When data in ADDR is read, the data should be read from the upper byte and lower byte in the order. When only the upper-byte data is read, the data is guaranteed. However, when only the lower-byte data is read, the data is not guaranteed.

Figure 17.2 shows data flow when accessing to ADDR.

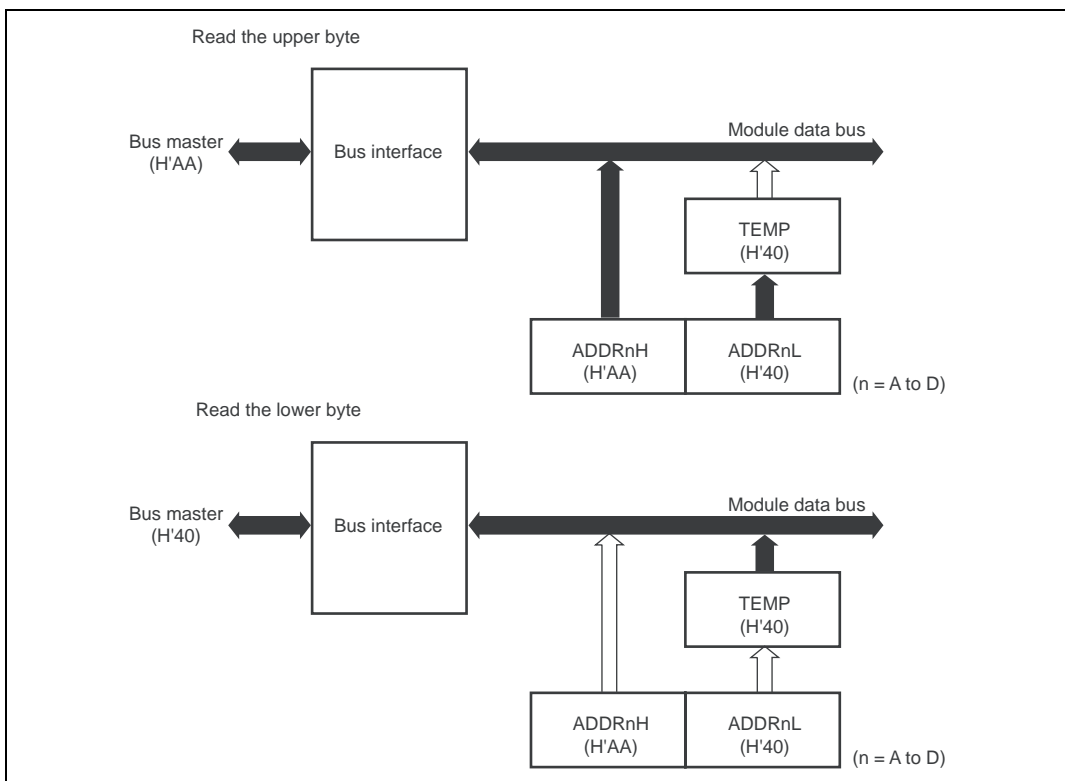


Figure 17.2 Access to ADDR (When Reading H'AA40)

17.5 Operation

The A/D converter operates by successive approximation with 10-bit resolution. It has two operating modes; single mode and scan mode. When changing the operating mode or analog input channel, in order to prevent incorrect operation, first clear the bit ADST to 0 in ADCSR. The ADST bit can be set at the same time as the operating mode or analog input channel is changed.

17.5.1 Single Mode

In single mode, A/D conversion is to be performed only once on the specified single channel. The operations are as follows.

1. A/D conversion is started when the ADST bit is set to 1, according to software, timer conversion start trigger, or external trigger input.
2. When A/D conversion is completed, the result is transferred to the corresponding A/D data register to the channel.
3. On completion of conversion, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit remains set to 1 during A/D conversion. When A/D conversion ends, the ADST bit is automatically cleared to 0 and the A/D converter enters the wait state.

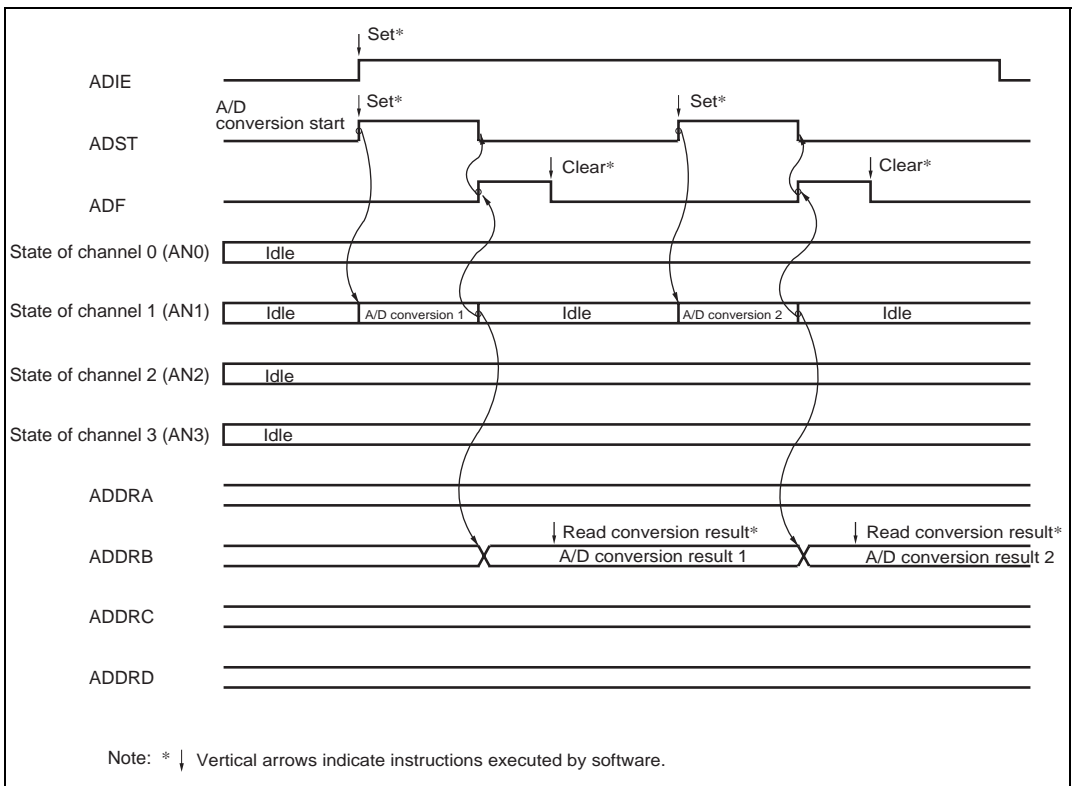
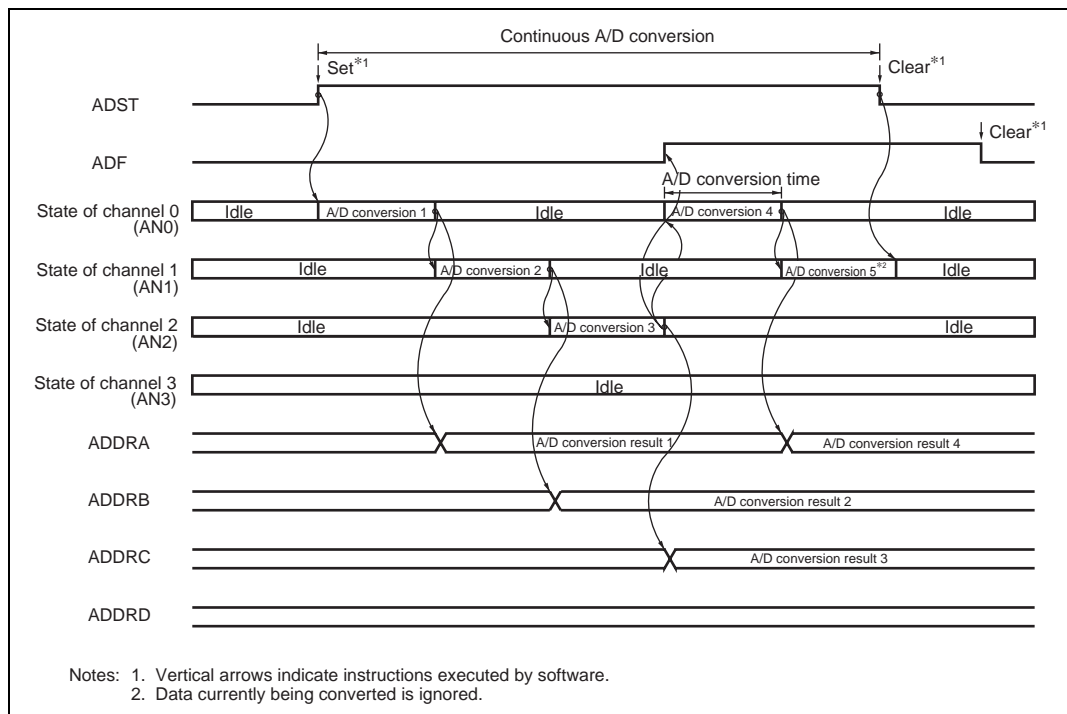


Figure 17.3 Example of A/D converter Operation (Single Mode, Channel 1 Selected)

17.5.2 Scan Mode

In scan mode, A/D conversion is to be performed sequentially on the specified channels (four channels maximum). The operations are as follows.

1. When the ADST bit is set to 1 by software, TPU, timer conversion start trigger, or external trigger, input, A/D conversion starts on the first channel in the group (AN0 when CH2 = 0, AN4 when CH2 = 1).
2. When A/D conversion for each channel is completed, the result is sequentially transferred to the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF flag is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested after A/D conversion ends. Conversion of the first channel in the group starts again.
4. Steps 2 to 3 are repeated as long as the ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion stops and the A/D converter enters the wait state.



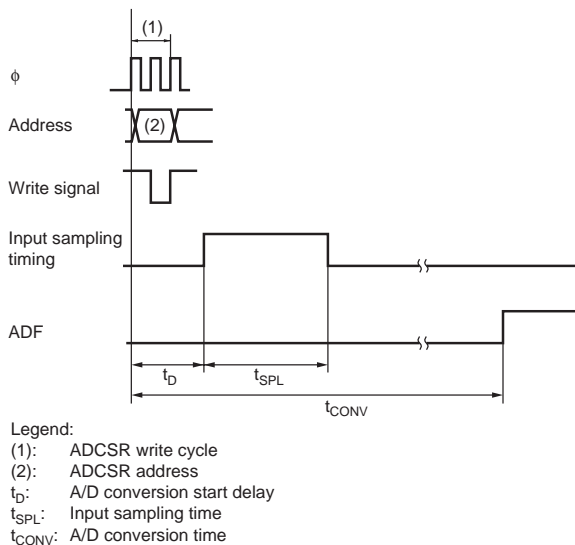
**Figure 17.4 Example of A/D Converter Operation
(Scan Mode, Channels AN0 to AN2 Selected)**

17.5.3 Input Sampling and A/D Conversion Time

The A/D converter has a built-in sample-and-hold circuit. The A/D converter samples the analog input when the A/D conversion start delay time (t_p) has passed after the ADST bit is set to 1, then starts conversion. Figure 17.5 shows the A/D conversion timing. Table 17.3 shows the A/D conversion time.

As indicated in figure 17.5, the A/D conversion time (t_{CONV}) includes t_p and the input sampling time (t_{SPL}). The length of t_p varies depending on the timing of the write access to ADCSR. The total conversion time therefore varies within the ranges indicated in table 17.3.

In scan mode, the values given in table 17.3 apply to the first conversion time. The values given in table 17.4 apply to the second and subsequent conversions.

**Figure 17.5 A/D Conversion Timing****Table 17.3 A/D Conversion Time (Single Mode)**

| Item | Symbol | CKS1 = 0 | | | | | | CKS1 = 1 | | | | | |
|----------------------------|------------|----------|-----|-----|----------|-----|-----|----------|-----|-----|----------|-----|-----|
| | | CKS0 = 0 | | | CKS0 = 1 | | | CKS0 = 0 | | | CKS0 = 1 | | |
| | | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max |
| A/D conversion start delay | t_D | 18 | — | 33 | 10 | — | 17 | 6 | — | 9 | 4 | — | 5 |
| Input sampling time | t_{SPL} | — | 127 | — | — | 63 | — | — | 31 | — | — | 15 | — |
| A/D conversion time | t_{CONV} | 515 | — | 530 | 259 | — | 266 | 131 | — | 134 | 67 | — | 68 |

Note: All values represent the number of states.

Table 17.4 A/D Conversion Time (Scan Mode)

| CKS1 | CKS0 | Conversion Time (State) |
|------|------|-------------------------|
| 0 | 0 | 512 (Fixed) |
| | 1 | 256 (Fixed) |
| 1 | 0 | 128 (Fixed) |
| | 1 | 64 (Fixed) |

17.5.4 External Trigger Input Timing

A/D conversion can be externally triggered. When the TRGS0 and TRGS1 bits are set to 11 in ADCR, external trigger input is enabled at the ADTRG pin. A falling edge at the ADTRG pin sets the ADST bit to 1 in ADCSR, starting A/D conversion. Other operations, in both single and scan modes, are the same as when the bit ADST has been set to 1 by software. Figure 17.6 shows the timing.

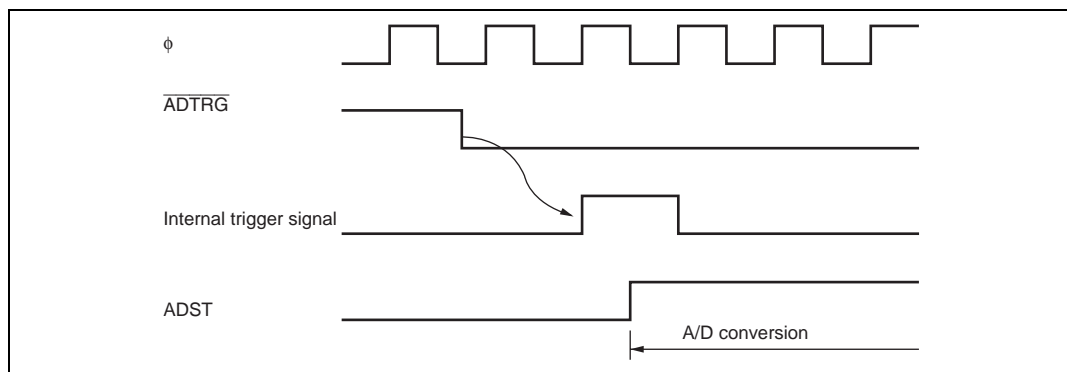


Figure 17.6 External Trigger Input Timing

17.6 Interrupt Source

The A/D converter generates an A/D conversion end interrupt (ADI) at the end of A/D conversion. Setting the ADIE bit to 1 enables ADI interrupt requests while the bit ADF in ADCSR is set to 1 after A/D conversion is completed. The DMAC* and the DTC can be activated by an ADI interrupt. Having the converted data read by the DMAC* or the DTC in response to an ADI interrupt enables continuous conversion without imposing a load on software.

Note: * Supported only by the H8S/2239 Group.

Table 17.5 A/D Converter Interrupt Source

| Name | Interrupt Source | Interrupt Source Flag | DTC Activation | DMAC* Activation |
|------|--------------------------|-----------------------|----------------|------------------|
| ADI | A/D conversion completed | ADF | Possible | Possible |

Note: * Supported only by the H8S/2239 Group.

17.7 A/D Conversion Accuracy Definitions

This LSI's A/D conversion accuracy definitions are given below.

- Resolution
The number of A/D converter digital output codes.
- Quantization error
The deviation inherent in the A/D converter, given by 1/2 LSB (see figure 17.7).
- Offset error
The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from the minimum voltage value B'0000000000 (H'000) to B'0000000001 (H'001) (see figure 17.8).
- Full-scale error
The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from B'1111111110 (H'3FE) to B'1111111111 (H'3FF) (see figure 17.8).
- Nonlinearity error
The error with respect to the ideal A/D conversion characteristic between zero voltage and full-scale voltage. Does not include offset error, full-scale error, or quantization error (see figure 17.8).
- Absolute accuracy
The deviation between the digital value and the analog input value. Includes offset error, full-scale error, quantization error, and nonlinearity error.

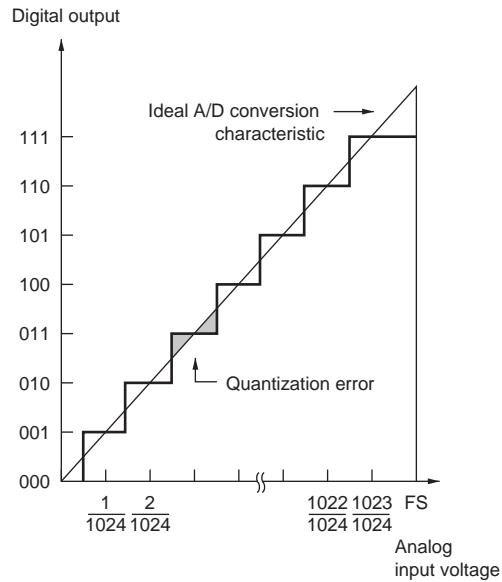


Figure 17.7 A/D Conversion Accuracy Definitions

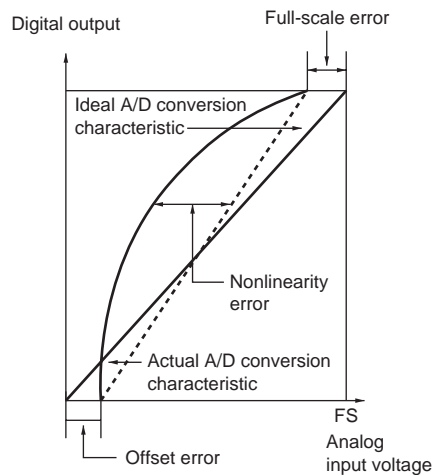


Figure 17.8 A/D Conversion Accuracy Definitions

17.8 Usage Notes

17.8.1 Module Stop Mode Setting

Operation of the A/D converter can be disabled or enabled using the module stop control register. The initial setting is for operation of the A/D converter to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 24, Power-Down Modes.

17.8.2 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion accuracy is guaranteed for an input signal for which the signal source impedance is $5\text{ k}\Omega$ or less. This specification is provided to enable the A/D converter's sample-and-hold circuit input capacitance to be charged within the sampling time; if the sensor output impedance exceeds $5\text{ k}\Omega$, charging may be insufficient and it may not be possible to guarantee A/D conversion accuracy. However, for A/D conversion in single mode with a large capacitance provided externally, the input load will essentially comprise only the internal input resistance of $10\text{ k}\Omega$, and the signal source impedance is ignored. However, as a low-pass filter effect is obtained in this case, it may not be possible to follow an analog signal with a large differential coefficient (e.g., $5\text{ mV}/\mu\text{s}$ or greater) (see figure 17.9). When converting a high-speed analog signal, a low-impedance buffer should be inserted.

17.8.3 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute accuracy. Be sure to make the connection to an electrically stable GND such as AVss.

Care is also required to insure that filter circuits do not communicate with digital signals on the mounting board (i.e., acting as antennas).

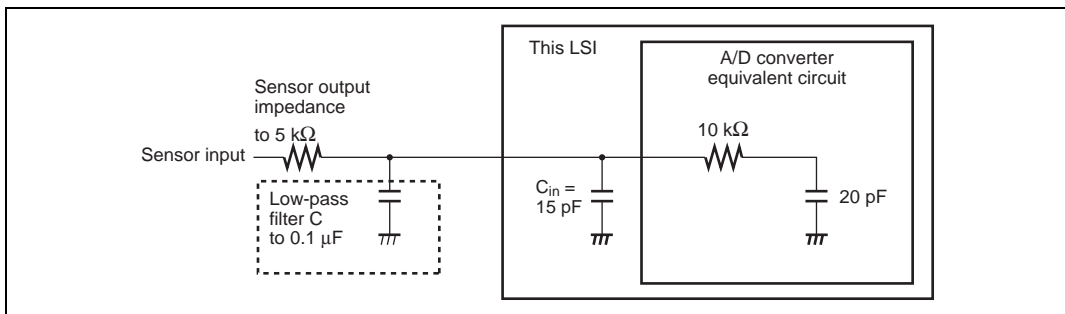


Figure 17.9 Example of Analog Input Circuit

17.8.4 Range of Analog Power Supply and Other Pin Settings

If the conditions below are not met, the reliability of the device may be adversely affected.

- Analog input voltage range
The voltage applied to analog input pin ANn during A/D conversion should be in the range $AV_{SS} \leq ANn \leq AV_{CC}$.
- Relationship between AV_{CC} , AV_{SS} and V_{CC} , V_{SS}
Set $AV_{SS} = V_{SS}$ as the relationship between AV_{CC} , AV_{SS} and V_{CC} , V_{SS} . If the A/D converter is not used, the AV_{CC} and AV_{SS} pins must not be left open. In addition, AN0 and AN1 may be used only when $V_{CC} = AV_{CC}$ in the case of the H8S/2239 Group, H8S/2227 Group, H8S/2238R, and H8S/2236R.
- Vref range
The reference voltage input from the Vref pin should be set to AV_{CC} or less.

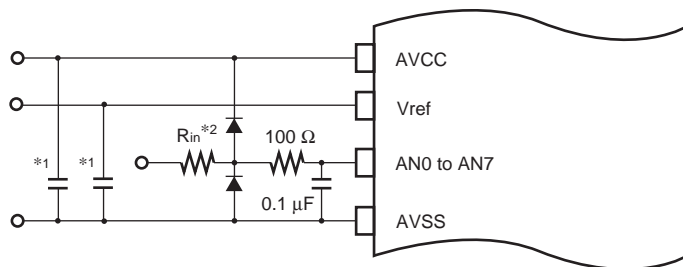
17.8.5 Notes on Board Design

In board design, digital circuitry and analog circuitry should be as mutually isolated as possible, and layout in which digital circuit signal lines and analog circuit signal lines cross or are in close proximity should be avoided as far as possible. Failure to do so may result in incorrect operation of the analog circuitry due to inductance, adversely affecting A/D conversion values. Also, digital circuitry must be isolated from the analog input signals (AN0 to AN7), and analog power supply (AV_{CC}) by the analog ground (AV_{SS}). Also, the analog ground (AV_{SS}) should be connected at one point to a stable digital ground (V_{SS}) on the board.

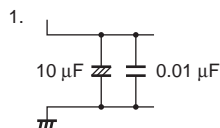
17.8.6 Notes on Noise Countermeasures

A protection circuit should be connected in order to prevent damage due to abnormal voltage, such as an excessive surge at the analog input pins (AN0 to AN7), between AV_{CC} and AV_{SS} , as shown in figure 17.10. Also, the bypass capacitors connected to AV_{CC} and the filter capacitor connected to AN0 to AN7 must be connected to AV_{SS} .

If a filter capacitor is connected, the input currents at the analog input pins (AN0 to AN7) are averaged, and so an error may arise. Also, when A/D conversion is performed frequently, as in scan mode, if the current charged and discharged by the capacitance of the sample-and-hold circuit in the A/D converter exceeds the current input via the input impedance (R_{in}), an error will arise in the analog input pin voltage. Careful consideration is therefore required when deciding circuit constants.



Notes: Values are reference values.

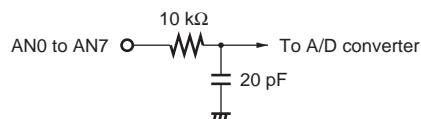


2. R_{in} : Input impedance

Figure 17.10 Example of Analog Input Protection Circuit

Table 17.6 Analog Pin Specifications

| Item | Min | Max | Unit |
|-------------------------------------|-----|-----|------|
| Analog input capacitance | — | 20 | pF |
| Permissible signal source impedance | — | 5 | kΩ |



Note: Values are reference values.

Figure 17.11 Analog Input Pin Equivalent Circuit

Section 18 D/A Converter

18.1 Features

- 8-bit resolution
- Two output channels
- Conversion time: 10 μ s, maximum (when load capacitance is 20 pF)
- Output voltage: 0 V to Vref
- Module stop mode can be set

Note: The D/A converter is not included in the H8S/2227 Group.

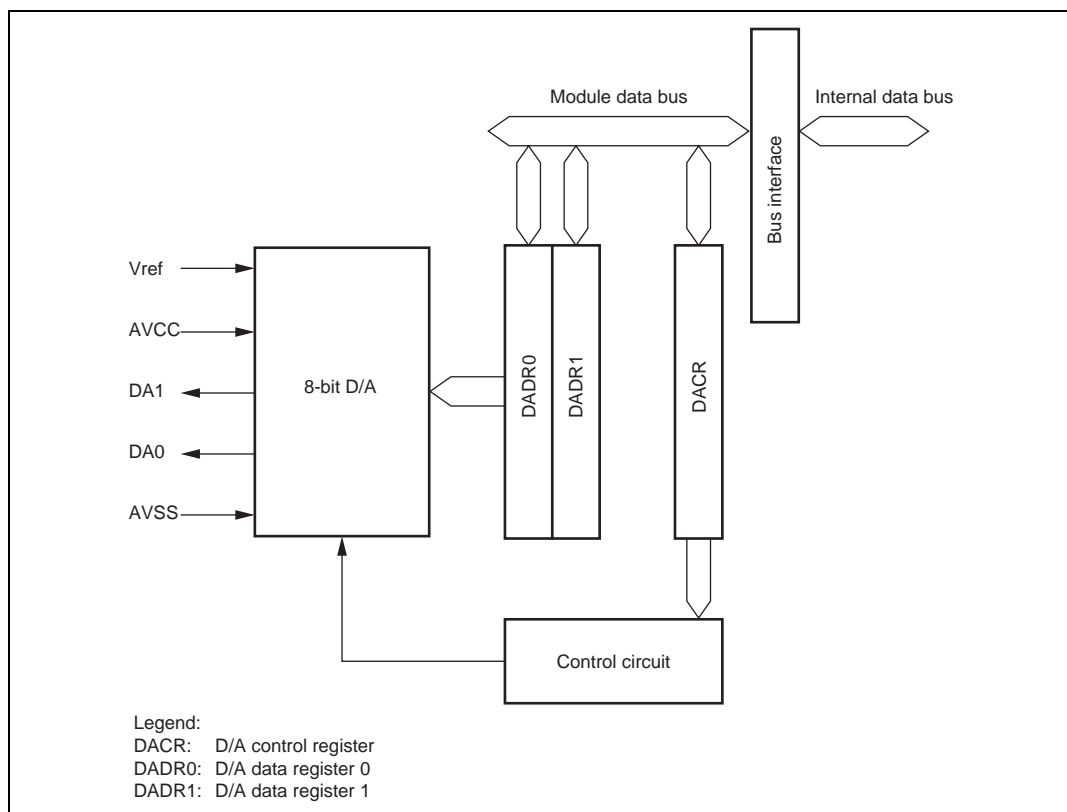


Figure 18.1 Block Diagram of D/A Converter

18.2 Input/Output Pins

Table 18.1 shows the pin configuration for the D/A converter.

Table 18.1 Pin Configuration

| Pin Name | Symbol | I/O | Function |
|-------------------------|-----------|--------|---|
| Analog power supply pin | AV_{CC} | Input | Analog block power supply |
| Analog ground pin | AV_{SS} | Input | Analog block ground and reference voltage |
| Analog output pin 0 | DA0 | Output | Channel 0 analog output pin |
| Analog output pin 1 | DA1 | Output | Channel 1 analog output pin |
| Reference voltage pin | Vref | Input | Reference voltage for analog block |

18.3 Register Description

The D/A converter has the following registers. For details on the module stop control register, refer to section 24.1.2, Module Stop Control Registers A to C (MSTPCRA to MSTPCRC).

- D/A data register 0 (DADR0)
- D/A data register 1 (DADR1)
- D/A control register (DACR)

18.3.1 D/A Data Registers 0 and 1 (DADR0 and DADR1)

DADR0 and DADR1 are 8-bit readable/writable registers that store data for D/A conversion. When analog output is permitted, D/A data register contents are converted and output to analog output pins.

18.3.2 D/A Control Register (DACR)

DACR controls D/A converter operation.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 | DAOE1 | 0 | R/W | D/A Output Enable 1 Controls D/A conversion and analog output 0: Analog output DA1 is disabled 1: D/A conversion for channel 1 and analog output DA1 are enabled |
| 6 | DAOE0 | 0 | R/W | D/A Output Enable 0 Controls D/A conversion and analog output 0: Analog output DA0 is disabled 1: D/A conversion for channel 0 and analog output DA0 are enabled |
| 5 | DAE | 0 | R/W | D/A Enable Controls D/A conversion in conjunction with the DAOE0 and DAOE1 bits. When the DAE bit is cleared to 0, D/A conversion for channels 0 and 1 are controlled individually. When DAE is set to 1, D/A conversion for channels 0 and 1 are controlled as one. Conversion result output is controlled by the DAOE0 and DAOE1 bits. For details, see table 18.2. |
| 4 to 0 | — | All 1 | — | Reserved These bits are always read as 1 and cannot be modified. |

Table 18.2 D/A Conversion Control

| Bit 5 | Bit 7 | Bit 6 | Description |
|-------|-------|-------|---|
| DAE | DAOE1 | DAOE0 | |
| 0 | 0 | 0 | Disables D/A Conversion |
| | | 1 | Enables D/A Conversion for channel 0 |
| | 1 | 0 | Enables D/A Conversion for channel 1 |
| | | 1 | Enables D/A Conversion for channels 0 and 1 |
| 1 | 0 | 0 | Disables D/A Conversion |
| | | 1 | Enables D/A Conversion for channels 0 and 1 |
| | 1 | 0 | |
| | | 1 | |

18.4 Operation

Two channels of the D/A converter can perform conversion individually.

When the DAOE bit in DACR is set to 1, D/A conversion is enabled and the conversion results are output.

An example of D/A conversion of channel 0 is shown below. The operation timing is shown in figure 18.2.

1. Write conversion data to DADR0.
2. When the DAOE0 bit in DACR is set to 1, D/A conversion starts. After the interval of t_{DCONV} , the conversion results are output from the analog output pin DA0. The conversion results are output continuously until DADR0 is modified or DAOE0 bit is cleared to 0. The output value is calculated by the following formula:

$$(\text{DADR contents}) / 256 \times V_{ref}$$
3. Conversion starts immediately after DADR0 is modified. After the interval of t_{DCONV} , conversion results are output.
4. When the DAOE bit is cleared to 0, analog output is disabled.

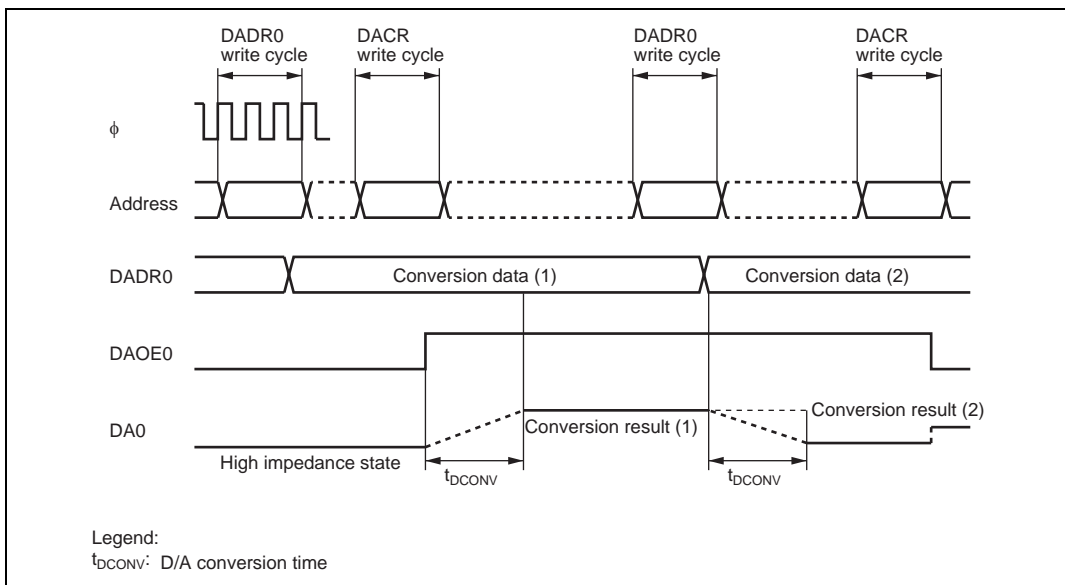


Figure 18.2 D/A Converter Operation Example

18.5 Usage Notes

18.5.1 Analog Power Supply Current in Power-Down Mode

If this LSI enters a power-down mode such as software standby, watch, subactive, subsleep, and module stop modes while D/A conversion is enabled, the D/A cannot retain analog outputs within the given D/A absolute accuracy* although it retains digital values. The analog power supply current is approximately the same as that during D/A conversion. To reduce analog power supply current in power-down mode, clear the DAOE0, DAOE1 and DAE bits to 0 to disable D/A outputs before entering the mode.

Note: * The H8S/2258 Group, H8S/2238B, and H8S/2236B satisfy the specified D/A absolute accuracy.

18.5.2 Setting for Module Stop Mode

It is possible to enable/disable the D/A converter operation using the module stop control register, the D/A converter does not operate by the initial value of the register. The register can be accessed by releasing the module stop mode. For more details, see section 24, Power-Down Modes.

Section 19 RAM

The H8S/2239 has 32 kbytes of on-chip high-speed static RAM. The H8S/2258, H8S/2238B, H8S/2238R, H8S/2237, and H8S/2227 have 16 kbytes of on-chip high-speed static RAM. The H8S/2256, H8S/2236B, H8S/2236R have 8 kbytes of on-chip high-speed static RAM. The H8S/2235, H8S/2233, H8S/2225, H8S/2224, and H8S/2223 have 4 kbytes of on-chip high-speed static RAM. The RAM is connected to the CPU by a 16-bit data bus, enabling one-state access by the CPU to both byte data and word data.

Section 20 Flash Memory (F-ZTAT Version)

The features of the flash memory are summarized below.

The block diagram of the flash memory is shown in figure 20.1.

20.1 Features

- Capacity
 - H8S/2239: 384 kbytes
 - H8S/2258: 256 kbytes
 - H8S/2238B: 256 kbytes
 - H8S/2238R: 256 kbytes
 - H8S/2227: 128 kbytes
- Programming/erase methods

The flash memory is programmed 128 bytes at a time. Erase is performed in single-block units. The flash memory of the H8S/2239 is configured as follows: 64 kbytes \times 5 blocks, 32 kbytes \times 1 block, and 4 kbytes \times 8 blocks. The flash memory of the H8S/2258, H8S/2238B, and H8S/2238R is configured as follows: 64 kbytes \times 3 blocks, 32 kbytes \times 1 block, and 4 kbytes \times 8 blocks. The flash memory of the H8S/2227 is configured as follows: 32 kbytes \times 2 blocks, 28 kbytes \times 1 block, 16 kbytes \times 1 block, 8 kbytes \times 2 blocks, and 1 kbyte \times 4 blocks. To erase the entire flash memory, each block must be erased in turn.
- Reprogramming capability

The flash memory can be reprogrammed for 100 times.
- Two programming modes
 - Boot mode
 - User program mode

On-board programming/erasing can be done in boot mode, in which the boot program built into the chip is started to erase or program of the entire flash memory. In normal user program mode, individual blocks can be erased or programmed.
- Automatic bit rate adjustment

For data transfer in boot mode, this LSI's bit rate can be automatically adjusted to match the transfer bit rate of the host.
- Programming/erasing protection

There are three protect modes, hardware, software, and error protect, which allow protected status to be designated for flash memory program/erase operations.

- Programmer mode

Flash memory can be programmed/erased in programmer mode using a PROM programmer, as well as in on-board programming mode.

- Emulation function for flash memory in RAM

The real-time emulation for programming of flash memory is possible by overlapping the flash memory to a part of RAM.

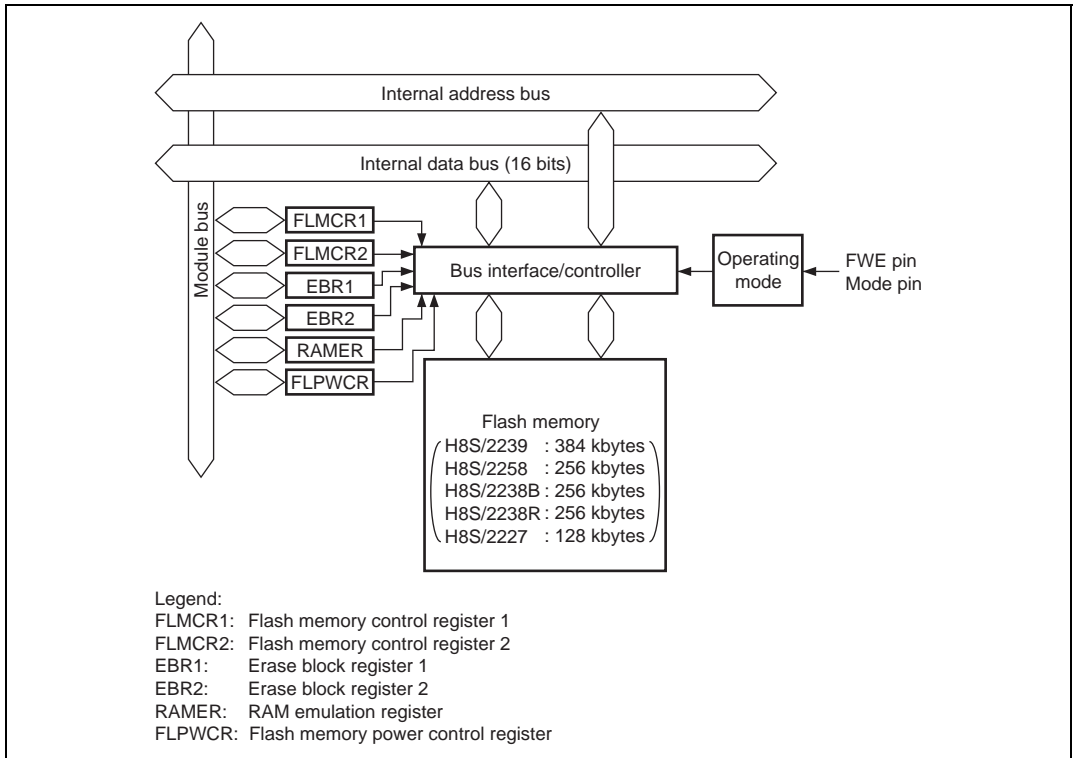


Figure 20.1 Block Diagram of Flash Memory

20.2 Mode Transitions

When the mode pins and the FWE pin are set in the reset state and a reset-start is executed, this LSI enters an operating mode as shown in figure 20.2. In user mode, flash memory can be read but not programmed or erased.

The boot, user program and programmer modes are provided as modes to write and erase the flash memory.

The differences between boot mode and user program mode are shown in table 20.1.

Figure 20.3 shows the operation flow for boot mode and figure 20.4 shows that for user program mode.

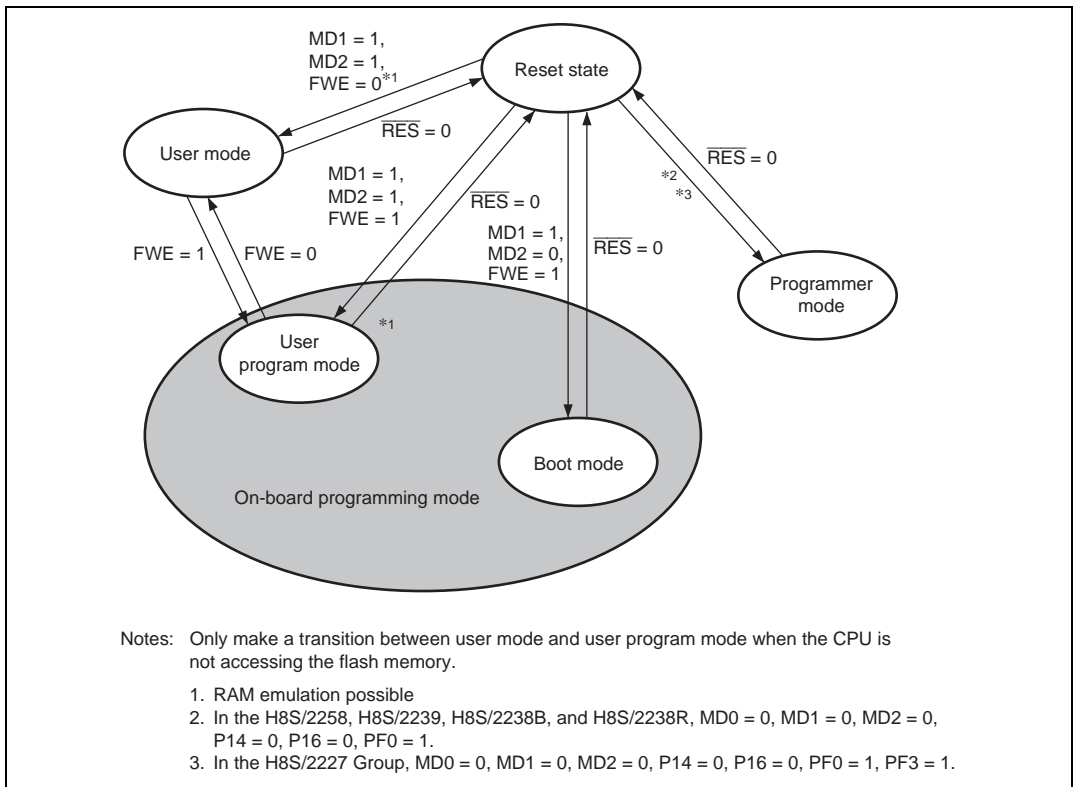


Figure 20.2 Flash Memory State Transitions

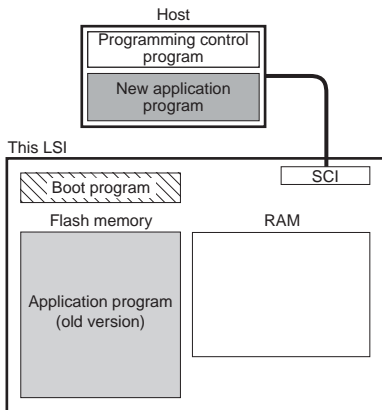
Table 20.1 Differences between Boot Mode and User Program Mode

| | Boot Mode | User Program Mode |
|------------------------------|------------------------|---|
| Total erase | Yes | Yes |
| Block erase | No | Yes |
| Programming control program* | Program/program-verify | Program/program-verify/erase/erase-verify/emulation |

Note: * To be provided by the user, in accordance with the recommended algorithm.

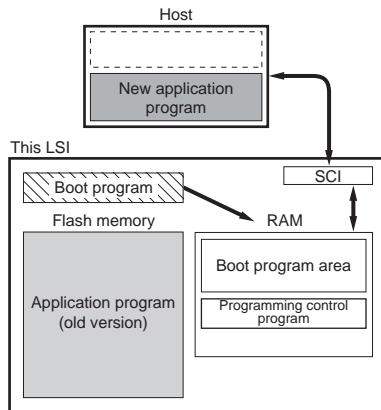
1. Initial state

The old program version or data remains written in the flash memory. The user should prepare the programming control program and new application program beforehand in the host.



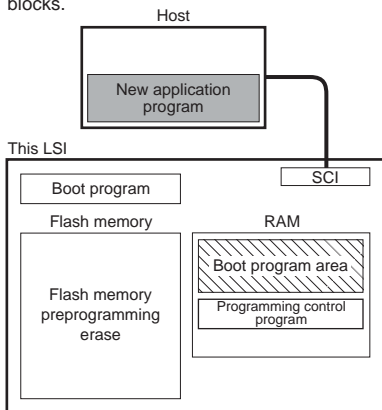
2. Programming control program transfer

When boot mode is entered, the boot program in this LSI (originally incorporated in the chip) is started and the programming control program in the host is transferred to RAM via SCI communication. The boot program required for flash memory erasing is automatically transferred to the RAM boot program area.



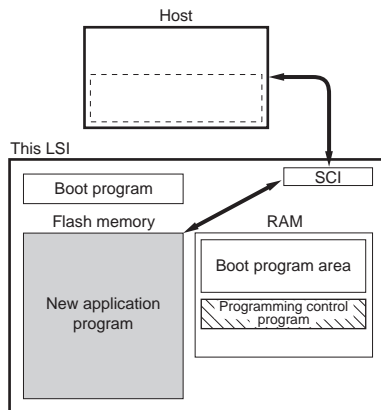
3. Flash memory initialization

The erase program in the boot program area (in RAM) is executed, and the flash memory is initialized (to H'FF). In boot mode, total flash memory erasure is performed, without regard to blocks.



4. Writing new application program

The programming control program transferred from the host to RAM is executed, and the new application program in the host is written into the flash memory.

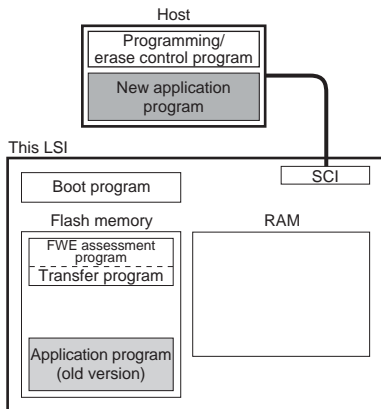


Program execution state

Figure 20.3 Boot Mode (Example)

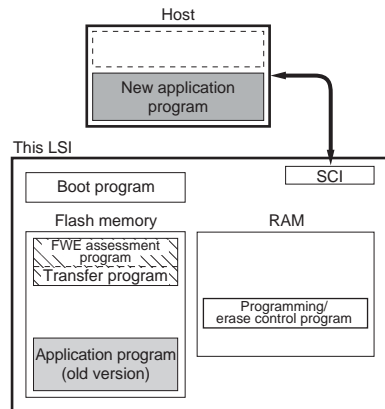
1. Initial state

The FWE assessment program that confirms that user program mode has been entered, and the program that will transfer the programming/erase control program from flash memory to on-chip RAM should be written into the flash memory by the user beforehand. The programming/erase control program should be prepared in the host or in the flash memory.



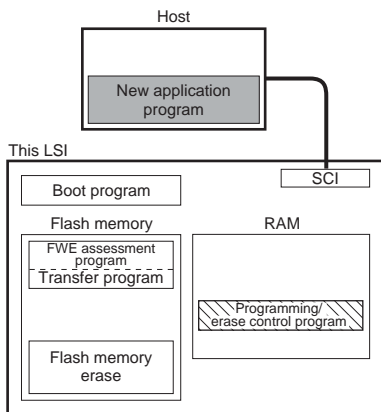
2. Programming/erase control program transfer

When user program mode is entered, user software confirms this fact, executes transfer program in the flash memory, and transfers the programming/erase control program to RAM.



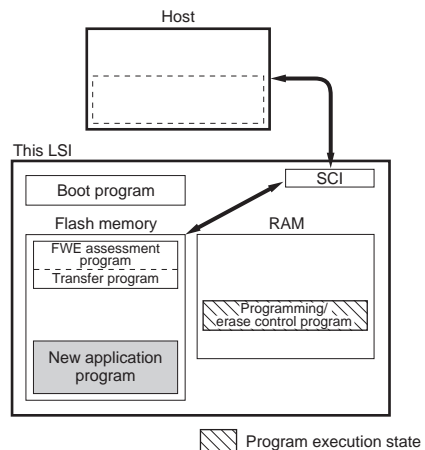
3. Flash memory initialization

The programming/erase program in RAM is executed, and the flash memory is initialized (to H'FF). Erasing can be performed in block units, but not in byte units.



4. Writing new application program

Next, the new application program in the host is written into the erased flash memory blocks. Do not write to unerased blocks.



Program execution state

Figure 20.4 User Program Mode (Example)

20.3 Block Configuration

Figure 20.5 shows the block configuration of 384-kbyte flash memory. Figure 20.6 shows the block configuration of 256-kbyte flash memory. Figure 20.7 shows the block configuration of 128-kbyte flash memory. The thick lines indicate erasing units, the narrow lines indicate programming units, and the values are addresses. The 384-kbyte flash memory is divided into 4 kbytes (8 blocks), 32 kbytes (1 block), and 64 kbytes (5 blocks). The 256-kbyte flash memory is divided into 4 kbytes (8 blocks), 32 kbytes (1 block), and 64 kbytes (3 blocks). The 128-kbyte flash memory is divided into 1 kbyte (4 blocks), 16 kbytes (1 block), 28 kbytes (1 block), 8 kbytes (2 blocks), and 32 kbytes (2 blocks). Erasing is performed in these units. Programming is performed in 128-byte units starting from an address with lower eight bits H'00 or H'80.

| | | | | | |
|---------------------------------|----------|----------|----------|---------------------------------|----------|
| EB0 Erase unit 4 kbytes | H'000000 | H'000001 | H'000002 | ← Programming unit: 128 bytes → | H'00007F |
| | | | | | H'000FFF |
| EB1 Erase unit 4 kbytes | H'001000 | H'001001 | H'001002 | ← Programming unit: 128 bytes → | H'00107F |
| | | | | | H'001FFF |
| EB2 Erase unit 4 kbytes | H'002000 | H'002001 | H'002002 | ← Programming unit: 128 bytes → | H'00207F |
| | | | | | H'002FFF |
| EB3 Erase unit 4 kbytes | H'003000 | H'003001 | H'003002 | ← Programming unit: 128 bytes → | H'00307F |
| | | | | | H'003FFF |
| EB4 Erase unit 4 kbytes | H'004000 | H'004001 | H'004002 | ← Programming unit: 128 bytes → | H'00407F |
| | | | | | H'004FFF |
| EB5 Erase unit 4 kbytes | H'005000 | H'005001 | H'005002 | ← Programming unit: 128 bytes → | H'00507F |
| | | | | | H'005FFF |
| EB6 Erase unit 4 kbytes | H'006000 | H'006001 | H'006002 | ← Programming unit: 128 bytes → | H'00607F |
| | | | | | H'006FFF |
| EB7 Erase unit 4 kbytes | H'007000 | H'007001 | H'007002 | ← Programming unit: 128 bytes → | H'00707F |
| | | | | | H'007FFF |
| EB8 Erase unit 32 kbytes | H'008000 | H'008001 | H'008002 | ← Programming unit: 128 bytes → | H'00807F |
| | | | | | H'00FFFF |
| EB9 Erase unit 64 kbytes | H'010000 | H'010001 | H'010002 | ← Programming unit: 128 bytes → | H'01007F |
| | | | | | H'01FFFF |
| EB10 Erase unit 64 kbytes | H'020000 | H'020001 | H'020002 | ← Programming unit: 128 bytes → | H'02007F |
| | | | | | H'02FFFF |
| EB11 Erase unit 64 kbytes | H'030000 | H'030001 | H'030002 | ← Programming unit: 128 bytes → | H'03007F |
| | | | | | H'03FFFF |
| EB12 Erase unit 64 kbytes | H'040000 | H'040001 | H'040002 | ← Programming unit: 128 bytes → | H'04007F |
| | | | | | H'04FFFF |
| EB13 Erase unit 64 kbytes | H'050000 | H'050001 | H'050002 | ← Programming unit: 128 bytes → | H'05007F |
| | | | | | H'05FFFF |

Figure 20.5 Block Configuration of 384-kbyte Flash Memory

| | | | | | |
|---------------------------------|----------|----------|----------|---------------------------------|----------|
| EB0 Erase unit 4 kbytes | H'000000 | H'000001 | H'000002 | ← Programming unit: 128 bytes → | H'00007F |
| | | | | | H'000FFF |
| | | | | | |
| EB1 Erase unit 4 kbytes | H'001000 | H'001001 | H'001002 | ← Programming unit: 128 bytes → | H'00107F |
| | | | | | H'001FFF |
| | | | | | |
| EB2 Erase unit 4 kbytes | H'002000 | H'002001 | H'002002 | ← Programming unit: 128 bytes → | H'00207F |
| | | | | | H'002FFF |
| | | | | | |
| EB3 Erase unit 4 kbytes | H'003000 | H'003001 | H'003002 | ← Programming unit: 128 bytes → | H'00307F |
| | | | | | H'003FFF |
| | | | | | |
| EB4 Erase unit 4 kbytes | H'004000 | H'004001 | H'004002 | ← Programming unit: 128 bytes → | H'00407F |
| | | | | | H'004FFF |
| | | | | | |
| EB5 Erase unit 4 kbytes | H'005000 | H'005001 | H'005002 | ← Programming unit: 128 bytes → | H'00507F |
| | | | | | H'005FFF |
| | | | | | |
| EB6 Erase unit 4 kbytes | H'006000 | H'006001 | H'006002 | ← Programming unit: 128 bytes → | H'00607F |
| | | | | | H'006FFF |
| | | | | | |
| EB7 Erase unit 4 kbytes | H'007000 | H'007001 | H'007002 | ← Programming unit: 128 bytes → | H'00707F |
| | | | | | H'007FFF |
| | | | | | |
| EB8 Erase unit 32 kbytes | H'008000 | H'008001 | H'008002 | ← Programming unit: 128 bytes → | H'00807F |
| | | | | | H'00FFFF |
| | | | | | |
| EB9 Erase unit 64 kbytes | H'010000 | H'010001 | H'010002 | ← Programming unit: 128 bytes → | H'01007F |
| | | | | | H'01FFFF |
| | | | | | |
| EB10 Erase unit 64 kbytes | H'020000 | H'020001 | H'020002 | ← Programming unit: 128 bytes → | H'02007F |
| | | | | | H'02FFFF |
| | | | | | |
| EB11 Erase unit 64 kbytes | H'030000 | H'030001 | H'030002 | ← Programming unit: 128 bytes → | H'03007F |
| | | | | | H'03FFFF |
| | | | | | |

Figure 20.6 Block Configuration of 256-kbyte Flash Memory

| | | | | | |
|--------------------------------|----------|----------|----------|---------------------------------|----------|
| EB0 Erase unit 1 kbyte | H'000000 | H'000001 | H'000002 | ← Programming unit: 128 bytes → | H'00007F |
| | H'000380 | H'000381 | H'000382 | ----- | H'0003FF |
| EB1 Erase unit 1 kbyte | H'000400 | H'000401 | H'000402 | ← Programming unit: 128 bytes → | H'00047F |
| | H'000780 | H'000781 | H'000782 | ----- | H'0007FF |
| EB2 Erase unit 1 kbyte | H'000800 | H'000801 | H'000802 | ← Programming unit: 128 bytes → | H'00087F |
| | H'000B80 | H'000B81 | H'000B82 | ----- | H'000BFF |
| EB3 Erase unit 1 kbyte | H'000C00 | H'000C01 | H'000C02 | ← Programming unit: 128 bytes → | H'000C7F |
| | H'000F80 | H'000F81 | H'000F82 | ----- | H'000FFF |
| EB4 Erase unit 28 kbytes | H'001000 | H'001001 | H'001002 | ← Programming unit: 128 bytes → | H'00107F |
| | H'007F80 | H'007F81 | H'007F82 | ----- | H'007FFF |
| EB5 Erase unit 16 kbytes | H'008000 | H'008001 | H'008002 | ← Programming unit: 128 bytes → | H'00807F |
| | H'00BF80 | H'00BF81 | H'00BF82 | ----- | H'00BFFF |
| EB6 Erase unit 8 kbytes | H'00C000 | H'00C001 | H'00C002 | ← Programming unit: 128 bytes → | H'00C07F |
| | H'00DF80 | H'00DF81 | H'00DF82 | ----- | H'00DFFF |
| EB7 Erase unit 8 kbytes | H'00E000 | H'00E001 | H'00E002 | ← Programming unit: 128 bytes → | H'00E07F |
| | H'00FF80 | H'00FF81 | H'00FF82 | ----- | H'00FFFF |
| EB8 Erase unit 32 kbytes | H'010000 | H'010001 | H'010002 | ← Programming unit: 128 bytes → | H'01007F |
| | H'017F80 | H'017F81 | H'017F82 | ----- | H'017FFF |
| EB9 Erase unit 32 kbytes | H'018000 | H'018001 | H'018002 | ← Programming unit: 128 bytes → | H'01807F |
| | H'01FF80 | H'01FF81 | H'01FF82 | ----- | H'01FFFF |

Figure 20.7 Block Configuration of 128-kbyte Flash Memory

20.4 Input/Output Pins

The flash memory is controlled by means of the pins shown in table 20.2.

Table 20.2 Pin Configuration

| Pin Name | I/O | Function |
|----------|--------|--|
| RES | Input | Reset |
| FWE | Input | Flash program/erase protection by hardware |
| MD2 | Input | Sets this LSI's operating mode |
| MD1 | Input | Sets this LSI's operating mode |
| MD0 | Input | Sets this LSI's operating mode |
| PF0 | Input | Sets MCU operating mode in programmer mode |
| P16 | Input | Sets MCU operating mode in programmer mode |
| P14 | Input | Sets MCU operating mode in programmer mode |
| TxD* | Output | Serial transmit data output |
| RxD* | Input | Serial receive data input |

Note: * SCI_2 (TxD2, RxD2) is used for the H8S/2258, H8S/2239, H8S/2238B, and H8S/2238R, and SCI_0 (TxD0, RxD0) for the H8S/2227.

20.5 Register Descriptions

The flash memory has the following registers.

- Flash memory control register 1 (FLMCR1)
- Flash memory control register 2 (FLMCR2)
- Erase block register 1 (EBR1)
- Erase block register 2 (EBR2)
- RAM emulation register (RAMER)
- Flash memory power control register (FLPWCR)
- Serial control register X (SCRX)

The registers described above are not present in the masked ROM version. If a register described above is read in the masked ROM version, an undefined value will be returned.

20.5.1 Flash Memory Control Register 1 (FLMCR1)

FLMCR1 is a register that makes the flash memory change to program mode, program-verify mode, erase mode, or erase-verify mode. For details on register setting, refer to section 20.8, Flash Memory Programming/Erasing.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | FWE | — | R | Flash Write Enable Bit Reflects the input level at the FWE pin. It is set to 1 when a low level is input to the FWE pin, and cleared to 0 when a high level is input. When this bit is cleared to 0, the flash memory changes to hardware protect mode. |
| 6 | SWE1 | 0 | R/W | Software Write Enable Bit When this bit is set to 1, flash memory programming/erasing is enabled. When this bit is cleared to 0, bits 5 to 0 in FLMCR1 register and all EBR1 and EBR2 bits cannot be set. [Setting condition] When FWE = 1 |
| 5 | ESU1 | 0 | R/W | Erase Setup Bit When this bit is set to 1, the flash memory changes to the erase setup state. When it is cleared to 0, the erase setup state is cancelled. Set this bit to 1 before setting the E1 bit in FLMCR1. [Setting condition] When FWE = 1 and SWE1 = 1 |
| 4 | PSU1 | 0 | R/W | Program Setup Bit When this bit is set to 1, the flash memory changes to the program setup state. When it is cleared to 0, the program setup state is cancelled. Set this bit to 1 before setting the P1 bit in FLMCR1. [Setting condition] When FWE = 1 and SWE1 = 1 |
| 3 | EV1 | 0 | R/W | Erase-Verify When this bit is set to 1, the flash memory changes to erase-verify mode. When it is cleared to 0, erase-verify mode is cancelled. [Setting condition] When FWE = 1 and SWE1 = 1 |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | PV1 | 0 | R/W | <p>Program-Verify</p> <p>When this bit is set to 1, the flash memory changes to program-verify mode. When it is cleared to 0, program-verify mode is cancelled.</p> <p>[Setting condition]</p> <p>When FWE = 1 and SWE1 = 1</p> |
| 1 | E1 | 0 | R/W | <p>Erase</p> <p>When this bit is set to 1, and while the SWE1 and ESU1 bits are 1, the flash memory changes to erase mode. When it is cleared to 0, erase mode is cancelled.</p> <p>[Setting condition]</p> <p>When FWE = 1, SWE1 = 1, and ESU1 = 1</p> |
| 0 | P1 | 0 | R/W | <p>Program</p> <p>When this bit is set to 1, and while the SWE1 and PSU1 bits are 1, the flash memory changes to program mode. When it is cleared to 0, program mode is cancelled.</p> <p>When FWE = 1, SWE1 = 1, and PSU1 = 1</p> |

20.5.2 Flash Memory Control Register 2 (FLMCR2)

FLMCR2 is a register that displays the state of flash memory programming/erasing. FLMCR2 is a read-only register, and should not be written to.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | FLER | 0 | R | <p>Indicates that an error has occurred during an operation on flash memory (programming or erasing). When FLER is set to 1, flash memory goes to the error-protection state.</p> <p>See section 20.9.3, Error Protection, for details.</p> |
| 6 to 0 | — | All 0 | R | <p>Reserved</p> <p>These bits are always read as 0.</p> |

20.5.3 Erase Block Register 1 (EBR1)

EBR1 specifies the flash memory erase area block. EBR1 is initialized to H'00 when the SWE1 bit in FLMCR1 is 0. Do not set more than one bit at a time, as this will cause all the bits in EBR1 and EBR2 to be automatically cleared to 0.

- 384-kbyte or 256-kbyte Flash Memory

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 7 | EB7 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB7 (H'007000 to H'007FFF) will be erased. |
| 6 | EB6 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB6 (H'006000 to H'006FFF) will be erased. |
| 5 | EB5 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB5 (H'005000 to H'005FFF) will be erased. |
| 4 | EB4 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB4 (H'004000 to H'004FFF) will be erased. |
| 3 | EB3 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB3 (H'003000 to H'003FFF) will be erased. |
| 2 | EB2 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB2 (H'002000 to H'002FFF) will be erased. |
| 1 | EB1 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB1 (H'001000 to H'001FFF) will be erased. |
| 0 | EB0 | 0 | R/W | When this bit is set to 1, 4 kbytes of EB0 (H'000000 to H'000FFF) will be erased. |

- 128-kbyte Flash Memory

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | EB7 | 0 | R/W | When this bit is set to 1, 8 kbytes of EB7 (H'00E000 to H'00FFFF) will be erased. |
| 6 | EB6 | 0 | R/W | When this bit is set to 1, 8 kbytes of EB6 (H'00C000 to H'00DFFF) will be erased. |
| 5 | EB5 | 0 | R/W | When this bit is set to 1, 16 kbytes of EB5 (H'008000 to H'00BFFF) will be erased. |
| 4 | EB4 | 0 | R/W | When this bit is set to 1, 28 kbytes of EB4 (H'001000 to H'007FFF) will be erased. |
| 3 | EB3 | 0 | R/W | When this bit is set to 1, 1 kbyte of EB3 (H'000C00 to H'000FFF) will be erased. |
| 2 | EB2 | 0 | R/W | When this bit is set to 1, 1 kbyte of EB2 (H'000800 to H'000BFF) will be erased. |
| 1 | EB1 | 0 | R/W | When this bit is set to 1, 1 kbyte of EB1 (H'000400 to H'0007FF) will be erased. |
| 0 | EB0 | 0 | R/W | When this bit is set to 1, 1 kbyte of EB0 (H'000000 to H'0003FF) will be erased. |

20.5.4 Erase Block Register 2 (EBR2)

EBR2 specifies the flash memory erase area block. EBR1 is initialized to H'00 when the SWE1 bit in FLMCR1 is 0. Do not set more than one bit at a time, as this will cause all the bits in EBR1 and EBR2 to be automatically cleared to 0.

- 384-kbyte Flash Memory

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 7, 6 | — | All 0 | R/W | Reserved These bits are always read as 0. The write value should always be 0. |
| 5 | EB13 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB13 (H'050000 to H'05FFFF) will be erased. |
| 4 | EB12 | 0 | R/W | When this bit is 1, 64 kbytes of EB12 (H'040000 to H'04FFFF) will be erased. |
| 3 | EB11 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB11 (H'030000 to H'03FFFF) will be erased. |
| 2 | EB10 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB10 (H'020000 to H'02FFFF) will be erased. |
| 1 | EB9 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB9 (H'010000 to H'01FFFF) will be erased. |
| 0 | EB8 | 0 | R/W | When this bit is set to 1, 32 kbytes of EB8 (H'008000 to H'00FFFF) will be erased. |

- 256-kbyte Flash Memory

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-------|---|
| 7 to 4 | — | All 0 | R/(W) | Reserved Initial values should not be changed. |
| 3 | EB11 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB11 (H'030000 to H'03FFFF) will be erased. |
| 2 | EB10 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB10 (H'020000 to H'02FFFF) will be erased. |
| 1 | EB9 | 0 | R/W | When this bit is set to 1, 64 kbytes of EB9 (H'010000 to H'01FFFF) will be erased. |
| 0 | EB8 | 0 | R/W | When this bit is set to 1, 32 kbytes of EB8 (H'008000 to H'00FFFF) will be erased. |

- 128-kbyte Flash Memory

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 2 | — | All 0 | R/W | Reserved Initial values should not be changed. |
| 1 | EB9 | 0 | R/W | When this bit is set to 1, 32 kbytes of EB9 (H'018000 to H'01FFFF) will be erased. |
| 0 | EB8 | 0 | R/W | When this bit is set to 1, 32 kbytes of EB8 (H'010000 to H'017FFF) will be erased. |

20.5.5 RAM Emulation Register (RAMER)

RAMER specifies the area of flash memory to be overlapped with part of RAM when emulating real-time flash memory programming. RAMER settings should be made in user mode or user program mode. To ensure correct operation of the emulation function, the ROM for which RAM emulation is performed should not be accessed immediately after this register has been modified. Normal execution of an access immediately after register modification is not guaranteed.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 7 to 5 | — | All 0 | R | Reserved These bits are always read as 0. |
| 4 | — | 0 | R/W | Reserved Only 0 should be written to this bit. |
| 3 | RAMS | 0 | R/W | RAM Select Specifies selection or non-selection of flash memory emulation in RAM. When RAMS = 1, the flash memory is overlapped with part of RAM, and all flash memory block are program/erase-protected. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | RAM2 | 0 | R/W | Flash Memory Area Selection |
| 1 | RAM1 | 0 | R/W | <p>When the RAMS bit is set to 1, one of the following flash memory areas is selected to overlap the RAM area. The areas correspond with 4-kbyte erase blocks for the 384-kbyte or 256-kbyte flash memory, 1-kbyte erase block for the 128-kbyte flash memory.</p> <p>384-kbyte or 256-kbyte flash memory</p> <p>000: H'000000 to H'000FFF (EB0)</p> <p>001: H'001000 to H'001FFF (EB1)</p> <p>010: H'002000 to H'002FFF (EB2)</p> <p>011: H'003000 to H'003FFF (EB3)</p> <p>100: H'004000 to H'004FFF (EB4)</p> <p>101: H'005000 to H'005FFF (EB5)</p> <p>110: H'006000 to H'006FFF (EB6)</p> <p>111: H'007000 to H'007FFF (EB7)</p> <p>128-kbyte flash memory</p> <p>000: H'000000 to H'0003FF (EB0)</p> <p>001: H'000400 to H'0007FF (EB1)</p> <p>010: H'000800 to H'000BFF (EB2)</p> <p>011: H'000C00 to H'000FFF (EB3)</p> <p>100: Setting prohibited</p> <p>101: Setting prohibited</p> <p>110: Setting prohibited</p> <p>111: Setting prohibited</p> |
| 0 | RAM0 | 0 | R/W | |

20.5.6 Flash Memory Power Control Register (FLPWCR)

FLPWCR enables/disables transition to power-down modes for the flash memory when this LSI enters sub-active mode.

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 7 | PDWND | 0 | R/W | Power Down Disable Enables/disables transition to power-down modes for the flash memory when this LSI enters sub-active mode. 0: Transition to power-down modes for the flash memory enabled. 1: Transition to power-down modes for the flash memory disabled. |
| 6 to 0 | — | All 0 | R | Reserved These bits are always read as 0. |

20.5.7 Serial Control Register X (SCRX)

SCRX performs register access control.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | — | 0 | R/W | Reserved Only 0 should be written to this bit. |
| 6 | IICX1 | 0 | R/W | I ² C Transfer Select 1, 0 |
| 5 | IICX0 | 0 | R/W | For details, see section 16.3.5, Serial Control Register X (SCRX). |
| 4 | IICE | 0 | R/W | I ² C Master Enable For details, see section 16.3.5, Serial Control Register X (SCRX). |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|--|
| 3 | FLSHE | 0 | R/W | Flash Memory Control Register Enable Controls for the CPU accessing to the control registers (FLMCR1, FLMCR2, EBR1, EBR2) of the flash memory. When this bit is set to 1, the flash memory control registers can be read/written to. When this bit is cleared to 0, the flash memory control registers are not selected. At this time, the contents of the flash memory control registers are retained. 0: Area at H'FFFA8 to H'FFFFAC not selected for the flash memory control registers. 1: Area at H'FFFA8 to H'FFFFAC selected for the flash memory control registers. |
| 2 to 0 | — | All 0 | R/W | Reserved Only 0 should be written to these bits. |

20.6 On-Board Programming Modes

When pins are set to on-board programming mode, program/erase/verify operations can be performed on the on-chip flash memory. There are two on-board programming modes: boot mode and user program mode. The pin settings for transition to each of these modes are shown in table 20.3. For a diagram of the transitions to the various flash memory modes, see figure 20.2.

Table 20.3 Setting On-Board Programming Modes

| Mode Setting | | FWE | MD2 | MD1 | MD0 |
|-------------------|------------------|-----|-----|-----|-----|
| Boot mode | Extended mode | 1 | 0 | 1 | 0 |
| | Single-chip mode | 1 | 0 | 1 | 1 |
| User program mode | Extended mode | 1 | 1 | 1 | 0 |
| | Single-chip mode | 1 | 1 | 1 | 1 |

20.6.1 Boot Mode

Table 20.4 shows the boot mode operations between reset end and branching to the programming control program.

1. When boot mode is used, the flash memory programming control program must be prepared in the host beforehand. Prepare a programming control program in accordance with the description in section 20.8, Flash Memory Programming/Erasing.

In boot mode, if any data has been programmed into the flash memory (if all data is not 1), all flash memory blocks are erased. Boot mode is for use when user program mode is unavailable, such as the first time on-board programming is performed, or if the program activated in user program mode is accidentally erased.

2. SCI should be set to asynchronous mode, and the transfer format as follows: 8-bit data, 1 stop bit, and no parity.
3. When the boot program is initiated, the chip measures the low-level period of asynchronous SCI communication data (H'00) transmitted continuously from the host. The chip then calculates the bit rate of transmission from the host, and adjusts the SCI bit rate to match that of the host. The reset should end with the RxD pin high. The RxD and TxD pins should be pulled up on the board if necessary. After the reset is complete, it takes approximately 100 states before the chip is ready to measure the low-level period.
4. After matching the bit rates, the chip transmits one H'00 byte to the host to indicate the completion of bit rate adjustment. The host should confirm that this adjustment end indication (H'00) has been received normally, and transmit one H'55 byte to the chip. If reception could not be performed normally, initiate boot mode again by a reset. Depending on the host's transfer bit rate and system clock frequency of this LSI, there will be a discrepancy between the bit rates of the host and the chip. To operate the SCI properly, set the host's transfer bit rate and system clock frequency of this LSI within the ranges listed in table 20.5.
5. In boot mode, a part of the on-chip RAM area is used by the boot program. The area H'FFC000 to H'FFDFFF is the area to which the programming control program is transferred from the host. The boot program area cannot be used until the execution state in boot mode switches to the programming control program.
6. Before branching to the programming control program, the chip terminates transfer operations by SCI (by clearing the RE and TE bits in SCR to 0), however the adjusted bit rate value remains set in BRR. Therefore, the programming control program can still use it for transfer of write data or verify data with the host. The TxD pin is high. The contents of the CPU general registers are undefined immediately after branching to the programming control program. These registers must be initialized at the beginning of the programming control program, as the stack pointer (SP), in particular, is used implicitly in subroutine calls, etc.
7. Boot mode can be cleared by driving the reset pin low, waiting at least 20 states, then setting the FWE pin and mode pins, and executing reset release*. Boot mode is also cleared when a WDT overflow occurs.
8. All interrupts are disabled during programming or erasing of the flash memory.

Note: * The input signals on the FWE and mode pins must satisfy the mode programming setup time ($t_{MDS} = 200 \text{ ns}$) at the reset release timing.

Table 20.4 Boot Mode Operation

| Item | Host Operation | Communications Contents | LSI Operation |
|--|--|--|---|
| | Processing Contents | | Processing Contents |
| Boot mode start | | | Branches to boot program at reset-start. <div>Boot program initiation</div> |
| Bit rate adjustment | Continuously transmits data H'00 at specified bit rate. Transmits data H'55 when data H'00 is received error-free. Receives data H'AA. | H'00, H'00 H'00 H'00 H'55 H'AA | <ul style="list-style-type: none"> Measures low-level period of receive data H'00. Calculates bit rate and sets it in BRR of SCI. Transmits data H'00 to host as adjustment end indication. Transmits data H'AA to host when data H'55 is received. |
| Transfer of programming control program | Transmits number of bytes (N) of programming control program to be transferred as 2-byte data (low-order byte following high-order byte) Transmits 1-byte of programming control program (repeated for N times) | High-order byte and low-order byte Echoback H'XX Echoback | Echobacks the 2-byte data received. Echobacks received data to host and also transfers it to RAM (repeated for N times) |
| Flash memory erase | Boot program erase error Receives data H'AA. | H'FF H'AA | Checks flash memory data, erases all flash memory blocks in case of written data existing, and transmits data H'AA to host. (If erase could not be done, transmits data H'FF to host and aborts operation.) |
| Execution of programming control program | | | Branches to programming control program transferred to on-chip RAM and starts execution. |

Table 20.5 System Clock Frequencies for Which Automatic Adjustment of LSI Bit Rate Is Possible

| Host Bit Rate | System Clock Frequency Range of This LSI | | |
|---------------|--|--------------------------------|-------------|
| | H8S/2258 | H8S/2238B, H8S/2238R, H8S/2227 | H8S/2239 |
| 19,200 bps | 10 to 13.5 MHz | 8 to 13.5 MHz | 8 to 20 MHz |
| 9,600 bps | | 4 to 13.5 MHz | 4 to 20 MHz |
| 4,800 bps | | 2 to 13.5 MHz | 2 to 20 MHz |

20.6.2 Programming/Erasing in User Program Mode

On-board programming/erasing of an individual flash memory block can also be performed in user program mode by branching to a user program/erase control program. The user must prepare on-board means for controlling FWE, on-board means of supplying programming data, and branching conditions. The flash memory must contain the user program/erase control program or a program that provides the user program/erase control program from external memory. As the flash memory itself cannot be read during programming/erasing, transfer the user program/erase control program to on-chip RAM, as in boot mode. Figure 20.8 shows a sample procedure for programming/erasing in user program mode. Prepare a user program/erase control program in accordance with the description in section 20.8, Flash Memory Programming/Erasing.

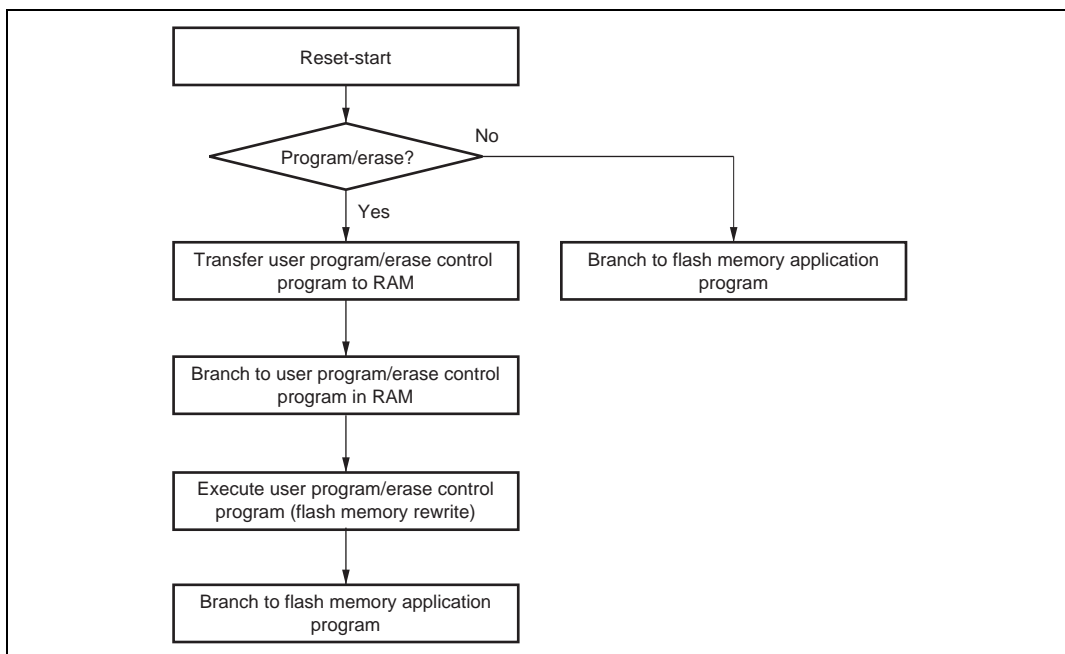


Figure 20.8 Programming/Erasing Flowchart Example in User Program Mode

20.7 Flash Memory Emulation in RAM

A setting in the RAM emulation register (RAMER) enables part of RAM to be overlapped onto the flash memory area so that data to be written to flash memory can be emulated in RAM in real time. Emulation can be performed in user mode or user program mode. Figure 20.9 shows an example of emulation of real-time flash memory programming.

1. Set RAMER to overlap part of RAM onto the area for which real-time programming is required.
2. Emulation is performed using the overlapping RAM.
3. After the program data has been confirmed, the RAMS bit is cleared, thus releasing the RAM overlap.
4. The data written in the overlapping RAM is written into the flash memory space.

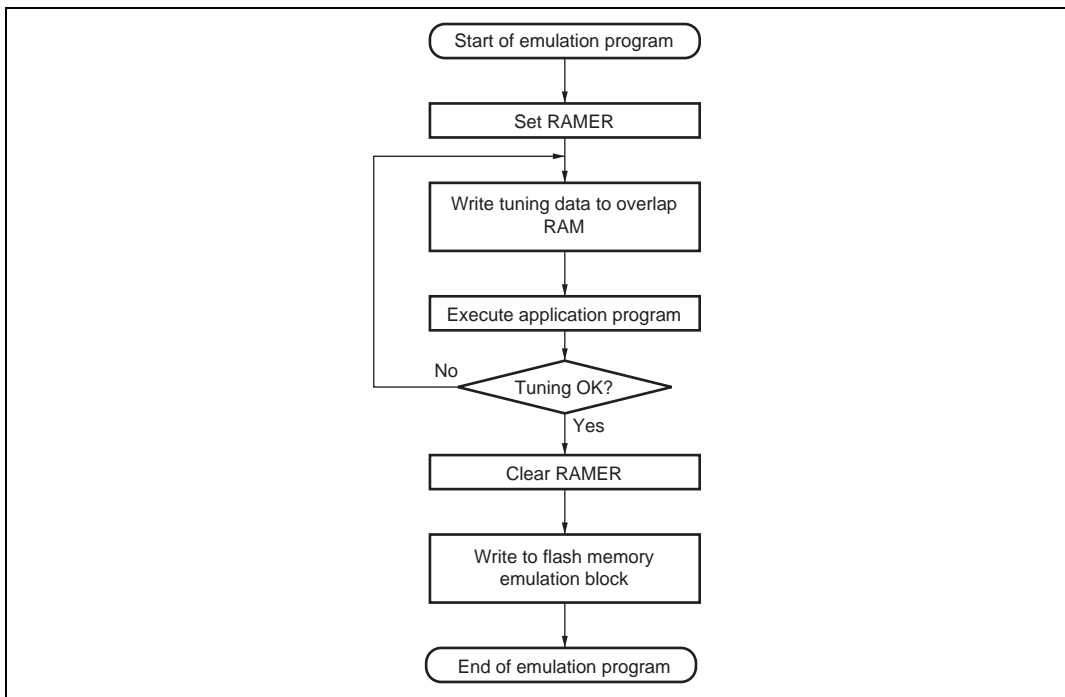


Figure 20.9 Flowchart for Flash Memory Emulation in RAM

An example in which flash memory block area EB1 is overlapped is shown in figure 20.10.

1. The RAM area to be overlapped is fixed at a 4-kbyte area in the range H'FFD000 to H'FFDFFF in the 384-kbyte or 256-kbyte flash memory. The RAM area to be overlapped is fixed at a 1-kbyte area in the range H'FFD000 to H'FFD3FF in the 128-kbyte flash memory.
2. The flash memory area to be overlapped is selected by RAMER from a 4-kbyte area of the EB0 to EB7 blocks.
3. The overlapped RAM area can be accessed from both the flash memory addresses and RAM addresses.

4. When the RAMS bit in RAMER is set to 1, program/erase protection is enabled for all flash memory blocks (emulation protection). In this state, setting the P1 or E1 bit in FLMCR1 to 1 does not cause a transition to program mode or erase mode.
5. A RAM area cannot be erased by execution of software in accordance with the erase algorithm.
6. Block area EB0 contains the vector table. When performing RAM emulation, the vector table is needed in the overlap RAM.

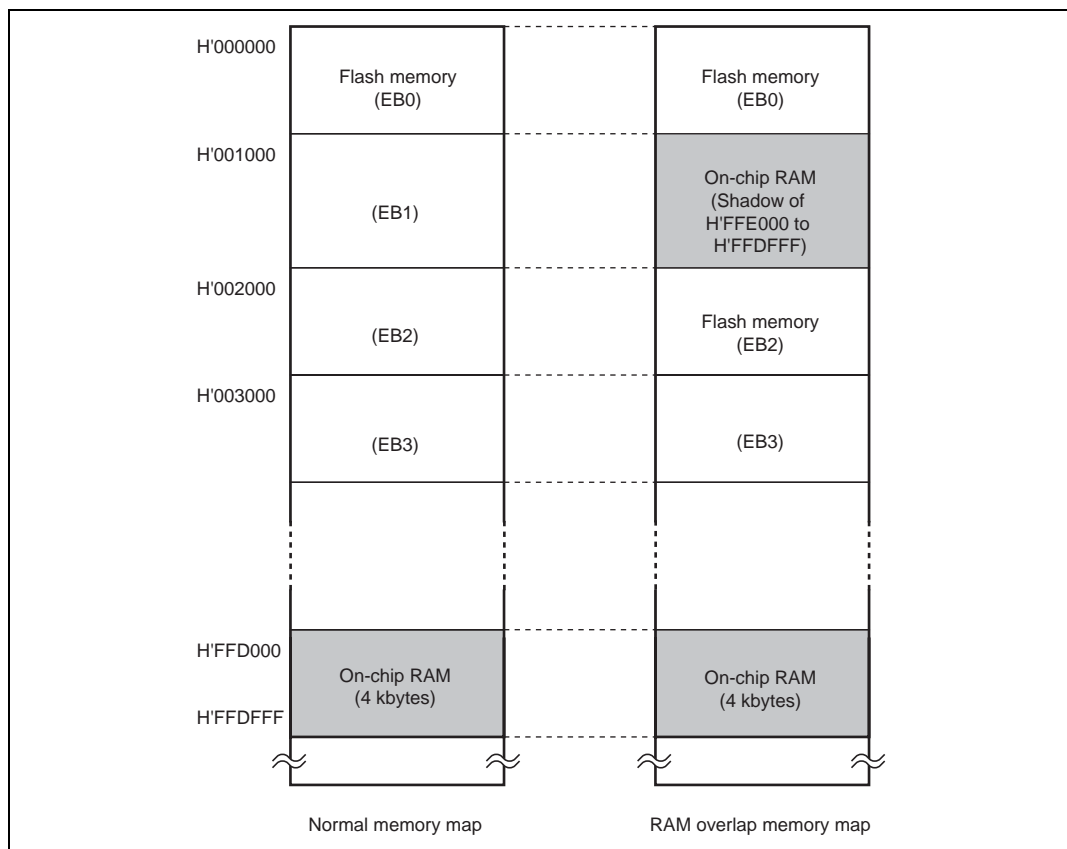


Figure 20.10 Example of RAM Overlap Operation

20.8 Flash Memory Programming/Erasing

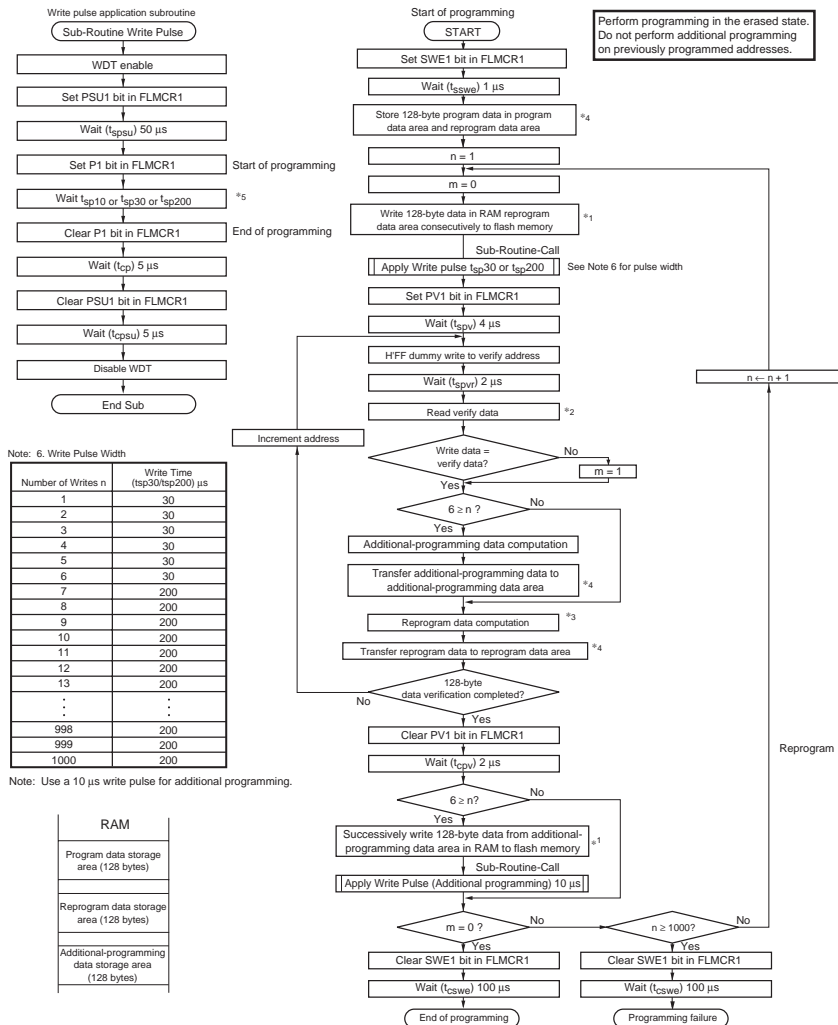
A software method using the CPU is employed to program and erase flash memory in the on-board programming modes. Depending on the FLMCR1 setting, the flash memory operates in one of the following four modes: Program mode, program-verify mode, erase mode, and erase-verify

mode. The programming control program in boot mode and the user program/erase control program in user program mode use these operating modes in combination to perform programming/erasing. Flash memory programming and erasing should be performed in accordance with the descriptions in section 20.8.1, Program/Program-Verify and section 20.8.2, Erase/Erase-Verify, respectively.

20.8.1 Program/Program-Verify

When writing data or programs to the flash memory, the program/program-verify flowchart shown in figure 20.11 should be followed. Performing programming operations according to this flowchart will enable data or programs to be written to the flash memory without subjecting the chip to voltage stress or sacrificing program data reliability.

1. Programming must be done to an empty address. Do not reprogram an address to which programming has already been performed.
2. Programming should be carried out 128 bytes at a time. A 128-byte data transfer must be performed even if writing fewer than 128 bytes. In this case, H'FF data must be written to the extra addresses.
3. Prepare the following data storage areas in RAM: A 128-byte programming data area, a 128-byte reprogramming data area, and a 128-byte additional-programming data area. Perform reprogramming data computation and additional programming data computation according to figure 20.11.
4. Consecutively transfer 128 bytes of data in byte units from the reprogramming data area or additional-programming data area to the flash memory. The program address and 128-byte data are latched in the flash memory. The lower 8 bits of the start address in the flash memory destination area must be H'00 or H'80.
5. The time during which the P1 bit is set to 1 is the programming time. Figure 20.11 shows the allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runaway, etc. Set a value greater than $(t_{\text{psu}} + t_{\text{sp200}} + t_{\text{cp}} + t_{\text{cpsu}})$ μs as the WDT overflow period.
7. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lower 1 bit are B'0. Verify data can be read in words from the address to which a dummy write was performed.
8. The maximum number of repetitions of the program/program-verify sequence of the same bit is (N).



Reprogram Data Computation Table

| Original Data (D) | Verify Data (V) | Reprogram Data (X) | Comments |
|-------------------|-----------------|--------------------|-----------------------------------|
| 0 | 0 | 1 | Programming completed |
| 0 | 1 | 0 | Programming incomplete; reprogram |
| 1 | 0 | 1 | |
| 1 | 1 | 1 | Still in erased state; no action |

Additional-Programming Data Computation Table

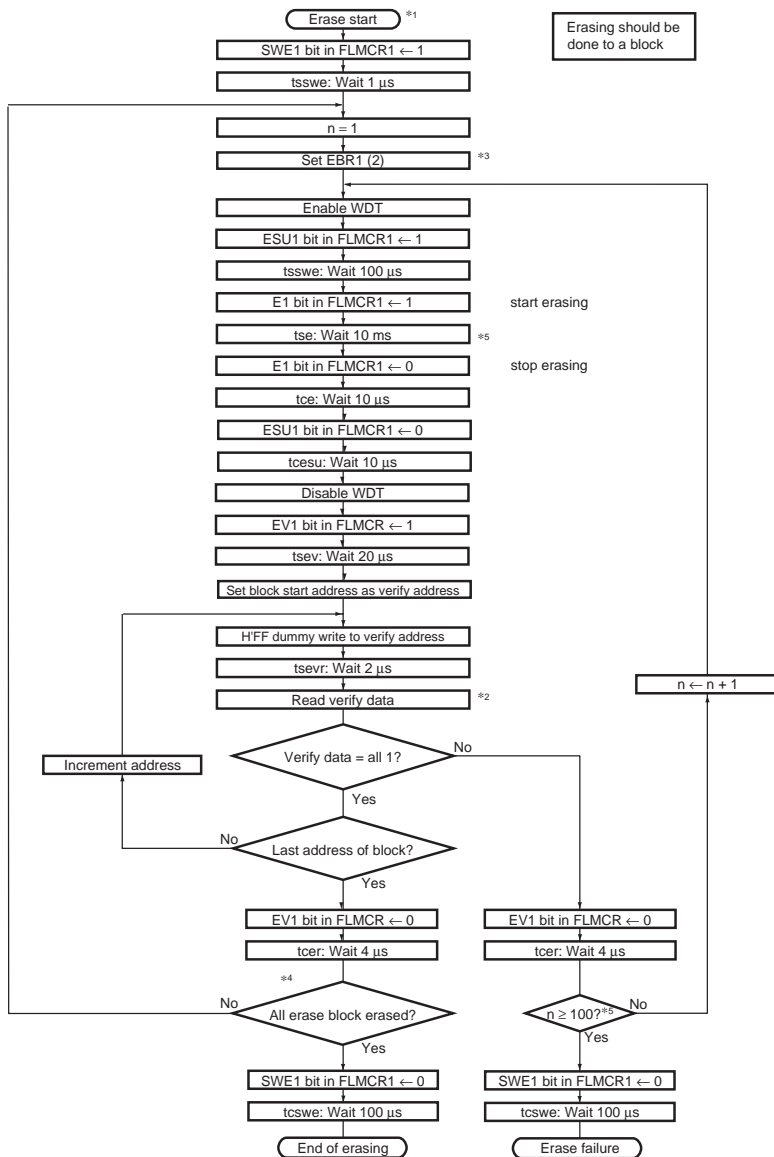
| Reprogram Data (X) | Verify Data (V) | Additional-Programming Data (Y) | Comments |
|--------------------|-----------------|---------------------------------|---|
| 0 | 0 | 0 | Additional programming to be executed |
| 0 | 1 | 1 | Additional programming not to be executed |
| 1 | 0 | 1 | Additional programming not to be executed |
| 1 | 1 | 1 | Additional programming not to be executed |

Figure 20.11 Program/Program-Verify Flowchart

20.8.2 Erase/Erase-Verify

When erasing flash memory, the erase/erase-verify flowchart shown in figure 20.12 should be followed.

1. Prewriting (setting erase block data to all 0) is not necessary.
2. Erasing is performed in block units. Make only a single-bit specification in the erase block register 1 and 2 (EBR1 and EBR2). To erase multiple blocks, each block must be erased in turn.
3. The time during which the E1 bit is set to 1 is the flash memory erase time.
4. The watchdog timer (WDT) is set to prevent overprogramming due to program runaway, etc. Set a value greater than $(t_{\text{sesu}} + t_{\text{se}} + t_{\text{ce}} + t_{\text{cesu}})$ ms as the WDT overflow period.
5. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lower 1 bit are B'0. Verify data can be read in words from the address to which a dummy write was performed.
5. If the read data is not erased successfully, set erase mode again, and repeat the erase/erase-verify sequence as before. The maximum number of repetitions of the erase/erase-verify sequence is (N).



- Notes:
1. Pre-writing (all erase block data are cleared to 0) is not necessary.
 2. Verify data is read out in 16 bit size (word access).
 3. Erasing block register (EBR) can be set about 1 bit at a time.
Do not specify 2 bits or more.
 4. Erasing is performed block by block. When multiple blocks must be erased, erase each lock one by one.
 5. This is a recommended value. To change it, consult tables 27.12, 27.25, 27.37, 27.49, and 27.59 and select a new value such that the erase time (tE), wait time after E1 bit setting (tse), and maximum erase count (N) do not exceed the maximum values indicated.

Figure 20.12 Erase/Erase-Verify Flowchart

20.9 Program/Erase Protection

There are three kinds of flash memory program/erase protection; hardware protection, software protection, and error protection.

20.9.1 Hardware Protection

Hardware protection refers to a state in which programming/erasing of flash memory is forcibly disabled or aborted because of a transition to reset or standby mode. Flash memory control register 1 (FLMCR1), flash memory control register 2 (FLMCR2), erase block register 1 (EBR1), and erase block register 2 (EBR2) are initialized. In a reset via the $\overline{\text{RES}}$ pin, the reset state is not entered unless the $\overline{\text{RES}}$ pin is held low until oscillation stabilizes after powering on. In the case of a reset during operation, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width specified in the AC Characteristics section.

20.9.2 Software Protection

Software protection can be implemented against programming/erasing of all flash memory blocks by clearing the SWE1 bit in FLMCR1. When software protection is in effect, setting the P1 or E1 bit in FLMCR1 does not cause a transition to program mode or erase mode. By setting the erase block register 1 and 2 (EBR1 and EBR2), erase protection can be set for individual blocks. When EBR1 and EBR2 are set to H'00, erase protection is set for all blocks. By setting bit RAMS in RAMER, programming/erase protection is set for all blocks.

20.9.3 Error Protection

In error protection, an error is detected when CPU runaway occurs during flash memory programming/erasing, or operation is not performed in accordance with the program/erase algorithm, and the program/erase operation is aborted. Aborting the program/erase operation prevents damage to the flash memory due to overprogramming or overerasing.

When the following errors are detected during programming/erasing of flash memory, the FLER bit in FLMCR2 is set to 1, and the error protection state is entered.

- When the flash memory of the relevant address area is read during programming/erasing (including vector read and instruction fetch)
- Immediately after exception handling (excluding a reset) during programming/erasing
- When a SLEEP instruction is executed during programming/erasing
- When the CPU releases the bus to the DMAC* or DTC during programming/erasing

Note: * Supported only by the H8S/2239 Group.

The FLMCR1, FLMCR2, EBR1, and EBR2 settings are retained, however program mode or erase mode is aborted at the point at which the error occurred. Program mode or erase mode cannot be re-entered by re-setting the P1 or E1 bit. However, PV1 and EV1 bit setting is enabled, and a transition can be made to verify mode. Error protection can be cleared only by a reset or in hardware standby.

20.10 Interrupt Handling When Programming/Erasing Flash Memory

All interrupts, including NMI input, are disabled when flash memory is being programmed or erased (when the P1 or E1 bit is set in FLMCR1), and while the boot program is executing in boot mode^{*1}, to give priority to the program or erase operation. There are three reasons for this:

1. Interrupt during programming or erasing might cause a violation of the programming or erasing algorithm, with the result that normal operation could not be assured.
2. In the interrupt exception handling sequence during programming or erasing, the vector would not be read correctly^{*2}, possibly resulting in CPU runaway.
3. If an interrupt occurred during boot program execution, it would not be possible to execute the normal boot mode sequence.

Notes: 1. Interrupt requests must be disabled inside and outside the CPU until the programming control program has completed programming.

2. The vector may not be read correctly in this case for the following two reasons:

- If flash memory is read while being programmed or erased (while the P1 or E1 bit is set in FLMCR1), correct read data will not be obtained (undetermined values will be returned).
- If the interrupt entry in the vector table has not been programmed yet, interrupt exception handling will not be executed correctly.

20.11 Programmer Mode

In programmer mode, a PROM programmer can be used to perform programming/erasing via a socket adapter, just as for a discrete flash memory. Use a PROM programmer which supports the Renesas Technology 512-kbyte, 256-kbyte, or 128-kbyte flash memory on-chip microcomputer device type. It requires the 12-MHz input clock.

The socket adapter pin correspondence diagram is shown in figure 20.13.

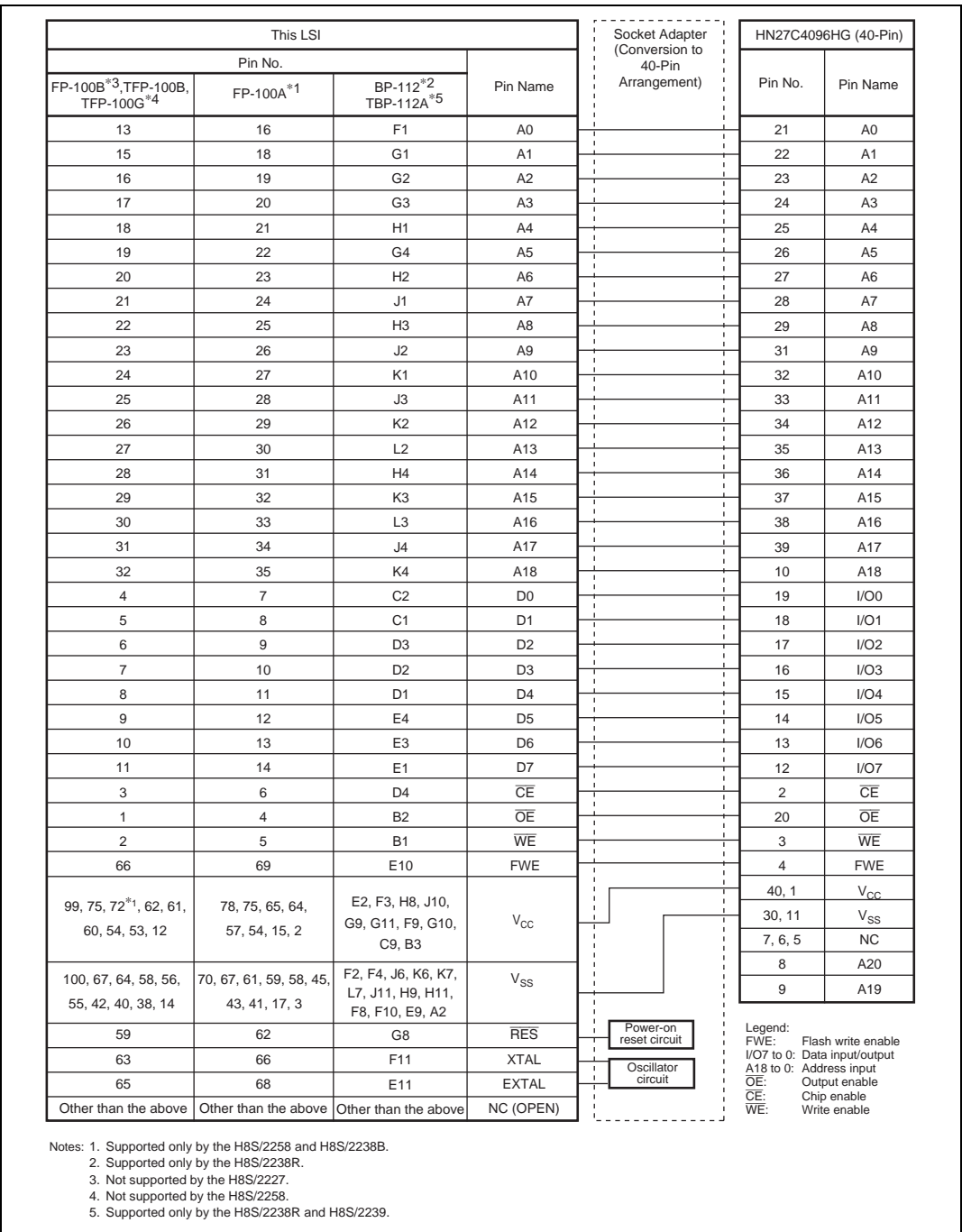


Figure 20.13 Socket Adapter Pin Correspondence Diagram

20.12 Power-Down States for Flash Memory

In user mode, the flash memory will operate in either of the following states:

- Normal operating mode
The flash memory can be read and written to at high speed.
- Power-down state
The flash memory can be read when part of the power circuit is halted and the LSI operates by subclocks.
- Standby mode
All flash memory circuits are halted.

Table 20.6 shows the correspondence between the operating modes of this LSI and the flash memory. When the flash memory returns to its normal operating state from standby mode, a period to stabilize the power supply circuits that were stopped is needed. When the flash memory returns to its normal operating state, bits STS2 to STS0 in SBYCR must be set to provide a wait time of at least 100 μ s, even when the external clock is being used.

Table 20.6 Flash Memory Operating States

| LSI Operating State | Flash Memory Operating State |
|---------------------|--|
| Active mode | Normal operating mode |
| Sleep mode | Normal operating mode |
| Watch mode | Standby mode |
| Standby mode | |
| Subactive mode | PDWND = 0: Power-down mode (read only) |
| Subsleep mode | PDWND = 1: Normal operating mode (read only) |

20.13 Flash Memory Programming and Erasing Precautions

Precautions concerning the use of on-board programming mode, the RAM emulation function, and programmer mode are summarized below.

Use the Specified Voltages and Timing for Programming and Erasing: Applied voltages in excess of the rating can permanently damage the device. Use a PROM programmer that supports the Renesas Technology flash memory on-chip microcomputer device type (FZTAT512V3A, FZTAT256V3A, or FZTAT128V3A).

Do not select the HN27C4096 setting for the PROM programmer, and only use the specified socket adapter. Failure to observe these points may result in damage to the device.

Powering On and Off (See Figures 20.14 to 20.16): Do not apply a high level to the FWE pin until VCC has stabilized. Also, drive the FWE pin low before turning off VCC.

When applying or disconnecting VCC power, fix the FWE pin low and place the flash memory in the hardware protection state.

The power-on and power-off timing requirements should also be satisfied in the event of a power failure and subsequent recovery.

FWE Application/Disconnection (See Figures 20.14 to 20.16): FWE application should be carried out when MCU operation is in a stable condition. If MCU operation is not stable, fix the FWE pin low and set the protection state.

The following points must be observed concerning FWE application and disconnection to prevent unintentional programming or erasing of flash memory:

- Apply FWE when the VCC voltage has stabilized within its rated voltage range.
- In boot mode, apply and disconnect FWE during a reset.
- In user program mode, FWE can be switched between high and low level regardless of the reset state. FWE input can also be switched during execution of a program in flash memory.
- Do not apply FWE if program runaway has occurred.
- Disconnect FWE only when the SWE1, ESU1, PSU1, EV1, PV1, P1, and E1 bits in FLMCR1 are cleared.

Make sure that the SWE1, ESU1, PSU1, EV1, PV1, P1, and E1 bits are not set by mistake when applying or disconnecting FWE.

Do Not Apply a Constant High Level to the FWE Pin: Apply a high level to the FWE pin only when programming or erasing flash memory. A system configuration in which a high level is constantly applied to the FWE pin should be avoided. Also, while a high level is applied to the FWE pin, the watchdog timer should be activated to prevent overprogramming or overerasing due to program runaway, etc.

Use the Recommended Algorithm when Programming and Erasing Flash Memory: The recommended algorithm enables programming and erasing to be carried out without subjecting the device to voltage stress or sacrificing program data reliability. When setting the P1 or E1 bit in FLMCR1, the watchdog timer should be set beforehand as a precaution against program runaway, etc.

Do Not Set or Clear the SWE1 Bit during Execution of a Program in Flash Memory: Wait for at least 100 μ s after clearing the SWE1 bit before executing a program or reading data in flash memory.

When the SWE1 bit is set, data in flash memory can be rewritten. Access flash memory only for verify operations (verification during programming/erasing). Also, do not clear the SWE1 bit during programming, erasing, or verifying. Similarly, when using the RAM emulation function while a high level is being input to the FWE pin, the SWE1 bit must be cleared before executing a program or reading data in flash memory.

However, the RAM area overlapping flash memory space can be read and written to regardless of whether the SWE1 bit is set or cleared.

Do Not Use Interrupts while Flash Memory Is Being Programmed or Erased: All interrupt requests, including NMI, should be disabled during FWE application to give priority to program/erase operations.

Do Not Perform Additional Programming. Erase the Memory before Reprogramming: In on-board programming, perform only one programming operation on a 128-byte programming unit block. In programmer mode, too, perform only one programming operation on a 128-byte programming unit block. Programming should be carried out with the entire programming unit block erased.

Before Programming, Check That the Chip Is Correctly Mounted in the PROM

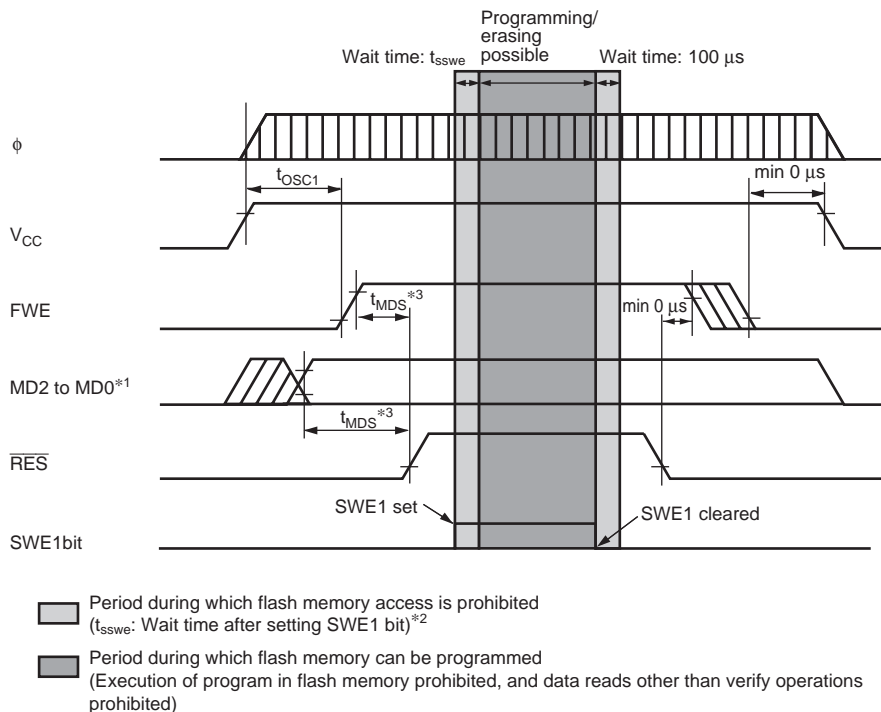
Programmer: Overcurrent damage to the device can result if the index marks on the PROM programmer socket, socket adapter, and chip are not correctly aligned.

Do Not Touch the Socket Adapter or Chip during Programming: Touching either of these can cause contact faults and write errors.

Reset the Flash Memory before Turning on the Power: To reset the flash memory during oscillation stabilization period, the reset signal must be input for at least 100 μ s.

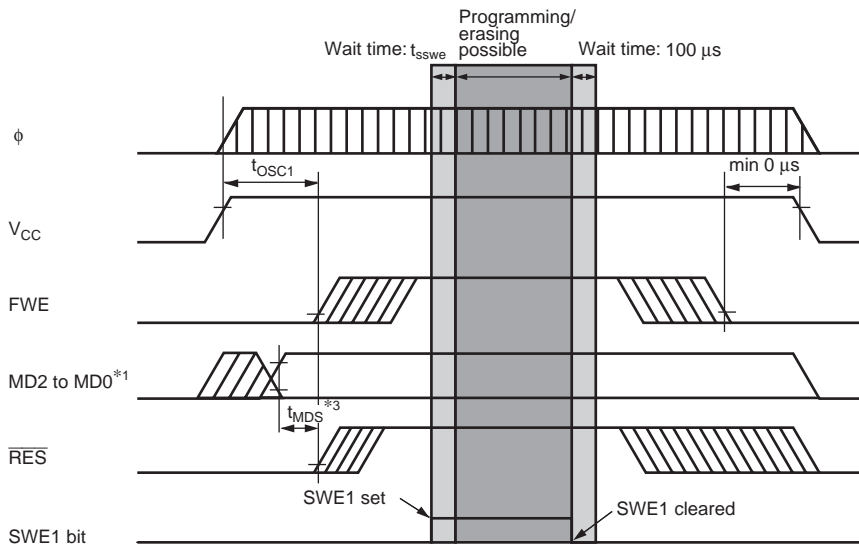
Apply the Reset Signal while SWE Is Low to Reset the Flash Memory during its operation:

The reset signal is applied at least 100 μ s after the SWE bit has been cleared.



- Notes:
1. Except when switching modes, the level of the mode pins (MD2 to MD0) must be fixed until power-off by pulling the pins up or down.
 2. See sections 27.2.6, 27.3.6, 27.4.6, 27.5.6, and 27.6.6, Flash Memory Characteristics.
 3. Mode programming setup time t_{MDS} (min) = 200 ns.

Figure 20.14 Power-On/Off Timing (Boot Mode)



Period during which flash memory access is prohibited
(t_{SSWE} : Wait time after setting SWE1 bit)*2

Period during which flash memory can be programmed/erased
(Execution of program in flash memory prohibited, and data reads other than verify operations prohibited)

- Notes: 1. Except when switching modes, the level of the mode pins (MD2 to MD0) must be fixed until power-off by pulling the pins up or down.
2. See sections 27.2.6, 27.3.6, 27.4.6, 27.5.6 and 27.6.6, Flash Memory Characteristics.
3. Mode programming setup time t_{MDS} (min) = 200 ns.

Figure 20.15 Power-On/Off Timing (User Program Mode)

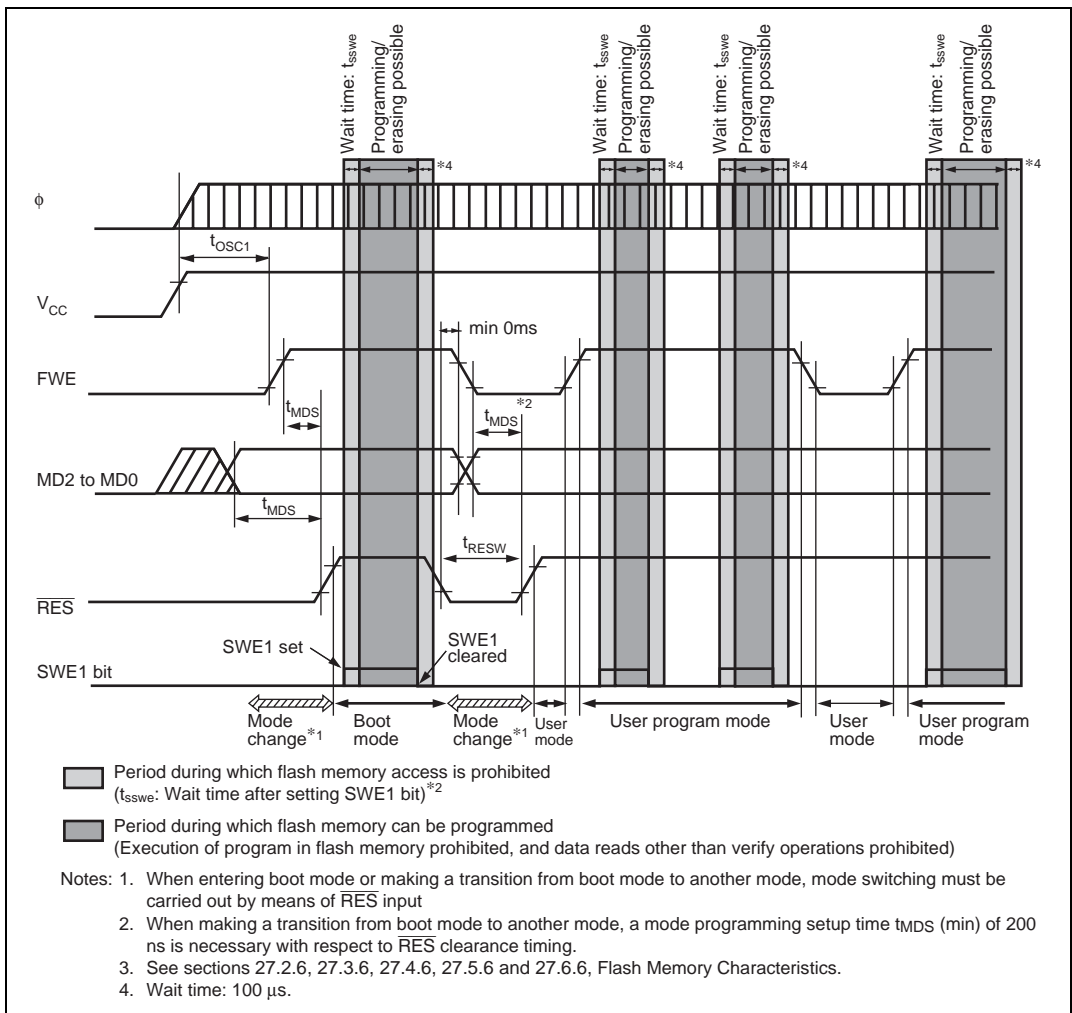


Figure 20.16 Mode Transition Timing
(Example: Boot Mode • User Mode • User Program Mode)

20.14 Note on Switching from F-ZTAT Version to Masked ROM Version

The masked ROM version does not have the internal registers for flash memory control that are provided in the F-ZTAT version. Table 20.7 lists the registers that are present in the F-ZTAT version but not in the masked ROM version. If a register listed in table 20.7 is read in the masked ROM version, an undefined value will be returned. Therefore, if application software developed on the F-ZTAT version is switched to a masked ROM version product, it must be modified to ensure that the registers in table 20.7 have no effect.

Table 20.7 Registers Present in F-ZTAT Version but Absent in Masked ROM Version

| Register | Abbreviation | Address |
|--|--------------|---------|
| Flash memory control register 1 | FLMCR1 | H'FFA8 |
| Flash memory control register 2 | FLMCR2 | H'FFA9 |
| Erase block register 1 | EBR1 | H'FFAA |
| Erase block register 2 | EBR2 | H'FFAB |
| RAM emulation register | RAMER | H'FEDB |
| Flash memory power control register | FLPWCR | H'FFAC |
| Serial control register X (Only bit 3) | SCRX | H'FDB4 |

Section 21 Masked ROM

This LSI incorporates a masked ROM which has the following features.

21.1 Features

- Size

| Product Class | | ROM Size | ROM Address (Modes 6 and 7) |
|----------------|-------------|------------|-----------------------------|
| H8S/2258 Group | HD6432258 | 256 kbytes | H'000000 to H'03FFFF |
| | HD6432256 | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432258W | 256 kbytes | H'000000 to H'03FFFF |
| | HD6432256W | 128 kbytes | H'000000 to H'01FFFF |
| H8S/2239 Group | HD6432239 | 384 kbytes | H'000000 to H'05FFFF |
| | HD6432239W | 384 kbytes | H'000000 to H'05FFFF |
| H8S/2238 Group | HD6432238B | 256 kbytes | H'000000 to H'03FFFF |
| | HD6432236B | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432238R | 256 kbytes | H'000000 to H'03FFFF |
| | HD6432236R | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432238BW | 256 kbytes | H'000000 to H'03FFFF |
| | HD6432236BW | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432238RW | 256 kbytes | H'000000 to H'03FFFF |
| | HD6432236RW | 128 kbytes | H'000000 to H'03FFFF |
| H8S/2237 Group | HD6432237 | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432235 | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432233 | 64 kbytes | H'000000 to H'00FFFF |
| H8S/2227 Group | HD6432227 | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432225 | 128 kbytes | H'000000 to H'01FFFF |
| | HD6432224 | 96 kbytes | H'000000 to H'017FFF |
| | HD6432223 | 64 kbytes | H'000000 to H'00FFFF |

- Connected to the bus master through 16-bit data bus, enabling one-state access to both byte data and word data.

Figure 21.1 shows a block diagram of the on-chip masked ROM.

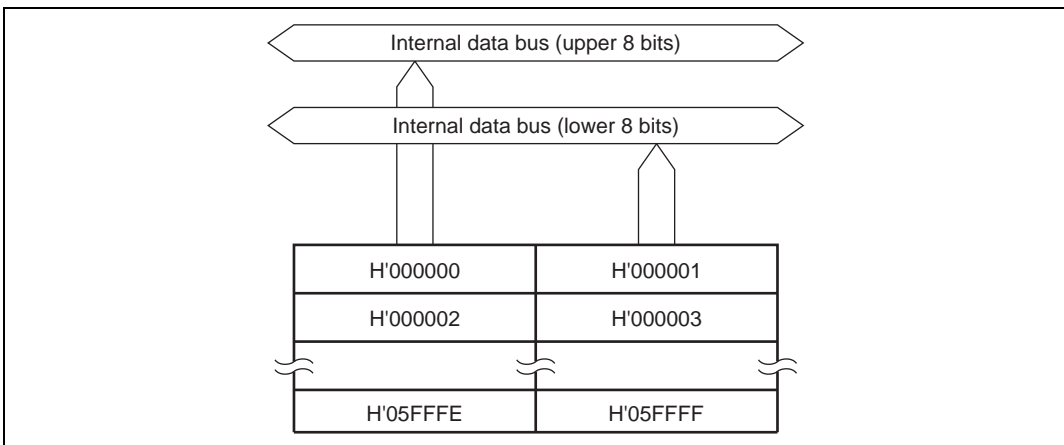


Figure 21.1 Block Diagram of On-Chip Masked ROM (384 kbytes)

Section 22 PROM

The PROM version can be set to PROM mode and programmed with a PROM programmer.

22.1 PROM Mode Setting

The PROM version (HD6472237) suspends its microcomputer functions when placed in PROM mode, enabling the on-chip PROM to be programmed. This programming can be done with a PROM programmer set up in the same way as for the HN27C101 ($V_{pp} = 12.5\text{ V}$) EPROM. Use of a socket adapter to convert from 100 pins to 32 pins enables programming with a commercial PROM programmer.

Caution is required when selecting the PROM programmer, as this LSI does not support page mode.

Table 22.1 shows how PROM mode is selected.

Table 22.1 Selecting PROM Mode

| Pin Names | Setting |
|--------------------------|---------|
| MD2, MD1, MD0 | Low |
| $\overline{\text{STBY}}$ | |
| PA2, PA1 | High |

22.2 Socket Adapter and Memory Map

Programs can be written and verified by attaching a socket adapter to convert from 100 pins to 32 pins to the PROM programmer. Figure 22.1 shows the wiring of the socket adapter, and table 22.2 gives ordering information for the socket adapter. Figure 22.2 shows the memory map in PROM mode.

HD6472237
(FP-100B, TFP-100B, TFP-100G)

| Pin No. | Pin Function |
|---------|--------------|
| 59 | RES |
| 4 | PD0 |
| 5 | PD1 |
| 6 | PD2 |
| 7 | PD3 |
| 8 | PD4 |
| 9 | PD5 |
| 10 | PD6 |
| 11 | PD7 |
| 13 | PC0 |
| 15 | PC1 |
| 16 | PC2 |
| 17 | PC3 |
| 18 | PC4 |
| 19 | PC5 |
| 20 | PC6 |
| 21 | PC7 |
| 22 | PB0 |
| 60 | NMI |
| 24 | PB2 |
| 25 | PB3 |
| 26 | PB4 |
| 27 | PB5 |
| 28 | PB6 |
| 29 | PB7 |
| 30 | PA0 |
| 73 | PF2 |
| 23 | PB1 |
| 74 | PF1 |
| 62, 12 | VCC |
| 54 | AVCC |
| 53 | Vref |
| 31 | PA1 |
| 32 | PA2 |
| 64, 14 | VSS |
| 42 | AVSS |
| 61 | STBY |
| 55 | MD0 |
| 56 | MD1 |
| 67 | MD2 |

EPROM socket

| Pin Function | HN27C101 (DIP-32) Pin No. |
|-----------------|------------------------------|
| VPP | 1 |
| EO0 | 13 |
| EO1 | 14 |
| EO2 | 15 |
| EO3 | 17 |
| EO4 | 18 |
| EO5 | 19 |
| EO6 | 20 |
| EO7 | 21 |
| EA0 | 12 |
| EA1 | 11 |
| EA2 | 10 |
| EA3 | 9 |
| EA4 | 8 |
| EA5 | 7 |
| EA6 | 6 |
| EA7 | 5 |
| EA8 | 27 |
| EA9 | 26 |
| EA10 | 23 |
| EA11 | 25 |
| EA12 | 4 |
| EA13 | 28 |
| EA14 | 29 |
| EA15 | 3 |
| EA16 | 2 |
| \overline{CE} | 22 |
| \overline{OE} | 24 |
| PGM | 31 |
| V _{CC} | 32 |
| V _{SS} | 16 |

Legend:

VPP: Programming power supply (12.5 V)
 EO7 to EO0: Data input/output
 EA16 to EA0: Address input
 \overline{OE} : Output enable
 \overline{CE} : Chip enable
 PGM: Program

Note: Pins not shown in this figure should be open.

**Figure 22.1 HD6472237 Socket Adapter Pin Correspondence Diagram
(FP-100B, TFP-100B, TFP-100G)**

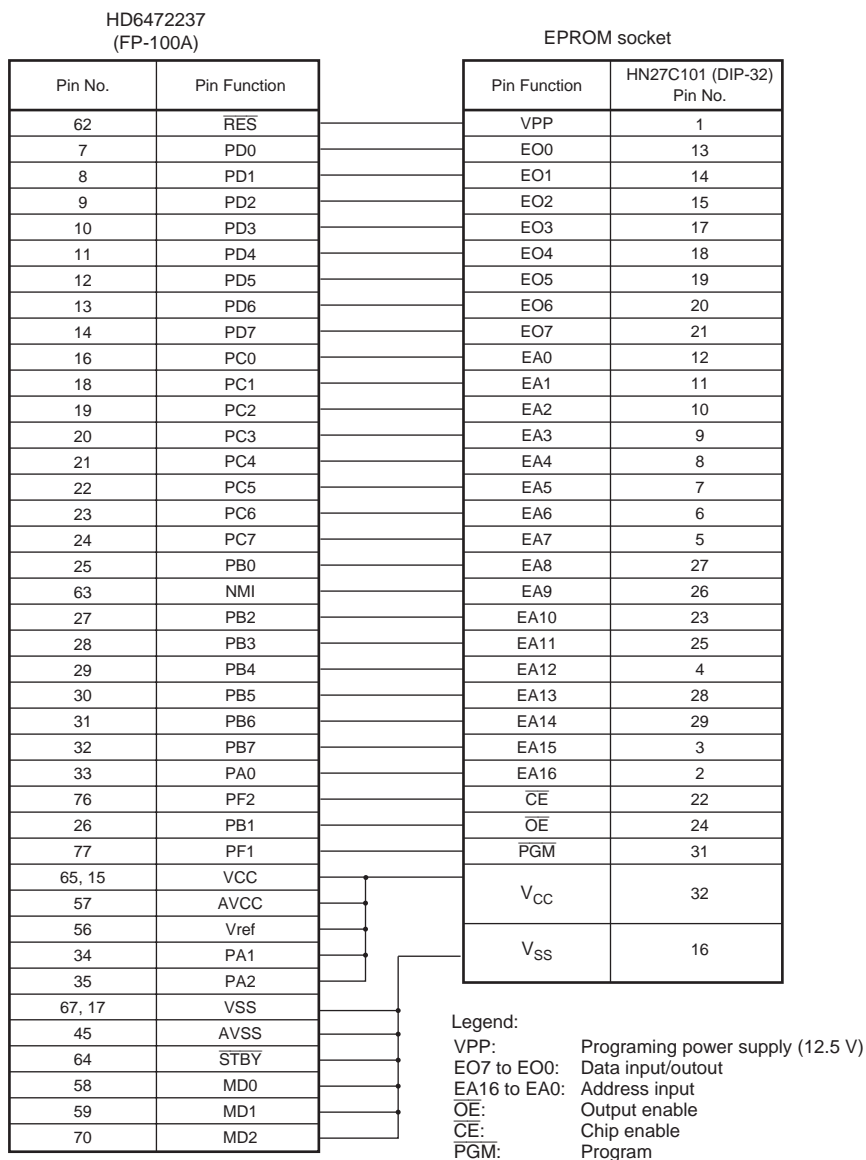
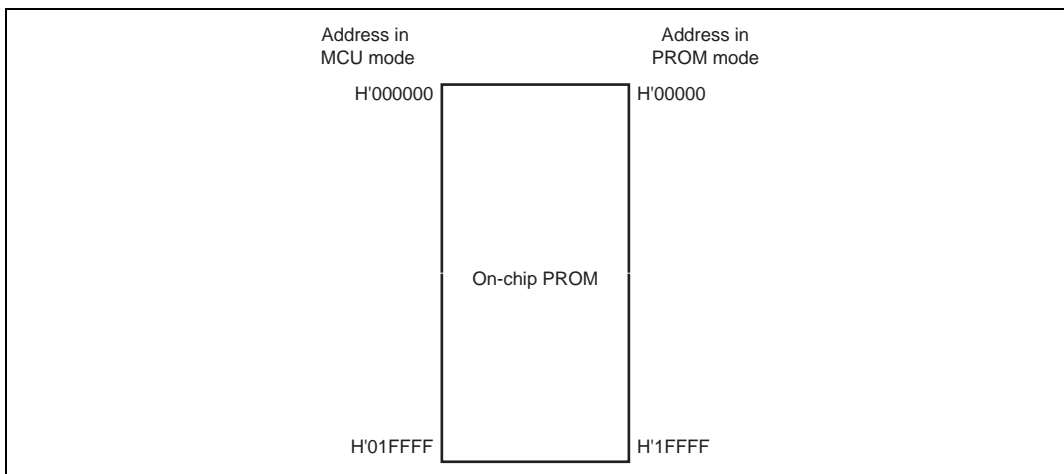


Figure 22.2 HD6472237 Socket Adapter Pin Correspondence Diagram (FP-100A)

Table 22.2 Socket Adapters

| Product Name | Package | Socket Adapter | |
|--------------|-------------------------|--------------------|-----------------|
| | | Minato Electronics | Data IO Japan |
| H8S/2237 | 100-pin TQFP (TFP-100B) | ME2237ESNS1H | H7223BT100D3201 |
| | 100-pin TQFP (TFP-100G) | ME2237ESMS1H | H7223GT100D3201 |
| | 100-pin QFP (FP-100A) | ME2237ESFS1H | H7223AQ100D3201 |
| | 100-pin QFP (FP-100B) | ME2237ESHS1H | H7223BQ100D3201 |

**Figure 22.3 Memory Map in PROM Mode**

22.3 Programming

Table 22.3 shows how to select the program, verify, and other modes in PROM mode.

Table 22.3 Mode Selection in PROM Mode

| Mode | Pins | | | | | | |
|------------------------|------------------------|------------------------|-------------------------|-----------------|-----------------|----------------|---------------|
| | $\overline{\text{CE}}$ | $\overline{\text{OE}}$ | $\overline{\text{PGM}}$ | V _{PP} | V _{CC} | E07 to E00 | EA16 to EA0 |
| Program | L | H | L | V _{PP} | V _{CC} | Data input | Address input |
| Verify | L | L | H | V _{PP} | V _{CC} | Data output | Address input |
| Programming prohibited | L | L | L | V _{PP} | V _{CC} | High impedance | Address input |
| | L | H | H | | | | |
| | H | L | L | | | | |
| | H | H | H | | | | |

Legend:

L: Low voltage level

H: High voltage level

V_{PP}: V_{PP} voltage level

V_{CC}: V_{CC} voltage level

Programming and verification should be carried out using the same specifications as for the standard HN27C101 EPROM.

However, do not set the PROM programmer to page mode does not support page programming. A PROM programmer that only supports page programming cannot be used. When choosing a PROM programmer, check that it supports high-speed programming in byte units. Always set addresses within the range H'00000 to H'1FFFF.

22.3.1 Programming and Verification

An efficient, high-speed programming procedure can be used to program and verify PROM data. This procedure writes data quickly without subjecting the chip to voltage stress or sacrificing data reliability. It leaves the data H'FF in unused addresses. Figure 22.4 shows the basic high-speed programming flowchart. Tables 22.4 and 22.5 list the electrical characteristics of the chip during programming. Figure 22.5 shows a timing chart.

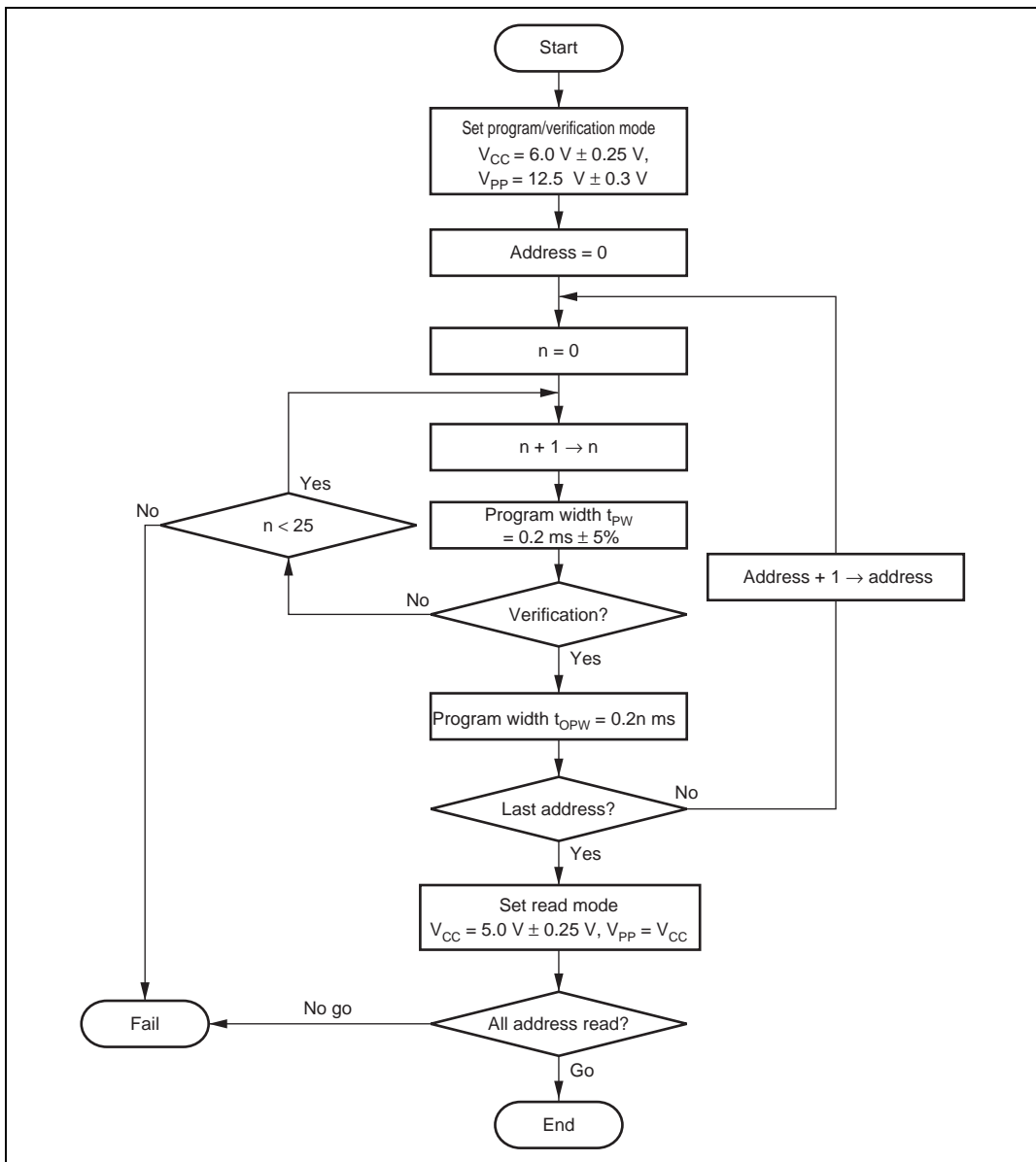


Figure 22.4 High-Speed Programming Flowchart

Table 22.4 DC Characteristics in PROM Mode(Conditions: $V_{CC} = 6.0\text{ V} \pm 0.25\text{ V}$, $V_{PP} = 12.5\text{ V} \pm 0.3\text{ V}$, $V_{SS} = 0\text{ V}$, $T_a = 25^\circ\text{C} \pm 5^\circ\text{C}$)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------|--|------------|------|-----|----------------|---------------|---------------------------------------|
| Input high voltage | EO7 to EO0, EA16 to EA0, $\overline{\text{OE}}$, $\overline{\text{CE}}$, PGM | V_{IH} | 2.4 | — | $V_{CC} + 0.3$ | V | |
| Input low voltage | EO7 to EO0, EA16 to EA0, $\overline{\text{OE}}$, $\overline{\text{CE}}$, PGM | V_{IL} | −0.3 | — | 0.8 | V | |
| Output high voltage | EO7 to EO0 | V_{OH} | 2.4 | — | — | V | $I_{OH} = -200\text{ }\mu\text{A}$ |
| Output low voltage | EO7 to EO0 | V_{OL} | — | — | 0.45 | V | $I_{OL} = 1.6\text{ mA}$ |
| Input leakage current | EO7 to EO0, EA16 to EA0, $\overline{\text{OE}}$, $\overline{\text{CE}}$, PGM | $ I_{LI} $ | — | — | 2 | μA | $V_{in} = 5.25\text{ V}/0.5\text{ V}$ |
| V_{CC} current | | I_{CC} | — | — | 40 | mA | |
| V_{PP} current | | I_{PP} | — | — | 40 | mA | |

Table 22.5 AC Characteristics in PROM Mode(Conditions: $V_{CC} = 6.0\text{ V} \pm 0.25\text{ V}$, $V_{PP} = 12.5\text{ V} \pm 0.3\text{ V}$, $T_a = 25^\circ\text{C} \pm 5^\circ\text{C}$)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|----------------|------|------|------|---------------|---------------------------|
| Address setup time | t_{AS} | 2 | — | — | μs | Figure 22.5 ^{*1} |
| $\overline{\text{OE}}$ setup time | t_{OES} | 2 | — | — | μs | |
| Data setup time | t_{DS} | 2 | — | — | μs | |
| Address hold time | t_{AH} | 0 | — | — | μs | |
| Data hold time | t_{DH} | 2 | — | — | μs | |
| Data output disable time | t_{DF}^{*2} | — | — | 130 | ns | |
| V_{PP} setup time | t_{VPS} | 2 | — | — | μs | |
| Programming pulse width | t_{PW} | 0.19 | 0.20 | 0.21 | ms | |
| PGM pulse width for overwrite programming | t_{OPW}^{*3} | 0.19 | — | 5.25 | ms | |
| V_{CC} setup time | t_{VCS} | 2 | — | — | μs | |
| $\overline{\text{CE}}$ setup time | t_{CES} | 2 | — | — | μs | |
| Data output delay time | t_{OE} | 0 | — | 150 | ns | |

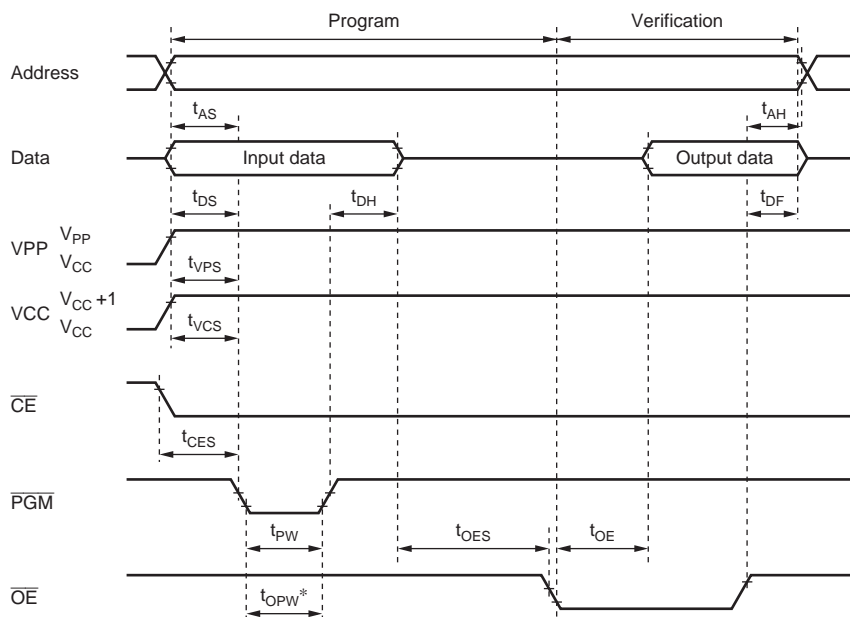
Notes: 1. Input pulse level: 0.8 V to 2.2 V

Input rise time/fall time $\leq 20\text{ ns}$

Timing reference levels: Input: 1.0 V, 2.0 V

Output: 0.8 V, 2.0 V

- t_{DF} is defined to be when output has reached the open state, and the output level can no longer be referenced.
- t_{OPW} is defined by the value shown in the flowchart.



Note: * t_{OPW} is defined by the value shown in the flowchart.

Figure 22.5 PROM Programming/Verification Timing

22.3.2 Programming Precautions

- Program using the specified voltages and timing.

The programming voltage (V_{pp}) in PROM mode is 12.5 V.

Applied voltages in excess of the specified values can permanently destroy the MCU. Be particularly careful about the PROM programmer's overshoot characteristics.

If the PROM programmer is set to Renesas Technology HN27C101 specifications, V_{pp} will be 12.5 V.

- Before programming, check that the MCU is correctly mounted in the PROM programmer. Overcurrent damage to the MCU can result if the index marks on the PROM programmer, socket adapter, and MCU are not correctly aligned.
- Do not touch the socket adapter or MCU while programming. Touching either of these can cause contact faults and programming errors.
- The MCU cannot be programmed in page programming mode. Select the programming mode carefully.

- The size of the PROM is 128 kbytes. Always set addresses within the range H'00000 to H'1FFFF. During programming, write H'FF to unused addresses to avoid verification errors.

22.3.3 Reliability of Programmed Data

An effective way to assure the data retention characteristics of the programmed chips is to bake them at 150°C, then screen them for data errors. This procedure quickly eliminates chips with PROM memory cells prone to early failure.

Figure 22.6 shows the recommended screening procedure.

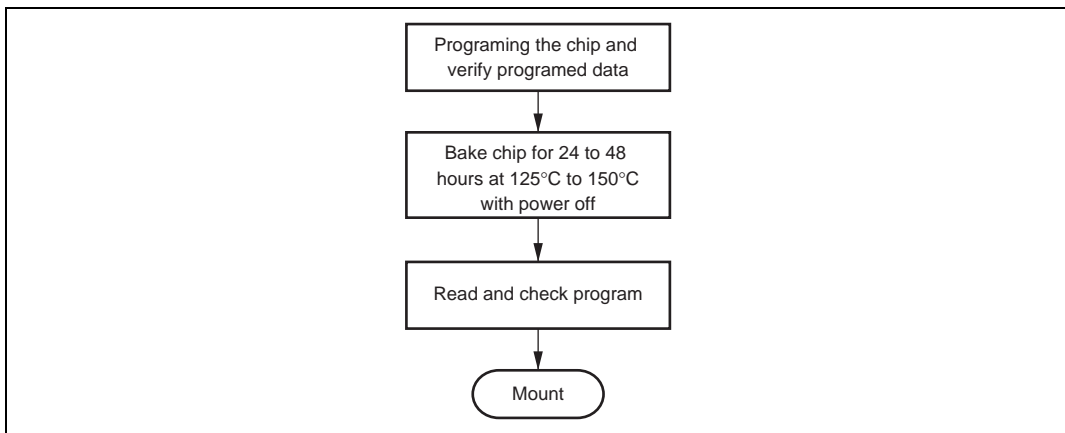


Figure 22.6 Recommended Screening Procedure

If a series of programming errors occurs while the same PROM programmer is being used, stop programming and check the PROM programmer and socket adapter for defects.

Please inform Renesas Technology of any abnormal conditions noted during or after programming or in screening of program data after high-temperature baking.

Section 23 Clock Pulse Generator

This LSI has an on-chip clock pulse generator that generates the system clock (ϕ), the bus master clock, and internal clocks. The clock pulse generator consists of an oscillator, duty adjustment circuit, clock selection circuit, medium-speed clock divider, bus master clock selection circuit, subclock oscillator, and wave formation circuit. A block diagram of the clock pulse generator is shown in figure 23.1.

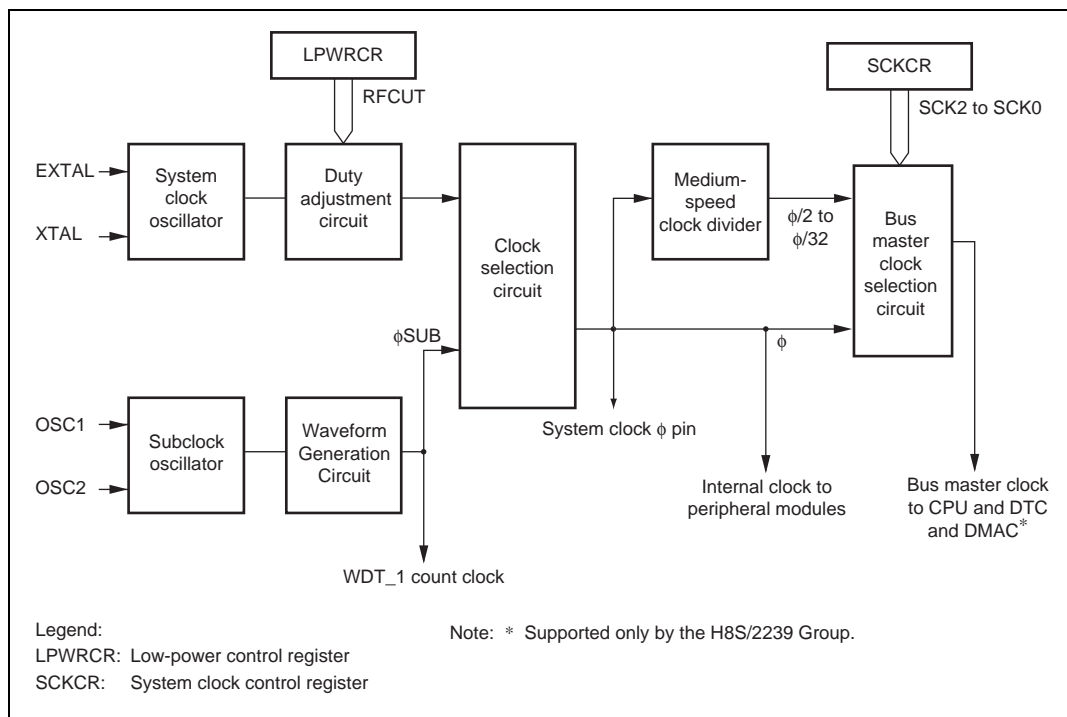


Figure 23.1 Block Diagram of Clock Pulse Generator

Frequency changes are performed by software by settings in the low-power control register (LPWRCR) and system clock control register (SCKCR).

23.1 Register Descriptions

The on-chip clock pulse generator has the following registers.

- System clock control register (SCKCR)
- Low-power control register (LPWRCR)

23.1.1 System Clock Control Register (SCKCR)

SCKCR performs medium-speed mode control.

| Bit | Bit Name | Initial Value | R/W | Description |
|------|----------|---------------|-----|---|
| 7 | PSTOP | 0 | R/W | ϕ Clock Output Prohibited Controls ϕ output. <ul style="list-style-type: none"> • High-speed mode, medium-speed mode, subactive mode, sleep mode, and subsleep mode 0: ϕ output 1: Fixed to high • Software standby mode, watch mode, and direct transition 0: Fixed to high 1: Fixed to high • Hardware standby mode 0: High impedance 1: High impedance |
| 6 | — | 0 | R/W | Reserved This bit is readable/writable, but the write value should always be 0. |
| 5, 4 | — | All 0 | — | Reserved These bits are always read as 0, and cannot be modified. |
| 3 | — | 0 | R/W | Reserved This bit is readable/writable, but the write value should always be 0. |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 2 | SCK2 | 0 | R/W | System Clock Select 2 to 0 |
| 1 | SCK1 | 0 | R/W | These bits select the bus master clock. |
| 0 | SCK0 | 0 | R/W | 000: High-speed mode 001: Medium-speed clock $\phi/2$ 010: Medium-speed clock $\phi/4$ 011: Medium-speed clock $\phi/8$ 100: Medium-speed clock $\phi/16$ 101: Medium-speed clock $\phi/32$ 11x: Setting prohibited |

Legend:

x: Don't care

23.1.2 Low-Power Control Register (LPWRCR)

LPWRCR performs down-mode control, selects sampling frequency for eliminating noise, performs subclock generation control, and specifies multiplication factor.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | DTON | 0 | R/W | <p>Direct Transfer ON Flag</p> <p>0: When the SLEEP instruction is executed in high-speed mode or medium-speed mode, operation shifts to sleep mode, software standby mode, or watch mode*.</p> <p>When the SLEEP instruction is executed in sub-active mode, operation shifts to sub-sleep mode or watch mode.</p> <p>1: When the SLEEP instruction is executed in high-speed mode or medium-speed mode, operation shifts directly to sub-active mode*, or shifts to sleep mode or software standby mode.</p> <p>When the SLEEP instruction is executed in sub-active mode, operation shifts directly to high-speed mode, or shifts to sub-sleep mode.</p> |
| 6 | LSON | 0 | R/W | <p>Low Speed ON Flag</p> <p>0: When the SLEEP instruction is executed in high-speed mode or medium-speed mode, operation shifts to sleep mode, software standby mode, or watch mode*.</p> <p>When the SLEEP instruction is executed in sub-active mode, operation shifts to watch mode* or shifts directly to high-speed mode.</p> <p>Operation shifts to high-speed mode when watch mode is cancelled.</p> <p>1: When the SLEEP instruction is executed in high-speed mode, operation shifts to watch mode or sub-active mode.</p> <p>When the SLEEP instruction is executed in sub-active mode, operation shifts to sub-sleep mode or watch mode.</p> <p>Operation shifts to sub-active mode when watch mode is cancelled.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 5 | NESEL | 0 | R/W | <p>Noise Elimination Sampling Frequency Select</p> <p>This bit selects the sampling frequency of the subclock (ϕ_{SUB}) generated by the subclock oscillator is sampled by the clock (ϕ) generated by the system clock oscillator</p> <p>Set 0 when ϕ is 5 MHz or higher. Set 1 when ϕ is 2.1 MHz or lower. Any value can be set when ϕ is 2.1 to 5 MHz.</p> <p>0: Sampling using $1/32 \times \phi$</p> <p>1: Sampling using $1/4 \times \phi$</p> |
| 4 | SUBSTP | 0 | R/W | <p>Subclock Enable</p> <p>This bit enables/disables subclock generation. This bit should be set to 1 when subclock is not used.</p> <p>0: Enables subclock generation.</p> <p>1: Disables subclock generation.</p> |
| 3 | RFCUT | 0 | R/W | <p>Oscillation Circuit Feedback Resistance Control Bit</p> <p>Selects whether or not built-in feedback resistance and duty adjustment circuit of the system clock generator are used when an external clock is input. Do not access when the crystal resonator is used.</p> <p>After setting this bit in the external clock input state, enter software standby mode, watch mode, or subactive mode. When software standby mode, watch mode, or subactive mode is entered, switch whether or not built-in feedback resistance and duty adjustment circuit are used.</p> <p>0: Built-in feedback resistance and duty adjustment circuit of the system clock generator used.</p> <p>1: Built-in feedback resistance and duty adjustment circuit of the system clock generator not used.</p> |
| 2 | — | 0 | R/W | <p>Reserved</p> <p>This bit is readable/writable, but the write value should always be 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|---|
| 1 | STC1 | 0 | R/W | Multiplication factor setting |
| 0 | STC0 | 0 | R/W | Specifies multiplication factor of the PLL circuit built in the evaluation chip. The specified multiplication factor becomes valid software standby mode, watch mode, or subactive mode is entered. These bits should be set to 11 in this LSI. Since the value becomes STC1 = STC0 = 0 after a reset, set STC1 = STC0 = 1. 00: $\times 1$ 01: $\times 2$ (setting prohibited) 10: $\times 4$ (setting prohibited) 11: PLL is bypass |

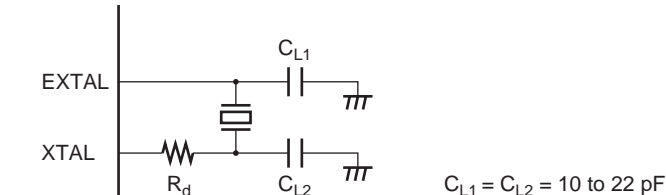
Note: * When watch mode or subactive mode is entered, set high-speed mode.

23.2 System Clock Oscillator

System clock pulses can be supplied by connecting a crystal resonator, or by input of an external clock.

23.2.1 Connecting a Crystal Resonator

A crystal resonator can be connected as shown in the example in figure 23.2. Select the damping resistance R_d according to table 23.1. An AT-cut parallel-resonance crystal should be used.



Note: C_{L1} and C_{L2} are reference values including the floating capacitance of the board.

Figure 23.2 Connection of Crystal Resonator (Example)

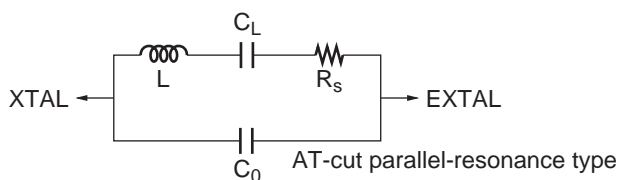
Table 23.1 Damping Resistance Value

| Frequency (MHz) | 2 ^{*1} | 4 ^{*1} | 6 ^{*1} | 8 ^{*1} | 10 | 12 | 16 ^{*2} | 20 ^{*2} |
|--------------------|-----------------|-----------------|-----------------|-----------------|-----|----|------------------|------------------|
| R _d (Ω) | 1 k | 500 | 300 | 200 | 100 | 0 | 0 | 0 |

Notes: 1. The H8S/2258 Group is out of operation.

2. Supported only by the H8S/2239 Group.

Figure 23.3 shows the equivalent circuit of the crystal resonator. Use a crystal resonator that has the characteristics shown in table 23.2.

**Figure 23.3 Crystal Resonator Equivalent Circuit****Table 23.2 Crystal Resonator Characteristics**

| Frequency (MHz) | 2 ^{*1} | 4 ^{*1} | 6 ^{*1} | 8 ^{*1} | 10 | 12 | 16 ^{*2} | 20 ^{*2} |
|-------------------------|-----------------|-----------------|-----------------|-----------------|----|----|------------------|------------------|
| R _s max (Ω) | 500 | 120 | 100 | 80 | 60 | 60 | 50 | 40 |
| C ₀ max (pF) | 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |

Notes: 1. The H8S/2258 Group is out of operation.

2. Supported only by the H8S/2239 Group.

23.2.2 External Clock Input

An external clock signal can be input as shown in the examples in figure 23.4. If the XTAL pin is left open, ensure that stray capacitance does not exceed 10 pF. When complementary clock is input to the XTAL pin, the external clock input should be fixed high in standby mode, subactive mode, subsleep mode, or watch mode.

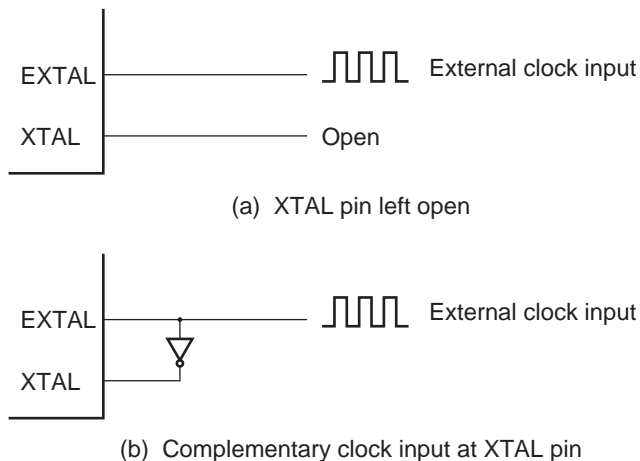
**Figure 23.4 External Clock Input (Examples)**

Table 23.3 shows the input conditions for the external clock. Table 23.4 shows the input conditions for the external clock when duty adjustment circuit is not used.

Table 23.3 External Clock Input Conditions (1) (H8S/2258 Group)

| Item | Symbol | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ | | Unit | Test Conditions |
|---------------------------------------|-----------|---|-----|-----------|-----------------|
| | | Min | Max | | |
| External clock input low pulse width | t_{EXL} | 30 | — | ns | Figure 23.5 |
| External clock input high pulse width | t_{EXH} | 30 | — | ns | |
| External clock rise time | t_{EXr} | — | 7 | ns | |
| External clock fall time | t_{EXf} | — | 7 | ns | Figure 27.10 |
| Clock low pulse width | t_{CL} | 0.4 | 0.6 | t_{CYC} | |
| Clock high pulse width | t_{CH} | 0.4 | 0.6 | t_{CYC} | |

Table 23.3 External Clock Input Conditions (2) (H8S/2238B, H8S/2236B)

| Item | Symbol | F-ZTAT | | Masked ROM | | Unit | Test Conditions | |
|---------------------------------------|-----------|---|-----|---|-----|-----------|--------------------------|--------------|
| | | $V_{cc} = 3.0\text{ V to }5.5\text{ V}$ | | $V_{cc} = 2.7\text{ V to }5.5\text{ V}$ | | | | |
| | | Min | Max | Min | Max | | | |
| External clock input low pulse width | t_{EXL} | 30 | — | 30 | — | ns | Figure 23.5 | |
| External clock input high pulse width | t_{EXH} | 30 | — | 30 | — | ns | | |
| External clock rise time | t_{EXr} | — | 7 | — | 7 | ns | | |
| External clock fall time | t_{EXf} | — | 7 | — | 7 | ns | | |
| Clock low pulse width | t_{CL} | 0.4 | 0.6 | 0.4 | 0.6 | t_{cyc} | $\phi \geq 5\text{ MHz}$ | Figure 27.10 |
| | | 80 | — | 80 | — | ns | $\phi < 5\text{ MHz}$ | |
| Clock high pulse width | t_{CH} | 0.4 | 0.6 | 0.4 | 0.6 | t_{cyc} | $\phi \geq 5\text{ MHz}$ | |
| | | 80 | — | 80 | — | ns | $\phi < 5\text{ MHz}$ | |

Table 23.3 External Clock Input Conditions (3) (H8S/2238R, H8S/2236R)

| Item | Symbol | F-ZTAT | | Masked ROM | | Unit | Test Conditions | |
|---------------------------------------|-----------|---|-----|---|------|-----------|--------------------------|--------------|
| | | $V_{cc} = 2.7\text{ V to }3.6\text{ V}$ | | $V_{cc} = 2.2\text{ V to }3.6\text{ V}$ | | | | |
| | | Min | Max | Min | Max | | | |
| External clock input low pulse width | t_{EXL} | 30 | — | 65 | — | ns | Figure 23.5 | |
| External clock input high pulse width | t_{EXH} | 30 | — | 65 | — | ns | | |
| External clock rise time | t_{EXr} | — | 7 | — | 15 | ns | | |
| External clock fall time | t_{EXf} | — | 7 | — | 15 | ns | | |
| Clock low pulse width | t_{CL} | 0.4 | 0.6 | 0.35 | 0.65 | t_{cyc} | $\phi \geq 5\text{ MHz}$ | Figure 27.10 |
| | | 80 | — | 70 | — | ns | $\phi < 5\text{ MHz}$ | |
| Clock high pulse width | t_{CH} | 0.4 | 0.6 | 0.35 | 0.65 | t_{cyc} | $\phi \geq 5\text{ MHz}$ | |
| | | 80 | — | 70 | — | ns | $\phi < 5\text{ MHz}$ | |

Table 23.3 External Clock Input Conditions (4) (H8S/2237 Group, H8S/2227 Group)

| Item | Symbol | F-ZTAT and Masked ROM | | Masked ROM | | ZTAT | | Unit | Test Conditions | |
|---------------------------------------|-----------|------------------------------|-----|------------------------------|------|------------------------------|-----|-----------|-------------------|--------------|
| | | $V_{CC} =$ 2.7 V to 3.6 V | | $V_{CC} =$ 2.2 V to 3.6 V | | $V_{CC} =$ 2.7 V to 3.6 V | | | | |
| | | Min | Max | Min | Max | Min | Max | | | |
| External clock input low pulse width | t_{EXL} | 30 | — | 65 | — | 40 | — | ns | Figure 23.5 | |
| External clock input high pulse width | t_{EXH} | 30 | — | 65 | — | 40 | — | ns | | |
| External clock rise time | t_{EXr} | — | 7 | — | 15 | — | 10 | ns | | |
| External clock fall time | t_{EXf} | — | 7 | — | 15 | — | 10 | ns | | |
| Clock low pulse width | t_{CL} | 0.4 | 0.6 | 0.35 | 0.65 | 0.4 | 0.6 | t_{cyc} | $\phi \geq 5$ MHz | Figure 27.10 |
| | | 80 | — | 70 | — | 80 | — | ns | $\phi < 5$ MHz | |
| Clock high pulse width | t_{CH} | 0.4 | 0.6 | 0.35 | 0.65 | 0.4 | 0.6 | t_{cyc} | $\phi \geq 5$ MHz | |
| | | 80 | — | 70 | — | 80 | — | ns | $\phi < 5$ MHz | |

Table 23.3 External Clock Input Conditions (5) (H8S/2239 Group)

| | | F-ZTAT and Masked ROM | | | | Masked ROM | | | | |
|---------------------------------------|-----------|------------------------------|-----|------------------------------|------|------------------------------|------|-----------|-------------------|--------------|
| | | $V_{cc} =$ 3.0 V to 3.6 V | | $V_{cc} =$ 2.7 V to 3.6 V | | $V_{cc} =$ 2.2 V to 3.6 V | | | | |
| Item | Symbol | Min | Max | Min | Max | Min | Max | Unit | Test Conditions | |
| External clock input low pulse width | t_{EXL} | 20 | — | 25 | — | 65 | — | ns | Figure 23.5 | |
| External clock input high pulse width | t_{EXH} | 20 | — | 25 | — | 65 | — | ns | | |
| External clock rise time | t_{EXr} | — | 5 | — | 6.25 | — | 15 | ns | | |
| External clock fall time | t_{EXf} | — | 5 | — | 6.25 | — | 15 | ns | | |
| Clock low pulse width | t_{CL} | 0.4 | 0.6 | 0.4 | 0.6 | 0.35 | 0.65 | t_{CYC} | $\phi \geq 5$ MHz | Figure 27.10 |
| | | — | — | 80 | — | 70 | — | ns | $\phi < 5$ MHz | |
| Clock high pulse width | t_{CH} | 0.4 | 0.6 | 0.4 | 0.6 | 0.35 | 0.65 | t_{CYC} | $\phi \geq 5$ MHz | |
| | | — | — | 80 | — | 70 | — | ns | $\phi < 5$ MHz | |

Table 23.4 External Clock Input Conditions (Duty Adjustment Circuit Unused) (1)
(H8S/2258 Group)

| Item | Symbol | $V_{CC} = 4.0 \text{ V to } 5.5 \text{ V}$ | | Unit | Test Conditions |
|---------------------------------------|-----------|--|-----|------|-----------------|
| | | Min | Max | | |
| External clock input low pulse width | t_{EXL} | 37 | — | ns | Figure 23.5 |
| External clock input high pulse width | t_{EXH} | 37 | — | ns | |
| External clock rise time | t_{EXr} | — | 7 | ns | |
| External clock fall time | t_{EXf} | — | 7 | ns | |

Note: If the duty adjustment circuit is not used, the maximum operating frequency will be lower to match the input waveform.

(Example: If $t_{EXL} = t_{EXH} = 37 \text{ ns}$ and $t_{EXr} = t_{EXf} = 7 \text{ ns}$, the clock cycle = 88 ns and the maximum operating frequency = 11.3 MHz)

Table 23.4 External Clock Input Conditions (Duty Adjustment Circuit Unused) (2)
(H8S/2238B, H8S/2236B)

| Item | Symbol | F-ZTAT | | Masked ROM | | Unit | Test Conditions |
|---------------------------------------|------------------|----------------------------------|-----|----------------------------------|-----|------|-----------------|
| | | V _{cc} = 3.0 V to 5.5 V | | V _{cc} = 2.7 V to 5.5 V | | | |
| | | Min | Max | Min | Max | | |
| External clock input low pulse width | t _{EXL} | 37 | — | 37 | — | ns | Figure 23.5 |
| External clock input high pulse width | t _{EXH} | 37 | — | 37 | — | ns | |
| External clock rise time | t _{EXr} | — | 7 | — | 7 | ns | |
| External clock fall time | t _{EXf} | — | 7 | — | 7 | ns | |

Note: If the duty adjustment circuit is not used, the maximum operating frequency will be lower to match the input waveform.

(Example: If $t_{EXL} = t_{EXH} = 37 \text{ ns}$ and $t_{EXr} = t_{EXf} = 7 \text{ ns}$, the clock cycle = 88 ns and the maximum operating frequency = 11.3 MHz)

Table 23.4 External Clock Input Conditions (Duty Adjustment Circuit Unused) (3)
(H8S/2238R, H8S/2236R)

| Item | Symbol | F-ZTAT | | F-ZTAT and Masked ROM | | Unit | Test Conditions |
|---------------------------------------|-----------|---|-----|---|-----|------|-----------------|
| | | $V_{cc} = 2.7\text{ V to }3.6\text{ V}$ | | $V_{cc} = 2.2\text{ V to }3.6\text{ V}$ | | | |
| | | Min | Max | Min | Max | | |
| External clock input low pulse width | t_{EXL} | 37 | — | 80 | — | ns | Figure 23.5 |
| External clock input high pulse width | t_{EXH} | 37 | — | 80 | — | ns | |
| External clock rise time | t_{EXr} | | 7 | — | 15 | ns | |
| External clock fall time | t_{EXf} | — | 7 | — | 15 | ns | |

Note: If the duty adjustment circuit is not used, the maximum operating frequency will be lower to match the input waveform.

(Example: If $t_{EXL} = t_{EXH} = 37\text{ ns}$ and $t_{EXr} = t_{EXf} = 7\text{ ns}$, the clock cycle = 88 ns and the maximum operating frequency = 11.3 MHz)

Table 23.4 External Clock Input Conditions (Duty Adjustment Circuit Unused) (4)
(H8S/2237 Group, H8S/2227 Group)

| Item | Symbol | F-ZTAT and Masked ROM | | Masked ROM | | ZTAT | | Unit | Test Conditions |
|---------------------------------------|-----------|---|-----|---|-----|---|-----|------|-----------------|
| | | $V_{cc} = 2.7\text{ V to }3.6\text{ V}$ | | $V_{cc} = 2.2\text{ V to }3.6\text{ V}$ | | $V_{cc} = 2.7\text{ V to }3.6\text{ V}$ | | | |
| | | Min | Max | Min | Max | Min | Max | | |
| External clock input low pulse width | t_{EXL} | 37 | — | 80 | — | 50 | — | ns | Figure 23.5 |
| External clock input high pulse width | t_{EXH} | 37 | — | 80 | — | 50 | — | ns | |
| External clock rise time | t_{EXr} | — | 7 | — | 15 | — | 10 | ns | |
| External clock fall time | t_{EXf} | — | 7 | — | 15 | — | 10 | ns | |

Note: If the duty adjustment circuit is not used, the maximum operating frequency will be lower to match the input waveform.

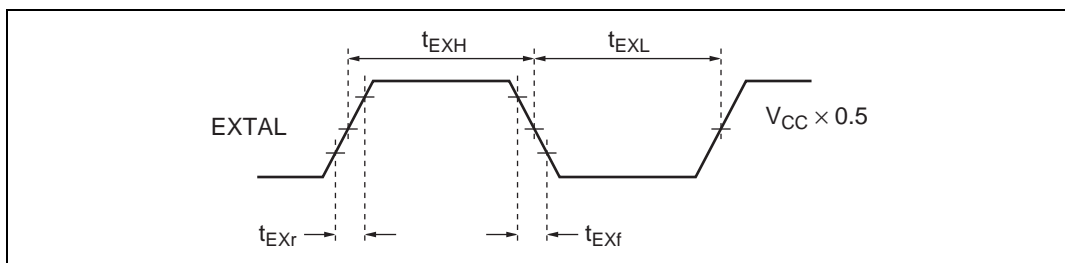
(Example: If $t_{EXL} = t_{EXH} = 37\text{ ns}$ and $t_{EXr} = t_{EXf} = 7\text{ ns}$, the clock cycle = 88 ns and the maximum operating frequency = 11.3 MHz)

**Table 23.4 External Clock Input Conditions (Duty Adjustment Circuit Unused) (5)
(H8S/2239 Group)**

| Item | Symbol | F-ZTAT and Masked ROM | | | | Masked ROM | | Unit | Test Conditions |
|---------------------------------------|-----------|--|-----|--|------|--|-----|------|-----------------|
| | | $V_{CC} = 3.0\text{ V to } 3.6\text{ V}$ | | $V_{CC} = 2.7\text{ V to } 3.6\text{ V}$ | | $V_{CC} = 2.2\text{ V to } 3.6\text{ V}$ | | | |
| | | Min | Max | Min | Max | Min | Max | | |
| External clock input low pulse width | t_{EXL} | 25 | — | 31.25 | — | 80 | — | ns | Figure 23.5 |
| External clock input high pulse width | t_{EXH} | 25 | — | 31.25 | — | 80 | — | ns | |
| External clock rise time | t_{EXr} | — | 5 | — | 6.25 | — | 15 | ns | |
| External clock fall time | t_{EXf} | — | 5 | — | 6.25 | — | 15 | ns | |

Note: When a duty adjustment circuit is not used, maximum operating frequency is lowered according to the input waveform.

(Example: When $t_{EXL} = t_{EXH} = 25 \text{ ns}$, $t_{EXr} = t_{EXf} = 5 \text{ ns}$, clock cycle time = 60 ns, and maximum operating frequency = 16.6 MHz)

**Figure 23.5 External Clock Input Timing**

23.2.3 Notes on Switching External Clock

When two or more external clocks (e.g.:10 MHz and 2 MHz) are used as the system clock, input clock should be switched in software standby mode.

An example of external clock switching circuit is shown in figure 23.6. An example of external clock switching timing is shown in figure 23.7.

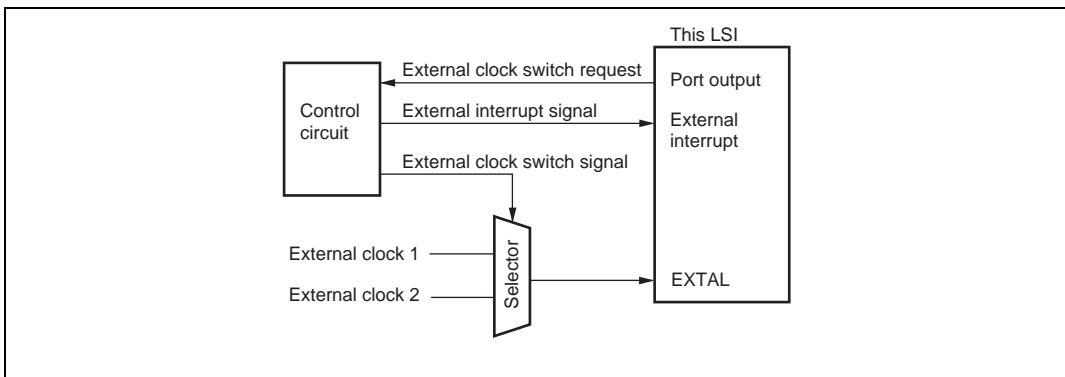


Figure 23.6 External Clock Switching Circuit (Example)

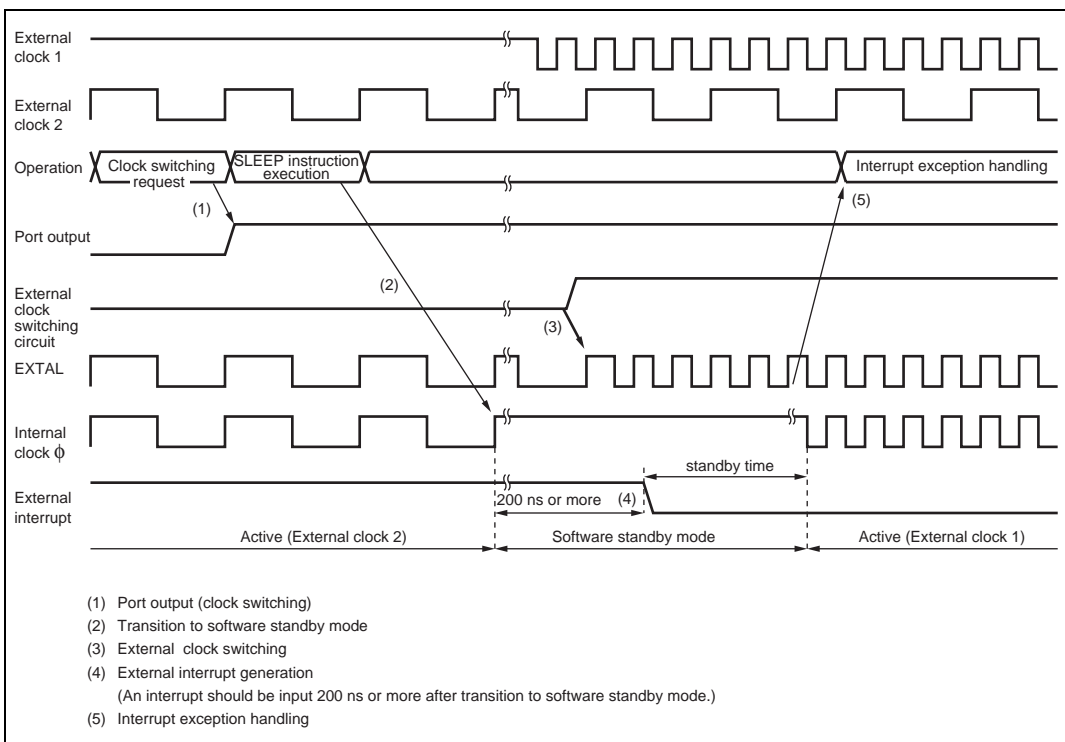


Figure 23.7 External Clock Switching Timing (Example)

23.3 Duty Adjustment Circuit

The duty adjustment circuit is valid when oscillation frequency is more than 5 MHz. The duty adjustment circuit adjusts clock output from the system clock oscillator to generate the system clock (ϕ).

23.4 Medium-Speed Clock Divider

The medium-speed clock divider divides the system clock to generate $\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, and $\phi/32$.

23.5 Bus Master Clock Selection Circuit

The bus master clock selection circuit selects the clock supplied to the bus master by setting the bits SCK2 to SCK0 in SCKCR. The bus master clock can be selected from system clock (ϕ), or medium-speed clocks ($\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, $\phi/32$).

23.6 System Clock when Using IEBus

When using the IEBus, the system clock must be set to either 12 MHz or 12.58 MHz. When the IEBus is not used, the system clock can be set to an arbitrary frequency between 10 MHz to 13.5 MHz.

Note: IEBus is supported only by the H8S/2258 Group.

23.7 Subclock Oscillator

23.7.1 Connecting 32.768-kHz Crystal Resonator

To supply a clock to the subclock divider, connect a 32.768-kHz crystal resonator, as shown in figure 23.8. Figure 23.9 shows the equivalence circuit for a 32.768-kHz oscillator.

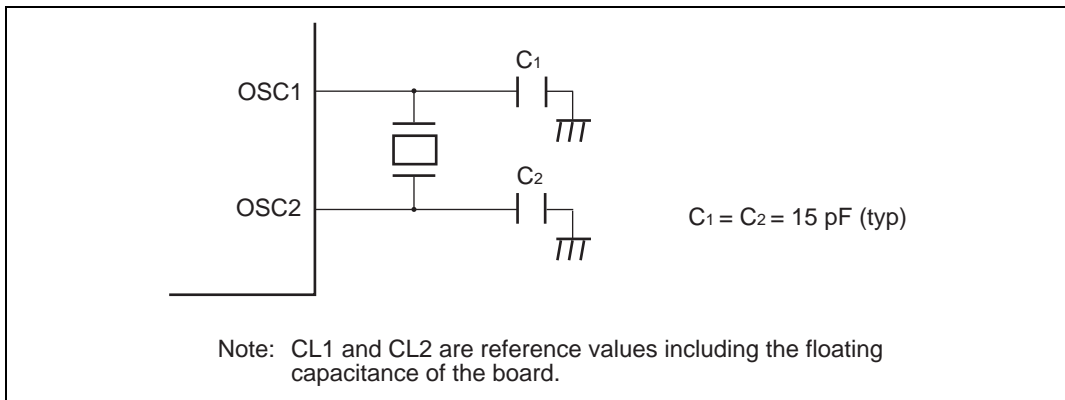


Figure 23.8 Connection Example of 32.768-kHz Quartz Oscillator

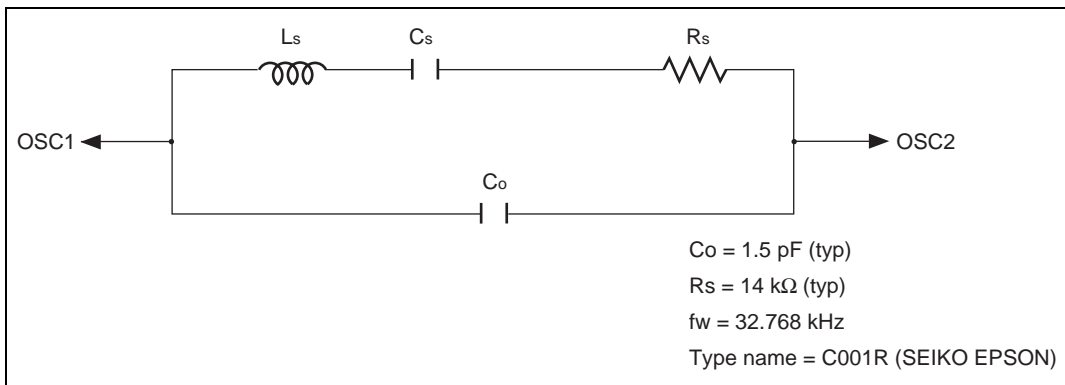


Figure 23.9 Equivalence Circuit for 32.768-kHz Oscillator

23.7.2 Handling Pins when Subclock Not Required

If no subclock is required, connect the OSC1 pin to Vss and leave OSC2 open, as shown in figure 23.10. The SUBSTP bit in LPWRCR must be set to 1. If the SUBSTP bit is not set to 1, transitions to the power-down modes may not complete normally.

On the H8S/2237 and H8S/2227 Group, the OSC1 pin should be connected to VCC.

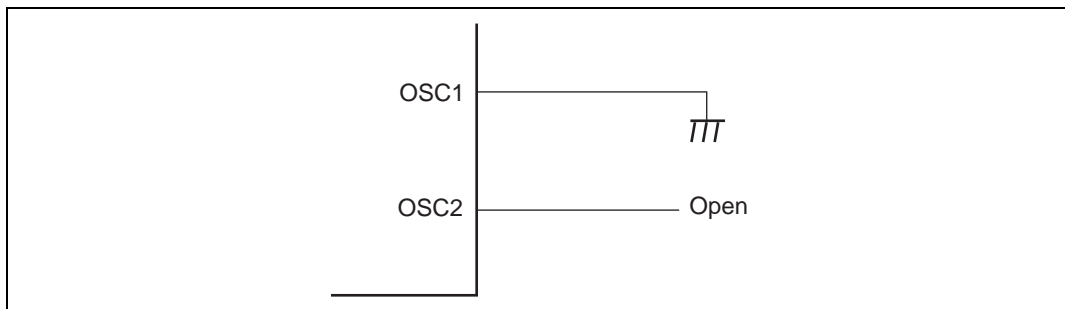


Figure 23.10 Pin Handling when Subclock Not Required

23.8 Subclock Waveform Generation Circuit

To eliminate noise from the subclock input to OSC1, the subclock is sampled using the dividing clock ϕ . The sampling frequency is set using the NESEL bit of LPWRCR. For details, see section 23.1.2, Low Power Control Register (LPWRCR).

No sampling is performed in sub-active mode, sub-sleep mode, or watch mode.

23.9 Usage Notes

23.9.1 Note on Crystal Resonator

As various characteristics related to the crystal resonator are closely linked to the user's board design, thorough evaluation is necessary on the user's part, using the resonator connection examples shown in this section as a guide. As the resonator circuit ratings will depend on the floating capacitance of the resonator and the mounting circuit, the ratings should be determined in consultation with the resonator manufacturer. The design must ensure that a voltage exceeding the maximum rating is not applied to the oscillator pin.

23.9.2 Note on Board Design

When designing the board, place the crystal resonator and its load capacitors as close as possible to the EXTAL, XTAL, OSC1, and OSC2 pins. Make wires as short as possible. Other signal lines should be routed away from the oscillator circuit, as shown in figure 23.11. This is to prevent induction from interfering with correct oscillation.

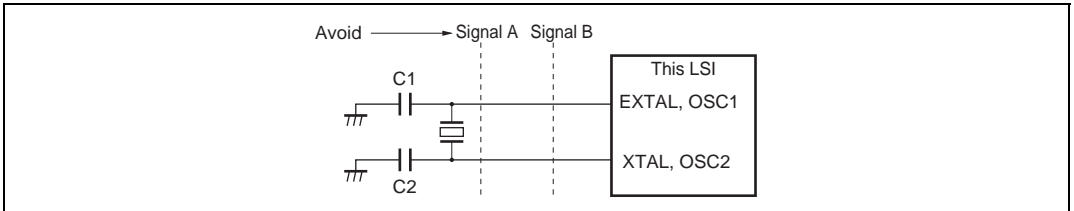


Figure 23.11 Note on Board Design of Oscillator Circuit

Section 24 Power-Down Modes

In addition to the normal program execution state, this LSI has nine power-down modes in which operation of the CPU and oscillator is halted and power dissipation is reduced. Low-power operation can be achieved by individually controlling the CPU, on-chip peripheral modules, and so on.

This LSI operating modes are as follows:

1. High-speed mode
2. Medium-speed mode
3. Subactive mode
4. Sleep mode
5. Subsleep mode
6. Watch mode
7. Module stop mode
8. Software standby mode
9. Hardware standby mode

2. to 9. are low power dissipation states. Sleep mode and subsleep mode are CPU states, medium-speed mode is a CPU and bus master state, subactive mode is a CPU and bus master and internal peripheral function state, and module stop mode is an internal peripheral function (including bus masters other than the CPU) state. Some of these states can be combined.

After a reset, the LSI is in high-speed mode with modules other than the DTC in module stop mode.

Table 24.1 shows the internal state of the LSI in the respective modes. Table 24.2 shows the conditions for shifting between the low power dissipation modes.

Figure 24.1 is a mode transition diagram.

Table 24.1 LSI Internal States in Each Mode

| Function | | High-Speed | Medium-Speed | Sleep | Module Stop | Watch | Sub active | Subsleep | Software Standby | Hardware Standby |
|------------------------------|--------------------|---------------------|------------------------|---------------------|--------------------------------|--------------------|--------------------|--------------------|---------------------|------------------|
| System clock pulse generator | | Function-ing | Function-ing | Function-ing | Function-ing | Halted | Halted | Halted | Halted | Halted |
| Subclock pulse generator | | Function-ing/halted | Function-ing/halted | Function-ing/halted | Function-ing/halted | Function-ing | Function-ing | Function-ing | Function-ing/halted | Halted |
| CPU | Instruc-tions | Function-ing | Medium-speed operation | Halted | Function-ing | Halted | Subclock operation | Halted | Halted | Halted |
| | Registers | | | Retained | | Retained | | Retained | | |
| RAM | | Function-ing | Function-ing | Function-ing (DTC) | Function-ing | Retained | Function-ing | Retained | Retained | Retained |
| I/O | | Function-ing | Function-ing | Function-ing | Function-ing | Retained | Function-ing | Function-ing | Retained | High impedance |
| External interrupts | NMI | Function-ing | Function-ing | Function-ing | Function-ing | Function-ing | Function-ing | Function-ing | Function-ing | Halted |
| | IRQn | | | | | | | | | |
| Peripheral functions | PBC | Function-ing | Medium-speed operation | Function-ing | Function-ing/halted (retained) | Halted (retained) | Subclock operation | Halted (retained) | Halted (retained) | Halted (reset) |
| | DTC | Function-ing | Medium-speed operation | Function-ing | Function-ing/halted (retained) | Halted (retained) | Halted (retained) | Halted (retained) | Halted (retained) | Halted (reset) |
| | DMAC*1 | | | | | | | | | |
| | WDT_1 | Function-ing | Function-ing | Function-ing | Function-ing | Subclock operation | Subclock operation | Subclock operation | Halted (retained) | Halted (reset) |
| | WDT_0 | Function-ing | Function-ing | Function-ing | Function-ing | Halted (retained) | Subclock operation | Subclock operation | Halted (retained) | Halted (reset) |
| | TMR | Function-ing | Function-ing | Function-ing | Function-ing/halted (retained) | Halted (retained) | Subclock operation | Subclock operation | Halted (retained) | Halted (reset) |
| | TPU | Function-ing | Function-ing | Function-ing | Function-ing/halted (retained) | Halted (retained) | Halted (retained) | Halted (retained) | Halted (retained) | Halted (reset) |
| | SCI | | | | | | | | | |
| | I ² C*2 | | | | | | | | | |
| | D/A*3*5 | | | | | | | | | |
| | A/D | Function-ing | Function-ing | Function-ing | Function-ing/halted (reset) | Halted (reset) | Halted (reset) | Halted (reset) | Halted (reset) | Halted (reset) |
| | IEB*4 | | | | | | | | | |

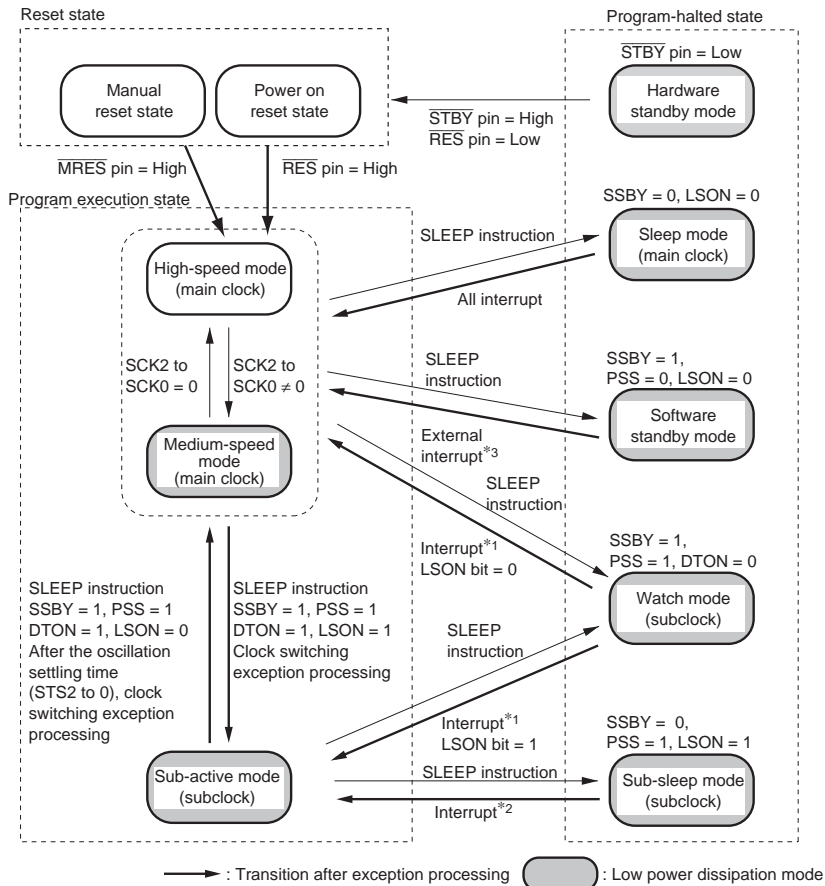
Notes: "Halted (retained)" means that internal register values are retained. The internal state is "operation suspended".

"Halted (reset)" means that internal register values and internal states are initialized.

In module stop mode, only modules for which a stop setting has been made are halted (reset or retained).

1. Supported only by the H8S/2239 Group.
2. Not available in the H8S/2237 Group and H8S/2227 Group.
3. Not available in the H8S/2227 Group.

4. Supported only by the H8S/2258 Group.
5. The analog output value does not satisfy the specified D/A absolute accuracy when D/A is halted (retained). However, the H8S/2258 Group, H8S/2238B, and H8S/2236B satisfy the specified D/A absolute accuracy.



Notes: When a transition is made between modes by means of an interrupt, the transition cannot be made on interrupt source generation alone. Ensure that interrupt handling is performed after accepting the interrupt request.

From any state except hardware standby mode, a transition to the reset state occurs when $\overline{\text{RES}}$ is driven Low. At any state except hardware standby mode and power-on reset state, a transition to the manual reset state occurs when the MRES pin is driven Low.

From any state, a transition to hardware standby mode occurs when $\overline{\text{STBY}}$ is driven low.

Always select high-speed mode before making a transition to watch mode or sub-active mode.

1. NMI, IRQ7 to IRQ0, and WDT1 interrupts.

2. NMI, IRQ7 to IRQ0, and WDT1, WDT0, and TMR3 to TMR0 interrupts.

3. NMI, IRQ7 to IRQ0

Figure 24.1 Mode Transition Diagram

Table 24.2 Low Power Dissipation Mode Transition Conditions

| Pre-Transition State | Status of Control Bit at Transition | | | | State After Transition Invoked by SLEEP Instruction | State After Transition Back from Low Power Mode Invoked by Interrupt |
|-----------------------------|-------------------------------------|-----|------|------|---|--|
| | SSBY | PSS | LSON | DTON | | |
| High-speed/ Medium-speed | 0 | × | 0 | × | Sleep | High-speed/medium-speed |
| | 0 | × | 1 | × | — | — |
| | 1 | 0 | 0 | × | Software standby | High-speed/medium-speed |
| | 1 | 0 | 1 | × | — | — |
| | 1 | 1 | 0 | 0 | Watch | High-speed |
| | 1 | 1 | 1 | 0 | Watch | Subactive |
| | 1 | 1 | 0 | 1 | — | — |
| | 1 | 1 | 1 | 1 | Subactive | — |
| Subactive | 0 | 0 | × | × | — | — |
| | 0 | 1 | 0 | × | — | — |
| | 0 | 1 | 1 | × | Subsleep | Subactive |
| | 1 | 0 | × | × | — | — |
| | 1 | 1 | 0 | 0 | Watch | High-speed |
| | 1 | 1 | 1 | 0 | Watch | Subactive |
| | 1 | 1 | 0 | 1 | High-speed | — |
| | 1 | 1 | 1 | 1 | — | — |

Legend:

×: Don't care

—: Don't set

24.1 Register Description

The following registers relate to the power-down modes. For details on system clock control register (SCKCR), refer to section 23.1.1, System Clock Control Register (SCKCR). For details on low power control register (LPWRCR), refer to section 23.1.2, Low Power Control Register (LPWRCR). For details on timer control status register (TCSR_1), refer to section 13.3.2, Timer Control/Status Register (TCSR).

- Standby control register (SBYCR)
- Module stop control register A (MSTPCRA)
- Module stop control register B (MSTPCRB)
- Module stop control register C (MSTPCRC)
- Low power control register (LPWRCR)
- System clock control register (SCKCR)
- Timer control status register (TCSR_1)

24.1.1 Standby Control Register (SBYCR)

SBYCR performs power-down mode control.

| Bit | Bit Name | Initial Value | R/W | Description |
|-----|----------|---------------|-----|--|
| 7 | SSBY | 0 | R/W | <p>Software Standby</p> <p>Specifies transition destination when the SLEEP instruction is executed.</p> <p>0: Shifts to sleep mode when the SLEEP instruction is executed in high-speed mode or medium-speed mode. Shifts to subsleep mode when the SLEEP instruction is executed in subactive mode.</p> <p>1: Shifts to software standby mode, subactive mode, and watch mode when the SLEEP instruction is executed in high-speed mode or medium-speed mode. Shifts to watch mode or high-speed mode when the SLEEP instruction is executed in subactive mode.</p> <p>Note that the value of the SSBY bit does not change even when software standby mode is canceled and making normal operation mode transition by executing an external interrupt. To clear this bit, 0 should be written to.</p> |

| Bit | Bit Name | Initial Value | R/W | Description |
|--------|----------|---------------|-----|---|
| 6 | STS2 | 0 | R/W | Standby Timer Select 2 to 0 |
| 5 | STS1 | 0 | R/W | These bits select the MCU wait time for clock settling to cancel software standby mode, watch mode, or subactive mode. With a crystal resonator (tables 24.3, 27.5, 27.17, 27.30, 27.42, 27.53), select a wait time of t_{osc2} ms (oscillation settling time) or more, depending on the operating frequency. With an external clock, there are no specific wait requirements. 000: Standby time = 8192 states 001: Standby time = 16384 states 010: Standby time = 32768 states 011: Standby time = 65536 states 100: Standby time = 131072 states 101: Standby time = 262144 states 110: Reserved 111: Standby time = 16 states* |
| 4 | STS0 | 0 | R/W | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| 3 | OPE | 1 | R/W | Output Port Enable Specifies whether the output of the address bus and bus control signals ($\overline{CS7}$ to $\overline{CS0}$, \overline{AS} , \overline{RD} , \overline{HWR} , and \overline{LWR}) should be retained or driven to the high impedance state, when shifting to software standby mode, watch mode, or direct transition. 0: High impedance 1: Output is retained. |
| 2 to 0 | — | All 0 | — | Reserved These bits are always read as 0 and cannot be modified. |

Note: * Don't set 16 states for standby time in the F-ZTAT version. 8192 states or more should be set.

24.1.2 Module Stop Control Registers A to C (MSTPCRA to MSTPCRC)

MSTPCR performs module stop mode control. When bits in MSTPCR registers are set to 1, module stop mode is set. When cleared to 0, module stop mode is cleared.

- MSTPCRA

| Bit | Bit Name | Initial Value | R/W | Target Module |
|-----|----------------------|---------------|-----|--|
| 7 | MSTPA7 | 0 | R/W | DMA controller (DMAC) ^{*2} |
| 6 | MSTPA6 | 0 | R/W | Data transfer controller (DTC) |
| 5 | MSTPA5 | 1 | R/W | 16-bit timer pulse unit (TPU) |
| 4 | MSTPA4 | 1 | R/W | 8-bit timer (TMR_0, TMR_1) |
| 3 | MSTPA3 ^{*1} | 1 | R/W | |
| 2 | MSTPA2 ^{*1} | 1 | R/W | |
| 1 | MSTPA1 | 1 | R/W | A/D converter |
| 0 | MSTPA0 | 1 | R/W | 8-bit timer (TMR_2 ^{*3} , TMR_3 ^{*3}) |

- MSTPCRB

| Bit | Bit Name | Initial Value | R/W | Target Module |
|-----|----------------------|---------------|-----|---|
| 7 | MSTPB7 | 1 | R/W | Serial communication interface 0 (SCI_0) |
| 6 | MSTPB6 | 1 | R/W | Serial communication interface 1 (SCI_1) |
| 5 | MSTPB5 | 1 | R/W | Serial communication interface 2 (SCI_2) ^{*4} |
| 4 | MSTPB4 | 1 | R/W | I ² C bus interface 0 (IIC_0) (optional) ^{*3} |
| 3 | MSTPB3 | 1 | R/W | I ² C bus interface 1 (IIC_1) (optional) ^{*3} |
| 2 | MSTPB2 ^{*1} | 1 | R/W | |
| 1 | MSTPB1 ^{*1} | 1 | R/W | |
| 0 | MSTPB0 ^{*1} | 1 | R/W | |

- MSTPCRC

| Bit | Bit Name | Initial Value | R/W | Target Module |
|-----|----------------------|---------------|-----|--|
| 7 | MSTPC7 | 1 | R/W | Serial communication interface 3 (SCI_3) |
| 6 | MSTPC6 ^{*1} | 1 | R/W | |
| 5 | MSTPC5 | 1 | R/W | D/A converter ^{*4} |
| 4 | MSTPC4 | 1 | R/W | PC break controller (PBC) |
| 3 | MSTPC3 | 1 | R/W | IEBus controller (IEB) ^{*5} |
| 2 | MSTPC2 ^{*1} | 1 | R/W | |
| 1 | MSTPC1 ^{*1} | 1 | R/W | |
| 0 | MSTPC0 ^{*1} | 1 | R/W | |

Notes: 1. Bits MSTPA3, MSTPA2, MSTPB5, MSTPB2 to MSTPB0, MSTPC6, MSTPC2 to MSTPC0 are readable/writable. The initial value of them is 1. The write value should always be 1.

2. Supported only by the H8S/2239 Group.

3. Not available in the H8S/2237 Group and H8S/2227 Group.

4. Not available in the H8S/2227 Group.

5. Supported only by the H8S/2258 Group.

24.2 Medium-Speed Mode

In high-speed mode, when the SCK2 to SCK0 bits in SCKCR are set to 1, the operating mode changes to medium-speed mode as soon as the current bus cycle ends. In medium-speed mode, the CPU operates on the operating clock ($\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, or $\phi/32$) specified by the SCK2 to SCK0 bits. The bus masters other than the CPU (DMAC* and DTC) also operate in medium-speed mode.

On-chip peripheral modules other than the bus masters always operate on the high-speed clock (ϕ).

In medium-speed mode, a bus access is executed in the specified number of states with respect to the bus master operating clock. For example, if $\phi/4$ is selected as the operating clock, on-chip memory is accessed in 4 states, and internal I/O registers in 8 states.

Medium-speed mode is cleared by clearing all of bits SCK2 to SCK0 to 0. A transition is made to high-speed mode and medium-speed mode is cleared at the end of the current bus cycle.

If a SLEEP instruction is executed when the SSBY bit in SBYCR is cleared to 0, and LSON bit in LPWRCR is cleared to 0, a transition is made to sleep mode. When sleep mode is cleared by an interrupt, medium-speed mode is restored.

When the SLEEP instruction is executed with the SSBY bit = 1, the LSON bit in LPWRCR = 0, and the PSS bit in TCSR_1 (WDT_1) = 0, operation shifts to the software standby mode. When software standby mode is cleared by an external interrupt, medium-speed mode is restored.

When the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin is set low and medium-speed mode is cancelled, operation shifts to the reset state. The same applies in the case of a reset caused by overflow of the watchdog timer.

When the $\overline{\text{STBY}}$ pin is driven low, a transition is made to hardware standby mode.

Figure 24.2 shows the timing for transition to and clearance of medium-speed mode.

Note: * Supported only by the H8S/2239 Group.

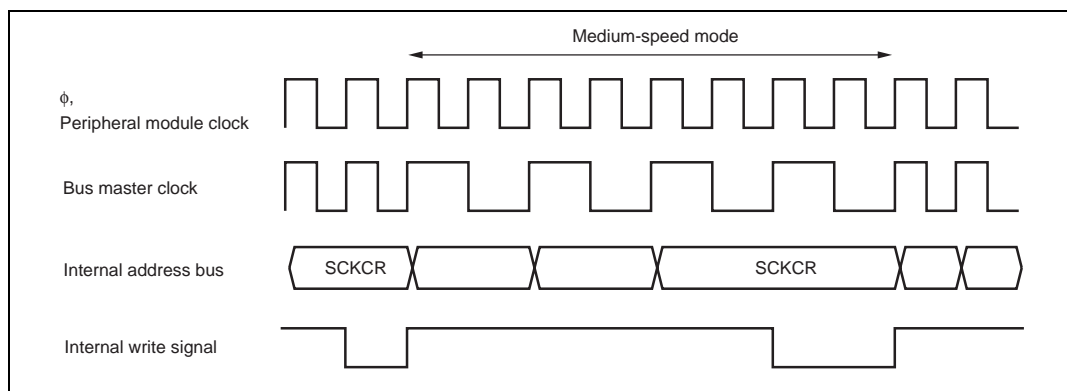


Figure 24.2 Medium-Speed Mode Transition and Clearance Timing

24.3 Sleep Mode

24.3.1 Transition to Sleep Mode

When the SLEEP instruction is executed while the SSBY bit in SBYCR = 0 and the LSON bit in LPWRCR = 0, the CPU enters the sleep mode. In sleep mode, CPU operation stops but the contents of the CPU's internal registers are retained. Other peripheral modules do not stop.

24.3.2 Exiting Sleep Mode

Sleep mode is exited by any interrupt, or signals at the $\overline{\text{RES}}$ pin, $\overline{\text{MRES}}$ pin, or $\overline{\text{STBY}}$ pin.

- **Exiting Sleep Mode by Interrupts**
When an interrupt occurs, sleep mode is exited and interrupt exception processing starts. Sleep mode is not exited if the interrupt is disabled, or interrupts other than NMI are masked by the CPU.
- **Exiting Sleep Mode by $\overline{\text{RES}}$ Pin or $\overline{\text{MRES}}$ Pin**
Setting the $\overline{\text{RES}}$ pin or $\overline{\text{MRES}}$ pin level low selects the reset state. After the stipulated reset input duration, driving the $\overline{\text{RES}}$ pin or $\overline{\text{MRES}}$ pin high starts the CPU performing reset exception processing.
- **Exiting Sleep Mode by $\overline{\text{STBY}}$ Pin**
When the $\overline{\text{STBY}}$ pin level is driven low, a transition is made to hardware standby mode.

24.4 Software Standby Mode

24.4.1 Transition to Software Standby Mode

A transition is made to software standby mode when the SLEEP instruction is executed while the SSBY bit in SBYCR = 1 and the LSON bit in LPWRCR = 0, and the PSS bit in TCSR_1 (WDT_1) = 0. In this mode, the CPU, on-chip peripheral modules, and system clock oscillator all stop. However, the contents of the CPU's internal registers, RAM data, and the states of on-chip peripheral modules other than SCI and the A/D converter, and the states of I/O ports are retained. In this mode the oscillator stops, and therefore power dissipation is significantly reduced.

24.4.2 Clearing Software Standby Mode

Software standby mode is cleared by an external interrupt (NMI pin, or pins $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$), or by means of the $\overline{\text{MRES}}$ pin or $\overline{\text{STBY}}$ pin.

- **Clearing with an Interrupt**
When an NMI, or IRQ7 to IRQ0 interrupt request signal is input, clock oscillation starts, and after the elapse of the time set in bits STS2 to STS0 in SYSCR, stable clocks are supplied to the entire this LSI chip, software standby mode is cleared, and interrupt exception handling is started.
When clearing software standby mode with an IRQ7 to IRQ0 interrupt, set the corresponding enable bit/pin function switching bit to 1 and ensure that no interrupt with a higher priority

than interrupts IRQ7 to IRQ0 is generated. Software standby mode cannot be cleared if the interrupt has been masked on the CPU side or has been designated as a DTC activation source.

- Clearing with the $\overline{\text{RES}}$ Pin or $\overline{\text{MRES}}$ Pin

When the $\overline{\text{RES}}$ pin or $\overline{\text{MRES}}$ pin is driven low, clock oscillation is started. At the same time as clock oscillation starts, clocks are supplied to the entire this LSI chip. Note that the $\overline{\text{RES}}$ pin or $\overline{\text{MRES}}$ pin must be held low until clock oscillation settles. When the $\overline{\text{RES}}$ pin or $\overline{\text{MRES}}$ pin goes high, the CPU begins reset exception handling.

- Clearing with the $\overline{\text{STBY}}$ Pin

When the $\overline{\text{STBY}}$ pin is driven low, a transition is made to hardware standby mode.

24.4.3 Oscillation Settling Time after Clearing Software Standby Mode

Bits STS2 to STS0 in SBYCR should be set as described below.

- Using a Crystal Oscillator

Set bits STS2 to STS0 so that the standby time is at least t_{OSC2} ms (the oscillation settling time). Table 24.3 shows the standby times for different operating frequencies and settings of bits STS2 to STS0.

- Using an External Clock

Any value can be set. Normally, minimum time is recommended.

Note: Do not set 16 states for standby time in the F-ZTAT version. 8192 states or more should be set.

Table 24.3 Oscillation Settling Time Settings

| STS2 | STS1 | STS0 | Standby Time | 20 MHz ^{*1} | 16 MHz ^{*1} | 13 MHz | 10 MHz | 8 MHz ^{*2} | 6 MHz ^{*2} | 4 MHz ^{*2} | 2 MHz ^{*2} | Unit |
|------|------|------|---------------|----------------------|----------------------|--------|--------|---------------------|---------------------|---------------------|---------------------|------|
| 0 | 0 | 0 | 8192 states | 0.41 | 0.51 | 0.6 | 0.8 | 1.0 | 1.4 | 2.0 | 4.1 | ms |
| | | 1 | 16384 states | 0.82 | 1.0 | 1.3 | 1.6 | 2.0 | 2.7 | 4.1 | 8.2 | |
| | 1 | 0 | 32768 states | 1.6 | 2.0 | 2.5 | 3.3 | 4.1 | 5.5 | 8.2 | 16.4 | |
| | | 1 | 65536 states | 3.3 | 4.1 | 5.0 | 6.6 | 8.2 | 10.9 | 16.4 | 32.8 | |
| 1 | 0 | 0 | 131072 states | 6.6 | 8.2 | 10.1 | 13.1 | 16.4 | 21.8 | 32.8 | 65.5 | |
| | | 1 | 262144 states | 13.1 | 16.4 | 20.2 | 26.2 | 32.8 | 43.7 | 65.5 | 131.1 | |
| | 1 | 0 | Reserved | — | — | — | — | — | — | — | — | |
| | | 1 | 16 states | 0.8 | 1.0 | 1.2 | 1.6 | 2.0 | 2.7 | 4.0 | 8.0 | |

 : Recommended time setting

- Notes: 1. Supported only by the H8S/2239 Group.
2. The H8S/2258 Group is out of operation.

24.4.4 Software Standby Mode Application Example

Figure 24.3 shows an example in which a transition is made to software standby mode at the falling edge on the NMI pin, and software standby mode is cleared at the rising edge on the NMI pin.

In this example, an NMI interrupt is accepted with the NMIEG bit in SYSCR cleared to 0 (falling edge specification), then the NMIEG bit is set to 1 (rising edge specification), the SSBY bit is set to 1, and a SLEEP instruction is executed, causing a transition to software standby mode.

Software standby mode is then cleared at the rising edge on the NMI pin.

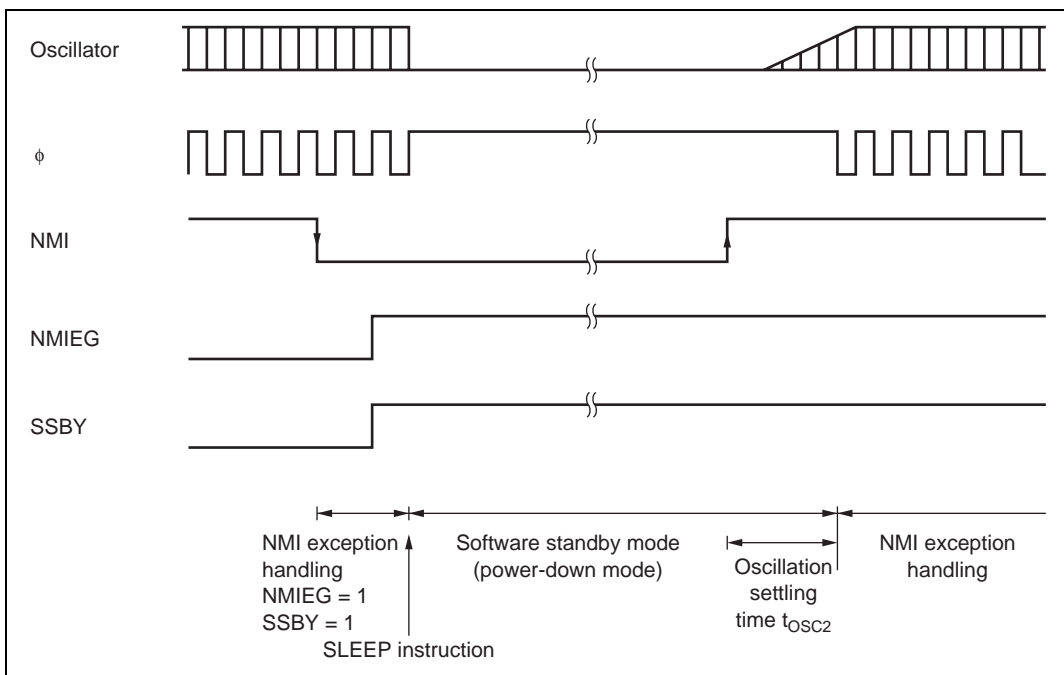


Figure 24.3 Software Standby Mode Application Example

24.5 Hardware Standby Mode

24.5.1 Transition to Hardware Standby Mode

When the $\overline{\text{STBY}}$ pin is driven low, a transition is made to hardware standby mode from any mode.

In hardware standby mode, all functions enter the reset state and stop operation, resulting in a significant reduction in power dissipation. As long as the prescribed voltage is supplied, on-chip RAM data is retained. I/O ports are set to the high-impedance state.

Do not change the state of the mode pins (MD2 to MD0) while this LSI is in hardware standby mode.

24.5.2 Clearing Hardware Standby Mode

Hardware standby mode is cleared by means of the $\overline{\text{STBY}}$ pin and the $\overline{\text{RES}}$ pin. When the $\overline{\text{STBY}}$ pin is driven high while the $\overline{\text{RES}}$ pin is low, the reset state is set and clock oscillation is started. Ensure that the $\overline{\text{RES}}$ pin is held low until the clock oscillator settles (at least t_{osc1} ms—the oscillation settling time—when using a crystal oscillator). When the $\overline{\text{RES}}$ pin is subsequently driven high, a transition is made to the program execution state via the reset exception handling state.

24.5.3 Hardware Standby Mode Timing

Figure 24.4 shows an example of hardware standby mode timing.

When the $\overline{\text{STBY}}$ pin is driven low after the $\overline{\text{RES}}$ pin has been driven low, a transition is made to hardware standby mode. Hardware standby mode is cleared by driving the $\overline{\text{STBY}}$ pin high, waiting for the oscillation settling time, then changing the $\overline{\text{RES}}$ pin from low to high.

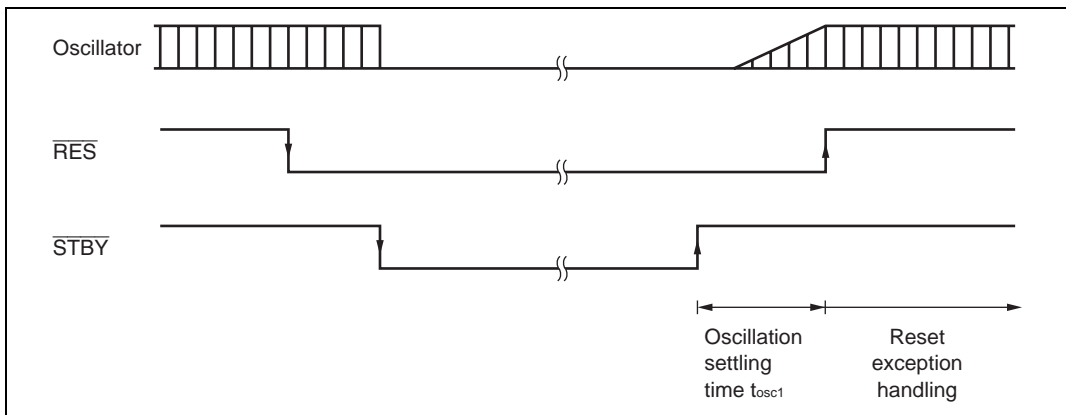


Figure 24.4 Hardware Standby Mode Timing

24.6 Module Stop Mode

Module stop mode can be set for individual on-chip peripheral modules.

When the corresponding MSTP bit in MSTPCR is set to 1, module operation stops at the end of the bus cycle and a transition is made to module stop mode. The CPU continues operating independently.

When the corresponding MSTP bit is cleared to 0, module stop mode is cleared and the module starts operating at the end of the bus cycle. In module stop mode, the internal states of modules other than SCI and the A/D converter are retained.

After reset clearance, all modules other than DMAC* and DTC are in module stop mode.

When an on-chip peripheral module is in module stop mode, read/write access to its registers is disabled.

Since the operations of the bus controller and I/O port are stopped when sleep mode is entered at the all-module stop state (MSTPCR = H'FFFFFFF), power consumption can further be reduced.

Note: * Supported only by the H8S/2239 Group.

24.7 Watch Mode

24.7.1 Transition to Watch Mode

CPU operation makes a transition to watch mode when the SLEEP instruction is executed in high-speed mode or subactive mode with SSBY in SBYCR = 1, DTON in LPWRCR = 0, and PSS in TCSR_1 (WDT_1) = 1.

In watch mode, the CPU is stopped and peripheral modules other than WDT_1 and system clock oscillator are also stopped. The contents of the CPU's internal registers, the data in internal RAM, and the statuses of the internal peripheral modules (excluding SCI and the A/D converter) and I/O ports are retained. To make a transition to watch mode, bits SCK2 to SCK0 in SCKCR must be set to 0.

24.7.2 Exiting Watch Mode

Watch mode is exited by any interrupt (WOVI_1 interrupt, NMI pin, or $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$), or signals at the $\overline{\text{RES}}$, $\overline{\text{MRES}}$, or $\overline{\text{STBY}}$ pin.

- Exiting Watch Mode by Interrupts

When an interrupt occurs, watch mode is exited and a transition is made to high-speed mode or medium-speed mode when the LPWRCR LSON bit = 0 or to subactive mode when the LSON bit = 1. When a transition is made to high-speed mode, a stable clock is supplied to all LSI circuits and interrupt exception processing starts after the time set in SBYCR STS2 to STS0 has elapsed. In the case of $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ interrupts, no transition is made from watch mode if the corresponding enable bit/pin function switching bit has been cleared to 0, and, in the case of interrupts from the internal peripheral modules, the interrupt enable register has been set to disable the reception of that interrupt, or is masked by the CPU.

See section 24.4.3, Oscillation Settling Time after Clearing Software Standby Mode, for how to set the oscillation settling time when making a transition from watch mode to high-speed mode.

- Exiting Watch Mode by $\overline{\text{RES}}$ Pin or $\overline{\text{MRES}}$ Pin

For exiting watch mode by the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin, see section 24.4.2, Clearing Software Standby Mode.

- Exiting Watch Mode by $\overline{\text{STBY}}$ Pin

When the $\overline{\text{STBY}}$ pin level is driven low, a transition is made to hardware standby mode.

24.8 Subsleep Mode

24.8.1 Transition to Subsleep Mode

When the SLEEP instruction is executed with the SSBY bit in SBYCR = 0, the LSON bit in LPWRCR = 1, and the PSS bit in TCSR_1 (WDT_1) = 1 in subactive mode, CPU operation shifts to subsleep mode.

In subsleep mode, the CPU is stopped. Peripheral modules other than TMR_0 to TMR3, WDT_0, and WDT_1 and system clock oscillator are also stopped. The contents of the CPU's internal registers, the data in internal RAM, and the statuses of the internal peripheral modules (excluding the SCI and the A/D converter) and I/O ports are retained.

24.8.2 Exiting Subsleep Mode

- Subsleep mode is exited by an interrupt (interrupts from internal peripheral modules, NMI pin, or $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$), or signals at the $\overline{\text{RES}}$ or $\overline{\text{STBY}}$ pin.

- Exiting Subsleep Mode by Interrupts

When an interrupt occurs, subsleep mode is exited and interrupt exception processing starts.

In the case of $\overline{\text{IRQ7}}$ to $\overline{\text{IRQ0}}$ interrupts, subsleep mode is not cancelled if the corresponding enable bit/pin function switching bit has been cleared to 0, and, in the case of interrupts from the internal peripheral modules, the interrupt enable register has been set to disable the reception of that interrupt, or is masked by the CPU.

- Exiting Subsleep Mode by $\overline{\text{RES}}$ Pin or $\overline{\text{MRES}}$ Pin

For exiting subsleep mode by the $\overline{\text{RES}}$ or $\overline{\text{MRES}}$ pin, see section 24.4.2, Clearing Software Standby Mode.

- Exiting Subsleep Mode by $\overline{\text{STBY}}$ Pin

When the $\overline{\text{STBY}}$ pin or $\overline{\text{MRES}}$ pin level is driven low, a transition is made to hardware standby mode.

24.9 Subactive Mode

24.9.1 Transition to Subactive Mode

When the SLEEP instruction is executed in high-speed mode with the SSBY bit in SBYCR = 1, the DTON bit in LPWRCR = 1, the LSON bit = 1, and the PSS bit in TCSR_1 (WDT_1) = 1, CPU operation shifts to subactive mode. When an interrupt occurs in watch mode, and if the LSON bit of LPWRCR is 1, a transition is made to subactive mode. And if an interrupt occurs in subsleep mode, a transition is made to subactive mode.

In subactive mode, the CPU operates at low speed on the subclock, and the program is executed step by step. Peripheral modules other than PBC, TMR_0 to TMR_3, WDT_0, and WDT_1, and system clock oscillator are also stopped.

When operating the CPU in subactive mode, the SCKCR SCK2 to SCK0 bits must be set to 0.

24.9.2 Exiting Subactive Mode

Subactive mode is exited by the SLEEP instruction or the $\overline{\text{RES}}$, $\overline{\text{MRES}}$ or $\overline{\text{STBY}}$ pin.

- Exiting Subactive Mode by SLEEP Instruction

When the SLEEP instruction is executed with the SSBY bit in SBYCR = 1, the DTON bit in LPWRCR = 0, and the PSS bit in TCSR_1 (WDT_1) = 1, the CPU exits subactive mode and a transition is made to watch mode. When the SLEEP instruction is executed with the SSBY bit in SBYCR = 0, the LSON bit in LPWRCR = 1, and the PSS bit in TCSR_1 (WDT_1) = 1, a transition is made to subsleep mode. Finally, when the SLEEP instruction is executed with the SSBY bit in SBYCR = 1, the DTON bit in LPWRCR = 1, the LSON bit = 0, and the PSS bit in TCSR_1 (WDT_1) = 1, a direct transition is made to high-speed mode (SCK2 to SCK0 all 0).

- Exiting Subactive Mode by $\overline{\text{RES}}$ Pin or $\overline{\text{MRES}}$ Pin

For exiting subactive mode by the RES or MRES pin, see section 24.4.2, Clearing Software Standby Mode.

- Exiting Subactive Mode by $\overline{\text{STBY}}$ Pin

When the $\overline{\text{STBY}}$ pin level is driven low, a transition is made to hardware standby mode.

24.10 Direct Transitions

There are three modes, high-speed, medium-speed, and subactive, in which the CPU executes programs. When a direct transition is made, there is no interruption of program execution when shifting between high-speed and subactive modes. Direct transitions are enabled by setting the LPWRCR DTON bit to 1, then executing the SLEEP instruction. After a transition, direct transition interrupt exception processing starts.

24.10.1 Direct Transitions from High-Speed Mode to Subactive Mode

Execute the SLEEP instruction in high-speed mode when the SSBY bit in SBYCR = 1, the LSON bit in LPWRCR = 1, and the DTON bit = 1, and the PSS bit in TSCR_1 (WDT_1) = 1 to make a transition to subactive mode.

24.10.2 Direct Transitions from Subactive Mode to High-Speed Mode

Execute the SLEEP instruction in subactive mode when the SSBY bit in SBYCR = 1, the LSON bit in LPWRCR = 0, and the DTON bit = 1, and the PSS bit in TSCR_1 (WDT_1) = 1 to make a direct transition to high-speed mode after the time set in STS2 to STS0 bits in SBYCR has elapsed.

24.11 ϕ Clock Output Enable

The PSTOP bit in SCKCR and the DDR of the corresponding port control the ϕ clock output. When the PSTOP bit is set to 1, ϕ clock stops at the end of the bus cycle and the ϕ clock output is fixed high. When the PSTOP bit is cleared to 0, the ϕ clock output is enabled. When the DDR of the corresponding port is cleared to 0, the ϕ clock output is disabled and it functions as an input port. Table 24.4 lists the ϕ pin states in respective process.

Table 24.4 ϕ Pin States in Respective Processes

| DDR | 0 | 1 | 1 |
|--|----------------|----------------|----------------|
| PSTOP | — | 0 | 1 |
| Hardware standby mode | High impedance | High impedance | High impedance |
| Software standby mode, watch mode, direct transition | High impedance | Fixed to high | Fixed high |
| Sleep mode, subsleep mode | High impedance | ϕ output | Fixed high |
| High-speed mode, medium-speed mode, subactive mode | High impedance | ϕ output | Fixed high |

24.12 Usage Notes

24.12.1 I/O Port Status

In software standby mode and watch mode, I/O port states are retained. Therefore, there is no reduction in current dissipation for the output current when a high-level signal is output.

24.12.2 Current Dissipation during Oscillation Settling Wait Period

Current dissipation increases during the oscillation settling wait period.

24.12.3 DTC and DMAC* Module Stop

Depending on the operating status of the DTC and DMAC*, the MSTPA6 bit and MSTPA7 bit may not be set to 1. Setting of the DTC and DMAC* module stop mode should be carried out only when the respective module is not activated.

For details, refer to section 8, DMA Controller (DMAC) and section 9, Data Transfer Controller (DTC).

Note: * Supported only by the H8S/2239 Group.

24.12.4 On-Chip Peripheral Module Interrupt

- Module Stop Mode

Relevant interrupt operations cannot be performed in module stop mode. Consequently, if module stop mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the DMAC*¹ or DTC activation source. Interrupts should therefore be disabled before entering module stop mode.

- Subactive Mode/Watch Mode

On-chip peripheral modules (DMAC*¹, DTC, TPU, IIC*²) that stop operation in subactive mode cannot clear interrupts in subactive mode. Therefore, if subactive mode is entered when an interrupt is requested, CPU interrupt factors cannot be cleared.

Interrupts should therefore be cleared before executing the SLEEP instruction and entering subactive or watch mode.

Notes: 1. Supported only by the H8S/2239 Group.

2. Not available in the H8S/2237 Group and H8S/2227 Group.

24.12.5 Writing to MSTPCR

MSTPCR should only be written to by the CPU.

24.12.6 Entering Subactive/Watch Mode and DMAC* and DTC Module Stop

To enter subactive or watch mode, set DMAC* and DTC to module stop (write 1 to the MSTPA6 bit and MSTPA7 bit) and reading the MSTPA6 bit and MSTPA7 bit as 1 before transiting mode. After transiting from subactive mode to active mode, clear module stop.

When DMAC* or DTC activation factor occurs in subactive mode, DMAC* or DTC is activated when module stop is cleared after active mode is entered.

Note: * Supported only by the H8S/2239 Group.

Section 25 Power Supply Circuit

25.1 Overview

The H8S/2258 Group, H8S/2238B, and H8S/2236B incorporates an internal power supply step-down circuit. Use of this circuit enables the internal power supply to be fixed at a constant level of approximately 3.0 V, independently of the voltage of the power supply connected to the external V_{cc} pin. As a result, the current consumed when an external power supply is used at 3.0 V or above can be held down to virtually the same low level as when used at approximately 3.0 V. If the external power supply is 3.0 V or below, the internal voltage will be practically the same as the external voltage.

The H8S/2239 Group, H8S/2238R, H8S/2236R, H8S/2237 Group, and H8S/2227 Group do not have an on-chip internal power supply voltage step-down circuit.

An external power supply should be connected to the V_{cc} and CV_{cc} pins.

25.2 Power Supply Connection for H8S/2258 Group, H8S/2238B, and H8S/2236B (On-Chip Internal Power Supply Step-Down Circuit)

Connect the external power supply to the V_{cc} pin, and connect a capacitance of approximately 0.1 μF between CV_{cc} and V_{ss} , as shown in figure 25.1. The internal step-down circuit is made effective simply by adding this external circuit. Permanent damage on the chip may result if the absolute maximum rating of CV_{cc} 4.3 V is exceeded. Must not connect the external power supply to the CV_{cc} pin.

- Notes:
1. In the external circuit interface, the external power supply voltage connected to V_{cc} and the GND potential connected to V_{ss} are the reference levels. For example, for port input/output levels, the V_{cc} level is the reference for the high level, and the V_{ss} level is that for the low level.
 2. The A/D converter and D/A converter analog power supply are not affected by internal step-down processing.

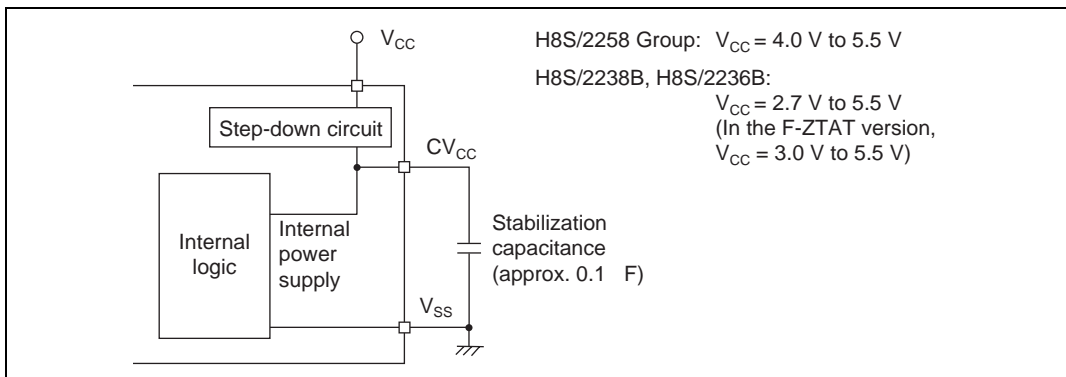


Figure 25.1 Power Supply Connection for H8S/2258 Group, H8S/2238B, and H8S/2236B (On-Chip Internal Power Supply Step-Down Circuit)

25.3 Power Supply Connection for H8S/2239 Group, H8S/2238R, H8S/2236R, H8S/2237 Group, and H8S/2227 Group (No Internal Power Supply Step-Down Circuit)

The H8S/2239 Group, H8S/2238R, H8S/2236R, H8S/2237 Group, and H8S/2227 Group do not have an on-chip internal power supply voltage step-down circuit. Connect the external power supply to the V_{CC} pin and CV_{CC} pin, as shown in figure 25.2. The external power supply is then input directly to the internal power supply.

Note: The permissible range for the power supply voltage is 2.2 V to 3.6 V (in the F-ZTAT version, 2.7 V to 3.6 V). Operation cannot be guaranteed if a voltage outside this range (less than 2.2 V or more than 3.6 V) is input.

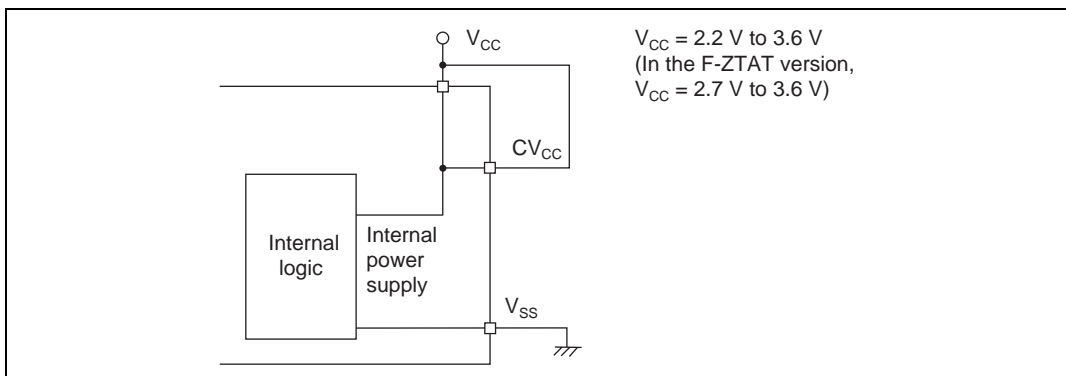


Figure 25.2 Power Supply Connection for H8S/2239 Group, H8S/2238R, H8S/2236R, H8S/2237 Group, and H8S/2227 Group (No Internal Power Supply Step-Down Circuit)

25.4 Note on Bypass Capacitor

A laminated ceramic capacitor of 0.01 μF to 0.1 μF should be inserted as a bypass capacitor in each pair of V_{ss} and V_{cc} .

The bypass capacitor should be placed as close as possible to the power supply pin of this LSI.

The capacitance value and frequency characteristics should be used according to the operating frequency of this LSI.

Section 26 List of Registers

This section gives information on the on-chip I/O registers and is configured as described below.

1. Register Addresses (In Address Order)
 - Descriptions by functional module, in ascending order of addresses
 - Descriptions by functional module
 - The number of access states are given
2. Register Bits
 - Bit configurations of the registers are described in the same order as the Register Addresses (In Address Order)
 - Reserved bits are indicated by “—” in the bit name
 - A blank in the bit name indicates that the corresponding whole register is allocated to the counter or data
3. Register States in Each Operating Mode
 - Register states are described in the same order as the Register Addresses (In Address Order)
 - 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

26.1 Register Addresses (In Address Order)

The data bus width indicates the number of bits by which the register is accessed.

The number of access states indicates the number of states based on the specified reference clock.

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|---|-------------------|---------|-----------------------|--------|---------------------|-----------------|
| DTC mode register A | MRA | 8 | H'EBC0 to H'EFBF | DTC | 16/32 ^{*2} | 2 |
| DTC mode register B | MRB | 8 | | DTC | 16/32 ^{*2} | 2 |
| DTC source address register | SAR | 24 | | DTC | 16/32 ^{*2} | 2 |
| DTC destination address register | DAR | 24 | | DTC | 16/32 ^{*2} | 2 |
| DTC transfer count register A | CRA | 16 | | DTC | 16/32 ^{*2} | 2 |
| DTC transfer count register B | CRB | 16 | | DTC | 16/32 ^{*2} | 2 |
| IEBus control register | IECTR | 8 | H'F800 to H'F816 | IEB | 8 | 2 |
| IEBus command register | IECMR | 8 | | IEB | 8 | 2 |
| IEBus master control register | IEMCR | 8 | | IEB | 8 | 2 |
| IEBus master unit address register 1 | IEAR1 | 8 | | IEB | 8 | 2 |
| IEBus master unit address register 2 | IEAR2 | 8 | | IEB | 8 | 2 |
| IEBus slave address setting register 1 | IESA1 | 8 | | IEB | 8 | 2 |
| IEBus slave address setting register 2 | IESA2 | 8 | | IEB | 8 | 2 |
| IEBus transmit message length register | IETBFL | 8 | | IEB | 8 | 2 |
| IEBus transmit buffer register | IETBR | 8 | | IEB | 8 | 2 |
| IEBus reception master address register 1 | IEMA1 | 8 | | IEB | 8 | 2 |
| IEBus reception master address register 2 | IEMA2 | 8 | | IEB | 8 | 2 |
| IEBus receive control field register | IERCTL | 8 | | IEB | 8 | 2 |
| IEBus receive message length register | IERBFL | 8 | | IEB | 8 | 2 |
| IEBus receive buffer register | IERBR | 8 | | IEB | 8 | 2 |
| IEBus lock address register 1 | IELA1 | 8 | | IEB | 8 | 2 |
| IEBus lock address register 2 | IELA2 | 8 | | IEB | 8 | 2 |
| IEBus general flag register | IEFLG | 8 | | IEB | 8 | 2 |
| IEBus transmit/runaway status register | IETSR | 8 | | IEB | 8 | 2 |
| IEBus transmit/runaway interrupt enable register | IEIET | 8 | | IEB | 8 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|---|-------------------|---------|-----------------------|------------------|-------------------|-----------------|
| IEBus transmit error flag register | IETEF | 8 | H'F800 to H'F816 | IEB | 8 | 2 |
| IEBus receive status register | IERSR | 8 | | IEB | 8 | 2 |
| IEBus receive interrupt enable register | IEIER | 8 | | IEB | 8 | 2 |
| IEBus receive error flag register | IEREF | 8 | | IEB | 8 | 2 |
| D/A data register_0 | DADR_0 | 8 | H'FDAC | D/A converter | 8 | 2 |
| D/A data register_1 | DADR_1 | 8 | H'FDAD | D/A converter | 8 | 2 |
| D/A control register | DACR | 8 | H'FDAE | D/A converter | 8 | 2 |
| Serial control register X | SCRX | 8 | H'FDB4 | IIC, FLASH | 8 | 2 |
| DDC switch register | DDCSWR | 8 | H'FDB5 | IIC | 8 | 2 |
| Timer control register_2 | TCR_2 | 8 | H'FDC0 | TMR_2 | 8 | 2 |
| Timer control register_3 | TCR_3 | 8 | H'FDC1 | TMR_3 | 8 | 2 |
| Timer control/status register_2 | TCSR_2 | 8 | H'FDC2 | TMR_2 | 8 | 2 |
| Timer control/status register_3 | TCSR_3 | 8 | H'FDC3 | TMR_3 | 8 | 2 |
| Time constant register A_2 | TCORA_2 | 8 | H'FDC4 | TMR_2 | 8/16 | 2 |
| Time constant register A_3 | TCORA_3 | 8 | H'FDC5 | TMR_3 | 8/16 | 2 |
| Time constant register B_2 | TCORB_2 | 8 | H'FDC6 | TMR_2 | 8/16 | 2 |
| Time constant register B_3 | TCORB_3 | 8 | H'FDC7 | TMR_3 | 8/16 | 2 |
| Timer counter_2 | TCNT_2 | 8 | H'FDC8 | TMR_2 | 8/16 | 2 |
| Timer counter_3 | TCNT_3 | 8 | H'FDC9 | TMR_3 | 8/16 | 2 |
| Serial mode register_3 | SMR_3 | 8 | H'FDD0 | SCI_3 | 8 | 2 |
| Bit rate register_3 | BRR_3 | 8 | H'FDD1 | SCI_3 | 8 | 2 |
| Serial control register_3 | SCR_3 | 8 | H'FDD2 | SCI_3 | 8 | 2 |
| Transmit data register_3 | TDR_3 | 8 | H'FDD3 | SCI_3 | 8 | 2 |
| Serial status register_3 | SSR_3 | 8 | H'FDD4 | SCI_3 | 8 | 2 |
| Receive data register_3 | RDR_3 | 8 | H'FDD5 | SCI_3 | 8 | 2 |
| Smart card mode register_3 | SCMR_3 | 8 | H'FDD6 | SCI_3 | 8 | 2 |
| Standby control register | SBYCR | 8 | H'FDE4 | SYSTEM | 8 | 2 |
| System control register | SYSCR | 8 | H'FDE5 | SYSTEM | 8 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|----------------------------------|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| System clock control register | SCKCR | 8 | H'FDE6 | SYSTEM | 8 | 2 |
| Mode control register | MDCR | 8 | H'FDE7 | SYSTEM | 8 | 2 |
| Module stop control register A | MSTPCRA | 8 | H'FDE8 | SYSTEM | 8 | 2 |
| Module stop control register B | MSTPCRB | 8 | H'FDE9 | SYSTEM | 8 | 2 |
| Module stop control register C | MSTPCRC | 8 | H'FDEA | SYSTEM | 8 | 2 |
| Pin function control register | PFCR | 8 | H'FDEB | BSC | 8 | 2 |
| Low power control register | LPWRCR | 8 | H'FDEC | SYSTEM | 8 | 2 |
| Serial expansion mode register 0 | SEMR_0 | 8 | H'FDF8 | SCI_0 | 8 | 2 |
| Break address register A | BARA | 32 | H'FE00 | PBC | 8/16 | 2 |
| Break address register B | BARB | 32 | H'FE04 | PBC | 8/16 | 2 |
| Break control register A | BCRA | 8 | H'FE08 | PBC | 8/16 | 2 |
| Break control register B | BCRB | 8 | H'FE09 | PBC | 8/16 | 2 |
| IRQ sense control register H | ISCRH | 8 | H'FE12 | INT | 8 | 2 |
| IRQ sense control register L | ISCR L | 8 | H'FE13 | INT | 8 | 2 |
| IRQ enable register | IER | 8 | H'FE14 | INT | 8 | 2 |
| IRQ status register | ISR | 8 | H'FE15 | INT | 8 | 2 |
| DTC enable register A | DTCERA | 8 | H'FE16 | DTC | 8 | 2 |
| DTC enable register B | DTCERB | 8 | H'FE17 | DTC | 8 | 2 |
| DTC enable register C | DTCERC | 8 | H'FE18 | DTC | 8 | 2 |
| DTC enable register D | DTCERD | 8 | H'FE19 | DTC | 8 | 2 |
| DTC enable register E | DTCERE | 8 | H'FE1A | DTC | 8 | 2 |
| DTC enable register F | DTCERF | 8 | H'FE1B | DTC | 8 | 2 |
| DTC enable register I | DTCERI | 8 | H'FE1E | DTC | 8 | 2 |
| DTC vector register | DTVECR | 8 | H'FE1F | DTC | 8 | 2 |
| Port 1 data direction register | P1DDR | 8 | H'FE30 | PORT | 8 | 2 |
| Port 3 data direction register | P3DDR | 8 | H'FE32 | PORT | 8 | 2 |
| Port 7 data direction register | P7DDR | 8 | H'FE36 | PORT | 8 | 2 |
| Port A data direction register | PADDR | 8 | H'FE39 | PORT | 8 | 2 |
| Port B data direction register | PBDDR | 8 | H'FE3A | PORT | 8 | 2 |
| Port C data direction register | PCDDR | 8 | H'FE3B | PORT | 8 | 2 |
| Port D data direction register | PDDDR | 8 | H'FE3C | PORT | 8 | 2 |
| Port E data direction register | PEDDR | 8 | H'FE3D | PORT | 8 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|-------------------------------------|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| Port F data direction register | PFDDR | 8 | H'FE3E | PORT | 8 | 2 |
| Port G data direction register | PGDDR | 8 | H'FE3F | PORT | 8 | 2 |
| Port A pull-up MOS control register | PAPCR | 8 | H'FE40 | PORT | 8 | 2 |
| Port B pull-up MOS control register | PBPCR | 8 | H'FE41 | PORT | 8 | 2 |
| Port C pull-up MOS control register | PCPCR | 8 | H'FE42 | PORT | 8 | 2 |
| Port D pull-up MOS control register | PDPCR | 8 | H'FE43 | PORT | 8 | 2 |
| Port E pull-up MOS control register | PEPCR | 8 | H'FE44 | PORT | 8 | 2 |
| Port 3 open drain control register | P3ODR | 8 | H'FE46 | PORT | 8 | 2 |
| Port A open drain control register | PAODR | 8 | H'FE47 | PORT | 8 | 2 |
| Timer control register_3 | TCR_3 | 8 | H'FE80 | TPU_3 | 8 | 2 |
| Timer mode register_3 | TMDR_3 | 8 | H'FE81 | TPU_3 | 8 | 2 |
| Timer I/O control register H_3 | TIORH_3 | 8 | H'FE82 | TPU_3 | 8 | 2 |
| Timer I/O control register L_3 | TIORL_3 | 8 | H'FE83 | TPU_3 | 8 | 2 |
| Timer interrupt enable register_3 | TIER_3 | 8 | H'FE84 | TPU_3 | 8 | 2 |
| Timer status register_3 | TSR_3 | 8 | H'FE85 | TPU_3 | 8 | 2 |
| Timer counter_3 | TCNT_3 | 16 | H'FE86 | TPU_3 | 16 | 2 |
| Timer general register A_3 | TGRA_3 | 16 | H'FE88 | TPU_3 | 16 | 2 |
| Timer general register B_3 | TGRB_3 | 16 | H'FE8A | TPU_3 | 16 | 2 |
| Timer general register C_3 | TGRC_3 | 16 | H'FE8C | TPU_3 | 16 | 2 |
| Timer general register D_3 | TGRD_3 | 16 | H'FE8E | TPU_3 | 16 | 2 |
| Timer control register_4 | TCR_4 | 8 | H'FE90 | TPU_4 | 8 | 2 |
| Timer mode register_4 | TMDR_4 | 8 | H'FE91 | TPU_4 | 8 | 2 |
| Timer I/O control register_4 | TIOR_4 | 8 | H'FE92 | TPU_4 | 8 | 2 |
| Timer interrupt enable register_4 | TIER_4 | 8 | H'FE94 | TPU_4 | 8 | 2 |
| Timer status register_4 | TSR_4 | 8 | H'FE95 | TPU_4 | 8 | 2 |
| Timer counter_4 | TCNT_4 | 16 | H'FE96 | TPU_4 | 16 | 2 |
| Timer general register A_4 | TGRA_4 | 16 | H'FE98 | TPU_4 | 16 | 2 |
| Timer general register B_4 | TGRB_4 | 16 | H'FE9A | TPU_4 | 16 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|-----------------------------------|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| Timer control register_5 | TCR_5 | 8 | H'FEA0 | TPU_5 | 8 | 2 |
| Timer mode register_5 | TMDR_5 | 8 | H'FEA1 | TPU_5 | 8 | 2 |
| Timer I/O control register_5 | TIOR_5 | 8 | H'FEA2 | TPU_5 | 8 | 2 |
| Timer interrupt enable register_5 | TIER_5 | 8 | H'FEA4 | TPU_5 | 8 | 2 |
| Timer status register_5 | TSR_5 | 8 | H'FEA5 | TPU_5 | 8 | 2 |
| Timer counter_5 | TCNT_5 | 16 | H'FEA6 | TPU_5 | 16 | 2 |
| Timer general register A_5 | TGRA_5 | 16 | H'FEA8 | TPU_5 | 16 | 2 |
| Timer general register B_5 | TGRB_5 | 16 | H'FEAA | TPU_5 | 16 | 2 |
| Timer start register | TSTR | 8 | H'FEB0 | TPU | 8 | 2 |
| Timer synchro register | TSYR | 8 | H'FEB1 | TPU | 8 | 2 |
| Interrupt priority register A | IPRA | 8 | H'FEC0 | INT | 8 | 2 |
| Interrupt priority register B | IPRB | 8 | H'FEC1 | INT | 8 | 2 |
| Interrupt priority register C | IPRC | 8 | H'FEC2 | INT | 8 | 2 |
| Interrupt priority register D | IPRD | 8 | H'FEC3 | INT | 8 | 2 |
| Interrupt priority register E | IPRE | 8 | H'FEC4 | INT | 8 | 2 |
| Interrupt priority register F | IPRF | 8 | H'FEC5 | INT | 8 | 2 |
| Interrupt priority register G | IPRG | 8 | H'FEC6 | INT | 8 | 2 |
| Interrupt priority register H | IPRH | 8 | H'FEC7 | INT | 8 | 2 |
| Interrupt priority register I | IPRI | 8 | H'FEC8 | INT | 8 | 2 |
| Interrupt priority register J | IPRJ | 8 | H'FEC9 | INT | 8 | 2 |
| Interrupt priority register K | IPRK | 8 | H'FECA | INT | 8 | 2 |
| Interrupt priority register L | IPRL | 8 | H'FECB | INT | 8 | 2 |
| Interrupt priority register O | IPRO | 8 | H'FECE | INT | 8 | 2 |
| Bus width control register | ABWCR | 8 | H'FED0 | BSC | 8 | 2 |
| Access state control register | ASTCR | 8 | H'FED1 | BSC | 8 | 2 |
| Wait control register H | WCRH | 8 | H'FED2 | BSC | 8 | 2 |
| Wait control register L | WCRL | 8 | H'FED3 | BSC | 8 | 2 |
| Bus control register H | BCRH | 8 | H'FED4 | BSC | 8 | 2 |
| Bus control register L | BCRL | 8 | H'FED5 | BSC | 8 | 2 |
| RAM emulation register | RAMER | 8 | H'FEDB | FLASH | 8 | 2 |
| Memory address register_0AH | MAR_0AH | 16 | H'FEE0 | DMAC | 16 | 2 |
| Memory address register_0AL | MAR_0AL | 16 | H'FEE2 | DMAC | 16 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|------------------------------------|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| I/O address register_0A | IOAR_0A | 16 | H'FEE4 | DMAC | 16 | 2 |
| Execute transfer count register_0A | ETCR_0A | 16 | H'FEE6 | DMAC | 16 | 2 |
| Memory address register_0BH | MAR_0BH | 16 | H'FEE8 | DMAC | 16 | 2 |
| Memory address register_0BL | MAR_0BL | 16 | H'FEEA | DMAC | 16 | 2 |
| I/O address register_0B | IOAR_0B | 16 | H'FEEC | DMAC | 16 | 2 |
| Execute transfer count register_0B | ETCR_0B | 16 | H'FEEE | DMAC | 16 | 2 |
| Memory address register_1AH | MAR_1AH | 16 | H'FEF0 | DMAC | 16 | 2 |
| Memory address register_1AL | MAR_1AL | 16 | H'FEF2 | DMAC | 16 | 2 |
| I/O address register_1A | IOAR_1A | 16 | H'FEF4 | DMAC | 16 | 2 |
| Execute transfer count register_1A | ETCR1A | 16 | H'FEF6 | DMAC | 16 | 2 |
| Memory address register_1BH | MAR_1BH | 16 | H'FEF8 | DMAC | 16 | 2 |
| Memory address register_1BL | MAR_1BL | 16 | H'FEFA | DMAC | 16 | 2 |
| I/O address register_1B | IOAR_1B | 16 | H'FEFC | DMAC | 16 | 2 |
| Execute transfer count register_1B | ETCR_1B | 16 | H'FEFE | DMAC | 16 | 2 |
| Port 1 data register | P1DR | 8 | H'FF00 | PORT | 8 | 2 |
| Port 3 data register | P3DR | 8 | H'FF02 | PORT | 8 | 2 |
| Port 7 data register | P7DR | 8 | H'FF06 | PORT | 8 | 2 |
| Port A data register | PADR | 8 | H'FF09 | PORT | 8 | 2 |
| Port B data register | PBDR | 8 | H'FF0A | PORT | 8 | 2 |
| Port C data register | PCDR | 8 | H'FF0B | PORT | 8 | 2 |
| Port D data register | PDDR | 8 | H'FF0C | PORT | 8 | 2 |
| Port E data register | PEDR | 8 | H'FF0D | PORT | 8 | 2 |
| Port F data register | PFDR | 8 | H'FF0E | PORT | 8 | 2 |
| Port G data register | PGDR | 8 | H'FF0F | PORT | 8 | 2 |
| Timer control register_0 | TCR_0 | 8 | H'FF10 | TPU_0 | 8 | 2 |
| Timer mode register_0 | TMDR_0 | 8 | H'FF11 | TPU_0 | 8 | 2 |
| Timer I/O control register H_0 | TIORH_0 | 8 | H'FF12 | TPU_0 | 8 | 2 |
| Timer I/O control register L_0 | TIORL_0 | 8 | H'FF13 | TPU_0 | 8 | 2 |
| Timer interrupt enable register_0 | TIER_0 | 8 | H'FF14 | TPU_0 | 8 | 2 |
| Timer status register_0 | TSR_0 | 8 | H'FF15 | TPU_0 | 8 | 2 |
| Timer counter_0 | TCNT_0 | 16 | H'FF16 | TPU_0 | 16 | 2 |
| Timer general register A_0 | TGRA_0 | 16 | H'FF18 | TPU_0 | 16 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|-----------------------------------|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| Timer general register B_0 | TGRB_0 | 16 | H'FF1A | TPU_0 | 16 | 2 |
| Timer general register C_0 | TGRC_0 | 16 | H'FF1C | TPU_0 | 16 | 2 |
| Timer general register D_0 | TGRD_0 | 16 | H'FF1E | TPU_0 | 16 | 2 |
| Timer control register_1 | TCR_1 | 8 | H'FF20 | TPU_1 | 8 | 2 |
| Timer mode register_1 | TMDR_1 | 8 | H'FF21 | TPU_1 | 8 | 2 |
| Timer I/O control register_1 | TIOR_1 | 8 | H'FF22 | TPU_1 | 8 | 2 |
| Timer interrupt enable register_1 | TIER_1 | 8 | H'FF24 | TPU_1 | 8 | 2 |
| Timer status register_1 | TSR_1 | 8 | H'FF25 | TPU_1 | 8 | 2 |
| Timer counter_1 | TCNT_1 | 16 | H'FF26 | TPU_1 | 16 | 2 |
| Timer general register A_1 | TGRA_1 | 16 | H'FF28 | TPU_1 | 16 | 2 |
| Timer general register B_1 | TGRB_1 | 16 | H'FF2A | TPU_1 | 16 | 2 |
| Timer control register_2 | TCR_2 | 8 | H'FF30 | TPU_2 | 8 | 2 |
| Timer mode register_2 | TMDR_2 | 8 | H'FF31 | TPU_2 | 8 | 2 |
| Timer I/O control register_2 | TIOR_2 | 8 | H'FF32 | TPU_2 | 8 | 2 |
| Timer interrupt enable register_2 | TIER_2 | 8 | H'FF34 | TPU_2 | 8 | 2 |
| Timer status register_2 | TSR_2 | 8 | H'FF35 | TPU_2 | 8 | 2 |
| Timer counter_2 | TCNT_2 | 16 | H'FF36 | TPU_2 | 16 | 2 |
| Timer general register A_2 | TGRA_2 | 16 | H'FF38 | TPU_2 | 16 | 2 |
| Timer general register B_2 | TGRB_2 | 16 | H'FF3A | TPU_2 | 16 | 2 |
| DMA write enable register | DMAWER | 8 | H'FF60 | DMAC | 8 | 2 |
| DMA terminal control register | DMATCR | 8 | H'FF61 | DMAC | 8 | 2 |
| DMA control register_0A | DMACR_0 A | 8 | H'FF62 | DMAC | 16 | 2 |
| DMA control register_0B | DMACR_0 B | 8 | H'FF63 | DMAC | 16 | 2 |
| DMA control register_1A | DMACR_1 A | 8 | H'FF64 | DMAC | 16 | 2 |
| DMA control register_1B | DMACR_1 B | 8 | H'FF65 | DMAC | 16 | 2 |
| DMA band control register H | DMABCRH | 8 | H'FF66 | DMAC | 16 | 2 |
| DMA band control register L | DMABCRL | 8 | H'FF67 | DMAC | 16 | 2 |
| Timer control register_0 | TCR_0 | 8 | H'FF68 | TMR_0 | 8 | 2 |
| Timer control register_1 | TCR_1 | 8 | H'FF69 | TMR_1 | 8 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|---|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| Timer control/status register_0 | TCSR_0 | 8 | H'FF6A | TMR_0 | 8 | 2 |
| Timer control/status register_1 | TCSR_1 | 8 | H'FF6B | TMR_1 | 8 | 2 |
| Time constant register A_0 | TCORA_0 | 8 | H'FF6C | TMR_0 | 8/16 | 2 |
| Time constant register A_1 | TCORA_1 | 8 | H'FF6D | TMR_1 | 8/16 | 2 |
| Time constant register B_0 | TCORB_0 | 8 | H'FF6E | TMR_0 | 8/16 | 2 |
| Time constant register B_1 | TCORB_1 | 8 | H'FF6F | TMR_1 | 8/16 | 2 |
| Timer counter_0 | TCNT_0 | 8 | H'FF70 | TMR_0 | 8/16 | 2 |
| Timer counter_1 | TCNT_1 | 8 | H'FF71 | TMR_1 | 8/16 | 2 |
| Timer control/status register_0 | TCSR_0 | 8 | H'FF74 | WDT_0 | 16 | 2 |
| Timer counter_0 | TCNT_0 | 8 | H'FF74 (write) | WDT_0 | 16 | 2 |
| Timer counter_0 | TCNT_0 | 8 | H'FF75 (read) | WDT_0 | 16 | 2 |
| Reset control/status register | RSTCSR | 8 | H'FF76 (write) | WDT_0 | 16 | 2 |
| Reset control/status register | RSTCSR | 8 | H'FF77 (read) | WDT_0 | 16 | 2 |
| Serial mode register_0 | SMR_0 | 8 | H'FF78 ^{*3} | SCI_0 | 8 | 2 |
| I ² C bus control register_0 | ICCR_0 | 8 | H'FF78 ^{*3} | IIC_0 | 8 | 2 |
| Bit rate register_0 | BRR_0 | 8 | H'FF79 ^{*3} | SCI_0 | 8 | 2 |
| I ² C bus status register_0 | ICSR_0 | 8 | H'FF79 ^{*3} | IIC_0 | 8 | 2 |
| Serial control register_0 | SCR_0 | 8 | H'FF7A | SCI_0 | 8 | 2 |
| Transmit data register_0 | TDR_0 | 8 | H'FF7B | SCI_0 | 8 | 2 |
| Serial status register_0 | SSR_0 | 8 | H'FF7C | SCI_0 | 8 | 2 |
| Receive data register_0 | RDR_0 | 8 | H'FF7D | SCI_0 | 8 | 2 |
| Smart card mode register_0 | SCMR_0 | 8 | H'FF7E ^{*3} | SCI_0 | 8 | 2 |
| I ² C bus data register_0 | ICDR_0 | 8 | H'FF7E ^{*3} | IIC_0 | 8 | 2 |
| Second slave address register_0 | SARX_0 | 8 | H'FF7E ^{*3} | IIC_0 | 8 | 2 |
| I ² C bus mode register_0 | ICMR_0 | 8 | H'FF7F | IIC_0 | 8 | 2 |
| Slave address register_0 | SAR_0 | 8 | H'FF7F | IIC_0 | 8 | 2 |
| Serial mode register_1 | SMR_1 | 8 | H'FF80 ^{*3} | SCI_1 | 8 | 2 |
| I ² C bus control register_1 | ICCR_1 | 8 | H'FF80 ^{*3} | IIC_1 | 8 | 2 |
| Bit rate register_1 | BRR_1 | 8 | H'FF81 ^{*3} | SCI_1 | 8 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|--|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| I ² C bus status register_1 | ICSR_1 | 8 | H'FF81 ^{*3} | IIC_1 | 8 | 2 |
| Serial control register_1 | SCR_1 | 8 | H'FF82 | SCI_1 | 8 | 2 |
| Transmit data register_1 | TDR_1 | 8 | H'FF83 | SCI_1 | 8 | 2 |
| Serial status register_1 | SSR_1 | 8 | H'FF84 | SCI_1 | 8 | 2 |
| Receive data register_1 | RDR_1 | 8 | H'FF85 | SCI_1 | 8 | 2 |
| Smart card mode register_1 | SCMR_1 | 8 | H'FF86 ^{*3} | SCI_1 | 8 | 2 |
| I ² C bus data register_1 | ICDR_1 | 8 | H'FF86 ^{*3} | IIC_1 | 8 | 2 |
| Second slave address register_1 | SARX_1 | 8 | H'FF86 ^{*3} | IIC_1 | 8 | 2 |
| I ² C bus mode register_1 | ICMR_1 | 8 | H'FF87 | IIC_1 | 8 | 2 |
| Slave address register_1 | SAR_1 | 8 | H'FF87 | IIC_1 | 8 | 2 |
| Serial mode register_2 | SMR_2 | 8 | H'FF88 | SCI_2 | 8 | 2 |
| Bit rate register_2 | BRR_2 | 8 | H'FF89 | SCI_2 | 8 | 2 |
| Serial control register_2 | SCR_2 | 8 | H'FF8A | SCI_2 | 8 | 2 |
| Transmit data register_2 | TDR_2 | 8 | H'FF8B | SCI_2 | 8 | 2 |
| Serial status register_2 | SSR_2 | 8 | H'FF8C | SCI_2 | 8 | 2 |
| Receive data register_2 | RDR_2 | 8 | H'FF8D | SCI_2 | 8 | 2 |
| Smart card mode register_2 | SCMR_2 | 8 | H'FF8E | SCI_2 | 8 | 2 |
| A/D data register AH | ADDRAH | 8 | H'FF90 | A/D | 8 | 2 |
| A/D data register AL | ADDRAL | 8 | H'FF91 | A/D | 8 | 2 |
| A/D data register BH | ADDRBH | 8 | H'FF92 | A/D | 8 | 2 |
| A/D data register BL | ADDRBL | 8 | H'FF93 | A/D | 8 | 2 |
| A/D data register CH | ADDRCH | 8 | H'FF94 | A/D | 8 | 2 |
| A/D data register CL | ADDRCL | 8 | H'FF95 | A/D | 8 | 2 |
| A/D data register DH | ADDRDH | 8 | H'FF96 | A/D | 8 | 2 |
| A/D data register DL | ADDRDL | 8 | H'FF97 | A/D | 8 | 2 |
| A/D control/status register | ADCSR | 8 | H'FF98 | A/D | 8 | 2 |
| A/D control register | ADCR | 8 | H'FF99 | A/D | 8 | 2 |
| Timer control/status register_1 | TCSR_1 | 8 | H'FFA2 | WDT_1 | 16 | 2 |
| Timer counter_1 | TCNT_1 | 8 | H'FFA2 (write) | WDT_1 | 16 | 2 |
| Timer counter_1 | TCNT_1 | 8 | H'FFA3 (read) | WDT_1 | 16 | 2 |
| Flash memory control register 1 | FLMCR1 | 8 | H'FFA8 | FLASH | 8 | 2 |

| Register Name | Abbrevia- tion | Bit No. | Address ^{*1} | Module | Data Bus Width | Access State |
|-------------------------------------|-------------------|---------|-----------------------|--------|-------------------|-----------------|
| Flash memory control register 2 | FLMCR2 | 8 | H'FFA9 | FLASH | 8 | 2 |
| Erase block register 1 | EBR1 | 8 | H'FFAA | FLASH | 8 | 2 |
| Erase block register 2 | EBR2 | 8 | H'FFAB | FLASH | 8 | 2 |
| Flash memory power control register | FLPWCR | 8 | H'FFAC | FLASH | 8 | 2 |
| Port 1 register | PORT1 | 8 | H'FFB0 | PORT | 8 | 2 |
| Port 3 register | PORT3 | 8 | H'FFB2 | PORT | 8 | 2 |
| Port 4 register | PORT4 | 8 | H'FFB3 | PORT | 8 | 2 |
| Port 7 register | PORT7 | 8 | H'FFB6 | PORT | 8 | 2 |
| Port 9 register | PORT9 | 8 | H'FFB8 | PORT | 8 | 2 |
| Port A register | PORTA | 8 | H'FFB9 | PORT | 8 | 2 |
| Port B register | PORTB | 8 | H'FFBA | PORT | 8 | 2 |
| Port C register | PORTC | 8 | H'FFBB | PORT | 8 | 2 |
| Port D register | PORTD | 8 | H'FFBC | PORT | 8 | 2 |
| Port E register | PORTE | 8 | H'FFBD | PORT | 8 | 2 |
| Port F register | PORTF | 8 | H'FFBE | PORT | 8 | 2 |
| Port G register | PORTG | 8 | H'FFBF | PORT | 8 | 2 |

Notes: 1. Lower 16 bits of the address.

2. Allocated on the on-chip RAM. 32-bit bus when DTC accesses as register information, and 16-bit in other cases.
3. Part of registers SCI_0 and SCI_1 and part of registers IIC_0 and IIC_1 are allocated to the same address. Use the IICE bit of the serial control register X (SCRX) to select the register.

26.2 Register Bits

Register addresses and bit names of the on-chip peripheral modules are described below.

Each line covers eight bits, and 16-bit register is shown as 2 lines.

| Register | | | | | | | | | |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| MRA | SM1 | SM0 | DM1 | DM0 | MD1 | MD0 | DTS | Sz | DTC |
| SAR | Bit 23 | Bit 22 | Bit 21 | Bit 20 | Bit 19 | Bit 18 | Bit 17 | Bit 16 | |
| | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| MRB | CHNE | DISEL | — | — | — | — | — | — | |
| DAR | Bit 23 | Bit 22 | Bit 21 | Bit 20 | Bit 19 | Bit 18 | Bit 17 | Bit 16 | |
| | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| CRA | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| CRB | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| IECTR | IEE | IOL | DEE | CK | RE | LUEE | — | — | IEB |
| IECMR | — | — | — | — | — | CMD2 | CMD1 | CMD0 | |
| IEMCR | SS | RN2 | RN1 | RN0 | CTL3 | CTL2 | CTL1 | CTL0 | |
| IEAR1 | IAR3 | IAR2 | IAR1 | IAR0 | IMD1 | IMD0 | — | STE | |
| IEAR2 | IAR11 | IAR10 | IAR9 | IAR8 | IAR7 | IAR6 | IAR5 | IAR4 | |
| IESA1 | ISA3 | ISA2 | ISA1 | ISA0 | — | — | — | — | |
| IESA2 | ISA11 | ISA10 | ISA9 | ISA8 | ISA7 | ISA6 | ISA5 | ISA4 | |
| IETBFL | TBFL7 | TBFL6 | TBFL5 | TBFL4 | TBFL3 | TBFL2 | TBFL1 | TBFL0 | |
| IETBR | TBR7 | TBR6 | TBR5 | TBR4 | TBR3 | TBR2 | TBR1 | TBR0 | |
| IEMA1 | IMA3 | IMA2 | IMA1 | IMA0 | — | — | — | — | |
| IEMA2 | IMA11 | IMA10 | IMA9 | IMA8 | IMA7 | IMA6 | IMA5 | IMA4 | |
| IERCTL | — | — | — | — | RCTL3 | RCTL2 | RCTL1 | RCTL0 | |
| IERBFL | RBFL7 | RBFL6 | RBFL5 | RBFL4 | RBFL3 | RBFL2 | RBFL1 | RBFL0 | |
| IERBR | RBR7 | RBR6 | RBR5 | RBR4 | RBR3 | RBR2 | RBR1 | RBR0 | |
| IELA1 | ILA7 | ILA6 | ILA5 | ILA4 | ILA3 | ILA2 | ILA1 | ILA0 | |

| Register | | | | | | | | | |
|---------------------|----------------------|----------------|----------------|---------------------------------|----------------|----------------|----------------|----------------|---------------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| IELA2 | — | — | — | — | ILA11 | ILA10 | ILA9 | ILA8 | IEB |
| IEFLG | CMX | MRQ | SRQ | SRE | LCK | — | RSS | GG | |
| IETSR | TxRDY | — | — | — | IRA | TxS | TxF | TxE | |
| IEIET | TxRDYE | — | — | — | IRAE | TxSE | TxFE | TxEE | |
| IETEF | — | — | — | AL | UE | TTME | RO | ACK | |
| IERSR | RxRDY | — | — | — | — | RxS | RxF | RxE | |
| IEIER | RxRDYE | — | — | — | — | RxSE | RxFE | RxEE | |
| IEREF | — | — | — | — | OVE | RTME | DLE | PE | |
| DADR_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | D/A converter |
| DADR_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| DACR | DAOE1 | DAOE0 | DAE | — | — | — | — | — | |
| SCRX | — | IICX1 | IICX0 | IICE | FLSHE | — | — | — | IIC, FLASH |
| DDCSWR | — | — | — | — | CLR3 | CLR2 | CLR1 | CLR0 | IIC |
| TCR_2 | CMIEB | CMIEA | OVIE | CCLR1 | CCLR0 | CKS2 | CKS1 | CKS0 | TMR_2 |
| TCR_3 | CMIEB | CMIEA | OVIE | CCLR1 | CCLR0 | CKS2 | CKS1 | CKS0 | TMR_3 |
| TCSR_2 | CMFB | CMFA | OVF | — | OS3 | OS2 | OS1 | OS0 | TMR_2 |
| TCSR_3 | CMFB | CMFA | OVF | — | OS3 | OS2 | OS1 | OS0 | TMR_3 |
| TCORA_2 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_2 |
| TCORA_3 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_3 |
| TCORB_2 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_2 |
| TCORB_3 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_3 |
| TCNT_2 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_2 |
| TCNT_3 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_3 |
| SMR_3 ^{*1} | C/ \bar{A} (GM) | CHR (BLK) | PE (PE) | O/ \bar{E} (O/ \bar{E}) | STOP (BCP1) | MP (BCP0) | CKS1 (CKS1) | CKS0 (CKS0) | SCI_3 |
| BRR_3 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SCR_3 | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 | |
| TDR_3 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SSR_3 ^{*1} | TDRE (TDRE) | RDRF (RDRF) | ORER (ORER) | FER (ERS) | PER (PER) | TEND (TEND) | MPB (MPB) | MPBT (MPBT) | |
| RDR_3 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SCMR_3 | — | — | — | — | SDIR | SINV | — | SMIF | |

Register

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------|---------|---------|---------|---------|---------|---------|---------|---------|--------|
| SBYCR | SSBY | STS2 | STS1 | STS0 | OPE | — | — | — | SYSTEM |
| SYSCR | — | — | INTM1 | INTM0 | NMIEG | MRESE | — | RAME | |
| SCKCR | PSTOP | — | — | — | — | SCK2 | SCK1 | SCK0 | |
| MDCR | — | — | — | — | — | MDS2 | MDS1 | MDS0 | |
| MSTPCRA | MSTPA7 | MSTPA6 | MSTPA5 | MSTPA4 | MSTPA3 | MSTPA2 | MSTPA1 | MSTPA0 | |
| MSTPCRB | MSTPB7 | MSTPB6 | MSTPB5 | MSTPB4 | MSTPB3 | MSTPB2 | MSTPB1 | MSTPB0 | |
| MSTPCRC | MSTPC7 | MSTPC6 | MSTPC5 | MSTPC4 | MSTPC3 | MSTPC2 | MSTPC1 | MSTPC0 | |
| PFCR | — | — | BUZZE | — | AE3 | AE2 | AE1 | AE0 | BSC |
| LPWRCR | DTON | LSON | NESEL | SUBSTP | RFCUT | — | STC1 | STC0 | SYSTEM |
| SEMR_0 | SSE | — | — | — | ABCS | ACS2 | ACS1 | ACS0 | SCI_0 |
| BARA | — | — | — | — | — | — | — | — | PBC |
| | BAA23 | BAA22 | BAA21 | BAA20 | BAA19 | BAA18 | BAA17 | BAA16 | |
| | BAA15 | BAA14 | BAA13 | BAA12 | BAA11 | BAA10 | BAA9 | BAA8 | |
| | BAA7 | BAA6 | BAA5 | BAA4 | BAA3 | BAA2 | BAA1 | BAA0 | |
| BARB | — | — | — | — | — | — | — | — | |
| | BAB23 | BAB22 | BAB21 | BAB20 | BAB19 | BAB18 | BAB17 | BAB16 | |
| | BAB15 | BAB14 | BAB13 | BAB12 | BAB11 | BAB10 | BAB9 | BAB8 | |
| | BAB7 | BAB6 | BAB5 | BAB4 | BAB3 | BAB2 | BAB1 | BAB0 | |
| BCRA | CMFA | CDA | BAMRA2 | BAMRA1 | BAMRA0 | CSELA1 | CSELA0 | BIEA | |
| BCRB | CMFB | CDB | BAMRB2 | BAMRB1 | BAMRB0 | CSELB1 | CSELB0 | BIEB | |
| ISCRH | IRQ7SCB | IRQ7SCA | IRQ6SCB | IRQ6SCA | IRQ5SCB | IRQ5SCA | IRQ4SCB | IRQ4SCA | INT |
| ISCTL | IRQ3SCB | IRQ3SCA | IRQ2SCB | IRQ2SCA | IRQ1SCB | IRQ1SCA | IRQ0SCB | IRQ0SCA | |
| IER | IRQ7E | IRQ6E | IRQ5E | IRQ4E | IRQ3E | IRQ2E | IRQ1E | IRQ0E | |
| ISR | IRQ7F | IRQ6F | IRQ5F | IRQ4F | IRQ3F | IRQ2F | IRQ1F | IRQ0F | |
| DTCERA | DTCEA7 | DTCEA6 | DTCEA5 | DTCEA4 | DTCEA3 | DTCEA2 | DTCEA1 | DTCEA0 | DTC |
| DTCERB | — | DTCEB6 | DTCEB5 | DTCEB4 | DTCEB3 | DTCEB2 | DTCEB1 | DTCEB0 | |
| DTCERC | DTCEC7 | DTCEC6 | DTCEC5 | DTCEC4 | DTCEC3 | DTCEC2 | DTCEC1 | DTCEC0 | |
| DTCERD | — | — | DTCED5 | DTCED4 | DTCED3 | DTCED2 | DTCED1 | DTCED0 | |
| DTCERE | DTCEE7 | DTCEE6 | DTCEE5 | DTCEE4 | DTCEE3 | DTCEE2 | DTCEE1 | DTCEE0 | |
| DTCERF | DTCEF7 | DTCEF6 | DTCEF5 | DTCEF4 | DTCEF3 | DTCEF2 | DTCEF1 | DTCEF0 | |
| DTCERI | DTCEI7 | DTCEI6 | — | — | — | — | — | — | |
| DTVECR | SWDTE | DTVEC6 | DTVEC5 | DTVEC4 | DTVEC3 | DTVEC2 | DTVEC1 | DTVEC0 | |

| Register | | | | | | | | | |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| P1DDR | P17DDR | P16DDR | P15DDR | P14DDR | P13DDR | P12DDR | P11DDR | P10DDR | PORT |
| P3DDR | — | P36DDR | P35DDR | P34DDR | P33DDR | P32DDR | P31DDR | P30DDR | |
| P7DDR | P77DDR | P76DDR | P75DDR | P74DDR | P73DDR | P72DDR | P71DDR | P70DDR | |
| PADDR | — | — | — | — | PA3DDR | PA2DDR | PA1DDR | PA0DDR | |
| PBDDR | PB7DDR | PB6DDR | PB5DDR | PB4DDR | PB3DDR | PB2DDR | PB1DDR | PB0DDR | |
| PCDDR | PC7DDR | PC6DDR | PC5DDR | PC4DDR | PC3DDR | PC2DDR | PC1DDR | PC0DDR | |
| PDDDR | PD7DDR | PD6DDR | PD5DDR | PD4DDR | PD3DDR | PD2DDR | PD1DDR | PD0DDR | |
| PEDDR | PE7DDR | PE6DDR | PE5DDR | PE4DDR | PE3DDR | PE2DDR | PE1DDR | PE0DDR | |
| PFDDR | PF7DDR | PF6DDR | PF5DDR | PF4DDR | PF3DDR | PF2DDR | PF1DDR | PF0DDR | |
| PGDDR | — | — | — | PG4DDR | PG3DDR | PG2DDR | PG1DDR | PG0DDR | |
| PAPCR | — | — | — | — | PA3PCR | PA2PCR | PA1PCR | PA0PCR | |
| PBPCR | PB7PCR | PB6PCR | PB5PCR | PB4PCR | PB3PCR | PB2PCR | PB1PCR | PB0PCR | |
| PCPCR | PC7PCR | PC6PCR | PC5PCR | PC4PCR | PC3PCR | PC2PCR | PC1PCR | PC0PCR | |
| PDPCR | PD7PCR | PD6PCR | PD5PCR | PD4PCR | PD3PCR | PD2PCR | PD1PCR | PD0PCR | |
| PEPCR | PE7PCR | PE6PCR | PE5PCR | PE4PCR | PE3PCR | PE2PCR | PE1PCR | PE0PCR | |
| P3ODR | — | P36ODR | P35ODR | P34ODR | P33ODR | P32ODR | P31ODR | P30ODR | |
| PAODR | — | — | — | — | PA3ODR | PA2ODR | PA1ODR | PA0ODR | |
| TCR_3 | CCLR2 | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU_3 |
| TMDR_3 | — | — | BFB | BFA | MD3 | MD2 | MD1 | MD0 | |
| TIORH_3 | IOB3 | IOB2 | IOB1 | IOB0 | IOA3 | IOA2 | IOA1 | IOA0 | |
| TIORL_3 | IOD3 | IOD2 | IOD1 | IOD0 | IOC3 | IOC2 | IOC1 | IOC0 | |
| TIER_3 | TTGE | — | — | TCIEV | TGIED | TGIEC | TGIEB | TGIEA | |
| TSR_3 | — | — | — | TCFV | TGFD | TGFC | TGFB | TGFA | |
| TCNT_3 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRA_3 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRB_3 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGR_3 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |

Register

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|--------|--------|--------|--------|--------|--------|--------|-------|-------|--------|
| TGRD_3 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | TPU_3 |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TCR_4 | — | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU_4 |
| TMDR_4 | — | — | — | — | MD3 | MD2 | MD1 | MD0 | |
| TIOR_4 | IOB3 | IOB2 | IOB1 | IOB0 | IOA3 | IOA2 | IOA1 | IOA0 | |
| TIER_4 | TTGE | — | TCIEU | TCIEV | — | — | TGIEB | TGIEA | |
| TSR_4 | TCFD | — | TCFU | TCFV | — | — | TGFB | TGFA | |
| TCNT_4 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRA_4 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRB_4 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TCR_5 | — | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU_5 |
| | — | — | — | — | MD3 | MD2 | MD1 | MD0 | |
| TIOR_5 | IOB3 | IOB2 | IOB1 | IOB0 | IOA3 | IOA2 | IOA1 | IOA0 | |
| TIER_5 | TTGE | — | TCIEU | TCIEV | — | — | TGIEB | TGIEA | |
| TSR_5 | TCFD | — | TCFU | TCFV | — | — | TGFB | TGFA | |
| TCNT_5 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRA_5 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRB_5 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TSTR | — | — | CST5 | CST4 | CST3 | CST2 | CST1 | CST0 | TPU |
| TSYR | — | — | SYNC5 | SYNC4 | SYNC3 | SYNC2 | SYNC1 | SYNC0 | |
| IPRA | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | INT |
| IPRB | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRC | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRD | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRE | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |

| Register | | | | | | | | | |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| IPRF | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | INT |
| IPRG | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRH | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRI | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRJ | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRK | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRL | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| IPRO | — | IPR6 | IPR5 | IPR4 | — | IPR2 | IPR1 | IPR0 | |
| ABWCR | ABW7 | ABW6 | ABW5 | ABW4 | ABW3 | ABW2 | ABW1 | ABW0 | BSC |
| ASTCR | AST7 | AST6 | AST5 | AST4 | AST3 | AST2 | AST1 | AST0 | |
| WCRH | W71 | W70 | W61 | W60 | W51 | W50 | W41 | W40 | |
| WCRL | W31 | W30 | W21 | W20 | W11 | W10 | W01 | W00 | |
| BCRH | ICIS1 | ICIS0 | BRSTRM | BRSTS1 | BRSTS0 | — | — | — | |
| BCRL | BRLE | — | — | — | — | — | — | WAITE | |
| RAMER | — | — | — | — | RAMS | RAM2 | RAM1 | RAM0 | |
| MAR_0A | — | — | — | — | — | — | — | — | DMAC |
| | Bit 23 | Bit 22 | Bit 21 | Bit 20 | Bit 19 | Bit 18 | Bit 17 | Bit 16 | |
| | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| IOAR_0A | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| ETCR_0A | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| MAR_0B | — | — | — | — | — | — | — | — | |
| | Bit 23 | Bit 22 | Bit 21 | Bit 20 | Bit 19 | Bit 18 | Bit 17 | Bit 16 | |
| | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| IOAR_0B | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| ETCR_0B | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |

| Register | | | | | | | | | |
|----------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| MAR_1A | — | — | — | — | — | — | — | — | DMAC |
| | Bit 23 | Bit 22 | Bit 21 | Bit 20 | Bit 19 | Bit 18 | Bit 17 | Bit 16 | |
| | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| IOAR_1A | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| ETCR_1A | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| MAR_1B | — | — | — | — | — | — | — | — | |
| | Bit 23 | Bit 22 | Bit 21 | Bit 20 | Bit 19 | Bit 18 | Bit 17 | Bit 16 | |
| | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| IOAR_1B | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| ETCR_1B | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| P1DR | P17DR | P16DR | P15DR | P14DR | P13DR | P12DR | P11DR | P10DR | PORT |
| P3DR | — | P36DR | P35DR | P34DR | P33DR | P32DR | P31DR | P30DR | |
| P7DR | P77DR | P76DR | P75DR | P74DR | P73DR | P72DR | P71DR | P70DR | |
| PADR | — | — | — | — | PA3DR | PA2DR | PA1DR | PA0DR | |
| PBDR | PB7DR | PB6DR | PB5DR | PB4DR | PB3DR | PB2DR | PB1DR | PB0DR | |
| PCDR | PC7DR | PC6DR | PC5DR | PC4DR | PC3DR | PC2DR | PC1DR | PC0DR | |
| PDDR | PD7DR | PD6DR | PD5DR | PD4DR | PD3DR | PD2DR | PD1DR | PD0DR | |
| PEDR | PE7DR | PE6DR | PE5DR | PE4DR | PE3DR | PE2DR | PE1DR | PE0DR | |
| PFDR | PF7DR | PF6DR | PF5DR | PF4DR | PF3DR | PF2DR | PF1DR | PF0DR | |
| PGDR | — | — | — | PG4DR | PG3DR | PG2DR | PG1DR | PG0DR | |
| TCR_0 | CCLR2 | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU_0 |
| TMDR_0 | — | — | BFB | BFA | MD3 | MD2 | MD1 | MD0 | |
| TIORH_0 | IOB3 | IOB2 | IOB1 | IOB0 | IOA3 | IOA2 | IOA1 | IOA0 | |
| TIORL_0 | IOD3 | IOD2 | IOD1 | IOD0 | IOC3 | IOC2 | IOC1 | IOC0 | |
| TIER_0 | TTGE | — | — | TCIEV | TGIED | TGIEC | TGIEB | TGIEA | |

| Register | | | | | | | | | |
|----------|--------|--------|--------|--------|--------|--------|-------|-------|--------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| TSR_0 | — | — | — | TCFV | TGFD | TGFC | TGFB | TGFA | TPU_0 |
| TCNT_0 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRA_0 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRB_0 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRC_0 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRD_0 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TCR_1 | — | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU_1 |
| TMDR_1 | — | — | — | — | MD3 | MD2 | MD1 | MD0 | |
| TIOR_1 | IOB3 | IOB2 | IOB1 | IOB0 | IOA3 | IOA2 | IOA1 | IOA0 | |
| TIER_1 | TTGE | — | TCIEU | TCIEV | — | — | TGIEB | TGIEA | |
| TSR_1 | TCFD | — | TCFU | TCFV | — | — | TGFB | TGFA | |
| TCNT_1 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRA_1 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRB_1 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TCR_2 | — | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU_2 |
| TMDR_2 | — | — | — | — | MD3 | MD2 | MD1 | MD0 | |
| TIOR_2 | IOB3 | IOB2 | IOB1 | IOB0 | IOA3 | IOA2 | IOA1 | IOA0 | |
| TIER_2 | TTGE | — | TCIEU | TCIEV | — | — | TGIEB | TGIEA | |
| TSR_2 | TCFD | — | TCFU | TCFV | — | — | TGFB | TGFA | |
| TCNT_2 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| TGRA_2 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |

Register

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|-----------------------------|-------------|--------------|------------|--------------|----------------|--------------|----------------|----------------|--------|
| TGRB_2 | Bit 15 | Bit 14 | Bit 13 | Bit 12 | Bit 11 | Bit 10 | Bit 9 | Bit 8 | TPU_2 |
| | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| DMAWER | — | — | — | — | WE1B | WE1A | WE0B | WE0A | DMAC |
| DMATCR | — | — | TEE1 | TEE0 | — | — | — | — | |
| DMACR_0A ^{*2} DTSZ | DTID | RPE | DTDIR | DTF3 | DTF2 | DTF1 | DTF0 | | |
| DMACR_0A ^{*3} DTSZ | SAID | SAIDE | BLKDIR | BLKE | — | — | — | | |
| DMACR_0B ^{*2} DTSZ | DTID | RPE | DTDIR | DTF3 | DTF2 | DTF1 | DTF0 | | |
| DMACR_0B ^{*3} — | DAID | DAIDE | — | DTF3 | DTF2 | DTF1 | DTF0 | | |
| DMACR_1A ^{*2} DTSZ | DTID | RPE | DTDIR | DTF3 | DTF2 | DTF1 | DTF0 | | |
| DMACR_1A ^{*3} DTSZ | SAID | SAIDE | BLKDIR | BLKE | — | — | — | | |
| DMACR_1B ^{*2} DTSZ | DTID | RPE | DTDIR | DTF3 | DTF2 | DTF1 | DTF0 | | |
| DMACR_1B ^{*3} — | DAID | DAIDE | — | DTF3 | DTF2 | DTF1 | DTF0 | | |
| DMABCRH ^{*2} FAE1 | FAE0 | SAE1 | SAE0 | DTA1B | DTA1A | DTA0B | DTA0A | | |
| DMABCRH ^{*3} FAE1 | FAE0 | — | — | DTA1 | — | DTA0 | — | | |
| DMABCR ^{*2} DTE1B | DTE1A | DTE0B | DTE0A | DTIE1B | DTIE1A | DTIE0B | DTIE0A | | |
| DMABCR ^{*3} DTME1 | DTE1 | DTME0 | DTE0 | DTIE1B | DTIE1A | DTIE0B | DTIE0A | | |
| TCR_0 | CMIEB | CMIEA | OVIE | CCLR1 | CCLR0 | CKS2 | CKS1 | CKS0 | TMR_0 |
| TCR_1 | CMIEB | CMIEA | OVIE | CCLR1 | CCLR0 | CKS2 | CKS1 | CKS0 | TMR_1 |
| TCSR_0 | CMFB | CMFA | OVF | ADTE | OS3 | OS2 | OS1 | OS0 | TMR_0 |
| TCSR_1 | CMFB | CMFA | OVF | — | OS3 | OS2 | OS1 | OS0 | TMR_1 |
| TCORA_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_0 |
| TCORA_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_1 |
| TCORB_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_0 |
| TCORB_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_1 |
| TCNT_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_0 |
| TCNT_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | TMR_1 |
| TCSR_0 | OVF | WT/IT | TME | — | — | CKS2 | CKS1 | CKS0 | WDT_0 |
| TCNT_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| RSTCSR | WOVF | RSTE | RSTS | — | — | — | — | — | |
| SMR_0 ^{*1} | C/Ā (GM) | CHR (BLK) | PE (PE) | O/Ē (O/Ē) | STOP (BCP1) | MP (BCP0) | CKS1 (CKS1) | CKS0 (CKS0) | SCI_0 |

| Register | | | | | | | | | |
|---------------------|--------------|--------|--------|-----------------|--------|--------|--------|--------|--------|
| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
| ICCR_0 | ICE | IEIC | MST | TRS | ACKE | BBSY | IRIC | SCP | IIC_0 |
| BRR_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | SCI_0 |
| ICSR_0 | ESTP | STOP | IRTR | AASX | AL | AAS | ADZ | ACKB | IIC_0 |
| SCR_0 | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 | SCI_0 |
| TDR_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SSR_0 ^{*1} | TDRE | RDRF | ORER | FER | PER | TEND | MPB | MPBT | |
| | (TDRE) | (RDRF) | (ORER) | (ERS) | (PER) | (TEND) | (MPB) | (MPBT) | |
| RDR_0 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SCMR_0 | — | — | — | — | SDIR | SINV | — | SMIF | |
| ICDR_0 | ICDR7 | ICDR6 | ICDR5 | ICDR4 | ICDR3 | ICDR2 | ICDR1 | ICDR0 | IIC_0 |
| SARX_0 | SVAX6 | SVAX5 | SVAX4 | SVAX3 | SVAX2 | SVAX1 | SVAX0 | FSX | |
| ICMR_0 | MLS | WAIT | CKS2 | CKS1 | CKS0 | BC2 | BC1 | BC0 | |
| SAR_0 | SVA6 | SVA5 | SVA4 | SVA3 | SVA2 | SVA1 | SVA0 | FS | |
| SMR_1 ^{*1} | C/ \bar{A} | CHR | PE | O/ \bar{E} | STOP | MP | CKS1 | CKS0 | SCI_1 |
| | (GM) | (BLK) | (PE) | (O/ \bar{E}) | (BCP1) | (BCP0) | (CKS1) | (CKS0) | |
| ICCR_1 | ICE | IEIC | MST | TRS | ACKE | BBSY | IRIC | SCP | IIC_1 |
| BRR_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | SCI_1 |
| ICSR_1 | ESTP | STOP | IRTR | AASX | AL | AAS | ADZ | ACKB | IIC_1 |
| SCR_1 | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 | SCI_1 |
| TDR_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SSR_1 ^{*1} | TDRE | RDRF | ORER | FER | PER | TEND | MPB | MPBT | |
| | (TDRE) | (RDRF) | (ORER) | (ERS) | (PER) | (TEND) | (MPB) | (MPBT) | |
| RDR_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SCMR_1 | — | — | — | — | SDIR | SINV | — | SMIF | |
| ICDR_1 | ICDR7 | ICDR6 | ICDR5 | ICDR4 | ICDR3 | ICDR2 | ICDR1 | ICDR0 | IIC_1 |
| SARX_1 | SVAX6 | SVAX5 | SVAX4 | SVAX3 | SVAX2 | SVAX1 | SVAX0 | FSX | |
| ICMR_1 | MLS | WAIT | CKS2 | CKS1 | CKS0 | BC2 | BC1 | BC0 | |
| SAR_1 | SVA6 | SVA5 | SVA4 | SVA3 | SVA2 | SVA1 | SVA0 | FS | |
| SMR_2 ^{*1} | C/ \bar{A} | CHR | PE | O/ \bar{E} | STOP | MP | CKS1 | CKS0 | SCI_2 |
| | (GM) | (BLK) | (PE) | (O/ \bar{E}) | (BCP1) | (BCP0) | (CKS1) | (CKS0) | |
| BRR_2 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SCR_2 | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 | |

Section 26 List of Registers

Register

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------------------|----------------|----------------|----------------|--------------|--------------|----------------|--------------|----------------|---------------|
| TDR_2 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | SCI_2 |
| SSR_2 ^{*1} | TDRE (TDRE) | RDRF (RDRF) | ORER (ORER) | FER (ERS) | PER (PER) | TEND (TEND) | MPB (MPB) | MPBT (MPBT) | |
| RDR_2 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| SCMR_2 | — | — | — | — | SDIR | SINV | — | SMIF | |
| ADDRAH | AD9 | AD8 | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | A/D converter |
| ADDRAL | AD1 | AD0 | — | — | — | — | — | — | |
| ADDRBH | AD9 | AD8 | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | |
| ADDRBL | AD1 | AD0 | — | — | — | — | — | — | |
| ADDRCH | AD9 | AD8 | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | |
| ADDRCL | AD1 | AD0 | — | — | — | — | — | — | |
| ADDRDH | AD9 | AD8 | AD7 | AD6 | AD5 | AD4 | AD3 | AD2 | |
| ADDRDL | AD1 | AD0 | — | — | — | — | — | — | |
| ADCSR | ADF | ADIE | ADST | SCAN | — | CH2 | CH1 | CH0 | |
| ADCR | TRGS1 | TRGS0 | — | — | CKS1 | CKS0 | — | — | |
| TCSR_1 | OVF | WT/IT | TME | PSS | RST/NMI | CKS2 | CKS1 | CKS0 | WDT_1 |
| TCNT_1 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
| FLMCR1 | FWE | SWE1 | ESU1 | PSU1 | EV1 | PV1 | E1 | P1 | FLASH |
| FLMCR2 | FLER | — | — | — | — | — | — | — | |
| EBR1 | EB7 | EB6 | EB5 | EB4 | EB3 | EB2 | EB1 | EB0 | |
| EBR2 | — | — | EB13 | EB12 | EB11 | EB10 | EB9 | EB8 | |
| FLPWCR | PDWND | — | — | — | — | — | — | — | |
| PORT1 | P17 | P16 | P15 | P14 | P13 | P12 | P11 | P10 | PORT |
| PORT3 | — | P36 | P35 | P34 | P33 | P32 | P31 | P30 | |
| PORT4 | P47 | P46 | P45 | P44 | P43 | P42 | P41 | P40 | |
| PORT7 | P77 | P76 | P75 | P74 | P73 | P72 | P71 | P70 | |
| PORT9 | P97 | P96 | — | — | — | — | — | — | |
| PORTA | — | — | — | — | PA3 | PA2 | PA1 | PA0 | |
| PORTB | PB7 | PB6 | PB5 | PB4 | PB3 | PB2 | PB1 | PB0 | |
| PORTC | PC7 | PC6 | PC5 | PC4 | PC3 | PC2 | PC1 | PC0 | |
| PORTD | PD7 | PD6 | PD5 | PD4 | PD3 | PD2 | PD1 | PD0 | |

Register

| Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|---------------|
| PORTE | PE7 | PE6 | PE5 | PE4 | PE3 | PE2 | PE1 | PE0 | PORT |
| PORTF | PF7 | PF6 | PF5 | PF4 | PF3 | PF2 | PF1 | PF0 | |
| PORTG | — | — | — | PG4 | PG3 | PG2 | PG1 | PG0 | |

Notes: 1. Some bit names differ depending on whether used in normal mode and Smart Card interface mode.

The name in () indicates the name in Smart Card interface mode.

2. Short address mode
3. Full address mode

26.3 Register States in Each Operating Mode

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------|------------|-----------|------------------|------------------|--------|
| MRA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | DTC |
| SAR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| MRB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DAR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| CRA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| CRB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IECTR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IEB |
| IECMR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEMCR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEAR1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEAR2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IESA1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IESA2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IETBFL | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IETBR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEMA1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEMA2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IERCTL | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IERBFL | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IERBR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IELA1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IELA2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEFLG | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IETSR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEIET | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IETEF | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IERSR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEIER | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IEREF | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------------|-------------|-------------|------------------|------------------|--------|
| DADR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | D/A |
| DADR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DACR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SCRX | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC |
| DDCSWR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_2 |
| TCR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_3 |
| TCSR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_2 |
| TCSR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_3 |
| TCORA_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_2 |
| TCORA_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_3 |
| TCORB_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_2 |
| TCORB_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_3 |
| TCNT_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_2 |
| TCNT_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_3 |
| SMR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_3 |
| BRR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SCR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TDR_3 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SSR_3 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| RDR_3 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SCMR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SBYCR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SYSTEM |
| SYSCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| SCKCR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| MDCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| MSTPCRA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| MSTPCRB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| MSTPCRC | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| PFCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | BSC |
| LPWRCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | SYSTEM |
| SEMR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_0 |

Section 26 List of Registers

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------|------------|-----------|------------------|------------------|--------|
| BARA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | PBC |
| BARB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| BCRA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| BCRB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ISCRH | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | INT |
| ISCRL | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IER | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ISR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTCERA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | DTC |
| DTCERB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTCERC | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTCERD | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTCERE | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTCERF | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTCERI | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DTVECR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| P1DDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | PORT |
| P3DDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| P7DDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PADDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PBDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PCDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PDDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PEDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PFDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PGDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PAPCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PBPCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PCPCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PDPCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PEPCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| P3ODR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------|------------|-----------|------------------|------------------|--------|
| PAODR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | PORT |
| TCR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU_3 |
| TMDR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIORH_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIORL_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIER_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSR_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCNT_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRA_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRB_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRC_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRD_3 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCR_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU_4 |
| TMDR_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIOR_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIER_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSR_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCNT_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRA_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRB_4 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCR_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU_5 |
| TMDR_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIOR_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIER_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSR_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCNT_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRA_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRB_5 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSTR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU |
| TSYR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |

Section 26 List of Registers

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------|------------|-----------|------------------|------------------|--------|
| IPRA | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | INT |
| IPRB | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRC | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRD | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRE | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRF | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRG | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRH | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRI | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRJ | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRK | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRL | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| IPRO | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ABWCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | BSC |
| ASTCR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| WCRH | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| WCRL | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| BCRH | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| BCRL | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| RAMER | Initialized | — | — | — | — | — | — | — | — | — | Initialized | FLASH |
| MAR_0A | — | — | — | — | — | — | — | — | — | — | — | DMAC |
| IOAR_0A | — | — | — | — | — | — | — | — | — | — | — | |
| ETCR_0A | — | — | — | — | — | — | — | — | — | — | — | |
| MAR_0B | — | — | — | — | — | — | — | — | — | — | — | |
| IOAR_0B | — | — | — | — | — | — | — | — | — | — | — | |
| ETCR_0B | — | — | — | — | — | — | — | — | — | — | — | |
| MAR_1A | — | — | — | — | — | — | — | — | — | — | — | |
| IOAR_1A | — | — | — | — | — | — | — | — | — | — | — | |
| ETCR_1A | — | — | — | — | — | — | — | — | — | — | — | |
| MAR_1B | — | — | — | — | — | — | — | — | — | — | — | |
| IOAR_1B | — | — | — | — | — | — | — | — | — | — | — | |
| ETCR_1B | — | — | — | — | — | — | — | — | — | — | — | |

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------|------------|-----------|------------------|------------------|--------|
| P1DR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | PORT |
| P3DR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| P7DR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PADR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PBDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PCDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PDDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PEDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PFDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PGDR | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| TCR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU_0 |
| TMDR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIORH_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIORL_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIER_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCNT_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRA_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRB_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRC_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRD_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU_1 |
| TMDR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIOR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIER_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCNT_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRA_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRB_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |

Section 26 List of Registers

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------|------------|-----------|------------------|------------------|--------|
| TCR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TPU_2 |
| TMDR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIOR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TIER_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TSR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCNT_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRA_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TGRB_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMAWER | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | DMAC |
| DMATCR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMACR_0A | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMACR_0B | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMACR_1A | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMACR_1B | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMABCRH | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| DMABCRL | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TCR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_0 |
| TCR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_1 |
| TCSR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_0 |
| TCSR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_1 |
| TCORA_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_0 |
| TCORA_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_1 |
| TCORB_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_0 |
| TCORB_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_1 |
| TCNT_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_0 |
| TCNT_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | TMR_1 |
| TCSR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | WDT_0 |
| TCNT_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| RSTCSR | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SMR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_0 |
| ICCR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC_0 |

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------------|-------------|-------------|------------------|------------------|--------|
| BRR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_0 |
| ICSR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC_0 |
| SCR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_0 |
| TDR_0 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SSR_0 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| RDR_0 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SCMR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ICDR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC_0 |
| SARX_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ICMR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SAR_0 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SMR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_1 |
| ICCR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC_1 |
| BRR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_1 |
| ICSR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC_1 |
| SCR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_1 |
| TDR_1 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SSR_1 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| RDR_1 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SCMR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ICDR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | IIC_1 |
| SARX_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| ICMR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SAR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SMR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | SCI_2 |
| BRR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| SCR_2 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| TDR_2 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SSR_2 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| RDR_2 | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| SCMR_2 | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |

Section 26 List of Registers

| Register Name | Reset | Manual Reset | High-speed | Medium-speed | Sleep | Module Stop | Watch | Sub-active | Sub-sleep | Software Standby | Hardware Standby | Module |
|---------------|-------------|--------------|------------|--------------|-------|-------------|-------------|-------------|-------------|------------------|------------------|--------|
| ADDRAH | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | A/D |
| ADDRAL | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADDRBH | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADDRBL | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADDRCH | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADDRCL | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADDRDH | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADDRDL | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADCSR | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| ADCR | Initialized | Initialized | — | — | — | Initialized | Initialized | Initialized | Initialized | Initialized | Initialized | |
| TCSR_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | WDT_1 |
| TCNT_1 | Initialized | Initialized | — | — | — | — | — | — | — | — | Initialized | |
| FLMCR1 | Initialized | — | — | — | — | — | — | — | — | Initialized | Initialized | FLASH |
| FLMCR2 | Initialized | — | — | — | — | — | — | — | — | Initialized | Initialized | |
| EBR1 | Initialized | — | — | — | — | — | — | — | — | Initialized | Initialized | |
| EBR2 | Initialized | — | — | — | — | — | — | — | — | Initialized | Initialized | |
| FLPWCR | Initialized | — | — | — | — | — | — | — | — | Initialized | Initialized | |
| PORT1 | Initialized | — | — | — | — | — | — | — | — | — | Initialized | PORT |
| PORT3 | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORT4 | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORT7 | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORT9 | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTA | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTB | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTC | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTD | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTE | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTF | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |
| PORTG | Initialized | — | — | — | — | — | — | — | — | — | Initialized | |

Note: — is not initialized.

Section 27 Electrical Characteristics

27.1 Power Supply Voltage and Operating Frequency Range

Figures 27.1, 27.2, 27.3, 27.4, and 27.5 show power supply voltage and operating frequency ranges (shaded areas) of the H8S/2258 Group, H8S/2239 Group, H8S/2238B, H8S/2236B, H8S/2238R, H8S/2236R, and H8S/2237 Group and H8S/2227 Group respectively.

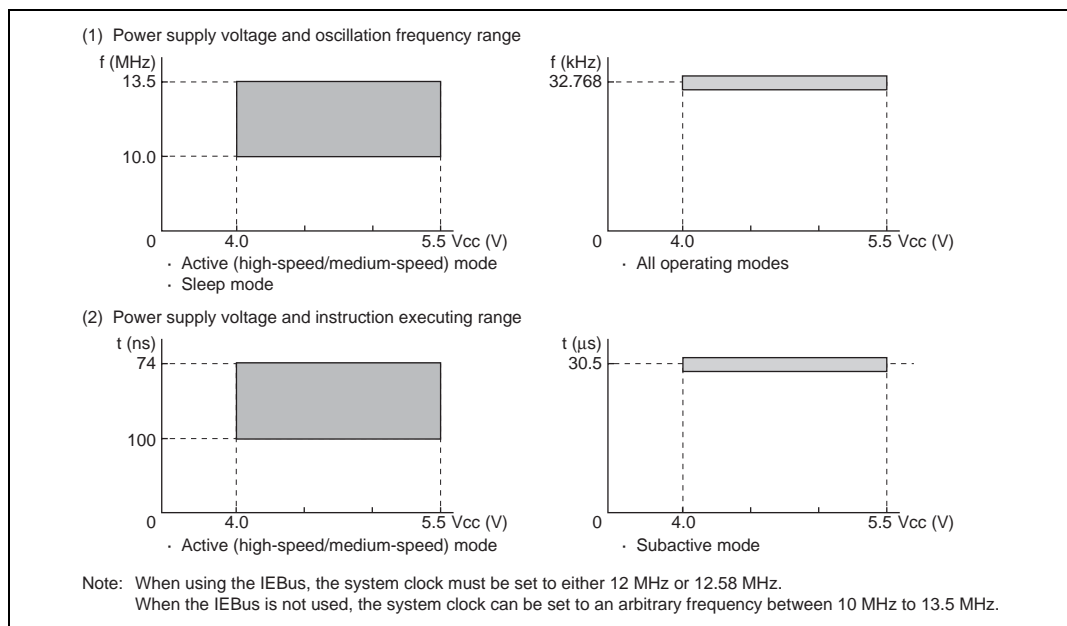
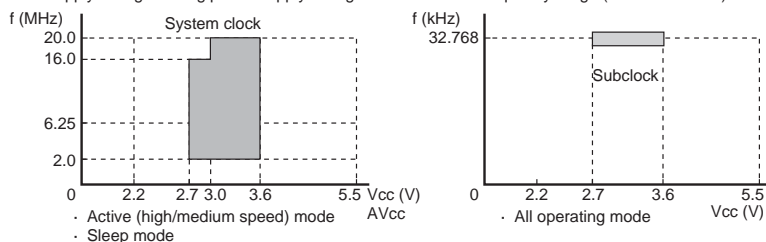
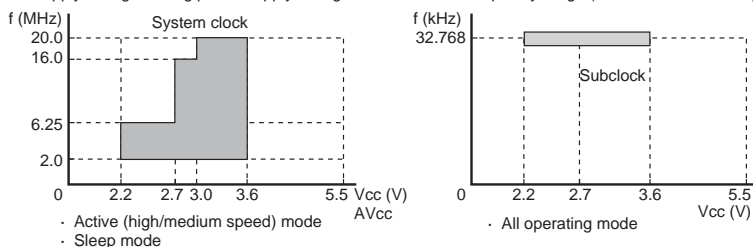


Figure 27.1 Power Supply Voltage and Operating Ranges (H8S/2258 Group)

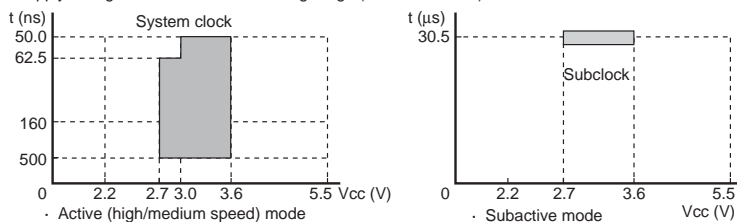
(1) Power supply voltage/analog power supply voltage and oscillation frequency range (F-ZTAT version)



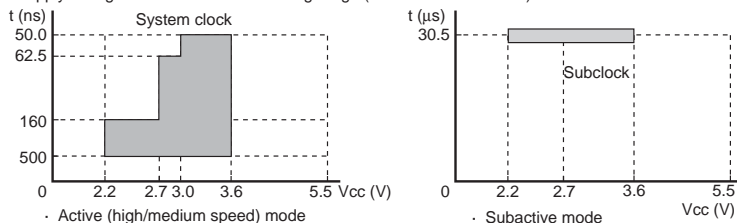
(2) Power supply voltage/analog power supply voltage and oscillation frequency range (Masked ROM version)



(3) Power supply voltage and instruction executing range (F-ZTAT version)



(4) Power supply voltage and instruction executing range (Masked ROM version)

**Figure 27.2 Power Supply Voltage and Operating Ranges (H8S/2239 Group)**

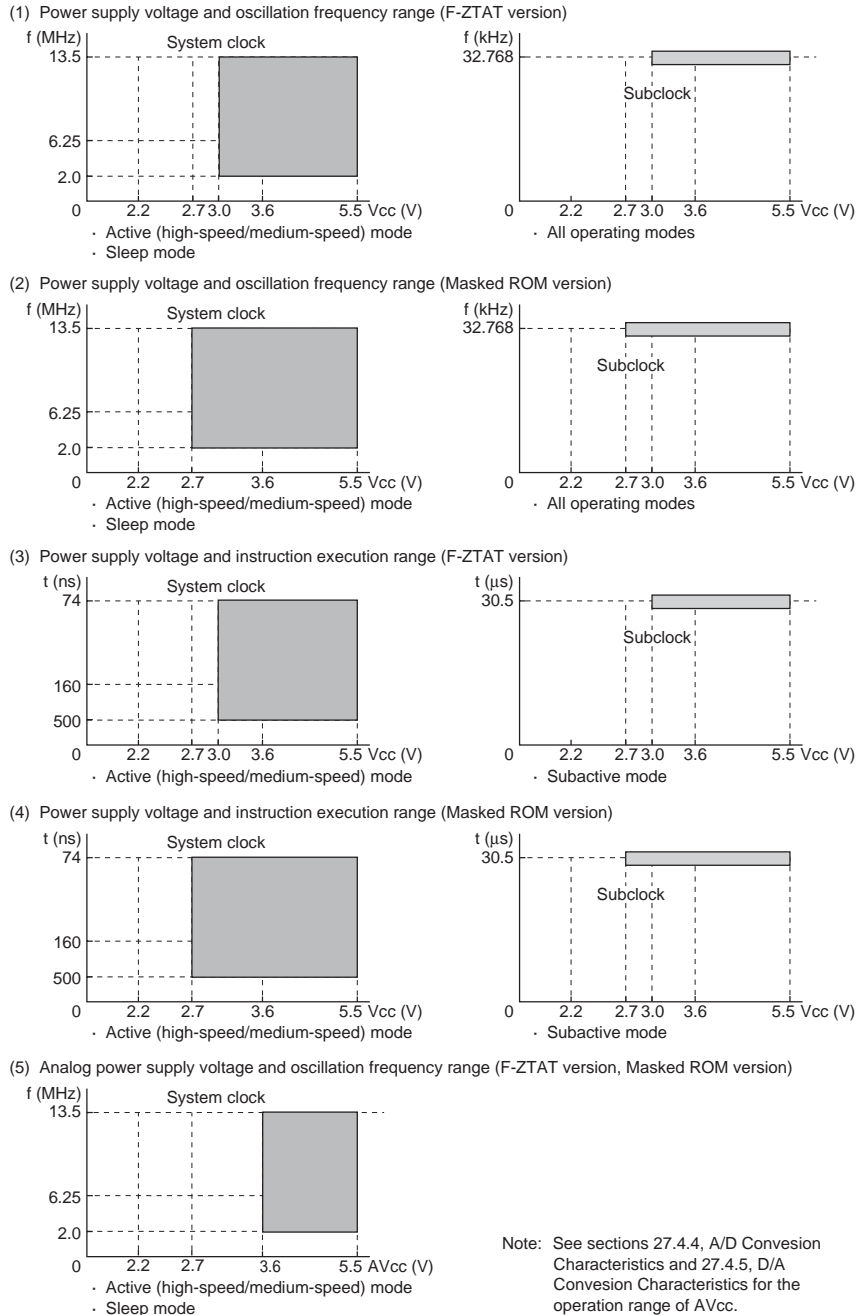
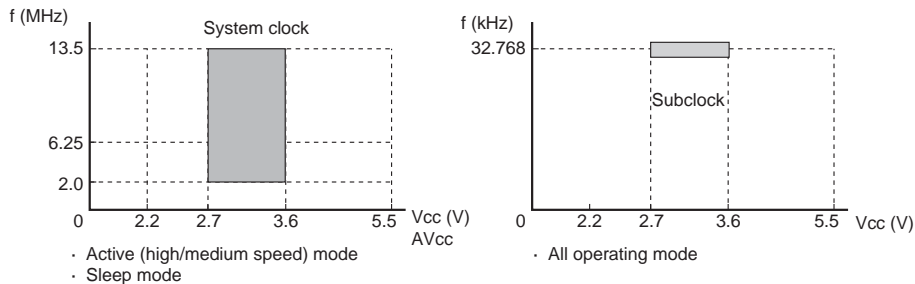
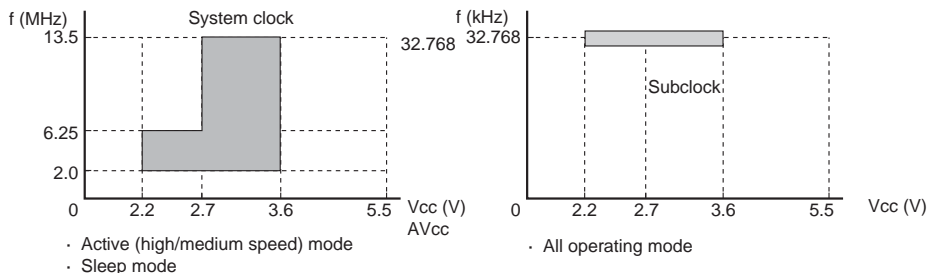


Figure 27.3 Power Supply Voltage and Operating Ranges (H8S/2238B and H8S/2236B)

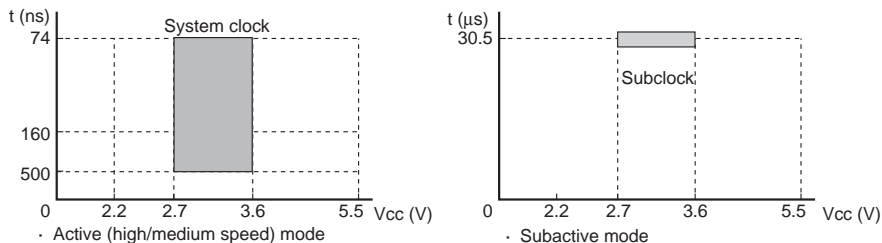
- (1) Power supply voltage/analog power supply voltage and oscillation frequency range
(F-ZTAT-version wide-range specifications)



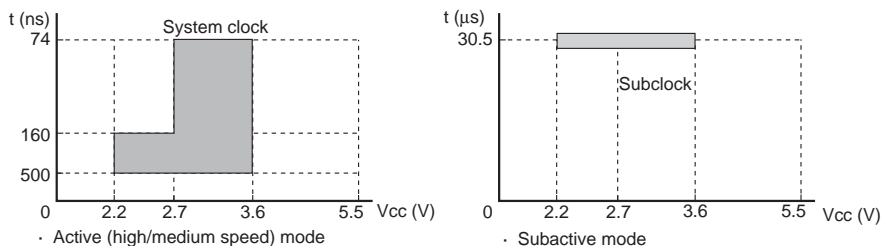
- (2) Power supply voltage/analog power supply voltage and oscillation frequency range
(F-ZTAT-version regular specifications/Masked ROM version)



- (3) Power supply voltage and instruction executing range (F-ZTAT-version wide-range specifications)

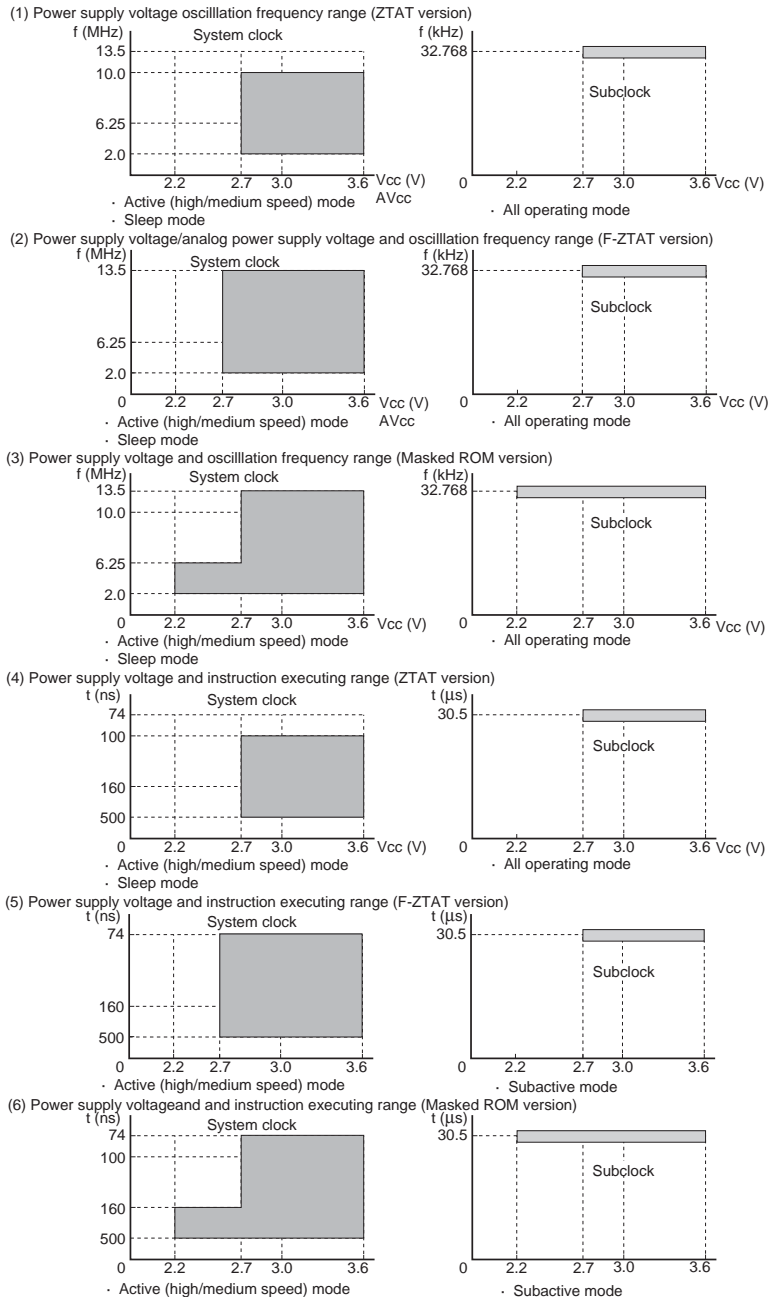


- (4) Power supply voltage and instruction executing range
(F-ZTAT-version regular specifications/Masked ROM version)



Note: The emulator does not operate at 2.2 V.

Figure 27.4 Power Supply Voltage and Operating Ranges (H8S/2238R and H8S/2236R)



**Figure 27.5 Power Supply Voltage and Operating Ranges
(H8S/2237 Group and H8S/2227 Group)**

27.2 Electrical Characteristics of H8S/2258 Group

27.2.1 Absolute Maximum Ratings

Table 27.1 lists the absolute maximum ratings.

Table 27.1 Absolute Maximum Ratings

| Item | Symbol | Value | Unit |
|--------------------------------------|-----------|--|------|
| Power supply voltage | V_{CC} | −0.3 to +7.0 | V |
| | CV_{CC} | −0.3 to +4.3 | V |
| Input voltage (except ports 4 and 9) | V_{in} | −0.3 to $V_{CC} + 0.3$ | V |
| Input voltage (ports 4 and 9) | V_{in} | −0.3 to $AV_{CC} + 0.3$ | V |
| Reference power supply voltage | V_{ref} | −0.3 to $AV_{CC} + 0.3$ | V |
| Analog power supply voltage | AV_{CC} | −0.3 to +7.0 | V |
| Analog input voltage | V_{AN} | −0.3 to $AV_{CC} + 0.3$ | V |
| Operating temperature | T_{opr} | Regular specifications: −20 to +75* | °C |
| | | Wide-range specifications: −40 to +85* | |
| Storage temperature | T_{stg} | −55 to +125 | °C |

Caution: Permanent damage to the chip may result if absolute maximum ratings are exceeded.

Note: * The operating temperature ranges for flash memory programming/erasing are $T_a = -20^{\circ}\text{C}$ to $+75^{\circ}\text{C}$ (regular specifications).

27.2.2 DC Characteristics

Table 27.2 lists the DC characteristics. Table 27.3 lists the permissible output currents. Table 27.4 lists the bus driving characteristics.

Table 27.2 DC Characteristics (1)

Conditions: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$
(wide-range specifications)*1

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-------------------------------|---|---------------|----------------------|-----|---------------------|------|------------------------------------|
| Schmitt trigger input voltage | $\overline{IRQ0}$ to $\overline{IRQ7}$ | VT^- | $V_{CC} \times 0.2$ | — | — | V | |
| | | VT^+ | — | — | $V_{CC} \times 0.8$ | V | |
| | | $VT^+ - VT^-$ | $V_{CC} \times 0.05$ | — | — | V | |
| Input high voltage | \overline{RES} , \overline{STBY} , NMI, MD2 to MD0, FWE | V_{IH} | $V_{CC} \times 0.9$ | — | $V_{CC} + 0.3$ | V | |
| | EXTAL, Ports 1, 3, 7, and A to G | | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4 and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3$ | V | |
| Input low voltage | \overline{RES} , \overline{STBY} , MD2 to MD0, FWE | V_{IL} | -0.3 | — | $V_{CC} \times 0.1$ | V | |
| | NMI, EXTAL, Ports 1, 3, 4, 7, 9, and A to G | | -0.3 | — | $V_{CC} \times 0.2$ | V | |
| Output high voltage | All output pins*3 except P34 and P35 | V_{OH} | $V_{CC} - 0.5$ | — | — | V | $I_{OH} = -200\text{ }\mu\text{A}$ |
| | | | $V_{CC} - 1.0$ | — | — | V | $I_{OH} = -1\text{ mA}$ |
| | P34 and P35*2 | | $V_{CC} - 2.7$ | — | — | V | $I_{OH} = -100\text{ }\mu\text{A}$ |
| Output low voltage | All output pins*3 | V_{OL} | — | — | 0.4 | V | $I_{OL} = 0.4\text{ mA}$ |
| | | | — | — | 0.4 | V | $I_{OL} = 0.8\text{ mA}$ |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|----------------------------|-------------|-----|-----|-----|---------|-------------------------------------|
| Input leakage current | RES | $ I_{in} $ | — | — | 1.0 | μA | $V_{in} = 0.5$ to $V_{cc} - 0.5 V$ |
| | STBY, NMI, MD2 to MD0, FWE | | — | — | 1.0 | μA | |
| | Ports 4 and 9 | | — | — | 1.0 | μA | $V_{in} = 0.5$ to $AV_{cc} - 0.5 V$ |
| Three states leakage current (off) | Ports 1, 3, 7, and A to G | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.5$ to $V_{cc} - 0.5 V$ |
| Input pull-up MOS current | Ports A to E | $-I_p$ | 10 | — | 300 | μA | $V_{in} = 0V$ |

- Notes: 1. If the A/D or D/A converter is not used, the AVCC, V_{ref} , and AVSS pins should not be open. Even if the A/D or D/A converter is not used, connect the AVCC and V_{ref} pins to V_{cc} and supply 4.0 V to 5.5 V. In this case, $V_{ref} \leq AV_{cc}$.
2. P35/SCK1/SCL0 and P34/SDA0 function as NMOS push-pull output. To output the high voltage from SCL0 and SDA0 (ICE = 1), connect an external pull-up resistor. NMOS controls P35/SCK1 and P34 (ICE = 0) to output the high voltage.
3. In the case when IICS = 0 and ICE = 0. Low voltage output of SCL1, SCL0, SDA1, and SDA0 with bus driving function is specified in table 27.4.

Table 27.2 DC Characteristics (2)

Conditions (F-ZTAT version): $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (regular specifications),
 $T_a = -40^{\circ}\text{C to }+85^{\circ}\text{C}$ (wide-range specifications)*¹

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|-------------------------------|-------------------------------|---------------|---|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 30 | pF | $V_{in} = 0\text{ V}$, $f = 1\text{ MHz}$, $T_a = 25^{\circ}\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 28 $V_{CC} = 5.0\text{ V}$ | 40 $V_{CC} = 5.5\text{ V}$ | mA | $f = 13.5\text{ MHz}$ |
| | Sleep mode | | — | 22 $V_{CC} = 5.0\text{ V}$ | 30 $V_{CC} = 5.5\text{ V}$ | mA | $f = 13.5\text{ MHz}$ |
| | All modules stopped | | — | 14 | — | mA | $f = 13.5\text{ MHz}$, $V_{CC} = 5.0\text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 17 | — | mA | $f = 13.5\text{ MHz}$, $V_{CC} = 5.0\text{ V}$ (reference value) |
| | Subactive mode | | — | 90 | 180 | μA | When 32.768 kHz crystal resonator is used, $V_{CC} = 5.0\text{ V}$ |
| | Subsleep mode | | — | 70 | 140 | μA | When 32.768 kHz crystal resonator is used, $V_{CC} = 5.0\text{ V}$ |
| | Watch mode | | — | 8 | 40 | μA | When 32.768 kHz crystal resonator is used, $V_{CC} = 5.0\text{ V}$ |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------------------|-----------------------------------|---------------|-----|------|-----|---------------|--|
| Current consumption*2 | Standby mode*3 | I_{CC}^{*4} | — | 1.5 | 10 | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | | — | — | 50 | | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D or D/A conversion | AI_{CC} | — | 0.4 | 1.5 | mA | |
| | Waiting for A/D or D/A conversion | | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D or D/A conversion | AI_{CC} | — | 2.1 | 3.5 | mA | |
| | Waiting for A/D or D/A conversion | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AVCC, V_{ref} , and AVSS pins should not be open. Even if the A/D or D/A converter is not used, connect the AVCC and V_{ref} pins to V_{CC} and supply 4.0 V to 5.5 V. In this case, $V_{ref} \leq AV_{CC}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{CC} - 0.5 \text{ V}$ and $V_{IL} \text{ max} = 0.5 \text{ V}$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 4.0 \text{ V}$, $V_{IH} \text{ min} = V_{CC} \times 0.9$, and $V_{IL} \text{ max} = 0.3 \text{ V}$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max.} = 2.0 \text{ (mA)} + 0.7 \text{ (mA/V)} \times V_{CC} + 1.4 \text{ (mA/MHz)} \times f + 0.20 \text{ (mA/(MHz}\cdot\text{V))} \times V_{CC} \times f$
(normal operation)
 $I_{CC} \text{ max.} = 1.5 \text{ (mA)} + 0.6 \text{ (mA/V)} \times V_{CC} + 1.1 \text{ (mA/MHz)} \times f + 0.15 \text{ (mA/(MHz}\cdot\text{V))} \times V_{CC} \times f$
(sleep mode)

Table 27.2 DC Characteristics (3)

Conditions (masked ROM version): $V_{CC} = 4.0 \text{ V to } 5.5 \text{ V}$, $AV_{CC} = 4.0 \text{ V to } 5.5 \text{ V}$,
 $V_{ref} = 4.0 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$
 (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*¹

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|--------------------------------|--------------------------------|---------------|---|
| Input capacitance | RES | C_{in} | — | — | 30 | pF | $V_{in} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 25 $V_{CC} = 5.0 \text{ V}$ | 40 $V_{CC} = 5.5 \text{ V}$ | mA | $f = 13.5 \text{ MHz}$ |
| | Sleep mode | | — | 20 $V_{CC} = 5.0 \text{ V}$ | 30 $V_{CC} = 5.5 \text{ V}$ | mA | $f = 13.5 \text{ MHz}$ |
| | All modules stopped | | — | 13 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 5.0 \text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 15 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 5.0 \text{ V}$ (reference value) |
| | Subactive mode | | — | 70 | 180 | μA | When 32.768 kHz crystal resonator is used, $V_{CC} = 5.0 \text{ V}$ |
| | Subsleep mode | | — | 50 | 100 | μA | When 32.768 kHz crystal resonator is used, $V_{CC} = 5.0 \text{ V}$ |
| | Watch mode | | — | 8 | 40 | μA | When 32.768 kHz crystal resonator is used, $V_{CC} = 5.0 \text{ V}$ |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|-----------------------------------|---------------|-----|------|-----|---------|--|
| Current consumption ^{*2} | Standby mode ^{*3} | I_{CC}^{*4} | — | 1.0 | 10 | μA | $T_a \leq 50^\circ C$, When 32.768 kHz crystal resonator is not used |
| | | | — | — | 50 | | $50^\circ C < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D or D/A conversion | AI_{CC} | — | 0.4 | 1.5 | mA | |
| | Waiting for A/D or D/A conversion | | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D or D/A conversion | AI_{CC} | — | 2.1 | 3.5 | mA | |
| | Waiting for A/D or D/A conversion | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AVCC, V_{ref} , and AVSS pins should not be open. Even if the A/D or D/A converter is not used, connect the AVCC and V_{ref} pins to V_{CC} and supply 4.0 V to 5.5 V. In this case, $V_{ref} \leq AV_{CC}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{CC} - 0.5 \text{ V}$ and $V_{IL} \text{ max} = 0.5 \text{ V}$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 4.0 \text{ V}$, $V_{IH} \text{ min} = V_{CC} \times 0.9$, and $V_{IL} \text{ max} = 0.3 \text{ V}$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max.} = 2.0 \text{ (mA)} + 0.7 \text{ (mA/V)} \times V_{CC} + 1.4 \text{ (mA/MHz)} \times f + 0.20 \text{ (mA/(MHz}\cdot\text{V))} \times V_{CC} \times f$
 (normal operation)
 $I_{CC} \text{ max.} = 1.5 \text{ (mA)} + 0.6 \text{ (mA/V)} \times V_{CC} + 1.1 \text{ (mA/MHz)} \times f + 0.15 \text{ (mA/(MHz}\cdot\text{V))} \times V_{CC} \times f$
 (sleep mode)

Table 27.3 Permissible Output Current

Conditions: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | | | Symbol | Min | Typ | Max | Unit |
|---|-----------------------------------|---|----------------|-----|-----|-----|------|
| Permissible output low current (per pin) | SCL1, SCL0, SDA1, SDA0 | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ | I_{OL} | — | — | 10 | mA |
| | Output pins other than above ones | | | — | — | 1.0 | |
| Permissible output low current (total) | Total of all output pins* | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ | $\sum I_{OL}$ | — | — | 60 | mA |
| Permissible output high current (per pin) | All output pins | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ | $-I_{OH}$ | — | — | 1.0 | mA |
| Permissible output high current (total) | Total of all output pins | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ | $\sum -I_{OH}$ | — | — | 30 | mA |

Note: * To protect chip reliability, do not exceed the output current values in table 27.3.

Table 27.4 Bus Driving Characteristics

Conditions: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)*

Objective pins: SCL1, SCL0, SDA1, and SDA0

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|---------------|---------------------|-----|---------------------|---------------|---|
| Schmitt trigger input voltage | VT^- | $V_{CC} \times 0.3$ | — | — | V | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| | VT^+ | — | — | $V_{CC} \times 0.7$ | | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| | $VT^+ - VT^-$ | 0.4 | — | — | | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| Input high voltage | V_{IH} | $V_{CC} \times 0.7$ | — | $V_{CC} + 0.5$ | V | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| Input low voltage | V_{IL} | -0.5 | — | $V_{CC} \times 0.3$ | V | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| Output low voltage | V_{OL} | — | — | 0.5 | V | $I_{OL} = 8\text{ mA}$ |
| | | — | — | 0.4 | | $I_{OL} = 3\text{ mA}$ |
| Input capacitance | C_{in} | — | — | 20 | pF | $V_{in} = 0\text{ V}$, $f = 1\text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| Three states leakage current (off) | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.5\text{ V to }V_{CC} - 0.5\text{ V}$ |
| SCL, SDA output falling time | t_{of} | $20 + 0.1 C_b$ | — | 250 | ns | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |

Note: * If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 4.0 V to 5.5 V. In this case, $V_{ref} \leq AV_{CC}$.

27.2.3 AC Characteristics

Figure 27.6 shows the test conditions for the AC characteristics.

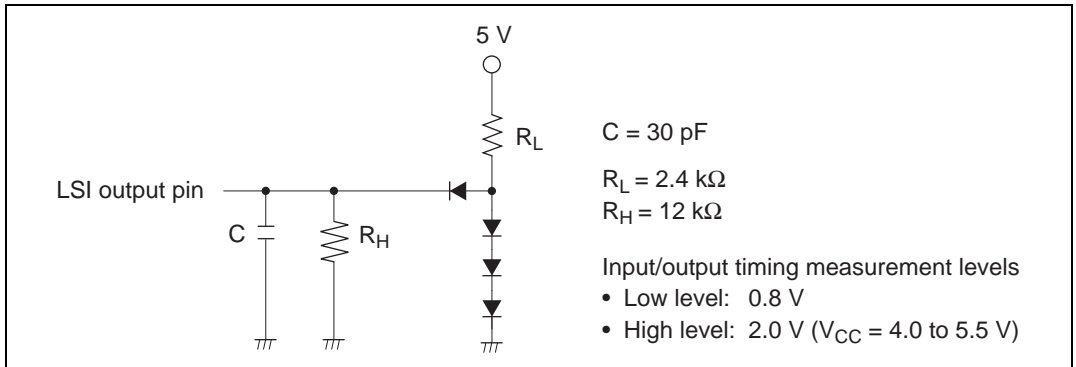


Figure 27.6 Output Load Circuit

(1) Clock Timing

Table 27.5 lists the clock timing.

Table 27.5 Clock Timing

Condition A: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$, 10 to 13.5 MHz, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Condition A | | Unit | Test Conditions |
|--|------------|-------------|--------|---------------|-----------------|
| | | Min | Max | | |
| Clock cycle time | t_{cyc} | 74 | 100 | ns | Figure 27.10 |
| Clock high pulse width | t_{CH} | 25 | — | ns | |
| Clock low pulse width | t_{CL} | 25 | — | ns | |
| Clock rise time | t_{Cr} | — | 10 | ns | |
| Clock fall time | t_{Cf} | — | 10 | ns | Figure 27.11 |
| Oscillation stabilization time at reset (crystal) | t_{OSC1} | 20 | — | ms | |
| Oscillation stabilization time in software standby (crystal) | t_{OSC2} | 8 | — | ms | |
| External clock output stabilization delay time | t_{DEXT} | 500 | — | μs | |
| 32-kHz clock oscillation stabilization time | t_{OSC3} | — | 2 | s | Figure 27.11 |
| Subclock oscillator frequency | f_{SUB} | 32.768 | 32.768 | kHz | |
| Subclock (ϕ_{SUB}) cycle time | t_{SUB} | 30.5 | 30.5 | μs | |

(2) Control Signal Timing

Table 27.6 lists the control signal timing.

Table 27.6 Control Signal Timing

Condition A: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$, 10 to 13.5 MHz, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Condition A | | Unit | Test Conditions |
|--|-------------|-------------|-----|-----------|-----------------|
| | | Min | Min | | |
| \overline{RES} setup time | t_{RESS} | 250 | — | ns | Figure 27.12 |
| \overline{RES} pulse width | t_{RESW} | 20 | — | t_{cyc} | |
| \overline{MRES} setup time | t_{MRESS} | 250 | — | ns | Figure 27.13 |
| \overline{MRES} pulse width | t_{MRESW} | 20 | — | t_{cyc} | |
| NMI setup time | t_{NMIS} | 250 | — | ns | |
| NMI hold time | t_{NMIH} | 10 | — | ns | |
| NMI pulse width (exiting software standby mode) | t_{NMIW} | 200 | — | ns | |
| \overline{IRQ} setup time | t_{IRQS} | 250 | — | ns | |
| \overline{IRQ} hold time | t_{IRQH} | 10 | — | ns | |
| \overline{IRQ} pulse width (exiting software standby mode) | t_{IRQW} | 200 | — | ns | |

(3) Bus Timing

Table 27.7 lists the bus timing.

Table 27.7 Bus Timing

Condition A: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 10\text{ to }13.5\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Condition A | | Unit | Test Conditions |
|-------------------------------|------------|---------------------------|---------------------------|------|------------------------|
| | | Min | Max | | |
| Address delay time | t_{AD} | — | 50 | ns | Figures 27.14 to 27.18 |
| Address setup time | t_{AS} | $0.5 \times t_{cyc} - 30$ | — | ns | |
| Address hold time | t_{AH} | $0.5 \times t_{cyc} - 15$ | — | ns | |
| \overline{CS} delay time | t_{CSD} | — | 50 | ns | |
| \overline{AS} delay time | t_{ASD} | — | 50 | ns | |
| \overline{RD} delay time 1 | t_{RSD1} | — | 50 | ns | |
| \overline{RD} delay time 2 | t_{RSD2} | — | 50 | ns | |
| Read data setup time | t_{RDS} | 30 | — | ns | |
| Read data hold time | t_{RDH} | 0 | — | ns | |
| Read data access time 1 | t_{ACC1} | — | $1.0 \times t_{cyc} - 65$ | ns | |
| Read data access time 2 | t_{ACC2} | — | $1.5 \times t_{cyc} - 65$ | ns | |
| Read data access time 3 | t_{ACC3} | — | $2.0 \times t_{cyc} - 65$ | ns | |
| Read data access time 4 | t_{ACC4} | — | $2.5 \times t_{cyc} - 65$ | ns | |
| Read data access time 5 | t_{ACC5} | — | $3.0 \times t_{cyc} - 65$ | ns | |
| \overline{WR} delay time 1 | t_{WRD1} | — | 50 | ns | |
| \overline{WR} delay time 2 | t_{WRD2} | — | 50 | ns | |
| \overline{WR} pulse width 1 | t_{WSW1} | $1.0 \times t_{cyc} - 30$ | — | ns | |
| \overline{WR} pulse width 2 | t_{WSW2} | $1.5 \times t_{cyc} - 30$ | — | ns | |
| Write data delay time | t_{WDD} | — | 70 | ns | |
| Write data setup time | t_{WDS} | $0.5 \times t_{cyc} - 37$ | — | ns | |
| Write data hold time | t_{WDH} | $0.5 \times t_{cyc} - 15$ | — | ns | |
| \overline{WAIT} setup time | t_{WTS} | 50 | — | ns | |
| \overline{WAIT} hold time | t_{WTH} | 10 | — | ns | |
| \overline{BREQ} setup time | t_{BRQS} | 50 | — | ns | Figure 27.19 |
| BACK delay time | t_{BACD} | — | 50 | ns | |
| Bus-floating time | t_{BZD} | — | 80 | ns | |

(4) Timing of On-Chip Peripheral Modules

Table 27.8 lists the timing of on-chip peripheral modules.

Table 27.8 Timing of On-Chip Peripheral Modules

Condition A: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$, 10 to 13.5 MHz, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| | | Symbol | Condition A | | Unit | Test Conditions |
|---------------|---------------------------------------|-------------------------|-------------|-----|------------|-----------------|
| Item | | | Min | Max | | |
| I/O ports | Output data delay time | t_{PVD} | — | 100 | ns | Figure 27.24 |
| | Input data setup time | t_{PRS} | 50 | — | | |
| | Input data hold time | t_{PRH} | 50 | — | | |
| TPU | Timer output delay time | t_{TODD} | — | 100 | ns | Figure 27.25 |
| | Timer input setup time | t_{TICS} | 40 | — | | |
| | Timer clock input setup time | t_{TCKS} | 40 | — | ns | Figure 27.26 |
| | Timer clock pulse width | Single edge t_{TCKWH} | 1.5 | — | t_{cyc} | |
| | | Both edges t_{TCKWL} | 2.5 | — | | |
| TMR | Timer output delay time | t_{TMOD} | — | 100 | ns | Figure 27.27 |
| | Timer reset input setup time | t_{TMRS} | 50 | — | ns | Figure 27.29 |
| | Timer clock input setup time | t_{TMCS} | 50 | — | ns | Figure 27.28 |
| | Timer clock pulse width | Single edge t_{TMCWH} | 1.5 | — | t_{cyc} | |
| | | Both edges t_{TMCWL} | 2.5 | — | | |
| WDT1 | BUZZ output delay time | t_{BUZD} | — | 100 | ns | Figure 27.30 |
| SCI | Input clock cycle | Asynchronous t_{Scyc} | 4 | — | t_{cyc} | Figure 27.31 |
| | | Synchronous | 6 | — | | |
| | Input clock pulse width | t_{SCKW} | 0.4 | 0.6 | t_{Scyc} | |
| | Input clock rise time | t_{SCKr} | — | 1.5 | t_{cyc} | |
| | Input clock fall time | t_{SCKf} | — | 1.5 | | |
| | Transmit data delay time | t_{TXD} | — | 100 | ns | Figure 27.32 |
| | Receive data setup time (synchronous) | t_{RXS} | 75 | — | ns | |
| | Receive data hold time (synchronous) | t_{RXH} | 75 | — | ns | |
| A/D converter | Trigger input setup time | t_{TRGS} | 40 | — | ns | Figure 27.33 |

Table 27.9 I²C Bus Timing

Conditions: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 5\text{ MHz}$ to maximum operating frequency,
 $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$

| Item | Symbol | Standard Value | | | Unit | Test Conditions |
|---|------------|----------------|-----|------|-----------|-----------------|
| | | Min | Typ | Max | | |
| SCL input cycle time | t_{SCL} | 12 | — | — | t_{cyc} | Figure 27.7 |
| SCL input high pulse width | t_{SCLH} | 3 | — | — | t_{cyc} | |
| SCL input low pulse width | t_{SCLL} | 5 | — | — | t_{cyc} | |
| SCL, SDA input rise time | t_{Sr} | — | — | 7.5* | t_{cyc} | |
| SCL, SDA input fall time | t_{Sf} | — | — | 300 | ns | |
| SCL, SDA input spike pulse elimination time | t_{SP} | — | — | 1 | t_{cyc} | |
| SDA input bus free time | t_{BUF} | 5 | — | — | t_{cyc} | |
| Start condition input hold time | t_{STAH} | 3 | — | — | t_{cyc} | |
| Retransmission start condition input setup time | t_{STAS} | 3 | — | — | t_{cyc} | |
| Stop condition input setup time | t_{STOS} | 3 | — | — | t_{cyc} | |
| Data input setup time | t_{SDAS} | 0.5 | — | — | t_{cyc} | |
| Data input hold time | t_{SDAH} | 0 | — | — | ns | |
| SCL, SDA load capacitance | C_b | — | — | 400 | pF | |

Note: * Can be $7.5\ t_{cyc}$ or $17.5\ t_{cyc}$ depending on the clock used in the I²C module. For details, see section 16.6, Usage Notes.

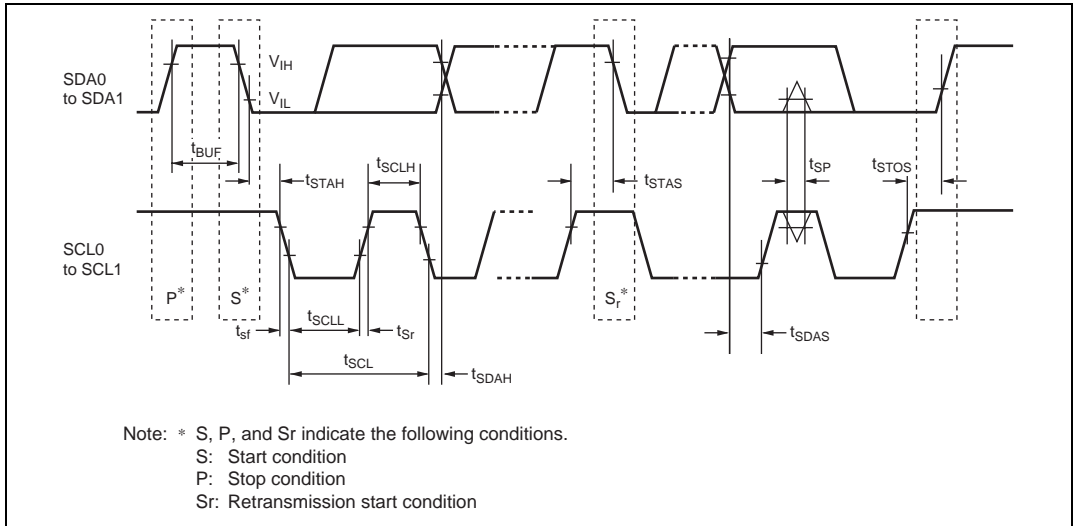


Figure 27.7 I²C Bus Interface Input/Output Timing (Optional)

27.2.4 A/D Conversion Characteristics

Table 27.10 lists the A/D conversion characteristics.

Table 27.10 A/D Conversion Characteristics

Condition A: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.0\text{ V to }3.6\text{ V}$, $V_{ref} = 3.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 10\text{ to }13.5\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular
specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Condition A | | | Unit |
|-------------------------------------|-------------|-----|-----------|---------------|
| | Min | Typ | Max | |
| Resolution | 10 | 10 | 10 | bits |
| Conversion time | 9.6 | — | — | μs |
| Analog input capacitance | — | — | 20 | pF |
| Permissible signal-source impedance | — | — | 5 | k Ω |
| Non-linearity error | — | — | ± 6.0 | LSB |
| Offset error | — | — | ± 4.0 | LSB |
| Full-scale error | — | — | ± 4.0 | LSB |
| Quantization error | — | — | ± 0.5 | LSB |
| Absolute accuracy | — | — | ± 8.0 | LSB |

27.2.5 D/A Conversion Characteristics

Table 27.11 lists the D/A conversion characteristics.

Table 27.11 D/A Conversion Characteristics

Condition A: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 10\text{ to }13.5\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular
specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Condition A | | | Unit | Test Conditions |
|-------------------|-------------|-----------|-----------|---------------|-------------------------------|
| | Min | Typ | Max | | |
| Resolution | 8 | 8 | 8 | bits | |
| Conversion time | — | — | 10 | μs | Load capacitance: 20 pF |
| Absolute accuracy | — | ± 2.0 | ± 3.0 | LSB | Load resistance: 2 M Ω |
| | — | — | ± 2.0 | LSB | Load resistance: 4 M Ω |

27.2.6 Flash Memory Characteristics

Table 27.12 Flash Memory Characteristics

Conditions: $V_{CC} = 4.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 4.0\text{ V to }5.5\text{ V}$, $V_{ref} = 4.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (Programming/erasing operating temperature range)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|--|-------------------|---------------------|------|-------------------|------------------------------------|
| Programming time ^{*1*2*4} | t_p | — | 40 | 200 | ms/ 128 bytes | |
| Erase time ^{*1*3*5} | t_E | — | 20 | 1000 | ms/block | |
| Reprogramming count | N_{WEC} | 100 ^{*6} | 10000 ^{*7} | — | Times | |
| Data hold time ^{*8} | t_{DRP} | 10 | — | — | Year | |
| Programming | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs |
| | Wait time after PSU1 bit setting ^{*1} | t_{spsu} | 50 | 50 | — | μs |
| | Wait time after P1 bit setting ^{*1*4} | t_{sp10} | 8 | 10 | 12 | μs |
| | | t_{sp30} | 28 | 30 | 32 | μs $1 \leq n \leq 6$ |
| | | t_{sp200} | 198 | 200 | 202 | μs $7 \leq n \leq 1000$ |
| | Wait time after P1 bit clear ^{*1} | t_{cp} | 5 | 5 | — | μs |
| | Wait time after PSU1 bit clear ^{*1} | t_{cpsu} | 5 | 5 | — | μs |
| | Wait time after PV1 bit setting ^{*1} | t_{spv} | 4 | 4 | — | μs |
| | Wait time after H'FF dummy write ^{*1} | t_{spvr} | 2 | 2 | — | μs |
| | Wait time after PV1 bit clear ^{*1} | t_{cpv} | 2 | 2 | — | μs |
| | Wait time after SWE1 bit clear ^{*1} | t_{cswe} | 100 | 100 | — | μs |
| | Maximum programming count ^{*1*4} | N1 | — | — | 6 ^{*4} | Times |
| | | N2 | — | — | 994 ^{*4} | |
| Erase | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs |
| | Wait time after ESU1 bit setting ^{*1} | t_{sesu} | 100 | 100 | — | μs |
| | Wait time after E1 bit setting ^{*1*5} | t_{se} | 10 | 10 | 100 | ms |
| | Wait time after E1 bit clear ^{*1} | t_{ce} | 10 | 10 | — | μs |
| | Wait time after ESU1 bit clear ^{*1} | t_{cesu} | 10 | 10 | — | μs |
| | Wait time after EV1 bit setting ^{*1} | t_{sev} | 20 | 20 | — | μs |
| | Wait time after H'FF dummy write ^{*1} | t_{sevr} | 2 | 2 | — | μs |
| | Wait time after EV1 bit clear ^{*1} | t_{cev} | 4 | 4 | — | μs |
| | Wait time after SWE1 bit clear ^{*1} | t_{cswe} | 100 | 100 | — | μs |
| | Maximum erase count ^{*1*5} | N | — | — | 100 | Times |

- Notes:
1. Follow the program/erase algorithms when making the time settings.
 2. Programming time per 128 bytes (Shows the total period for which the P1 bit in the flash memory control register 1 (FLMCR1) is set. It does not include the program verification time.)
 3. Erase block time (Shows the total period for which the E1 bit in FLMCR1 is set. It does not include the erase verification time.)
 4. Maximum programming time

$$t_p(\text{max}) = \text{Wait time after P1 bit setting } (t_{sp}) \times \text{Maximum programming count (N)}$$

$$(t_{sp30} + t_{sp10}) \times 6 + (t_{sp200}) \times 994$$
 5. For the maximum erase time ($t_E(\text{max})$), the following relationship applies between the wait time after E1 bit setting (z) and the maximum erase count (N):

$$t_E(\text{max}) = \text{Wait time after E1 bit setting } (t_{se}) \times \text{Maximum erase count (N)}$$
 6. The minimum times that all characteristics after reprogramming are guaranteed. (The range between 1 and a minimum value is guaranteed.)
 7. Reference value at 25°C. (Normally, it is a reference that rewriting is enabled up to this value.)
 8. Data hold characteristics are when reprogramming is performed within the range of specifications including a minimum value.

27.3 Electrical Characteristics of H8S/2239 Group

27.3.1 Absolute Maximum Ratings

Table 27.13 lists the absolute maximum ratings.

Table 27.13 Absolute Maximum Ratings

| Item | Symbol | Value | Unit |
|--------------------------------------|-----------|--|------|
| Power supply voltage | V_{CC} | −0.3 to +4.3 | V |
| | CV_{CC} | −0.3 to +4.3 | V |
| Input voltage (except ports 4 and 9) | V_{in} | −0.3 to $V_{CC} + 0.3$ | V |
| Input voltage (ports 4 and 9) | V_{in} | −0.3 to $AV_{CC} + 0.3$ | V |
| Reference power supply voltage | V_{ref} | −0.3 to $AV_{CC} + 0.3$ | V |
| Analog power supply voltage | AV_{CC} | −0.3 to +4.3 | V |
| Analog input voltage | V_{AN} | −0.3 to $AV_{CC} + 0.3$ | V |
| Operating temperature | T_{opr} | Regular specifications: −20 to +75* | °C |
| | | Wide-range specifications: −40 to +85* | |
| Storage temperature | T_{stg} | −55 to +125 | °C |

Caution: Permanent damage to the chip may result if absolute maximum rating are exceeded.

Note: * The operating temperature ranges for flash memory programming/erasing are $T_a = -20^{\circ}\text{C}$ to $+50^{\circ}\text{C}$ (regular specifications).

27.3.2 DC Characteristics

Table 27.14 lists the DC characteristics. Table 27.15 lists the permissible output currents. Table 27.16 lists the bus driving characteristics.

Table 27.14 DC Characteristics (1)

| | | | | | | |
|--|--|---|--|--|--|--|
| Condition A (F-ZTAT version): | | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications)* ¹ | | | | |
| Condition B (Masked ROM version): | | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)* ¹ | | | | |
| Condition C (F-ZTAT version and masked ROM version): | | $V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 3.0 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications) | | | | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--|---|----------------------|---------------------|-----|--------------------------------|------|-----------------|
| Schmitt trigger IRQ0 to IRQ7 input voltage | $\overline{VT^-}$ | $V_{CC} \times 0.2$ | — | — | — | V | |
| | $\overline{VT^+}$ | — | — | — | $V_{CC} \times 0.8$ | V | |
| | $\overline{VT^+} - \overline{VT^-}$ | $V_{CC} \times 0.05$ | — | — | — | V | |
| Input high voltage | \overline{RES} , \overline{STBY} , \overline{NMI} , \overline{FWE} , MD2 to MD0 | V_{IH} | $V_{CC} \times 0.9$ | — | $V_{CC} + 0.3$ | V | |
| | EXTAL, Ports 1, 3, 7, and A to G | | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4* ⁵ and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3$ * ⁵ | V | |
| Input low voltage | \overline{RES} , \overline{STBY} , \overline{FWE} , MD2 to MD0 | V_{IL} | -0.3 | — | $V_{CC} \times 0.1$ | V | |
| | \overline{NMI} , EXTAL, Ports 1, 3, 4, 7, 9, and A to G | | -0.3 | — | $V_{CC} \times 0.2$ | V | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|--|-------------|----------------|-----|-----|---------|---|
| Output high voltage | All output pins* ⁴ except P34 and P35 | V_{OH} | $V_{CC} - 0.5$ | — | — | V | $I_{OH} = -200 \mu A$ |
| | | | $V_{CC} - 1.0$ | — | — | V | $I_{OH} = -1 \text{ mA}^{*2}$ |
| | P34 and P35* ³ | | $V_{CC} - 2.0$ | — | — | V | $I_{OH} = -100 \mu A$ (reference value) |
| Output low voltage | All output pins* ⁴ | V_{OL} | — | — | 0.4 | V | $I_{OL} = 0.4 \text{ mA}$ |
| | | | — | — | 0.4 | V | $I_{OL} = 0.8 \text{ mA}^{*2}$ |
| Input leakage current | RES | $ I_{in} $ | — | — | 1.0 | μA | $V_{in} = 0.2$ to $V_{CC} - 0.2 \text{ V}$ |
| | STBY, NMI, FWE, MD2 to MD0 | | — | — | 1.0 | μA | |
| | Ports 4, 9 | | — | — | 1.0 | μA | $V_{in} = 0.2$ to $AV_{CC} - 0.2 \text{ V}$ |
| Three states leakage current (off) | Ports 1, 3, 7, and A to G | $ I_{TSL} $ | — | — | 1.0 | μA | $V_{in} = 0.2$ to $V_{CC} - 0.2 \text{ V}$ |
| Input pull-up MOS current | Ports A to E | $-I_p$ | 10 | — | 300 | μA | $V_{in} = 0 \text{ V}$ |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.
2. $V_{CC} = 2.7 \text{ V}$ to 3.6 V
3. P35/SCK1 and P34 function as NMOS push-pull output. To output the high voltage, connect an external pull-up resistor.
4. In the case when ICE = 0. Low voltage output with bus driving function is specified in table 27.16.
5. When $V_{CC} < AV_{CC}$, the maximum value for P40 and P41 is $V_{CC} + 0.3 \text{ V}$.

Table 27.14 DC Characteristics (2)

Condition A (F-ZTAT version): $V_{CC} = 2.7 \text{ V}$ to 3.6 V , $AV_{CC} = 2.7 \text{ V}$ to 3.6 V ,
 $V_{ref} = 2.7 \text{ V}$ to AV_{CC} , $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C}$ to $+75^\circ\text{C}$
 (regular specifications)*¹

Condition C (F-ZTAT version): $V_{CC} = 3.0 \text{ V}$ to 3.6 V , $AV_{CC} = 3.0 \text{ V}$ to 3.6 V ,
 $V_{ref} = 3.0 \text{ V}$ to AV_{CC} , $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C}$ to $+75^\circ\text{C}$
 (regular specifications), $T_a = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (wide-range specifications)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|--------------------------|--------------------------|------|---|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 30 | pF | $V_{in} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 29 | 55 | mA | $f = 20.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 25 | 42 | mA | $f = 16.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | Sleep mode | | — | 19 | 43 | mA | $f = 20.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 17 | 32 | mA | $f = 16.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | All modules stopped | | — | 16 | — | mA | $f = 20.0 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | | | — | 15 | — | mA | $f = 16.0 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 15 | — | mA | $f = 20.0 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | | | — | 13 | — | mA | $f = 16.0 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------------------|-----------------------|---------------|-----|--------------------------------|-------------------------------|---------------|---|
| Current consumption*2 | Subactive mode | I_{CC}^{*4} | — | 70 | 180 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 50 | 130 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Watch mode | | — | 8 | 40 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Standby mode*3 | | — | 1.0 $V_{CC} = 3.0\text{ V}$ | 10 $V_{CC} = 3.6\text{ V}$ | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | | — | — | 50 $V_{CC} = 3.6\text{ V}$ | μA | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D conversion | AI_{CC} | — | 0.5 | 1.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D conversion | AI_{CC} | — | 1.3 | 2.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.

- Current consumption values are for $V_{IH\text{ min}} = V_{CC} - 0.2\text{ V}$ and $V_{IL\text{ max}} = 0.2\text{ V}$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
- The values are for $V_{RAM} \leq V_{CC} < 2.7\text{ V}$, $V_{IH\text{ min}} = V_{CC} - 0.2$, and $V_{IL\text{ max}} = 0.2\text{ V}$.
- I_{CC} depends on V_{CC} and f as follows:
 $I_{CC\text{ max}} = 1.0\text{ (mA)} + 0.74\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (normal operation)
 $I_{CC\text{ max}} = 1.0\text{ (mA)} + 0.58\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (sleep mode)

Table 27.14 DC Characteristics (3)

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*¹

Condition C (F-ZTAT version): $V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 3.0 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|--------------------------|--------------------------|------|---|
| Input capacitance | RES | C_{in} | — | — | 30 | pF | $V_{in} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 29 | 55 | mA | $f = 20.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 25 | 42 | mA | $f = 16.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | Sleep mode | | — | 10 | 18 | mA | $f = 6.25 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 19 | 43 | mA | $f = 20.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 17 | 32 | mA | $f = 16.0 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | All modules stopped | | — | 7.5 | 14 | mA | $f = 6.25 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 16 | — | mA | $f = 20.0 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | | | — | 15 | — | mA | $f = 16.0 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------------------|---------------------------------|---------------|-----|-------------------------|------------------------|---------|---|
| Current consumption*2 | Medium-speed mode ($\phi/32$) | I_{CC}^{*4} | — | 15 | — | mA | $f = 20.0$ MHz, $V_{CC} = 3.0$ V (reference value) |
| | | | — | 13 | — | mA | $f = 16.0$ MHz, $V_{CC} = 3.0$ V (reference value) |
| | Subactive mode | | — | 45 | 180 | μ A | $V_{CC} = 3.0$ V, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 30 | 100 | μ A | $V_{CC} = 3.0$ V, When 32.768 kHz crystal resonator is used |
| | Watch mode | | — | 8 | 40 | μ A | $V_{CC} = 3.0$ V, When 32.768 kHz crystal resonator is used |
| | Standby mode*3 | | — | 0.5 $V_{CC} = 3.0$ V | 10 $V_{CC} = 3.6$ V | μ A | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | | | | 50 $V_{CC} = 3.6$ V | μ A | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D conversion | AI_{CC} | — | 0.5 | 1.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μ A | |
| Reference power supply current | During A/D conversion | AI_{CC} | — | 1.3 | 2.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μ A | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.

2. Current consumption values are for V_{IH} min = $V_{CC} - 0.2$ V and V_{IL} max = 0.2 V, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.

3. The values are for $V_{RAM} \leq V_{CC} < 2.2$ V, V_{IH} min = $V_{CC} - 0.2$, and V_{IL} max = 0.2 V.

4. I_{CC} depends on V_{CC} and f as follows:

$$I_{CC} \text{ max} = 1.0 \text{ (mA)} + 0.74 \text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f \text{ (normal operation)}$$

$$I_{CC} \text{ max} = 1.0 \text{ (mA)} + 0.58 \text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f \text{ (sleep mode)}$$

Table 27.15 Permissible Output Currents

| | |
|-----------------------------------|--|
| Condition A (F-ZTAT version): | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications) |
| Condition B (Masked ROM version): | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications) $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications) |
| Condition C (F-ZTAT version): | $V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 3.0 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications) |

| Item | | Symbol | Min | Typ | Max | Unit |
|---|-----------------------------------|---|-----|-----|-----|------|
| Permissible output low current (per pin) | SCL1, SCL0, SDA1, SDA0 | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ I_{OL} | — | — | 10 | mA |
| | Output pins other than above ones | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$ I_{OL} | — | — | 0.5 | |
| | | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ | — | — | 1.0 | |
| Permissible output low current (total) | Total of all output pins | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$ $\sum I_{OL}$ | — | — | 30 | mA |
| | | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ | — | — | 60 | |
| Permissible output high current (per pin) | All output pins | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$ $-I_{OH}$ | — | — | 0.5 | mA |
| | | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ | — | — | 1.0 | |
| Permissible output high current (total) | Total of all output pins | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$ $\sum -I_{OH}$ | — | — | 15 | mA |
| | | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ | — | — | 30 | |

Note: To protect chip reliability, do not exceed the output current values in table 27.15.

Table 27.16 Bus Driving Characteristics

Conditions: $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.7 \text{ V to } AV_{CC}$,
 $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*,
Objective pins: SCL1, SCL0, SDA1, SDA0

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|---------------|----------------------|-----|---------------------|---------------|---|
| Schmitt trigger input voltage | VT^- | $V_{CC} \times 0.3$ | — | — | V | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ |
| | VT^+ | — | — | $V_{CC} \times 0.7$ | V | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ |
| | $VT^+ - VT^-$ | $V_{CC} \times 0.05$ | — | — | V | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ |
| Input high voltage | V_{IH} | $V_{CC} \times 0.7$ | — | $V_{CC} + 0.5$ | V | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ |
| Input low voltage | V_{IL} | -0.5 | — | $V_{CC} \times 0.3$ | V | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ |
| Output low voltage | V_{OL} | — | — | 0.5 | V | $I_{OL} = 6 \text{ mA}$, $V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$ |
| | | — | — | 0.4 | V | $I_{OL} = 3 \text{ mA}$ |
| Input capacitance | C_{in} | — | — | 20 | pF | $V_{in} = 0 \text{ V}$ $f = 1 \text{ MHz}$ $T_a = 25^\circ\text{C}$ |
| Three states leakage current (off) | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.5 \text{ V to } V_{CC} - 0.5 \text{ V}$ |
| SCL, SDA output falling time | t_{of} | $20 + 0.1 C_b$ | — | 250 | ns | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$ |

Note: * If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.

27.3.3 AC Characteristics

Figure 27.8 shows the test conditions for the AC characteristics.

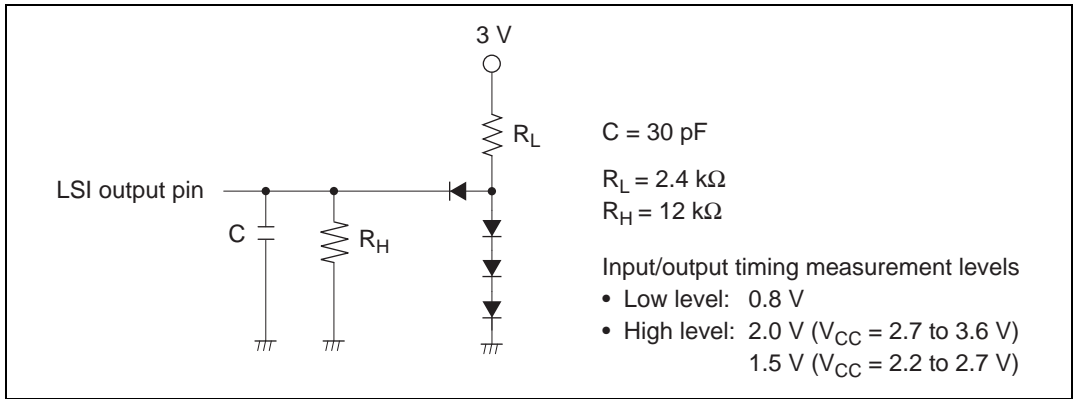


Figure 27.8 Output Load Circuit

(1) Clock Timing

Table 27.17 lists the clock timing.

Table 27.17 Clock Timing

Condition A (F-ZTAT version and masked ROM version):

$V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $\phi = 32.768 \text{ kHz}$,
 $2 \text{ to } 16.0 \text{ MHz}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular
specifications)

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $\phi = 32.768 \text{ kHz}$,
 $2 \text{ to } 6.25 \text{ MHz}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular
specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range
specifications)

Condition C (F-ZTAT version and masked ROM version):

$V_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 3.0 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $\phi = 32.768 \text{ kHz}$,
 $10.0 \text{ to } 20.0 \text{ MHz}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular
specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range
specifications)

| Item | Symbol | Condition A | | | Condition B | | | Condition C | | | Unit | Test Conditions |
|---|------------|-------------|-----|-----|-------------|-----|-----|-------------|-----|-----|------|-----------------|
| | | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | | |
| Clock cycle time | t_{cy} | 62.5 | — | 500 | 160 | — | 500 | 50 | — | 100 | ns | Figure 27.10 |
| Clock high pulse width | t_{CH} | 20 | — | — | 50 | — | — | 17 | — | — | ns | |
| Clock low pulse width | t_{CL} | 20 | — | — | 50 | — | — | 17 | — | — | ns | |
| Clock rise time | t_{Cr} | — | — | 10 | — | — | 25 | — | — | 10 | ns | Figure 27.11 |
| Clock fall time | t_{Cf} | — | — | 10 | — | — | 25 | — | — | 10 | ns | |
| Oscillation stabilization time at reset (crystal) | t_{OSC1} | 20 | — | — | 40 | — | — | 20 | — | — | ms | |

| Item | Symbol | Condition A | | | Condition B | | | Condition C | | | Unit | Test Conditions |
|--|------------|-------------|--------|-----|-------------|--------|-----|-------------|--------|-----|---------|-----------------|
| | | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | | |
| Oscillation stabilization time in software standby (crystal) | t_{OSC2} | 8 | — | — | 16 | — | — | 8 | — | — | ms | |
| External clock output stabilization delay time | t_{DEXT} | 500 | — | — | 1000 | — | — | 500 | — | — | μ s | Figure 27.11 |
| Subclock oscillation stabilization time | t_{OSC3} | — | — | 2 | — | — | 4 | — | — | 2 | s | |
| Subclock oscillator frequency | f_{SUB} | — | 32.768 | — | — | 32.768 | — | — | 32.768 | — | kHz | |
| Subclock (ϕ_{SUB}) cycle time | t_{SUB} | — | 30.5 | — | — | 30.5 | — | — | 30.5 | — | μ s | |

(2) Control Signal Timing

Table 27.18 lists the control signal timing.

Table 27.18 Control Signal Timing

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 32.768 \text{ kHz}, \\ &2 \text{ to } 16.0 \text{ MHz}, T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$
 $V_{ref} = 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 32.768 \text{ kHz}, 2 \text{ to } 6.25 \text{ MHz},$
 $T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$

Condition C (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 3.0 \text{ V to } 3.6 \text{ V}, AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 3.0 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 32.768 \text{ kHz}, 10.0 \\ &\text{to } 20.0 \text{ MHz}, T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ &T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Symbol | Conditions A, C | | Condition B | | Unit | Test Conditions |
|---|-------------|-----------------|-----|-------------|-----|-----------|-----------------|
| | | Min | Max | Min | Max | | |
| RES setup time | t_{RESS} | 250 | — | 350 | — | ns | Figure 27.12 |
| RES pulse width | t_{RESW} | 20 | — | 20 | — | t_{cyc} | |
| MRES setup time | t_{MRESS} | 250 | — | 350 | — | ns | |
| MRES pulse width | t_{MRESW} | 20 | — | 20 | — | t_{cyc} | |
| NMI setup time | t_{NMIS} | 250 | — | 350 | — | ns | Figure 27.13 |
| NMI hold time | t_{NMIH} | 10 | — | 10 | — | ns | |
| NMI pulse width (exiting software standby mode) | t_{NMIW} | 200 | — | 300 | — | ns | |
| IRQ setup time | t_{IRQS} | 250 | — | 350 | — | ns | |
| IRQ hold time | t_{IRQH} | 10 | — | 10 | — | ns | |
| IRQ pulse width (exiting software standby mode) | t_{IRQW} | 200 | — | 300 | — | ns | |

(3) Bus Timing

Table 27.19 lists the bus timing.

Table 27.19 Bus Timing

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 16.0 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition C (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 3.0 \text{ V to } 3.6 \text{ V}, AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 3.0 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 10.0 \text{ MHz to } \\ &20.0 \text{ MHz}, T_a = 20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= 40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|------------------------------|------------|------------------------------|------------------------------|------------------------------|------------------------------|------------------------------|-----|------|------------------------|
| | | Min | Max | Min | Max | Min | Max | | |
| Address delay time | t_{AD} | — | 40 | — | 90 | — | 35 | ns | Figures 27.14 to 27.18 |
| Address setup time | t_{AS} | $0.5 \times t_{cyc}$ – 42 | — | $0.5 \times t_{cyc}$ – 60 | — | $0.5 \times t_{cyc}$ – 35 | — | ns | |
| Address hold time | t_{AH} | $0.5 \times t_{cyc}$ – 10 | — | $0.5 \times t_{cyc}$ – 30 | — | $0.5 \times t_{cyc}$ – 5 | — | ns | |
| \overline{CS} delay time | t_{CSD} | — | 40 | — | 90 | — | 35 | ns | |
| \overline{AS} delay time | t_{ASD} | — | 40 | — | 90 | — | 25 | ns | |
| \overline{RD} delay time 1 | t_{RSD1} | — | 40 | — | 90 | — | 25 | ns | |
| \overline{RD} delay time 2 | t_{RSD2} | — | 40 | — | 90 | — | 25 | ns | |
| Read data setup time | t_{RDS} | 30 | — | 50 | — | 15 | — | ns | |
| Read data hold time | t_{RDH} | 0 | — | 0 | — | 0 | — | ns | |
| Read data access time 1 | t_{ACC1} | — | $1.0 \times t_{cyc}$ – 55 | — | $1.0 \times t_{cyc}$ – 90 | — | — | ns | |

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|-------------------------|------------|------------------------------|------------------------------|------------------------------|------------------------------|------------------------------|------------------------------|------|------------------------|
| | | Min | Max | Min | Max | Min | Max | | |
| Read data access time 2 | t_{ACC2} | — | $1.5 \times t_{cyc}$ – 50 | — | $1.5 \times t_{cyc}$ – 90 | — | $1.5 \times t_{cyc}$ – 40 | ns | Figures 27.14 to 27.18 |
| Read data access time 3 | t_{ACC3} | — | $2.0 \times t_{cyc}$ – 55 | — | $2.0 \times t_{cyc}$ – 90 | — | $2.0 \times t_{cyc}$ – 50 | ns | |
| Read data access time 4 | t_{ACC4} | — | $2.5 \times t_{cyc}$ – 50 | — | $2.5 \times t_{cyc}$ – 90 | — | $2.5 \times t_{cyc}$ – 40 | ns | |
| Read data access time 5 | t_{ACC5} | — | $3.0 \times t_{cyc}$ – 55 | — | $3.0 \times t_{cyc}$ – 90 | — | $3.0 \times t_{cyc}$ – 50 | ns | |
| WR delay time 1 | t_{WRD1} | — | 40 | — | 90 | — | 25 | ns | |
| WR delay time 2 | t_{WRD2} | — | 40 | — | 90 | — | 25 | ns | |
| WR pulse width 1 | t_{WSW1} | $1.0 \times t_{cyc}$ – 20 | — | $1.0 \times t_{cyc}$ – 60 | — | $1.0 \times t_{cyc}$ – 20 | — | ns | |
| WR pulse width 2 | t_{WSW2} | $1.5 \times t_{cyc}$ – 20 | — | $1.5 \times t_{cyc}$ – 60 | — | $1.5 \times t_{cyc}$ – 20 | — | ns | |
| Write data delay time | t_{WDD} | — | 60 | — | 100 | — | 40 | ns | |
| Write data setup time | t_{WDS} | $0.5 \times t_{cyc}$ – 57 | — | $0.5 \times t_{cyc}$ – 80 | — | $0.5 \times t_{cyc}$ – 65 | — | ns | |
| Write data hold time | t_{WDH} | $0.5 \times t_{cyc}$ – 27 | — | $0.5 \times t_{cyc}$ – 60 | — | $0.5 \times t_{cyc}$ – 20 | — | ns | |
| WAIT setup time | t_{WTS} | 40 | — | 90 | — | 25 | — | ns | Figure 27.16 |
| WAIT hold time | t_{WTH} | 10 | — | 10 | — | 10 | — | ns | |
| BREQ setup time | t_{BRQS} | 40 | — | 90 | — | 25 | — | ns | Figure 27.19 |
| BACK delay time | t_{BACD} | — | 40 | — | 90 | — | 40 | ns | |
| Bus-floating time | t_{BZD} | — | 60 | — | 160 | — | 50 | ns | |

(4) DMAC Timing

Table 27.20 lists the DMAC timing.

Table 27.20 DMAC Timing

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V, } \phi = 2 \text{ to } 16.0 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V,}$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V, } \phi = 2 \text{ to } 6.25 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition C (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 3.0 \text{ V to } 3.6 \text{ V, } AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 3.0 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V, } \phi = 10.0 \text{ MHz to } \\ &20.0 \text{ MHz, } T_a = 20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= 40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|-------------------|-------------|-------------|-----|-------------|-----|-------------|-----|------|-----------------|
| | | Min | Max | Min | Max | Min | Max | | |
| DREQ setup time | t_{DRQS} | 40 | — | 60 | — | 30 | — | ns | Figure 27.23 |
| DREQ hold time | t_{DRQH} | 10 | — | 20 | — | 10 | — | ns | |
| TEND delay time | t_{TED} | — | 30 | — | 50 | — | 30 | ns | Figure 27.22 |
| DACK delay time 1 | t_{DACD1} | — | 30 | — | 50 | — | 30 | ns | Figure 27.20 |
| DACK delay time 2 | t_{DACD2} | — | 30 | — | 50 | — | 30 | ns | Figure 27.21 |

(5) Timing of On-Chip Peripheral Modules

Table 27.21 lists the timing of on-chip peripheral modules. Table 27.22 lists the I²C bus timing.

Table 27.21 Timing of On-Chip Peripheral Modules

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V, } \phi = 32.768 \text{ kHz,} \\ &2 \text{ to } 16.0 \text{ MHz, } T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V,}$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V,} \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 6.25 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition C (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 3.0 \text{ V to } 3.6 \text{ V, } AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 3.0 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V,} \\ \phi &= 32.768 \text{ kHz, } 10.0 \text{ MHz to } 20.0 \text{ MHz,} \\ T_a &= 20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= 40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|-----------|-------------------------------------|-------------|-------------|-----|-------------|-----|-------------|-----|-----------|-----------------|
| | | | Min | Max | Min | Max | Min | Max | | |
| I/O port* | Output data delay time | t_{PWD} | — | 70 | — | 150 | — | 50 | ns | Figure 27.24 |
| | Input data setup time | t_{PRS} | 50 | — | 80 | — | 30 | — | | |
| | Input data hold time | t_{PRH} | 50 | — | 80 | — | 30 | — | | |
| TPU | Timer output delay time | t_{TOD} | — | 70 | — | 150 | — | 50 | ns | Figure 27.25 |
| | Timer input setup time | t_{TICS} | 40 | — | 60 | — | 30 | — | | |
| | Timer clock input setup time | t_{TCKS} | 40 | — | 60 | — | 30 | — | ns | Figure 27.26 |
| | Timer clock Single pulse width edge | t_{TCKWH} | 1.5 | — | 1.5 | — | 1.5 | — | t_{cyc} | |
| | | Both edges | t_{TCKWL} | 2.5 | — | 2.5 | — | 2.5 | — | |

| Item | | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|---------------|---------------------------------------|----------------------------|-------------|-----|-------------|-----|-------------|-----|------------|-----------------|
| | | | Min | Max | Min | Max | Min | Max | | |
| TMR | Timer output delay time | t_{TMOD} | — | 70 | — | 150 | — | 50 | ns | Figure 27.27 |
| | Timer reset input setup time | t_{TMRs} | 50 | — | 80 | — | 30 | — | ns | Figure 27.29 |
| | Timer clock input setup time | t_{TMCS} | 50 | — | 80 | — | 30 | — | ns | Figure 27.28 |
| | Timer clock pulse width | Single edge t_{TMCWH} | 1.5 | — | 1.5 | — | 1.5 | — | t_{cyc} | |
| | | Both edges t_{TMCWL} | 2.5 | — | 2.5 | — | 2.5 | — | | |
| WDT_1 | BUZZ output delay time | t_{BUZD} | — | 70 | — | 150 | — | 50 | ns | Figure 27.30 |
| SCI* | Input clock cycle | Asynchronous t_{Scyc} | 4 | — | 4 | — | 4 | — | t_{cyc} | Figure 27.31 |
| | | Synchronous | 6 | — | 6 | — | 6 | — | | |
| | Input clock pulse width | t_{SCKW} | 0.4 | 0.6 | 0.4 | 0.6 | 0.4 | 0.6 | t_{Scyc} | |
| | Input clock rise time | t_{SCKr} | — | 1.5 | — | 1.5 | — | 1.5 | t_{cyc} | |
| | Input clock fall time | t_{SCKf} | — | 1.5 | — | 1.5 | — | 1.5 | | |
| | Transmit data delay time | t_{TXD} | — | 75 | — | 150 | — | 50 | ns | Figure 27.32 |
| | Receive data setup time (synchronous) | t_{RXS} | 75 | — | 150 | — | 50 | — | ns | |
| | Receive data hold time (synchronous) | t_{RXH} | 75 | — | 150 | — | 50 | — | ns | |
| A/D converter | Trigger input setup time | t_{TRGS} | 40 | — | 60 | — | 30 | — | ns | Figure 27.33 |

Note: * NMOS controls P35/SCK1 and P34 to output the high voltage. To output the high voltage from P35/SCK1 and P34, connect an external pull-up resistor.

Table 27.22 I²C Bus Timing

Conditions: $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{SS} = 0 \text{ V}$, $\phi = 5 \text{ MHz}$ to maximum operating frequency,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|------------|---------------|-----|-----------------|------|-----------------|
| SCL input cycle time | t_{SCL} | $12 t_{cyc}$ | — | — | ns | Figure 27.34 |
| SCL input high pulse width | t_{SCLH} | $3 t_{cyc}$ | — | — | ns | |
| SCL input low pulse width | t_{SCLL} | $5 t_{cyc}$ | — | — | ns | |
| SCL, SDA input rise time | t_{Sr} | — | — | $7.5 t_{cyc}^*$ | ns | |
| SCL, SDA input fall time | t_{Sf} | — | — | 300 | ns | |
| SCL, SDA input spike pulse delete time | t_{Sp} | — | — | $1 t_{cyc}$ | ns | |
| SDA input bus free time | t_{BUF} | $5 t_{cyc}$ | — | — | ns | |
| Operating condition input hold time | t_{STAH} | $3 t_{cyc}$ | — | — | ns | |
| Retransmitting operating condition input setup time | t_{STAS} | $3 t_{cyc}$ | — | — | ns | |
| Stop condition input setup time | t_{STOS} | $3 t_{cyc}$ | — | — | ns | |
| Data input setup time | t_{SDAS} | $0.5 t_{cyc}$ | — | — | ns | |
| Data input hold time | t_{SDAH} | 0 | — | — | ns | |
| SCL, SDA capacitor load | C_b | — | — | 400 | pF | |

Note: * Maximum SCL and SDA input rise time $7.5 t_{cyc}$ or $17.5 t_{cyc}$ can be selected depending on the clock that is used in the I²C module. For detail, see section 16.6, Usage Notes.

27.3.4 A/D Conversion Characteristics

Table 27.23 lists the A/D conversion characteristics.

Table 27.23 A/D Conversion Characteristics

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}^*, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 16.0 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*,$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \phi = 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition C (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 3.0 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V}^*, \\ V_{ref} &= 3.0 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 10.0 \text{ to } 20.0 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Conditions A, C | | | Condition B | | | Unit |
|-------------------------------------|-----------------|-----|-----------|-------------|-----|-----------|---------------|
| | Min | Typ | Max | Min | Typ | Max | |
| Resolution | 10 | 10 | 10 | 10 | 10 | 10 | bits |
| Conversion time | 8.1 | — | — | 20.9 | — | — | μs |
| Analog input capacitance | — | — | 20 | — | — | 20 | pF |
| Permissible signal-source impedance | — | — | 5 | — | — | 5 | k Ω |
| Nonlinearity error | — | — | ± 6.0 | — | — | ± 6.0 | LSB |
| Offset error | — | — | ± 4.0 | — | — | ± 4.0 | LSB |
| Full-scale error | — | — | ± 4.0 | — | — | ± 4.0 | LSB |
| Quantization error | — | — | ± 0.5 | — | — | ± 0.5 | LSB |
| Absolute accuracy | — | — | ± 8.0 | — | — | ± 8.0 | LSB |

Note: * AN0 and AN1 can be used only when $V_{CC} = AV_{CC}$.

27.3.5 D/A Conversion Characteristics

Table 27.24 lists the D/A conversion characteristics.

Table 27.24 D/A Conversion Characteristics

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V, } \phi = 2 \text{ to } 16.0 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition B (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V,}$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V, } \phi = 2 \text{ to } 6.25 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition C (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 3.0 \text{ V to } 3.6 \text{ V, } AV_{CC} = 3.0 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 3.0 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V,} \\ \phi &= 10.0 \text{ to } 20.0 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Conditions A, C | | | Condition B | | | Unit | Test Condition |
|--------------------|-----------------|------|------|-------------|------|------|------|-----------------------------|
| | Min | Typ | Max | Min | Typ | Max | | |
| Resolution | 8 | 8 | 8 | 8 | 8 | 8 | bits | |
| Conversion time | — | — | 10 | — | — | 10 | μs | Load capacitance = 20 pF |
| Absolute accuracy* | — | ±2.0 | ±3.0 | — | ±3.0 | ±4.0 | LSB | Load resistance = 2 MΩ |
| | — | — | ±2.0 | — | — | ±3.0 | LSB | Load resistance = 4 MΩ |

Note: * Does not apply in module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode.

27.3.6 Flash Memory Characteristics

Table 27.25 lists the flash memory characteristics.

Table 27.25 Flash Memory Characteristics

Conditions: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{ref} = 2.7\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $V_{CC} = 3.0\text{ V to }3.6\text{ V}$ (Programming/erasing operating voltage range), $T_a = -20^{\circ}\text{C to }+50^{\circ}\text{C}$ (Programming/erasing operating temperature range; regular specifications, wide-range specifications)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|--|-------------------|---------------------|-------------------|--------------|------------------------------------|
| Programming time ^{*1*2*4} | t_p | — | 10 | 200 | ms/128 bytes | |
| Erase time ^{*1*3*5} | t_E | — | 100 | 1200 | ms/block | |
| Reprogramming count | N_{WEC} | 100 ^{*6} | 10000 ^{*7} | — | Times | |
| Data hold time ^{*8} | t_{DRP} | 10 | — | — | year | |
| Programming | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs |
| | Wait time after PSU1 bit setting ^{*1} | t_{spsu} | 50 | 50 | — | μs |
| | Wait time after P1 bit setting ^{*1*4} | t_{sp10} | 8 | 10 | 12 | μs |
| | | t_{sp30} | 28 | 30 | 32 | μs $1 \leq n \leq 6$ |
| | | t_{sp200} | 198 | 200 | 202 | μs $7 \leq n \leq 1000$ |
| | Wait time after P1 bit clear ^{*1} | t_{cp} | 5 | 5 | — | μs |
| | Wait time after PSU1 bit clear ^{*1} | t_{cpsu} | 5 | 5 | — | μs |
| | Wait time after PV1 bit setting ^{*1} | t_{spv} | 4 | 4 | — | μs |
| | Wait time after H'FF dummy write ^{*1} | t_{spvr} | 2 | 2 | — | μs |
| | Wait time after PV1 bit clear ^{*1} | t_{cpv} | 2 | 2 | — | μs |
| | Wait time after SWE1 bit clear | t_{cswe} | 100 | 100 | — | μs |
| Maximum programming count ^{*1*4} | N1 | — | — | 6 ^{*4} | Times | |
| | N2 | — | — | 994 ^{*4} | | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-------|--|------------|-----|-----|-----|---------|-----------------|
| Erase | Wait time after SWE1 bit setting ^{*1} | t_{SSWE} | 1 | 1 | — | μs | |
| | Wait time after ESU1 bit setting ^{*1} | t_{SESU} | 100 | 100 | — | μs | |
| | Wait time after E1 bit setting ^{*1*5} | t_{SE} | 10 | 10 | 100 | ms | |
| | Wait time after E1 bit clear ^{*1} | t_{CE} | 10 | 10 | — | μs | |
| | Wait time after ESU1 bit clear ^{*1} | t_{CESU} | 10 | 10 | — | μs | |
| | Wait time after EV1 bit setting ^{*1} | t_{SEV} | 20 | 20 | — | μs | |
| | Wait time after H'FF dummy write ^{*1} | t_{SEVT} | 2 | 2 | — | μs | |
| | Wait time after EV1 bit clear ^{*1} | t_{CEV} | 4 | 4 | — | μs | |
| | Wait time after SWE1 bit clear | t_{CSWE} | 100 | 100 | — | μs | |
| | Maximum erase count ^{*1*5} | N | — | — | 100 | Times | |

- Notes: 1. Make each time setting in accordance with the program/program-verify flowchart or erase/erase-verify flowchart.
2. Programming time per 128 bytes (Shows the total period for which the P1 bit in the flash memory control register 1 (FLMCR1) is set. It does not include the program verification time.)
3. Block erase time (Shows the total period for which the E1 bit in FLMCR1 is set. It does not include the erase verification time.)
4. Maximum programming time value

$$t_p(\text{max}) = \text{Wait time after P1 bit setting } (t_{sp}) \times \text{maximum program count } (N)$$

$$(t_{sp30} + t_{sp10}) \times 6 + (t_{sp200}) \times 994$$
5. Relationship among the maximum erase time ($t_E(\text{max})$), the wait time after E1 bit setting (t_{SE}), and the maximum erase count (N) is shown below.

$$t_E(\text{max}) = \text{Wait time after E1 bit setting } (t_{SE}) \times \text{maximum erase count } (N)$$
6. The minimum times that all characteristics after rewriting are guaranteed. (A range between 1 and minimum value is guaranteed.)
7. The reference value at 25°C. (Normally, it is a reference that rewriting is enabled up to this value.)
8. Data hold characteristics when rewriting is performed within the range of specifications including minimum value.

27.4 Electrical Characteristics of H8S/2238B and H8S/2236B

27.4.1 Absolute Maximum Ratings

Table 27.26 lists the absolute maximum ratings.

Table 27.26 Absolute Maximum Ratings

| Item | Symbol | Value | Unit |
|--------------------------------------|-----------|--|------|
| Power supply voltage | V_{CC} | −0.3 to +7.0 | V |
| | CV_{CC} | −0.3 to +4.3 | V |
| Input voltage (except ports 4 and 9) | V_{in} | −0.3 to $V_{CC} + 0.3$ | V |
| Input voltage (ports 4 and 9) | V_{in} | −0.3 to $AV_{CC} + 0.3$ | V |
| Reference voltage | V_{ref} | −0.3 to $AV_{CC} + 0.3$ | V |
| Analog power supply voltage | AV_{CC} | −0.3 to +7.0 | V |
| Analog input voltage | V_{AN} | −0.3 to $AV_{CC} + 0.3$ | V |
| Operating temperature | T_{opr} | Regular specifications: −20 to +75* | °C |
| | | Wide-range specifications: −40 to +85* | °C |
| Storage temperature | T_{stg} | −55 to +125 | °C |

Caution: Permanent damage to the chip may result if absolute maximum rating are exceeded.

Note: * The operating temperature ranges for flash memory programming/erasing are $T_a = -20^{\circ}\text{C}$ to $+75^{\circ}\text{C}$.

27.4.2 DC Characteristics

Table 27.27 lists the DC characteristics. Table 27.28 lists the permissible output currents. Table 27.29 lists the bus drive characteristics.

Table 27.27 DC Characteristics (1)

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)*¹

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)*¹

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-------------------------------|--|----------------------|----------------------|-----|--------------------------|------|---|
| Schmitt trigger input voltage | $\overline{IRQ7}$ to $\overline{IRQ0}$ | VT | $V_{CC} \times 0.2$ | — | — | V | |
| | | VT ⁺ | — | — | $V_{CC} \times 0.8$ | V | |
| | | VT ⁺ – VT | $V_{CC} \times 0.05$ | — | — | V | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| | | | $V_{CC} \times 0.04$ | — | — | V | $V_{CC} = 2.7\text{ V to }4.0\text{ V}$ |
| Input high voltage | RES, \overline{STBY} , NMI, MD2 to MD0, FWE | V_{IH} | $V_{CC} \times 0.9$ | — | $V_{CC} + 0.3$ | V | |
| | EXTAL Ports 1, 3, 7, A to G | | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4 and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3\text{ V}$ | V | |
| Input low voltage | RES, \overline{STBY} , MD2 to MD0, FWE | V_{IL} | –0.3 | — | $V_{CC} \times 0.1$ | V | |
| | NMI, EXTAL Ports 1, 3, 4, 7, 9, A to G | | –0.3 | — | $V_{CC} \times 0.2$ | V | |
| Output high voltage | All output pins except P34 and P35* ³ | V_{OH} | $V_{CC} - 0.5$ | — | — | V | $I_{OH} = -200\text{ }\mu\text{A}$ |
| | | | $V_{CC} - 1.0$ | — | — | V | $I_{OH} = -1\text{ mA}$ |
| | P34 and P35* ² | | $V_{CC} - 2.7$ | — | — | V | $I_{OH} = -100\text{ }\mu\text{A}$, $V_{CC} = 4.5\text{ V to }5.5\text{ V}$ |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|----------------------------|-------------|-----|-----|-----|---------------|--|
| Output low voltage | All output pins*3 | V_{OL} | — | — | 0.4 | V | $I_{OL} = 0.4 \text{ mA}$ |
| | | | — | — | 0.4 | V | $I_{OL} = 0.8 \text{ mA}$ |
| Input leakage current | \overline{RES} | $ I_{in} $ | — | — | 1.0 | μA | $V_{in} = 0.5 \text{ to } V_{CC} - 0.5 \text{ V}$ |
| | STBY, NMI, MD2 to MD0, FWE | | — | — | 1.0 | μA | |
| | Ports 4, 9 | | — | — | 1.0 | μA | $V_{in} = 0.5 \text{ to } AV_{CC} - 0.5 \text{ V}$ |
| Three-state leakage current (off state) | Ports 1, 3, 7, A to G | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.5 \text{ to } V_{CC} - 0.5 \text{ V}$ |
| Input pull-up MOS current | Ports A to E | $-I_p$ | 10 | — | 300 | μA | $V_{in} = 0 \text{ V}$ |

- Notes: 1. If the A/D and D/A converters are not used, do not leave the AV_{CC} , V_{ref} , and AV_{SS} pins open. Apply a voltage between 2.0 V and 5.5 V to the AV_{CC} and V_{ref} pins by connecting them to V_{CC} , for instance. Set $V_{ref} \leq AV_{CC}$.
2. P35/SCK1/SCL0 and P34/SDA0 are NMOS push-pull outputs.
 In order to output a high level from SCL0 and SDA0 ($ICE = 1$), a pull-up resistance must be connected externally.
 The high level of P35/SCK1 and P34 ($ICE = 0$) is driven by NMOS.
 In order to output a high level, a pull-up resistance must be connected externally.
3. This is the case when $IICE = 0$ and $ICE = 0$. Low-level output when the bus drive function is selected will be determined in table 27.29.

Table 27.27 DC Characteristics (2)

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V}$ to 5.5 V , $AV_{CC} = 3.6\text{ V}$ to 5.5 V ,
 $V_{ref} = 3.6\text{ V}$ to AV_{CC} , $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C}$ to $+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (wide-range specifications)*1

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------|---------------------------------|-------------|-----|-------------------------------|-------------------------------|---------------|---|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 30 | pF | $V_{in} = 0\text{ V}$, $f = 1\text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins except the above | | — | — | 15 | pF | |
| Current dissipation*2 | Normal operation | I_{CC} *4 | — | 23 $V_{CC} = 3.0\text{ V}$ | 40 $V_{CC} = 5.5\text{ V}$ | mA | $f = 13.5\text{ MHz}$ |
| | Sleep mode | | — | 18 $V_{CC} = 3.0\text{ V}$ | 30 $V_{CC} = 5.5\text{ V}$ | mA | $f = 13.5\text{ MHz}$ |
| | All modules stopped | | — | 13 | — | mA | $f = 13.5\text{ MHz}$, $V_{CC} = 3.0\text{ V}$ (reference values) |
| | Medium-speed mode ($\phi/32$) | | — | 13 | — | mA | $f = 13.5\text{ MHz}$, $V_{CC} = 3.0\text{ V}$ (reference values) |
| | Subactive mode | | — | 80 | 180 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 60 | 130 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Watch mode | | — | 8 | 40 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | | | | | | | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|-------------------------------|---------------|-----|--------------------------------|-------------------------------|---------------|--|
| Current dissipation ^{*2} | Standby mode ^{*3} | I_{CC}^{*4} | — | 1.0 $V_{CC} = 3.0\text{ V}$ | 10 $V_{CC} = 5.5\text{ V}$ | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | | — | — | 50 $V_{CC} = 5.5\text{ V}$ | | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D and D/A conversion | AI_{CC} | — | 0.3 | 1.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| Reference current | During A/D and D/A conversion | AI_{CC} | — | 1.3 | 3.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D and D/A converters are not used, do not leave the AV_{CC} , V_{ref} , and AV_{SS} pins open. Apply a voltage between 2.0 V and 5.5 V to the AV_{CC} and V_{ref} pins by connecting them to V_{CC} , for instance. Set $V_{ref} \leq AV_{CC}$.
2. Current dissipation values are for $V_{IH} \text{ min} = V_{CC} - 0.5\text{ V}$, $V_{IL} \text{ max} = 0.5\text{ V}$ with all output pins unloaded and the on-chip pull-up resistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 3.0\text{ V}$, $V_{IH} \text{ min} = V_{CC} \times 0.9$, and $V_{IL} \text{ max} = 0.3\text{ V}$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max} = 2.0\text{ (mA)} + 0.7\text{ (mA/V)} \times V_{CC} + 1.4\text{ (mA/MHz)} \times f + 0.20\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (normal operation)
 $I_{CC} \text{ max} = 1.5\text{ (mA)} + 0.6\text{ (mA/V)} \times V_{CC} + 1.1\text{ (mA/MHz)} \times f + 0.15\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (sleep mode)

Table 27.27 DC Characteristics (3)

Condition B (Masked ROM version): $V_{CC} = 2.7 \text{ V to } 5.5 \text{ V}$, $AV_{CC} = 3.6 \text{ V to } 5.5 \text{ V}$,
 $V_{ref} = 3.6 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*1

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------|---------------------------------|-------------|-----|---------------------------------|--------------------------------|---------------|--|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 30 | pF | $V_{in} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins except the above | | — | — | 15 | pF | |
| Current dissipation*2 | Normal operation | I_{CC} *4 | — | 22 $V_{CC} = 3.0 \text{ V}$ | 40 $V_{CC} = 5.5 \text{ V}$ | mA | $f = 13.5 \text{ MHz}$ |
| | Sleep mode | | — | 16 $V_{CC} = 3.0 \text{ V}$ | 30 $V_{CC} = 5.5 \text{ V}$ | mA | $f = 13.5 \text{ MHz}$ |
| | All modules stopped | | — | 13 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference values) |
| | Medium-speed mode ($\phi/32$) | | — | 13 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference values) |
| | Subactive mode | | — | 60 | 180 | μA | $V_{CC} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 35 | 100 | μA | $V_{CC} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used |
| | Watch mode | | — | 8 | 40 | μA | $V_{CC} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used |
| | Standby mode*3 | | — | 0.5 $V_{CC} = 3.0 \text{ V}$ | 10 $V_{CC} = 5.5 \text{ V}$ | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | | — | — | 50 $V_{CC} = 5.5 \text{ V}$ | | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------|-------------------------------|-----------|-----|------|-----|------|-----------------|
| Analog power supply current | During A/D and D/A conversion | I_{CC} | — | 0.3 | 1.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| Reference current | During A/D and D/A conversion | I_{CC} | — | 1.3 | 3.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D and D/A converters are not used, do not leave the AV_{CC} , V_{ref} , and AV_{SS} pins open. Apply a voltage between 2.0 V and 5.5 V to the AV_{CC} and V_{ref} pins by connecting them to V_{CC} , for instance. Set $V_{ref} \leq AV_{CC}$.
2. Current dissipation values are for $V_{IH} \text{ min} = V_{CC} - 0.5 \text{ V}$, $V_{IL} \text{ max} = 0.5 \text{ V}$ with all output pins unloaded and the on-chip pull-up resistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 2.7 \text{ V}$, $V_{IH} \text{ min} = V_{CC} \times 0.9$, and $V_{IL} \text{ max} = 0.3 \text{ V}$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max} = 2.0 \text{ (mA)} + 0.7 \text{ (mA/V)} \times V_{CC} + 1.4 \text{ (mA/MHz)} \times f + 0.20 \text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f \text{ (normal mode)}$
 $I_{CC} \text{ max} = 1.5 \text{ (mA)} + 0.6 \text{ (mA/V)} \times V_{CC} + 1.1 \text{ (mA/MHz)} \times f + 0.15 \text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f \text{ (sleep mode)}$

Table 27.28 Permissible Output Currents

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | | Symbol | Min | Typ | Max | Unit |
|---|----------------------------------|----------------|-----|-----|-----|------|
| Permissible output low current (per pin) | SCL1, SCL0, SDA1, SDA0 | I_{OL} | — | — | 10 | mA |
| | All output pins except the above | | — | — | 1.0 | |
| Permissible output low current (total) | Total of all output pins | $\sum I_{OL}$ | — | — | 60 | mA |
| Permissible output high current (per pin) | All output pins | $-I_{OH}$ | — | — | 1.0 | mA |
| Permissible output high current (total) | Total of all output pins | $\sum -I_{OH}$ | — | — | 30 | mA |

Note: To protect chip reliability, do not exceed the output current values in table 27.28.

Table 27.29 Bus Drive Characteristics

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (regular specifications),
 $T_a = -40^{\circ}\text{C to }+85^{\circ}\text{C}$ (wide-range specifications)*

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (regular specifications),
 $T_a = -40^{\circ}\text{C to }+85^{\circ}\text{C}$ (wide-range specifications)*

Applicable Pins: SCL1 and SCL0, SDA1 and SDA0

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|-------------------|----------------------|-----|---------------------|---------------|--|
| Schmitt trigger input voltage | VT^{-} | $V_{CC} \times 0.3$ | — | — | V | $V_{CC} = 2.7\text{ V to }5.5\text{ V}$ |
| | VT^{+} | — | — | $V_{CC} \times 0.7$ | | $V_{CC} = 2.7\text{ V to }5.5\text{ V}$ |
| | $VT^{+} - VT^{-}$ | 0.4 | — | — | | $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| | | $V_{CC} \times 0.05$ | — | — | | $V_{CC} = 2.7\text{ V to }4.0\text{ V}$ |
| Input high voltage | V_{IH} | $V_{CC} \times 0.7$ | — | $V_{CC} + 0.5$ | V | $V_{CC} = 2.7\text{ V to }5.5\text{ V}$ |
| Input low voltage | V_{IL} | -0.5 | — | $V_{CC} \times 0.3$ | V | $V_{CC} = 2.7\text{ V to }5.5\text{ V}$ |
| Output low voltage | V_{OL} | — | — | 0.5 | V | $I_{OL} = 8\text{ mA}$, $V_{CC} = 4.0\text{ V to }5.5\text{ V}$ |
| | | | | 0.4 | | $I_{OL} = 3\text{ mA}$ |
| Input capacitance | C_{in} | — | — | 20 | pF | $V_{in} = 0\text{ V}$, $f = 1\text{ MHz}$, $T_a = 25^{\circ}\text{C}$ |
| Three-state leakage current (off state) | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.5\text{ to }V_{CC} - 0.5\text{ V}$ |
| SCL, SDA output fall time | t_{Of} | $20 + 0.1 C_b$ | — | 250 | ns | $V_{CC} = 2.7\text{ V to }5.5\text{ V}$ |

Note: * If the A/D and D/A converters are not used, do not leave the AV_{CC} , V_{ref} , and AV_{SS} pins open. Apply a voltage between 2.0 V and 5.5 V to the AV_{CC} and V_{ref} pins by connecting them to V_{CC} , for instance. Set $V_{ref} \leq AV_{CC}$.

27.4.3 AC Characteristics

Figure 27.9 shows the test conditions for the AC characteristics.

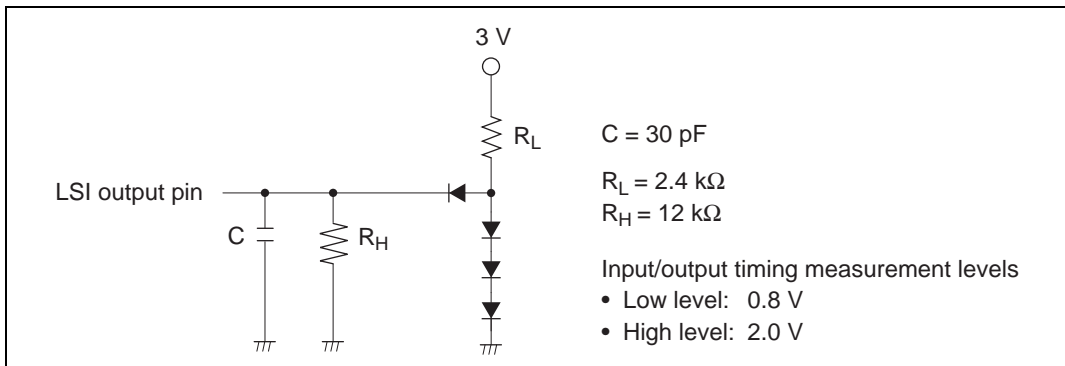


Figure 27.9 Output Load Circuit

(1) Clock Timing

Table 27.30 lists the clock timing

Table 27.30 Clock Timing

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz, }2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz, }2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Conditions A, B | | | Unit | Test Conditions |
|---|------------|-----------------|--------|-----|---------------|-----------------|
| | | Min | Typ | Max | | |
| Clock cycle time | t_{cyc} | 74 | — | 500 | ns | Figure 27.10 |
| Clock high pulse width | t_{CH} | 25 | — | — | ns | |
| Clock low pulse width | t_{CL} | 25 | — | — | ns | |
| Clock rise time | t_{Cr} | — | — | 10 | ns | |
| Clock fall time | t_{Cf} | — | — | 10 | ns | |
| Reset oscillation stabilization time (crystal) | t_{OSC1} | 20 | — | — | ms | Figure 27.11 |
| Software standby oscillation stabilization time (crystal) | t_{OSC2} | 8 | — | — | ms | |
| External clock output stabilization delay time | t_{DEXT} | 500 | — | — | μs | Figure 27.11 |
| Subclock oscillation stabilization time | t_{OSC3} | — | — | 2 | s | |
| Subclock oscillator frequency | f_{SUB} | — | 32.768 | — | kHz | |
| Subclock (ϕ_{SUB}) cycle time | t_{SUB} | — | 30.5 | — | μs | |

(2) Control Signal Timing

Table 27.31 lists the control signal timing.

Table 27.31 Control Signal Timing

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, $2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, $2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Conditions A, B | | Unit | Test Conditions |
|---|--------------------|-----------------|-----|------------------|-----------------|
| | | Min | Max | | |
| $\overline{\text{RES}}$ setup time | t_{RESS} | 250 | — | ns | Figure 27.12 |
| $\overline{\text{RES}}$ pulse width | t_{RESW} | 20 | — | t_{cyc} | |
| $\overline{\text{MRES}}$ setup time | t_{MRESS} | 250 | — | ns | Figure 27.13 |
| $\overline{\text{MRES}}$ pulse width | t_{MRESW} | 20 | — | t_{cyc} | |
| $\overline{\text{NMI}}$ setup time | t_{NMIS} | 250 | — | ns | |
| $\overline{\text{NMI}}$ hold time | t_{NMIH} | 10 | — | ns | |
| $\overline{\text{NMI}}$ pulse width (exiting software standby mode) | t_{NMIW} | 200 | — | ns | |
| $\overline{\text{IRQ}}$ setup time | t_{IRQS} | 250 | — | ns | |
| $\overline{\text{IRQ}}$ hold time | t_{IRQH} | 10 | — | ns | |
| $\overline{\text{IRQ}}$ pulse width (exiting software standby mode) | t_{IRQW} | 200 | — | ns | |

(3) Bus Timing

Table 27.32 lists the bus timing.

Table 27.32 Bus Timing

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Conditions A, B | | Unit | Test Conditions |
|------------------------------|------------|---------------------------|---------------------------|------|------------------------|
| | | Min | Max | | |
| Address delay time | t_{AD} | — | 50 | ns | Figures 27.14 to 27.18 |
| Address setup time | t_{AS} | $0.5 \times t_{cyc} - 30$ | — | ns | |
| Address hold time | t_{AH} | $0.5 \times t_{cyc} - 15$ | — | ns | |
| \overline{CS} delay time | t_{CSD} | — | 50 | ns | |
| \overline{AS} delay time | t_{ASD} | — | 50 | ns | |
| \overline{RD} delay time 1 | t_{RSD1} | — | 50 | ns | |
| \overline{RD} delay time 2 | t_{RSD2} | — | 50 | ns | |
| Read data setup time | t_{RDS} | 30 | — | ns | |
| Read data hold time | t_{RDH} | 0 | — | ns | |
| Read data access time 1 | t_{ACC1} | — | $1.0 \times t_{cyc} - 65$ | ns | |
| Read data access time 2 | t_{ACC2} | — | $1.5 \times t_{cyc} - 65$ | ns | |
| Read data access time 3 | t_{ACC3} | — | $2.0 \times t_{cyc} - 65$ | ns | |
| Read data access time 4 | t_{ACC4} | — | $2.5 \times t_{cyc} - 65$ | ns | |
| Read data access time 5 | t_{ACC5} | — | $3.0 \times t_{cyc} - 65$ | ns | |

| Item | Symbol | Conditions A, B | | Unit | Test Conditions |
|-------------------------------|------------|---------------------------|-----|------|------------------------|
| | | Min | Max | | |
| \overline{WR} delay time 1 | t_{WRD1} | — | 50 | ns | Figures 27.14 to 27.18 |
| \overline{WR} delay time 2 | t_{WRD2} | — | 50 | ns | |
| \overline{WR} pulse width 1 | t_{WSW1} | $1.0 \times t_{cyc} - 30$ | — | ns | |
| \overline{WR} pulse width 2 | t_{WSW2} | $1.5 \times t_{cyc} - 30$ | — | ns | |
| Write data delay time | t_{WDD} | — | 70 | ns | |
| Write data setup time | t_{WDS} | $0.5 \times t_{cyc} - 37$ | — | ns | |
| Write data hold time | t_{WDH} | $0.5 \times t_{cyc} - 15$ | — | ns | |
| \overline{WAIT} setup time | t_{WTS} | 50 | — | ns | |
| \overline{WAIT} hold time | t_{WTH} | 10 | — | ns | Figure 27.16 |
| \overline{BREQ} setup time | t_{BRQS} | 50 | — | ns | |
| \overline{BACK} delay time | t_{BACD} | — | 50 | ns | Figure 27.19 |
| Bus-floating time | t_{BZD} | — | 80 | ns | |

(4) Timing of On-Chip Peripheral Modules

Table 27.33 shows the timing of on-chip peripheral modules, and table 27.34 shows the I²C bus timing.

Table 27.33 Timing of On-Chip Peripheral Modules

Condition A (F-ZTAT version): $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, $2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (regular specifications),
 $T_a = -40^{\circ}\text{C to }+85^{\circ}\text{C}$ (wide-range specifications)

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$,
 $V_{ref} = 3.6\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, $2\text{ MHz to }13.5\text{ MHz}$,
 $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (regular specifications),
 $T_a = -40^{\circ}\text{C to }+85^{\circ}\text{C}$ (wide-range specifications)

| Item | | Symbol | Conditions A, B | | Unit | Test Conditions |
|-----------|------------------------------|----------------------------|-----------------|-----|-----------|-----------------|
| | | | Min | Max | | |
| I/O port* | Output data delay time | t_{PWD} | — | 100 | ns | Figure 27.24 |
| | Input data setup time | t_{PRS} | 50 | — | | |
| | Input data hold time | t_{PRH} | 50 | — | | |
| TPU | Timer output delay time | t_{TOD} | — | 100 | ns | Figure 27.25 |
| | Timer input setup time | t_{TICS} | 40 | — | | |
| | Timer clock input setup time | t_{TCKS} | 40 | — | ns | Figure 27.26 |
| | Timer clock pulse width | Single edge t_{TCKWH} | 1.5 | — | t_{cyc} | |
| | | Both edges t_{TCKWL} | 2.5 | — | | |
| TMR | Timer output delay time | t_{TMOD} | — | 100 | ns | Figure 27.27 |
| | Timer reset input setup time | t_{TMRS} | 50 | — | ns | Figure 27.29 |
| | Timer clock input setup time | t_{TMCS} | 50 | — | ns | Figure 27.28 |
| | Timer clock pulse width | Single edge t_{TMCWH} | 1.5 | — | t_{cyc} | |
| | | Both edges t_{TMCWL} | 2.5 | — | | |

| Item | | | Symbol | Conditions A, B | | Unit | Test Conditions |
|---------------|---------------------------------------|--------------|-------------------|-----------------|-----|-------------------|-----------------|
| | | | | Min | Max | | |
| WDT1 | BUZZ output delay time | | t _{BUZD} | — | 100 | ns | Figure 27.30 |
| SCI* | Input clock cycle | Asynchronous | t _{Scyc} | 4 | — | t _{cyc} | Figure 27.31 |
| | | Synchronous | | 6 | — | | |
| | Input clock pulse width | | t _{SCKW} | 0.4 | 0.6 | t _{Scyc} | Figure 27.32 |
| | Input clock rise time | | t _{SCKr} | — | 1.5 | t _{cyc} | |
| | Input clock fall time | | t _{SCKf} | — | 1.5 | | |
| | Transmit data delay time | | t _{TXD} | — | 100 | ns | |
| | Receive data setup time (synchronous) | | t _{RXS} | 75 | — | ns | |
| | Receive data hold time (synchronous) | | t _{RXH} | 75 | — | ns | |
| A/D converter | Trigger input setup time | | t _{TRGS} | 40 | — | ns | Figure 27.33 |

Note: * The high level of P35/SCK1 and P34 is driven by NMOS. In order to output a high level at V_{CC} = 4.5 V or below, a pull-up resistance must be connected externally.

Table 27.34 I²C Bus Timing

Condition A (F-ZTAT version): $V_{CC} = 3.0 \text{ V to } 5.5 \text{ V}$, $V_{SS} = 0 \text{ V}$, $\phi = 5 \text{ MHz}$ to maximum operating frequency, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$

Condition B (Masked ROM version): $V_{CC} = 2.7 \text{ V to } 5.5 \text{ V}$, $V_{SS} = 0 \text{ V}$, $\phi = 5 \text{ MHz}$ to maximum operating frequency, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$

| Item | Symbol | Conditions A, B | | | Unit | Test Conditions |
|---|------------|-----------------|-----|-----------------|------|-----------------|
| | | Min | Typ | Max | | |
| SCL input cycle time | t_{SCL} | 12 t_{cyc} | — | — | ns | Figure 27.34 |
| SCL input high pulse width | t_{SCLH} | 3 t_{cyc} | — | — | ns | |
| SCL input low pulse width | t_{SCLL} | 5 t_{cyc} | — | — | ns | |
| SCL, SDA input rise time | t_{Sr} | — | — | 7.5 t_{cyc}^* | ns | |
| SCL, SDA input fall time | t_{Sf} | — | — | 300 | ns | |
| SCL, SDA input spike pulse elimination time | t_{SP} | — | — | 1 t_{cyc} | ns | |
| SDA input bus free time | t_{BUF} | 5 t_{cyc} | — | — | ns | |
| Start condition input hold time | t_{STAH} | 3 t_{cyc} | — | — | ns | |
| Retransmission start condition input setup time | t_{STAS} | 3 t_{cyc} | — | — | ns | |
| Stop condition input setup time | t_{STOS} | 3 t_{cyc} | — | — | ns | |
| Data input setup time | t_{SDAS} | 0.5 t_{cyc} | — | — | ns | |
| Data input hold time | t_{SDAH} | 0 | — | — | ns | |
| SCL, SDA capacitive load | C_b | — | — | 400 | pF | |

Note: * 7.5 t_{cyc} and 17.5 t_{cyc} can be set according to the clock selected for use by the I²C module. For details, see section 16.6, Usage Notes.

27.4.4 A/D Conversion Characteristics

A/D converter characteristics for the F-ZTAT and masked ROM versions are shown in table 27.35.

Table 27.35 A/D Conversion Characteristics (F-ZTAT and Masked ROM Versions)

Condition: $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$, $V_{ref} = 3.6\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz to }13.5\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Condition | | | Unit |
|-------------------------------------|-----------|-----|-----------|---------------|
| | Min | Typ | Max | |
| Resolution | 10 | 10 | 10 | bit |
| Conversion time | 9.6 | — | — | μs |
| Analog input capacitance | — | — | 20 | pF |
| Permissible signal-source impedance | — | — | 5 | k Ω |
| Nonlinearity error | — | — | ± 6.0 | LSB |
| Offset error | — | — | ± 4.0 | LSB |
| Full-scale error | — | — | ± 4.0 | LSB |
| Quantization | — | — | ± 0.5 | LSB |
| Absolute accuracy | — | — | ± 8.0 | LSB |

27.4.5 D/A Conversion Characteristics

Table 27.36 lists the D/A conversion characteristics.

Table 27.36 D/A Conversion Characteristics (F-ZTAT and Masked ROM Versions)

Condition: $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.6\text{ V to }5.5\text{ V}$, $V_{ref} = 3.6\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ MHz to }13.5\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Condition | | | Unit | Test Conditions |
|-------------------|-----------|-----------|-----------|---------------|-----------------------------|
| | Min | Typ | Max | | |
| Resolution | 8 | 8 | 8 | bit | |
| Conversion time | — | — | 10 | μs | 20-pF capacitive load |
| Absolute accuracy | — | ± 2.0 | ± 3.0 | LSB | 2-M Ω resistive load |
| | — | — | ± 2.0 | LSB | 4-M Ω resistive load |

27.4.6 Flash Memory Characteristics

Table 27.37 lists the flash memory characteristics.

Table 27.37 Flash Memory Characteristics

Conditions: $V_{CC} = 3.0\text{ V to }5.5\text{ V}$, $AV_{CC} = 3.0\text{ V to }5.5\text{ V}$, $V_{ref} = 3.0\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (program/erase operating temperature range;
 regular specifications), $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$ (program/erase operating temperature
 range; wide-range specifications)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|--|-------------------|---------------------|------|-------------------|------------------------------------|
| Programming time ^{*1*2*4} | t_p | — | 10 | 200 | ms/ 128 bytes | |
| Erase time ^{*1*3*5} | t_E | — | 100 | 1200 | ms/block | |
| Rewrite times | N_{WEC} | 100 ^{*6} | 10000 ^{*7} | — | Times | |
| Data holding time ^{*8} | t_{DRP} | 10 | — | — | Years | |
| Programming | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs |
| | Wait time after PSU1 bit setting ^{*1} | t_{spsu} | 50 | 50 | — | μs |
| | Wait time after P1 bit setting ^{*1*4} | t_{sp10} | 8 | 10 | 12 | μs |
| | | t_{sp30} | 28 | 30 | 32 | μs $1 \leq n \leq 6$ |
| | | t_{sp200} | 198 | 200 | 202 | μs $7 \leq n \leq 1000$ |
| | Wait time after P1 bit clearing ^{*1} | t_{cp} | 5 | 5 | — | μs |
| | Wait time after PSU1 bit clearing ^{*1} | t_{cpsu} | 5 | 5 | — | μs |
| | Wait time after PV1 bit setting ^{*1} | t_{spv} | 4 | 4 | — | μs |
| | Wait time after H'FF dummy write ^{*1} | t_{spvr} | 2 | 2 | — | μs |
| | Wait time after PV1 bit clearing ^{*1} | t_{cpv} | 2 | 2 | — | μs |
| | Wait time after SWE1 bit clearing | t_{cswe} | 100 | 100 | — | μs |
| | Maximum number of programming operations ^{*1*4} | N1 | — | — | 6 ^{*4} | Times |
| | | N2 | — | — | 994 ^{*4} | |
| Erasing | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs |
| | Wait time after ESU1 bit setting ^{*1} | t_{sesu} | 100 | 100 | — | μs |
| | Wait time after E1 bit setting ^{*1*5} | t_{se} | 10 | 10 | 100 | ms |
| | Wait time after E1 bit clearing ^{*1} | t_{ce} | 10 | 10 | — | μs |
| | Wait time after ESU1 bit clearing ^{*1} | t_{cesu} | 10 | 10 | — | μs |
| | Wait time after EV1 bit setting ^{*1} | t_{sev} | 20 | 20 | — | μs |
| | Wait time after H'FF dummy write ^{*1} | t_{sevr} | 2 | 2 | — | μs |
| | Wait time after EV1 bit clearing ^{*1} | t_{cev} | 4 | 4 | — | μs |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---------|--|------------|-----|-----|-----|---------|-----------------|
| Erasing | Wait time after SWE1 bit clearing | t_{cswc} | 100 | 100 | — | μs | |
| | Maximum number of erases ^{*1*5} | N | — | — | 100 | Times | |

- Notes: 1. Follow the program/erase algorithms when making the time settings.
2. Programming time per 128 bytes (Indicates the total time during which the P1 bit is set in flash memory control register 1 (FLMCR1). Does not include the program-verify time.)
3. Time to erase one block (Indicates the time during which the E1 bit is set in FLMCR1. Does not include the erase-verify time.)
4. Maximum programming time
 $(t_p(\max) = \text{Wait time after P1 bit setting } (t_{sp}) \times \text{maximum number of writes (N)})$
 $(t_{sp30} + t_{sp10}) \times 6 + (t_{sp200}) \times 994$
5. For the maximum erase time ($t_e(\max)$), the following relationship applies between the wait time after E1 bit setting (z) and the maximum number of erases (N):
 $t_e(\max) = \text{Wait time after E1 bit setting } (t_{se}) \times \text{maximum number of erases (N)}$
6. The minimum times that all characteristics after rewriting are guaranteed. (A range between 1 and minimum value is guaranteed.)
7. The reference value at 25°C. (Normally, it is a reference that rewriting is enabled up to this value.)
8. Data hold characteristics when rewriting is performed within the range of specifications including minimum value.

27.5 Electrical Characteristics of H8S/2238R and H8S/2236R

27.5.1 Absolute Maximum Ratings

Table 27.38 lists the absolute maximum ratings.

Table 27.38 Absolute Maximum Ratings

| Item | Symbol | Value | Unit |
|--------------------------------------|-----------|---|------|
| Power supply voltage | V_{CC} | −0.3 to +4.3 | V |
| | CV_{CC} | −0.3 to +4.3 | V |
| Input voltage (except ports 4 and 9) | V_{in} | −0.3 to $V_{CC} + 0.3$ | V |
| Input voltage (ports 4 and 9) | V_{in} | −0.3 to $AV_{CC} + 0.3$ | V |
| Reference power supply voltage | V_{ref} | −0.3 to $AV_{CC} + 0.3$ | V |
| Analog power supply voltage | AV_{CC} | −0.3 to +4.3 | V |
| Analog input voltage | V_{AN} | −0.3 to $AV_{CC} + 0.3$ | V |
| Operating temperature | T_{opr} | Regular specifications: −20 to +75 ^{*1} | °C |
| | | Wide-range specifications: −40 to +85 ^{*2} | |
| Storage temperature | T_{stg} | −55 to +125 | °C |

Caution: Permanent damage to the chip may result if absolute maximum rating are exceeded.

- Notes: 1. When the operating voltage in read is $V_{CC} = 2.7$ V to 3.6 V, the operating temperature ranges for flash memory programming/erasing are $T_a = -20^{\circ}\text{C}$ to $+75^{\circ}\text{C}$. When the operating voltage in read is $V_{CC} = 2.2$ V to 3.6 V, the operating temperature ranges for flash memory programming/erasing are $T_a = -20^{\circ}\text{C}$ to $+50^{\circ}\text{C}$.
2. The operating temperature ranges for flash memory programming/erasing are $T_a = -40^{\circ}\text{C}$ to $+80^{\circ}\text{C}$ (regular specifications).

27.5.2 DC Characteristics

Table 27.39 lists the DC characteristics. Table 27.40 lists the permissible output currents. Table 27.41 lists the bus driving characteristics.

Table 27.39 DC Characteristics (1)

| | |
|-----------------------------------|--|
| Condition A (F-ZTAT version): | $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications) $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*1 |
| Condition B (F-ZTAT version): | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications) |
| Condition C (Masked ROM version): | $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$, $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications), $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*1 |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--|---|---------------|----------------------|-----|----------------------|------|-----------------|
| Schmitt trigger IRQ0 to IRQ7 input voltage | | VT^- | $V_{CC} \times 0.2$ | — | — | V | |
| | | VT^+ | — | — | $V_{CC} \times 0.8$ | V | |
| | | $VT^+ - VT^-$ | $V_{CC} \times 0.05$ | — | — | V | |
| Input high voltage | \overline{RES} , \overline{STBY} , NMI, FWE, MD2 to MD0 | V_{IH} | $V_{CC} \times 0.9$ | — | $V_{CC} + 0.3$ | V | |
| | EXTAL, Ports 1, 3, 7, and A to G | | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4*5 and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3^{*5}$ | V | |
| Input low voltage | \overline{RES} , \overline{STBY} , FWE, MD2 to MD0 | V_{IL} | -0.3 | — | $V_{CC} \times 0.1$ | V | |
| | NMI, EXTAL, Ports 1, 3, 4, 7, 9, and A to G | | -0.3 | — | $V_{CC} \times 0.2$ | V | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|--|-------------|----------------|-----|-----|---------|--|
| Output high voltage | All output pins* ⁴ except P34 and P35 | V_{OH} | $V_{CC} - 0.5$ | — | — | V | $I_{OH} = -200 \mu A$ |
| | | | $V_{CC} - 1.0$ | — | — | V | $I_{OH} = -1 \text{ mA}^{*2}$ |
| | P34 and P35* ³ | | $V_{CC} - 2.0$ | — | — | V | $I_{OH} = -100 \mu A$ (reference value) |
| Output low voltage | All output pins* ⁴ | V_{OL} | — | — | 0.4 | V | $I_{OL} = 0.4 \text{ mA}$ |
| | | | — | — | 0.4 | V | $I_{OL} = 0.8 \text{ mA}^{*2}$ |
| Input leakage current | \overline{RES} | $ I_{in} $ | — | — | 1.0 | μA | $V_{in} = 0.2 \text{ to } V_{CC} - 0.2 \text{ V}$ |
| | \overline{STBY} , NMI, FWE, MD2 to MD0 | | — | — | 1.0 | μA | |
| | Ports 4, 9 | | — | — | 1.0 | μA | $V_{in} = 0.2 \text{ to } AV_{CC} - 0.2 \text{ V}$ |
| Three states leakage current (off) | Ports 1, 3, 7, and A to G | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.2 \text{ to } V_{CC} - 0.2 \text{ V}$ |
| Input pull-up MOS current | Ports A to E | $-I_p$ | 10 | — | 300 | μA | $V_{in} = 0V$ |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.
2. $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$
3. P35/SCK1 and P34 function as NMOS push-pull output. To output the high voltage, connect an external pull-up resistor.
4. In the case when ICE = 0. Low voltage output with bus driving function is specified in table 27.41.
5. When $V_{CC} < AV_{CC}$, the maximum value for P40 and P41 is $V_{CC} + 0.3 \text{ V}$.

Table 27.39 DC Characteristics (2)

Condition A (F-ZTAT version): $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications)
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*¹

Condition B (F-ZTAT version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|--------------------------|--------------------------|---------------|---|
| Input capacitance | RES | C_{in} | — | — | 30 | pF | $V_{in} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 20 | 37 | mA | $f = 13.5 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | Sleep mode | | — | 10 | 18 | mA | $f = 6.25 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 15 | 29 | mA | $f = 13.5 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 7.5 | 14 | mA | $f = 6.25 \text{ MHz}$ |
| | | | | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | All modules stopped | | — | 15 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 13 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | Subactive mode | | — | 70 | 180 | μA | $V_{CC} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 50 | 130 | μA | $V_{CC} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|----------------------------|---------------|-----|--------------------------------|-------------------------------|---------------|---|
| Current consumption ^{*2} | Watch mode | I_{CC}^{*4} | — | 8 | 40 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Standby mode ^{*3} | | — | 1.0 $V_{CC} = 3.0\text{ V}$ | 10 $V_{CC} = 3.6\text{ V}$ | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | | — | — | 50 $V_{CC} = 3.6\text{ V}$ | μA | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D conversion | AI_{CC} | — | 0.5 | 1.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D conversion | AI_{CC} | — | 1.3 | 2.5 | mA | |
| | Idle | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{CC} - 0.2\text{ V}$ and $V_{IL} \text{ max} = 0.2\text{ V}$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 2.2\text{ V}$, $V_{IH} \text{ min} = V_{CC} - 0.2$, and $V_{IL} \text{ max} = 0.2\text{ V}$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max} = 1.0\text{ (mA)} + 0.74\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (normal operation)
 $I_{CC} \text{ max} = 1.0\text{ (mA)} + 0.58\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (sleep mode)

Table 27.39 DC Characteristics (3)

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*1

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------|--------------------------------------|-------------|-----|--------------------------|--------------------------|---------------|---|
| Input capacitance | RES | C_{in} | — | — | 30 | pF | $V_{in} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_a = 25^\circ\text{C}$ |
| | NMI | | — | — | 30 | pF | |
| | P32 to P35 | | — | — | 20 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption*2 | Normal operation | I_{CC} *4 | — | 20 | 37 | mA | $f = 13.5 \text{ MHz}$ |
| | | | — | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 10 | 18 | mA | $f = 6.25 \text{ MHz}$ |
| | | | — | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | Sleep mode | | — | 15 | 29 | mA | $f = 13.5 \text{ MHz}$ |
| | | | — | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | | | — | 7.5 | 14 | mA | $f = 6.25 \text{ MHz}$ |
| | | | — | $V_{CC} = 3.0 \text{ V}$ | $V_{CC} = 3.6 \text{ V}$ | | |
| | All modules stopped | | — | 15 | — | mA | $f = 13.5 \text{ MHz}$, $V_{CC} = 3.0 \text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 13 | — | | |
| | Subactive mode | | — | 45 | 180 | μA | $V_{CC} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|----------------------------|---------------|-----|------------------|------------------|---------|---|
| Current consumption ^{*2} | Subsleep mode | I_{CC}^{*4} | — | 30 | 100 | μA | $V_{CC} = 3.0 V$, When 32.768 kHz crystal resonator is used |
| | Watch mode | — | — | 8 | 40 | μA | $V_{CC} = 3.0 V$, When 32.768 kHz crystal resonator is used |
| | Standby mode ^{*3} | — | — | 0.5 | 10 | μA | $T_a \leq 50^\circ C$, When 32.768 kHz crystal resonator is not used |
| | | — | — | $V_{CC} = 3.0 V$ | $V_{CC} = 3.6 V$ | μA | $50^\circ C < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D conversion | AI_{CC} | — | 0.5 | 1.5 | mA | |
| | Idle | — | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D conversion | AI_{CC} | — | 1.3 | 2.5 | mA | |
| | Idle | — | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{CC} - 0.2 V$ and $V_{IL} \text{ max} = 0.2 V$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 2.2 V$, $V_{IH} \text{ min} = V_{CC} - 0.2$, and $V_{IL} \text{ max} = 0.2 V$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max} = 1.0 \text{ (mA)} + 0.74 \text{ (mA/(MHz} \times V)) \times V_{CC} \times f$ (normal operation)
 $I_{CC} \text{ max} = 1.0 \text{ (mA)} + 0.58 \text{ (mA/(MHz} \times V)) \times V_{CC} \times f$ (sleep mode)

Table 27.40 Permissible Output Currents

Condition A (F-ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (F-ZTAT version): $V_{CC} = 2.2\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.2\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.2\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | | Symbol | Min | Typ | Max | Unit |
|---|-----------------------------------|---|------------------|-----|-----|------|
| Permissible output low current (per pin) | SCL1, SCL0, SDA1, SDA0 | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ | I_{OL} | — | 10 | mA |
| | Output pins other than above ones | $V_{CC} = 2.2\text{ V to }3.6\text{ V}$ | I_{OL} | — | 0.5 | |
| | | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ | | — | 1.0 | |
| Permissible output low current (total) | Total of all output pins | $V_{CC} = 2.2\text{ V to }3.6\text{ V}$ | ΣI_{OL} | — | 30 | mA |
| | | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ | | — | 60 | |
| Permissible output high current (per pin) | All output pins | $V_{CC} = 2.2\text{ V to }3.6\text{ V}$ | $-I_{OH}$ | — | 0.5 | mA |
| | | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ | | — | 1.0 | |
| Permissible output high current (total) | Total of all output pins | $V_{CC} = 2.2\text{ V to }3.6\text{ V}$ | $\Sigma -I_{OH}$ | — | 15 | mA |
| | | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ | | — | 30 | |

Note: To protect chip reliability, do not exceed the output current values in table 27.40.

Table 27.41 Bus Driving Characteristics

Conditions: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{ref} = 2.7\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)*,
Objective pins: SCL1 and 0 and SDA1 and 0

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|---------------|----------------------|-----|---------------------|---------------|---|
| Schmitt trigger input voltage | VT^- | $V_{CC} \times 0.3$ | — | — | V | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ |
| | VT^+ | — | — | $V_{CC} \times 0.7$ | V | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ |
| | $VT^+ - VT^-$ | $V_{CC} \times 0.05$ | — | — | V | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ |
| Input high voltage | V_{IH} | $V_{CC} \times 0.7$ | — | $V_{CC} + 0.5$ | V | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ |
| Input low voltage | V_{IL} | -0.5 | — | $V_{CC} \times 0.3$ | V | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ |
| Output low voltage | V_{OL} | — | — | 0.5 | V | $I_{OL} = 6\text{ mA}$, $V_{CC} = 3.0\text{ V to }3.6\text{ V}$ |
| | | — | — | 0.4 | V | $I_{OL} = 3\text{ mA}$ |
| Input capacitance | C_{in} | — | — | 20 | pF | $V_{in} = 0\text{ V}$ $f = 1\text{ MHz}$ $T_a = 25^\circ\text{C}$ |
| Three states leakage current (off) | $ I_{TSI} $ | — | — | 1.0 | μA | $V_{in} = 0.5\text{ V to }V_{CC} - 0.5\text{ V}$ |
| SCL, SDA output falling time | t_{of} | $20 + 0.1 C_b$ | — | 250 | ns | $V_{CC} = 2.7\text{ V to }3.6\text{ V}$ |

Note: * If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.

27.5.3 AC Characteristics

Figure 27.8 shows the test conditions for the AC characteristics.

(1) Clock Timing

Table 27.42 lists the clock timing.

Table 27.42 Clock Timing

Condition A (F-ZTAT version and masked ROM version):

$$V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V},$$

$$V_{ref} = 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V},$$

$$\phi = 32.768 \text{ kHz}, 2 \text{ to } 13.5 \text{ MHz},$$

$$T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)},$$

$$T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$$

Condition B (F-ZTAT version):

$$V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$$

$$V_{ref} = 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V},$$

$$\phi = 32.768 \text{ kHz}, 2 \text{ to } 6.25 \text{ MHz},$$

$$T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}$$

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$

$$V_{ref} = 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V},$$

$$\phi = 32.768 \text{ kHz}, 2 \text{ to } 6.25 \text{ MHz},$$

$$T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)},$$

$$T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$$

| Item | Symbol | Condition A | | | Conditions B, C | | | Unit | Test Conditions |
|--|------------|-------------|--------|-----|-----------------|--------|-----|---------|-----------------|
| | | Min | Typ | Max | Min | Typ | Max | | |
| Clock cycle time | t_{cyc} | 74 | — | 500 | 160 | — | 500 | ns | Figure 27.10 |
| Clock high pulse width | t_{CH} | 25 | — | — | 50 | — | — | ns | |
| Clock low pulse width | t_{CL} | 25 | — | — | 50 | — | — | ns | |
| Clock rise time | t_{Cr} | — | — | 10 | — | — | 25 | ns | |
| Clock fall time | t_{Cf} | — | — | 10 | — | — | 25 | ns | |
| Oscillation stabilization time at reset (crystal) | t_{OSC1} | 20 | — | — | 40 | — | — | ms | Figure 27.11 |
| Oscillation stabilization time in software standby (crystal) | t_{OSC2} | 8 | — | — | 16 | — | — | ms | |
| External clock output stabilization delay time | t_{DEXT} | 500 | — | — | 1000 | — | — | μ s | Figure 27.11 |
| Subclock oscillation stabilization time | t_{OSC3} | — | — | 2 | — | — | 4 | s | |
| Subclock oscillator frequency | f_{SUB} | — | 32.768 | — | — | 32.768 | — | kHz | |
| Subclock (ϕ_{SUB}) cycle time | t_{SUB} | — | 30.5 | — | — | 30.5 | — | μ s | |

(2) Control Signal Timing

Table 27.43 lists the control signal timing.

Table 27.43 Control Signal Timing

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V,} \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 13.5 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition B (F-ZTAT version):

$$\begin{aligned} V_{CC} &= 2.2 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V,} \\ V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V,} \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 6.25 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V, } AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V,}$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V,} \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 6.25 \text{ MHz,} \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications),} \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Symbol | Condition A | | Conditions B, C | | Unit | Test Conditions |
|---|--------------------|-------------|-----|-----------------|-----|------------------|-----------------|
| | | Min | Max | Min | Max | | |
| $\overline{\text{RES}}$ setup time | t_{RESS} | 250 | — | 350 | — | ns | Figure 27.12 |
| $\overline{\text{RES}}$ pulse width | t_{RESW} | 20 | — | 20 | — | t_{cyc} | |
| $\overline{\text{MRES}}$ setup time | t_{MRESS} | 250 | — | 350 | — | ns | Figure 27.13 |
| $\overline{\text{MRES}}$ pulse width | t_{MRESW} | 20 | — | 20 | — | t_{cyc} | |
| NMI setup time | t_{NMIS} | 250 | — | 350 | — | ns | |
| NMI hold time | t_{NMIH} | 10 | — | 10 | — | ns | |
| NMI pulse width (exiting software standby mode) | t_{NMIW} | 200 | — | 300 | — | ns | |
| $\overline{\text{IRQ}}$ setup time | t_{IRQS} | 250 | — | 350 | — | ns | |
| $\overline{\text{IRQ}}$ hold time | t_{IRQH} | 10 | — | 10 | — | ns | |
| $\overline{\text{IRQ}}$ pulse width (exiting software standby mode) | t_{IRQW} | 200 | — | 300 | — | ns | |

(3) Bus Timing

Table 27.44 lists the bus timing.

Table 27.44 Bus Timing

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 13.5 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition B (F-ZTAT version):

$$\begin{aligned} V_{CC} &= 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Symbol | Condition A | | Conditions B, C | | Unit | Test Conditions |
|------------------------------|------------|---------------------------|---------------------------|---------------------------|---------------------------|------|------------------------|
| | | Min | Max | Min | Max | | |
| Address delay time | t_{AD} | — | 50 | — | 90 | ns | Figures 27.14 to 27.18 |
| Address setup time | t_{AS} | $0.5 \times t_{cyc} - 30$ | — | $0.5 \times t_{cyc} - 60$ | — | ns | |
| Address hold time | t_{AH} | $0.5 \times t_{cyc} - 15$ | — | $0.5 \times t_{cyc} - 30$ | — | ns | |
| \overline{CS} delay time | t_{CSD} | — | 50 | — | 90 | ns | |
| \overline{AS} delay time | t_{ASD} | — | 50 | — | 90 | ns | |
| \overline{RD} delay time 1 | t_{RSD1} | — | 50 | — | 90 | ns | |
| \overline{RD} delay time 2 | t_{RSD2} | — | 50 | — | 90 | ns | |
| Read data setup time | t_{RDS} | 30 | — | 50 | — | ns | |
| Read data hold time | t_{RDH} | 0 | — | 0 | — | ns | |
| Read data access time 1 | t_{ACC1} | — | $1.0 \times t_{cyc} - 65$ | — | $1.0 \times t_{cyc} - 90$ | ns | |

| Item | Symbol | Condition A | | Conditions B, C | | Unit | Test Conditions |
|-------------------------------|------------|------------------------------|------------------------------|------------------------------|------------------------------|------|------------------------|
| | | Min | Max | Min | Max | | |
| Read data access time 2 | t_{ACC2} | — | $1.5 \times t_{cyc}$ – 65 | — | $1.5 \times t_{cyc}$ – 90 | ns | Figures 27.14 to 27.18 |
| Read data access time 3 | t_{ACC3} | — | $2.0 \times t_{cyc}$ – 65 | — | $2.0 \times t_{cyc}$ – 90 | ns | |
| Read data access time 4 | t_{ACC4} | — | $2.5 \times t_{cyc}$ – 65 | — | $2.5 \times t_{cyc}$ – 90 | ns | |
| Read data access time 5 | t_{ACC5} | — | $3.0 \times t_{cyc}$ – 65 | — | $3.0 \times t_{cyc}$ – 90 | ns | |
| \overline{WR} delay time 1 | t_{WRD1} | — | 50 | — | 90 | ns | |
| \overline{WR} delay time 2 | t_{WRD2} | — | 50 | — | 90 | ns | Figure 27.16 |
| \overline{WR} pulse width 1 | t_{WSW1} | $1.0 \times t_{cyc}$ – 30 | — | $1.0 \times t_{cyc}$ – 60 | — | ns | |
| \overline{WR} pulse width 2 | t_{WSW2} | $1.5 \times t_{cyc}$ – 30 | — | $1.5 \times t_{cyc}$ – 60 | — | ns | |
| Write data delay time | t_{WDD} | — | 70 | — | 100 | ns | |
| Write data setup time | t_{WDS} | $0.5 \times t_{cyc}$ – 37 | — | $0.5 \times t_{cyc}$ – 80 | — | ns | |
| Write data hold time | t_{WDH} | $0.5 \times t_{cyc}$ – 15 | — | $0.5 \times t_{cyc}$ – 60 | — | ns | Figure 27.19 |
| \overline{WAIT} setup time | t_{WTS} | 50 | — | 90 | — | ns | |
| \overline{WAIT} hold time | t_{WTH} | 10 | — | 10 | — | ns | |
| \overline{BREQ} setup time | t_{BRQS} | 50 | — | 90 | — | ns | |
| \overline{BACK} delay time | t_{BACD} | — | 50 | — | 90 | ns | |
| Bus-floating time | t_{BZD} | — | 80 | — | 160 | ns | |

(4) Timing of On-Chip Peripheral Modules

Table 27.45 lists the timing of on-chip peripheral modules. Table 27.46 lists the I²C bus timing.

Table 27.45 Timing of On-Chip Peripheral Modules

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 13.5 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition B (F-ZTAT version):

$$\begin{aligned} V_{CC} &= 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 32.768 \text{ kHz, } 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | | Symbol | Condition A | | Conditions B, C | | Unit | Test Conditions | |
|-----------|------------------------------|-------------------|--------------------|-----|-----------------|-----|------|-----------------|------------------|
| | | | Min | Max | Min | Max | | | |
| I/O port* | Output data delay time | t _{PWD} | — | 100 | — | 150 | ns | Figure 27.24 | |
| | Input data setup time | t _{PRS} | 50 | — | 80 | — | | | |
| | Input data hold time | t _{PRH} | 50 | — | 80 | — | | | |
| TPU | Timer output delay time | t _{TOCD} | — | 100 | — | 150 | ns | Figure 27.25 | |
| | Timer input setup time | t _{TICS} | 40 | — | 60 | — | | | |
| | Timer clock input setup time | t _{TCKS} | 40 | — | 60 | — | ns | Figure 27.26 | |
| | Timer clock pulse width | Single edge | t _{TCKWH} | 1.5 | — | 1.5 | — | | t _{cyc} |
| | | Both edges | t _{TCKWL} | 2.5 | — | 2.5 | — | | |
| TMR | Timer output delay time | t _{TMOD} | — | 100 | — | 150 | ns | Figure 27.27 | |
| | Timer reset input setup time | t _{TMRS} | 50 | — | 80 | — | ns | Figure 27.29 | |
| | Timer clock input setup time | t _{TMCS} | 50 | — | 80 | — | ns | Figure 27.28 | |

| Item | | Symbol | Condition A | | Conditions B, C | | Unit | Test Conditions | |
|---------------|---------------------------------------|--------------|-------------|-----|-----------------|-----|------------|-----------------|--------------|
| | | | Min | Max | Min | Max | | | |
| TMR | Timer clock pulse width | Single edge | t_{TMCWH} | 1.5 | — | 1.5 | — | t_{cyc} | Figure 27.28 |
| | | Both edges | t_{TMCWL} | 2.5 | — | 2.5 | — | | |
| WDT_1 | BUZZ output delay time | t_{BUZD} | — | 100 | — | 150 | ns | Figure 27.30 | |
| SCI* | Input clock cycle | Asynchronous | t_{Scyc} | 4 | — | 4 | — | t_{cyc} | Figure 27.31 |
| | | Synchronous | | 6 | — | 6 | — | | |
| | Input clock pulse width | t_{SCKW} | 0.4 | 0.6 | 0.4 | 0.6 | t_{Scyc} | | |
| | Input clock rise time | t_{SCKr} | — | 1.5 | — | 1.5 | t_{cyc} | | |
| | Input clock fall time | t_{SCKf} | — | 1.5 | — | 1.5 | | | |
| | Transmit data delay time | t_{TXD} | — | 100 | — | 150 | ns | Figure 27.32 | |
| | Receive data setup time (synchronous) | t_{RXS} | 75 | — | 150 | — | ns | | |
| | Receive data hold time (synchronous) | t_{RXH} | 75 | — | 150 | — | ns | | |
| A/D converter | Trigger input setup time | t_{TRGS} | 40 | — | 60 | — | ns | Figure 27.33 | |

Note: * NMOS controls P35/SCK1 and P34 to output the high voltage. To output the high voltage from P35/SCK1 and P34, connect an external pull-up resistor.

Table 27.46 I²C Bus Timing

Conditions: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{SS} = 0\text{ V}$, $\phi = 5\text{ MHz}$ to maximum operating frequency,
 $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|------------|---------------|-----|-------------------|------|-----------------|
| SCL input cycle time | t_{SCL} | $12 t_{cyc}$ | — | — | ns | Figure 27.34 |
| SCL input high pulse width | t_{SCLH} | $3 t_{cyc}$ | — | — | ns | |
| SCL input low pulse width | t_{SCLL} | $5 t_{cyc}$ | — | — | ns | |
| SCL, SDA input rise time | t_{Sr} | — | — | $7.5 t_{cyc}^{*}$ | ns | |
| SCL, SDA input fall time | t_{Sf} | — | — | 300 | ns | |
| SCL, SDA input spike pulse delete time | t_{SP} | — | — | $1 t_{cyc}$ | ns | |
| SDA input bus free time | t_{BUF} | $5 t_{cyc}$ | — | — | ns | |
| Operating condition input hold time | t_{STAH} | $3 t_{cyc}$ | — | — | ns | |
| Retransmitting operating condition input setup time | t_{STAS} | $3 t_{cyc}$ | — | — | ns | |
| Stop condition input setup time | t_{STOS} | $3 t_{cyc}$ | — | — | ns | |
| Data input setup time | t_{SDAS} | $0.5 t_{cyc}$ | — | — | ns | |
| Data input hold time | t_{SDAH} | 0 | — | — | ns | |
| SCL, SDA capacitor load | C_b | — | — | 400 | pF | |

Note: * Maximum SCL and SDA input rise time $7.5 t_{cyc}$ or $17.5 t_{cyc}$ can be selected depending on the clock that is used in the I²C module. For detail, see section 16.6, Usage Notes.

27.5.4 A/D Conversion Characteristics

Table 27.47 lists the A/D conversion characteristics.

Table 27.47 A/D Conversion Characteristics

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}^*, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 13.5 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition B (F-ZTAT version):

$$\begin{aligned} V_{CC} &= 2.2 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*, \\ V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}^*,$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Condition A | | | Conditions B, C | | | Unit |
|-------------------------------------|-------------|-----|-----------|-----------------|-----|-----------|---------------|
| | Min | Typ | Max | Min | Typ | Max | |
| Resolution | 10 | 10 | 10 | 10 | 10 | 10 | bits |
| Conversion time | 9.6 | — | — | 20.9 | — | — | μs |
| Analog input capacitance | — | — | 20 | — | — | 20 | pF |
| Permissible signal-source impedance | — | — | 5 | — | — | 5 | k Ω |
| Nonlinearity error | — | — | ± 6.0 | — | — | ± 6.0 | LSB |
| Offset error | — | — | ± 4.0 | — | — | ± 4.0 | LSB |
| Full-scale error | — | — | ± 4.0 | — | — | ± 4.0 | LSB |
| Quantization error | — | — | ± 0.5 | — | — | ± 0.5 | LSB |
| Absolute accuracy | — | — | ± 8.0 | — | — | ± 8.0 | LSB |

Note: * AN0 and AN1 can be used only when $V_{CC} = AV_{CC}$.

27.5.5 D/A Conversion Characteristics

Table 27.48 lists the D/A conversion characteristics.

Table 27.48 D/A Conversion Characteristics

Condition A (F-ZTAT version and masked ROM version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 13.5 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

Condition B (F-ZTAT version):

$$\begin{aligned} V_{CC} &= 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)} \end{aligned}$$

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$

$$\begin{aligned} V_{ref} &= 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ \phi &= 2 \text{ to } 6.25 \text{ MHz}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)} \end{aligned}$$

| Item | Condition A | | | Conditions B, C | | | Unit | Test Condition |
|--------------------|-------------|------|------|-----------------|------|------|------|--------------------------|
| | Min | Typ | Max | Min | Typ | Max | | |
| Resolution | 8 | 8 | 8 | 8 | 8 | 8 | bits | |
| Conversion time | — | — | 10 | — | — | 10 | μs | Load capacitance = 20 pF |
| Absolute accuracy* | — | ±2.0 | ±3.0 | — | ±3.0 | ±4.0 | LSB | Load resistance = 2 MΩ |
| | — | — | ±2.0 | — | — | ±3.0 | LSB | Load resistance = 4 MΩ |

Note: * Does not apply in module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode.

27.5.6 Flash Memory Characteristics

Table 27.49 lists the flash memory characteristics.

Table 27.49 Flash Memory Characteristics

Condition A: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{ref} = 2.7\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $V_{CC} = 3.0\text{ V to }3.6\text{ V}$ (Programming/erasing operating voltage range), $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (Programming/erasing operating temperature range; regular specifications), $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (Programming/erasing operating temperature range; wide-range specifications)

Condition B: $V_{CC} = 2.2\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}$, $V_{ref} = 2.2\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $V_{CC} = 3.0\text{ V to }3.6\text{ V}$ (Programming/erasing operating voltage range), $T_a = -20^\circ\text{C to }+50^\circ\text{C}$ (Programming/erasing operating temperature range; regular specifications)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--|-------------|-------------------|---------------------|------|---------------|----------------------|
| Programming time ^{*1*2*4} | t_p | — | 10 | 200 | ms/128 bytes | |
| Erase time ^{*1*3*5} | t_E | — | 100 | 1200 | ms/block | |
| Reprogramming count | N_{WEC} | 100 ^{*6} | 10000 ^{*7} | — | Times | |
| Data holding time ^{*8} | t_{DRP} | 10 | — | — | year | |
| Programming Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs | |
| Wait time after PSU1 bit setting ^{*1} | t_{psu} | 50 | 50 | — | μs | |
| Wait time after P1 bit setting ^{*1*4} | t_{sp10} | 8 | 10 | 12 | μs | |
| | t_{sp30} | 28 | 30 | 32 | μs | $1 \leq n \leq 6$ |
| | t_{sp200} | 198 | 200 | 202 | μs | $7 \leq n \leq 1000$ |
| Wait time after P1 bit clear ^{*1} | t_{cp} | 5 | 5 | — | μs | |
| Wait time after PSU1 bit clear ^{*1} | t_{cpsu} | 5 | 5 | — | μs | |
| Wait time after PV1 bit setting ^{*1} | t_{spv} | 4 | 4 | — | μs | |
| Wait time after H'FF dummy write ^{*1} | t_{spvr} | 2 | 2 | — | μs | |
| Wait time after PV1 bit clear ^{*1} | t_{cpv} | 2 | 2 | — | μs | |
| Wait time after SWE1 bit clear | t_{cswe} | 100 | 100 | — | μs | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-------------|--|------------|-----|-----|-------------------|-------|-----------------|
| Programming | Maximum programming count ^{*1*4} | N1 | — | — | 6 ^{*4} | Times | |
| | | N2 | — | — | 994 ^{*4} | | |
| Erase | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs | |
| | Wait time after ESU1 bit setting ^{*1} | t_{sesu} | 100 | 100 | — | μs | |
| | Wait time after E1 bit setting ^{*1*5} | t_{se} | 10 | 10 | 100 | ms | |
| | Wait time after E1 bit clear ^{*1} | t_{ce} | 10 | 10 | — | μs | |
| | Wait time after ESU1 bit clear ^{*1} | t_{cesu} | 10 | 10 | — | μs | |
| | Wait time after EV1 bit setting ^{*1} | t_{sev} | 20 | 20 | — | μs | |
| | Wait time after H'FF dummy write ^{*1} | t_{sevr} | 2 | 2 | — | μs | |
| | Wait time after EV1 bit clear ^{*1} | t_{cev} | 4 | 4 | — | μs | |
| | Wait time after SWE1 bit clear | t_{cswe} | 100 | 100 | — | μs | |
| | Maximum erase count ^{*1*5} | N | — | — | 100 | Times | |

- Notes: 1. Make each time setting in accordance with the program/program-verify flowchart or erase/erase-verify flowchart.
2. Programming time per 128 bytes (Shows the total period for which the P1 bit in the flash memory control register 1 (FLMCR1) is set. It does not include the program verification time.)
3. Block erase time (Shows the total period for which the E1 bit in FLMCR1 is set. It does not include the erase verification time.)
4. Maximum programming time value
 $t_p(\text{max}) = \text{Wait time after P1 bit setting } (t_{sp}) \times \text{maximum program count } (N)$
 $(t_{sp30} + t_{sp10}) \times 6 + (t_{sp200}) \times 994$
5. Relationship among the maximum erase time ($t_e(\text{max})$), the wait time after E1 bit setting (t_{se}), and the maximum erase count (N) is shown below.
 $t_e(\text{max}) = \text{Wait time after E1 bit setting } (t_{se}) \times \text{maximum erase count } (N)$
6. The minimum times that all characteristics after reprogramming are guaranteed. (The range between 1 and a minimum value is guaranteed.)
7. Reference value at 25°C. (Normally, it is a reference that rewriting is enabled up to this value.)
8. Data hold characteristics are when reprogramming is performed within the range of specifications including a minimum value.

27.6 Electrical Characteristics of H8S/2237 Group and H8S/2227 Group

27.6.1 Absolute Maximum Ratings

Table 27.50 lists the absolute maximum ratings.

Table 27.50 Absolute Maximum Ratings

| Item | Symbol | Value | Unit |
|-------------------------------------|-----------|---------------------------------------|------|
| Power supply voltage | V_{CC} | −0.3 to +4.3 | V |
| Program voltage* | V_{PP} | −0.3 to +13.5 | V |
| Input voltage (except ports4 and 9) | V_{in} | −0.3 to $V_{CC} + 0.3$ | V |
| Input voltage (ports 4 and 9) | V_{in} | −0.3 to $AV_{CC} + 0.3$ | V |
| Reference power supply voltage | V_{ref} | −0.3 to $AV_{CC} + 0.3$ | V |
| Analog power supply voltage | AV_{CC} | −0.3 to +4.3 | V |
| Analog input voltage | V_{AN} | −0.3 to $AV_{CC} + 0.3$ | V |
| Operating temperature | T_{opr} | Regular specifications: −20 to +75 | °C |
| | | Wide-range specifications: −40 to +85 | |
| Storage temperature | T_{stg} | −55 to +125 | °C |

Caution: Permanent damage to the chip may result if absolute maximum rating are exceeded.

Notes: The operating temperature ranges for flash memory programming/erasing are $T_a = -20^{\circ}\text{C}$ to $+75^{\circ}\text{C}$ (regular specifications) and $T_a = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$ (wide-range specifications).

* Supported in the HD6472237.

27.6.2 DC Characteristics

Table 27.51 lists the DC characteristics. Table 27.52 lists the permissible output currents.

Table 27.51 DC Characteristics (1)

Conditions (ZTAT version and F-ZTAT version):

$$\begin{aligned} V_{CC} &= 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\ V_{ref} &= 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\ T_a &= -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\ T_a &= -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}^{*1} \end{aligned}$$

Conditions (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}$,
 $T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}^{*1}$

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-------------------------------|---|---------------|----------------------|-----|----------------------|------|----------------------------------|
| Schmitt trigger input voltage | IRQ0 to IRQ7 | VT^- | $V_{CC} \times 0.2$ | — | — | V | |
| | | VT^+ | — | — | $V_{CC} \times 0.8$ | V | |
| | | $VT^+ - VT^-$ | $V_{CC} \times 0.07$ | — | — | V | ZTAT version, masked ROM version |
| | | $VT^+ - VT^-$ | $V_{CC} \times 0.05$ | — | — | V | F-ZTAT version |
| Input high voltage | RES, STBY, NMI, MD2 to MD0, FWE | V_{IH} | $V_{CC} \times 0.9$ | — | $V_{CC} + 0.3$ | V | |
| | EXTAL, Ports 1, 3, 7, and A to G | | $V_{CC} \times 0.8$ | — | $V_{CC} + 0.3$ | V | |
| | Ports 4 ^{*5} and 9 | | $V_{CC} \times 0.8$ | — | $AV_{CC} + 0.3^{*5}$ | V | |
| | | | | | | | |
| Input low voltage | RES, STBY, FWE, MD2 to MD0 | V_{IL} | -0.3 | — | $V_{CC} \times 0.1$ | V | |
| | NMI, EXTAL, Ports 1, 3, 4, 7, 9, and A to G | | -0.3 | — | $V_{CC} \times 0.2$ | V | |
| | | | | | | | |
| | | | | | | | |
| Output high voltage | All output pins | V_{OH} | $V_{CC} - 0.5$ | — | — | V | $I_{OH} = -200 \mu\text{A}$ |
| | | | $V_{CC} - 1.0$ | — | — | V | $I_{OH} = -1 \text{ mA}^{*2}$ |
| | | V_{OL} | — | — | 0.4 | V | $I_{OL} = 0.4 \text{ mA}$ |
| | | | — | — | 0.4 | V | $I_{OL} = 0.8 \text{ mA}^{*2}$ |

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|--|--------------------|-----|-----|---------------|--|
| Input leakage current | $\overline{\text{RES}}$ | $ I_{\text{in}} $ | — | 1.0 | μA | $V_{\text{in}} = 0.5 \text{ to } V_{\text{CC}}$ $-0.5 V^{*3}$ |
| | $\overline{\text{STBY}}$, NMI, FWE, MD2 to MD0 | — | — | 1.0 | μA | $V_{\text{in}} = 0.2 \text{ to } V_{\text{CC}}$ $-0.2 V^{*4}$ |
| | Ports 4, 9 | — | — | 1.0 | μA | $V_{\text{in}} = 0.5 \text{ to } AV_{\text{CC}}$ $-0.5 V^{*3}$ $V_{\text{in}} = 0.2 \text{ to } AV_{\text{CC}}$ $-0.2 V^{*4}$ |
| Three states leakage current (off) | Ports 1, 3, 7, and A to G | $ I_{\text{TSI}} $ | — | 1.0 | μA | $V_{\text{in}} = 0.5 \text{ to } V_{\text{CC}} - 0.5 V^{*3}$ $V_{\text{in}} = 0.2 \text{ to } V_{\text{CC}} - 0.2 V^{*4}$ |
| Input pull-up MOS current | Ports A to E | $-I_{\text{p}}$ | 10 | 300 | μA | $V_{\text{in}} = 0\text{V}$ |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{\text{ref}} \leq AV_{\text{CC}}$.
- $V_{\text{CC}} = 2.7 \text{ V to } 3.6 \text{ V}$
 - For ZTAT version and masked ROM version
 - For F-ZTAT version
 - When $V_{\text{CC}} < AV_{\text{CC}}$, the maximum value for P40 and P41 is $V_{\text{CC}} + 0.3 \text{ V}$.

Table 27.51 DC Characteristics (2)

Conditions (F-ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{ref} = 2.7\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)*¹

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|-------------------------------|-------------------------------|---------------|---|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 30 | pF | $V_{in} = 0\text{ V}$ |
| | NMI | | — | — | 30 | pF | $f = 1\text{ MHz}$ |
| | All input pins other than above ones | | — | — | 15 | pF | $T_a = 25^\circ\text{C}$ |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 20 $V_{CC} = 3.0\text{ V}$ | 37 $V_{CC} = 3.6\text{ V}$ | mA | $f = 13.5\text{ MHz}$ |
| | Sleep mode | | — | 15 $V_{CC} = 3.0\text{ V}$ | 29 $V_{CC} = 3.6\text{ V}$ | mA | $f = 13.5\text{ MHz}$ |
| | All modules stopped | | — | 15 | — | mA | $f = 13.5\text{ MHz}$, $V_{CC} = 3.0\text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 11 | — | mA | $f = 13.5\text{ MHz}$, $V_{CC} = 3.0\text{ V}$ (reference value) |
| | Subactive mode | | — | 60 | 160 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 35 | 90 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Watch mode | | — | 8 | 40 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------------------|----------------------------|------------|-----|------------------|------------------|---------|--|
| Current consumption | Standby mode ^{*2} | I_{cc} | — | 1.0 | 10 | μA | $T_a \leq 50^\circ C$, When 32.768 kHz crystal resonator is not used |
| | mode ^{*3} | | | $V_{cc} = 3.0 V$ | $V_{cc} = 3.6 V$ | | |
| | | | — | — | 50 | μA | $50^\circ C < T_a$, When 32.768 kHz crystal resonator is not used |
| | | | | | $V_{cc} = 3.6 V$ | | |
| Analog power supply current | During A/D conversion | $A I_{cc}$ | — | 0.8 | 1.5 | mA | $AV_{cc} = 3.0 V$ |
| | Idle | | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D conversion | $A I_{cc}$ | — | 1.3 | 2.5 | mA | $V_{ref} = 3.0 V$ |
| | Idle | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{cc} , V_{ref} , and AV_{ss} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{cc} and V_{ref} pins to V_{cc} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{cc}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{cc} - 0.2 V$ and $V_{IL} \text{ max} = 0.2 V$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{cc} < 2.7 V$, $V_{IH} \text{ min} = V_{cc} \times 0.9$, and $V_{IL} \text{ max} = 0.3 V$.
4. I_{cc} depends on V_{cc} and f as follows:
 $I_{cc} \text{ max} = 1.0 \text{ (mA)} + 0.74 \text{ (mA/(MHz} \times V)) \times V_{cc} \times f \text{ (normal operation)}$
 $I_{cc} \text{ max} = 1.0 \text{ (mA)} + 0.58 \text{ (mA/(MHz} \times V)) \times V_{cc} \times f \text{ (sleep mode)}$

Table 27.51 DC Characteristics (3)

Conditions (ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)*¹

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|--------------------------------------|-------------------------|-----|-------------------------------|-------------------------------|---------------|--|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 80 | pF | $V_{in} = 0\text{ V}$ |
| | NMI | | — | — | 50 | pF | $f = 1\text{ MHz}$ |
| | All input pins other than above ones | | — | — | 15 | pF | $T_a = 25^\circ\text{C}$ |
| Current consumption* ² | Normal operation | I_{CC} * ⁴ | — | 16 $V_{CC} = 3.0\text{ V}$ | 28 $V_{CC} = 3.6\text{ V}$ | mA | $f = 10\text{ MHz}$ |
| | Sleep mode | | — | 12 $V_{CC} = 3.0\text{ V}$ | 22 $V_{CC} = 3.6\text{ V}$ | mA | $f = 10\text{ MHz}$ |
| | All modules stopped | | — | 12 | — | mA | $f = 10\text{ MHz}$, $V_{CC} = 3.0\text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 8.5 | — | mA | $f = 10\text{ MHz}$, $V_{CC} = 3.0\text{ V}$ (reference value) |
| | Subactive mode | | — | 80 | 120 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Subsleep mode | | — | 60 | 90 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Watch mode | | — | 8 | 12 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|--------------------------------|----------------------------|-----------|-----|------|------|---------------|--|
| Current consumption | Standby mode ^{*2} | I_{cc} | — | 0.01 | 5.0 | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | mode ^{*3} | | — | — | 20.0 | μA | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D conversion | I_{cc} | — | 0.2 | 1.0 | mA | $AV_{cc} = 3.0\text{ V}$ |
| | Idle | | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D conversion | I_{cc} | — | 1.3 | 2.5 | mA | $V_{ref} = 3.0\text{ V}$ |
| | Idle | | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{cc} , V_{ref} , and AV_{ss} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{cc} and V_{ref} pins to V_{cc} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{cc}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{cc} - 0.5\text{ V}$ and $V_{IL} \text{ max} = 0.5\text{ V}$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{cc} < 2.7\text{ V}$, $V_{IH} \text{ min} = V_{cc} \times 0.9$, and $V_{IL} \text{ max} = 0.3\text{ V}$.
4. I_{cc} depends on V_{cc} and f as follows:
 $I_{cc} \text{ max} = 1.0\text{ (mA)} + 0.74\text{ (mA/(MHz} \times \text{V))} \times V_{cc} \times f$ (normal operation)
 $I_{cc} \text{ max} = 1.0\text{ (mA)} + 0.58\text{ (mA/(MHz} \times \text{V))} \times V_{cc} \times f$ (sleep mode)

Table 27.51 DC Characteristics (4)

Conditions (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)*¹

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------|--------------------------------------|--------------------|-----|---------------------------------|--|------|--|
| Input capacitance | $\overline{\text{RES}}$ | C_{in} | — | — | 80 | pF | $V_{\text{in}} = 0 \text{ V}$, $f = 1 \text{ MHz}$, $T_{\text{a}} = 25 \text{ }^{\circ}\text{C}$ |
| | NMI | | — | — | 50 | pF | |
| | All input pins other than above ones | | — | — | 15 | pF | |
| Current consumption*2 | Normal operation | I_{cc} *4 | — | 20 | 37 | mA | $f = 13.5 \text{ MHz}$ |
| | | | | $V_{\text{cc}} = 3.0 \text{ V}$ | $V_{\text{cc}} = 3.6 \text{ V}$ | | |
| | | | — | 10 | 18 | mA | $f = 6.25 \text{ MHz}$ |
| | | | | $V_{\text{cc}} = 3.0 \text{ V}$ | $V_{\text{cc}} = 3.6 \text{ V}$ | | |
| | Sleep mode | | — | 15 | 29 | mA | $f = 13.5 \text{ MHz}$ |
| | | | | $V_{\text{cc}} = 3.0 \text{ V}$ | $V_{\text{cc}} = 3.6 \text{ V}$ | | |
| | | | — | 7.5 | 14 | mA | $f = 6.25 \text{ MHz}$ |
| | | | | $V_{\text{cc}} = 3.0 \text{ V}$ | $V_{\text{cc}} = 3.6 \text{ V}$ | | |
| | All modules stopped | | — | 15 | — | mA | $f = 13.5 \text{ MHz}$, $V_{\text{cc}} = 3.0 \text{ V}$ (reference value) |
| | Medium-speed mode ($\phi/32$) | | — | 11 | — | mA | $f = 13.5 \text{ MHz}$, $V_{\text{cc}} = 3.0 \text{ V}$ (reference value) |
| Subactive mode | — | 60 | 160 | μA | $V_{\text{cc}} = 3.0 \text{ V}$, When 32.768 kHz crystal resonator is used | | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-----------------------------------|----------------------------|-----------|------|------|-----|------|--|
| Current consumption ^{*2} | Subsleep mode | — | — | 35 | 90 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Watch mode | — | — | 8 | 40 | μA | $V_{CC} = 3.0\text{ V}$, When 32.768 kHz crystal resonator is used |
| | Standby mode ^{*3} | — | 0.01 | 10 | — | μA | $T_a \leq 50^\circ\text{C}$, When 32.768 kHz crystal resonator is not used |
| | | — | — | — | 50 | μA | $50^\circ\text{C} < T_a$, When 32.768 kHz crystal resonator is not used |
| Analog power supply current | During A/D conversion | I_{CC} | — | 0.8 | 1.5 | mA | $AV_{CC} = 3.0\text{ V}$ |
| | Idle | — | — | 0.01 | 5.0 | μA | |
| Reference power supply current | During A/D conversion | I_{CC} | — | 1.3 | 2.5 | mA | $V_{ref} = 3.0\text{ V}$ |
| | Idle | — | — | 0.01 | 5.0 | μA | |
| RAM standby voltage | | V_{RAM} | 2.0 | — | — | V | |

- Notes: 1. If the A/D or D/A converter is not used, the AV_{CC} , V_{ref} , and AV_{SS} pins should not be open. Even if the A/D or D/A converter is not used, connect the AV_{CC} and V_{ref} pins to V_{CC} and supply 2.0 V to 3.6 V. In this case, $V_{ref} \leq AV_{CC}$.
2. Current consumption values are for $V_{IH} \text{ min} = V_{CC} - 0.5\text{ V}$ and $V_{IL} \text{ max} = 0.5\text{ V}$, with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.
3. The values are for $V_{RAM} \leq V_{CC} < 2.2\text{ V}$, $V_{IH} \text{ min} = V_{CC} \times 0.9$, and $V_{IL} \text{ max} = 0.3\text{ V}$.
4. I_{CC} depends on V_{CC} and f as follows:
 $I_{CC} \text{ max} = 1.0\text{ (mA)} + 0.74\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (normal operation)
 $I_{CC} \text{ max} = 1.0\text{ (mA)} + 0.58\text{ (mA/(MHz} \times \text{V))} \times V_{CC} \times f$ (sleep mode)

Table 27.52 Permissible Output Currents

Conditions (ZTAT version and F-ZTAT version):

$$\begin{aligned}
 &V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}, \\
 &V_{ref} = 2.7 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V}, \\
 &T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)}, \\
 &T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}
 \end{aligned}$$

Conditions (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}, AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V},$
 $V_{ref} = 2.2 \text{ V to } AV_{CC}, V_{SS} = AV_{SS} = 0 \text{ V},$
 $T_a = -20^\circ\text{C to } +75^\circ\text{C (regular specifications)},$
 $T_a = -40^\circ\text{C to } +85^\circ\text{C (wide-range specifications)}$

| Item | | | Symbol | Min | Typ | Max | Unit |
|---|--------------------------|--|------------------|-----|-----|-----|------|
| Permissible output low current (per pin) | Output pins | $V_{CC} = 2.2\text{ V to } 3.6\text{ V}$ | I_{OL} | — | — | 0.5 | mA |
| | | $V_{CC} = 2.7\text{ V to } 3.6\text{ V}$ | | — | — | 1.0 | |
| Permissible output low current (total) | Total of all output pins | $V_{CC} = 2.2\text{ V to } 3.6\text{ V}$ | ΣI_{OL} | — | — | 30 | mA |
| | | $V_{CC} = 2.7\text{ V to } 3.6\text{ V}$ | | — | — | 60 | |
| Permissible output high current (per pin) | All output pins | $V_{CC} = 2.2\text{ V to } 3.6\text{ V}$ | $-I_{OH}$ | — | — | 0.5 | mA |
| | | $V_{CC} = 2.7\text{ V to } 3.6\text{ V}$ | | — | — | 1.0 | |
| Permissible output high current (total) | Total of all output pins | $V_{CC} = 2.2\text{ V to } 3.6\text{ V}$ | $\Sigma -I_{OH}$ | — | — | 15 | mA |
| | | $V_{CC} = 2.7\text{ V to } 3.6\text{ V}$ | | — | — | 30 | |

Note: To protect chip reliability, do not exceed the output current values in table 27.52.

27.6.3 AC Characteristics

Figure 27.9 shows the test conditions for the AC characteristics.

(1) Clock Timing

Table 27.53 lists the clock timing.

Table 27.53 Clock Timing

Condition A (ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, 2 to 10 MHz,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (F-ZTAT version, masked ROM version):
 $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, 2 to 13.5MHz,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.2\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 32.768\text{ kHz}$, 2 to 6.25 MHz,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|--|------------|-------------|-----|-------------|-----|-------------|-----|------|-----------------|
| | | Min | Max | Min | Max | Min | Max | | |
| Clock cycle time | t_{cyc} | 100 | 500 | 74 | 500 | 160 | 500 | ns | Figure 27.10 |
| Clock high pulse width | t_{CH} | 35 | — | 25 | — | 50 | — | ns | |
| Clock low pulse width | t_{CL} | 35 | — | 25 | — | 50 | — | ns | |
| Clock rise time | t_{Cr} | — | 15 | — | 10 | — | 25 | ns | |
| Clock fall time | t_{Cf} | — | 15 | — | 10 | — | 25 | ns | |
| Oscillation stabilization time at reset (crystal) | t_{OSC1} | 20 | — | 20 | — | 40 | — | ms | Figure 27.11 |
| Oscillation stabilization time in software standby (crystal) | t_{OSC2} | 8 | — | 8 | — | 16 | — | ms | |

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|--|-------------------|-------------|--------|-------------|--------|-------------|--------|---------------|-----------------|
| | | Min | Max | Min | Max | Min | Max | | |
| External clock output stabilization delay time | t_{DEXT} | 500 | — | 500 | — | 1000 | — | μs | Figure 27.11 |
| Subclock oscillation stabilization time | t_{OSC3} | — | 2 | — | 2 | — | 3 | s | |
| Subclock oscillator frequency | f_{SUB} | 32.768 | 32.768 | 32.768 | 32.768 | 32.768 | 32.768 | kHz | |
| Subclock (ϕ_{SUB}) cycle time | t_{SUB} | 30.5 | 30.5 | 30.5 | 30.5 | 30.5 | 30.5 | μs | |

(2) Control Signal Timing

Table 27.54 lists the control signal timing.

Table 27.54 Control Signal Timing

Condition A (ZTAT version): $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $\phi = 32.768 \text{ kHz}$, 2 to 10 MHz,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

Condition B (F-ZTAT version, masked ROM version):
 $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $\phi = 32.768 \text{ kHz}$, 2 to 13.5MHz,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $\phi = 32.768 \text{ kHz}$, 2 to 6.25 MHz,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|---|--------------------|-------------|-----|-------------|-----|-------------|-----|------------------|-----------------|
| | | Min | Max | Min | Max | Min | Max | | |
| $\overline{\text{RES}}$ setup time | t_{RESS} | 250 | — | 250 | — | 350 | — | ns | Figure 27.12 |
| $\overline{\text{RES}}$ pulse width | t_{RESW} | 20 | — | 20 | — | 20 | — | t_{cyc} | |
| $\overline{\text{MRES}}$ setup time | t_{MRESS} | 250 | — | 250 | — | 350 | — | ns | |
| $\overline{\text{MRES}}$ pulse width | t_{MRESW} | 20 | — | 20 | — | 20 | — | t_{cyc} | Figure 27.13 |
| NMI setup time | t_{NMIS} | 250 | — | 250 | — | 350 | — | ns | |
| NMI hold time | t_{NMIH} | 10 | — | 10 | — | 10 | — | ns | |
| NMI pulse width (exiting software standby mode) | t_{NMIW} | 200 | — | 200 | — | 300 | — | ns | |
| $\overline{\text{IRQ}}$ setup time | t_{IRQS} | 250 | — | 250 | — | 350 | — | ns | |
| $\overline{\text{IRQ}}$ hold time | t_{IRQH} | 10 | — | 10 | — | 10 | — | ns | |
| $\overline{\text{IRQ}}$ pulse width (exiting software standby mode) | t_{IRQW} | 200 | — | 200 | — | 300 | — | ns | |

(3) Bus Timing

Table 27.55 lists the bus timing.

Table 27.55 Bus Timing

Condition A (ZTAT version): $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $\phi = 2 \text{ to } 10 \text{ MHz}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

Condition B (F-ZTAT version, masked ROM version):
 $V_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.7 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.7 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $\phi = 2 \text{ to } 13.5 \text{ MHz}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$, $AV_{CC} = 2.2 \text{ V to } 3.6 \text{ V}$,
 $V_{ref} = 2.2 \text{ V to } AV_{CC}$, $V_{SS} = AV_{SS} = 0 \text{ V}$,
 $\phi = 2 \text{ to } 6.25 \text{ MHz}$,
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to } +85^\circ\text{C}$ (wide-range specifications)

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|------------------------------|------------|------------------------|-----|------------------------|-----|------------------------|-----|------|------------------------|
| | | Min | Max | Min | Max | Min | Max | | |
| Address delay time | t_{AD} | — | 60 | — | 50 | — | 90 | ns | Figures 27.14 to 27.18 |
| Address setup time | t_{AS} | $0.5 \times t_{cyc}$ — | — | $0.5 \times t_{cyc}$ — | — | $0.5 \times t_{cyc}$ — | — | ns | |
| | | 40 | | 30 | | 60 | | | |
| Address hold time | t_{AH} | $0.5 \times t_{cyc}$ — | — | $0.5 \times t_{cyc}$ — | — | $0.5 \times t_{cyc}$ — | — | ns | |
| | | 20 | | 15 | | 30 | | | |
| \overline{CS} delay time | t_{CSD} | — | 60 | — | 50 | — | 90 | ns | |
| \overline{AS} delay time | t_{ASD} | — | 60 | — | 50 | — | 90 | ns | |
| \overline{RD} delay time 1 | t_{RSD1} | — | 60 | — | 50 | — | 90 | ns | |
| \overline{RD} delay time 2 | t_{RSD2} | — | 60 | — | 50 | — | 90 | ns | |
| Read data setup time | t_{RDS} | 30 | — | 30 | — | 50 | — | ns | |
| Read data hold time | t_{RDH} | 0 | — | 0 | — | 0 | — | ns | |

| Item | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|-------------------------------|------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|------|------------------------|
| | | Min | Max | Min | Max | Min | Max | | |
| Read data access time 1 | t_{ACC1} | — | $1.0 \times t_{cyc} - 65$ | — | $1.0 \times t_{cyc} - 65$ | — | $1.0 \times t_{cyc} - 90$ | ns | Figures 27.14 to 27.18 |
| Read data access time 2 | t_{ACC2} | — | $1.5 \times t_{cyc} - 65$ | — | $1.5 \times t_{cyc} - 65$ | — | $1.5 \times t_{cyc} - 90$ | ns | |
| Read data access time 3 | t_{ACC3} | — | $2.0 \times t_{cyc} - 65$ | — | $2.0 \times t_{cyc} - 65$ | — | $2.0 \times t_{cyc} - 90$ | ns | |
| Read data access time 4 | t_{ACC4} | — | $2.5 \times t_{cyc} - 65$ | — | $2.5 \times t_{cyc} - 65$ | — | $2.5 \times t_{cyc} - 90$ | ns | |
| Read data access time 5 | t_{ACC5} | — | $3.0 \times t_{cyc} - 65$ | — | $3.0 \times t_{cyc} - 65$ | — | $3.0 \times t_{cyc} - 90$ | ns | |
| \overline{WR} delay time 1 | t_{WRD1} | — | 60 | — | 50 | — | 90 | ns | |
| \overline{WR} delay time 2 | t_{WRD2} | — | 60 | — | 50 | — | 90 | ns | |
| \overline{WR} pulse width 1 | t_{WSW1} | $1.0 \times t_{cyc} - 40$ | — | $1.0 \times t_{cyc} - 30$ | — | $1.0 \times t_{cyc} - 60$ | — | ns | |
| \overline{WR} pulse width 2 | t_{WSW2} | $1.5 \times t_{cyc} - 40$ | — | $1.5 \times t_{cyc} - 30$ | — | $1.5 \times t_{cyc} - 60$ | — | ns | |
| Write data delay time | t_{WDD} | — | 80 | — | 70 | — | 100 | ns | |
| Write data setup time | t_{WDS} | $0.5 \times t_{cyc} - 50$ | — | $0.5 \times t_{cyc} - 37$ | — | $0.5 \times t_{cyc} - 80$ | — | ns | |
| Write data hold time | t_{WDH} | $0.5 \times t_{cyc} - 30$ | — | $0.5 \times t_{cyc} - 15$ | — | $0.5 \times t_{cyc} - 60$ | — | ns | |
| \overline{WAIT} setup time | t_{WTS} | 60 | — | 50 | — | 90 | — | ns | |
| \overline{WAIT} hold time | t_{WTH} | 10 | — | 10 | — | 10 | — | ns | |
| \overline{BREQ} setup time | t_{BRQS} | 60 | — | 50 | — | 90 | — | ns | |
| \overline{BACK} delay time | t_{BACD} | — | 60 | — | 50 | — | 90 | ns | Figure 27.19 |
| Bus-floating time | t_{BZD} | — | 100 | — | 80 | — | 160 | ns | |

(4) Timing of On-Chip Peripheral Modules

Table 27.56 lists the timing of on-chip peripheral modules.

Table 27.56 Timing of On-Chip Peripheral Modules

Condition A (ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$,
 $2\text{ to }10\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (F-ZTAT version, masked ROM version):
 $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$,
 $2\text{ to }13.5\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.2\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 32.768\text{ kHz}$,
 $2\text{ to }6.25\text{ MHz}$, $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions |
|----------|------------------------------|-------------|-------------|-----|-------------|-----|-------------|-----|------|-----------------|
| | | | Min | Max | Min | Max | Min | Max | | |
| I/O port | Output data delay time | t_{PWD} | — | 100 | — | 100 | — | 150 | ns | Figure 27.24 |
| | Input data setup time | t_{PRS} | 50 | — | 50 | — | 80 | — | | |
| | Input data hold time | t_{PRH} | 50 | — | 50 | — | 80 | — | | |
| TPU | Timer output delay time | t_{TODD} | — | 100 | — | 100 | — | 150 | ns | Figure 27.25 |
| | Timer input setup time | t_{TICS} | 50 | — | 40 | — | 60 | — | | |
| | Timer clock input setup time | t_{TKS} | 50 | — | 40 | — | 60 | — | ns | Figure 27.26 |
| | Timer clock | Single edge | t_{TCKWH} | 1.5 | — | 1.5 | — | 1.5 | — | |
| | pulse width | Both edges | t_{TCKWL} | 2.5 | — | 2.5 | — | 2.5 | — | |

| Item | | Symbol | Condition A | | Condition B | | Condition C | | Unit | Test Conditions | |
|---------------|---------------------------------------|--------------------|-------------------|-----|-------------|-----|-------------|-----|-------------------|------------------|--------------|
| | | | Min | Max | Min | Max | Min | Max | | | |
| TMR | Timer output delay time | t _{TMOD} | — | 100 | — | 100 | — | 150 | ns | Figure 27.27 | |
| | Timer reset input setup time | t _{TMRS} | 50 | — | 50 | — | 80 | — | ns | Figure 27.29 | |
| | Timer clock input setup time | t _{TMCS} | 50 | — | 50 | — | 80 | — | ns | Figure 27.28 | |
| | Timer clock Single edge | t _{TMCWH} | 1.5 | — | 1.5 | — | 1.5 | — | t _{cyc} | | |
| | pulse width Both edges | t _{TMCWL} | 2.5 | — | 2.5 | — | 2.5 | — | | | |
| WDT_1 | BUZZ output delay time | t _{BUZD} | — | 100 | — | 100 | — | 150 | ns | Figure 27.30 | |
| SCI* | Input clock cycle | Asynchronous | t _{Scyc} | 4 | — | 4 | — | 4 | — | t _{cyc} | Figure 27.31 |
| | | Synchronous | | 6 | — | 6 | — | 6 | — | | |
| | Input clock pulse width | t _{SCKW} | 0.4 | 0.6 | 0.4 | 0.6 | 0.4 | 0.6 | t _{Scyc} | | |
| | Input clock rise time | t _{SCKr} | — | 1.5 | — | 1.5 | — | 1.5 | t _{cyc} | | |
| | Input clock fall time | t _{SCKf} | — | 1.5 | — | 1.5 | — | 1.5 | | | |
| | Transmit data delay time | t _{TXD} | — | 100 | — | 100 | — | 150 | ns | Figure 27.32 | |
| | Receive data setup time (synchronous) | t _{RXS} | 100 | — | 75 | — | 150 | — | ns | | |
| | Receive data hold time (synchronous) | t _{RXH} | 100 | — | 75 | — | 150 | — | ns | | |
| A/D converter | Trigger input setup time | t _{TRGS} | 50 | — | 40 | — | 60 | — | ns | Figure 27.33 | |

27.6.4 A/D Conversion Characteristics

Table 27.57 lists the A/D conversion characteristics.

Table 27.57 A/D Conversion Characteristics

Condition A (ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}^*$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}^*$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ to }10\text{ MHz}$, T_a
 $= -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (F-ZTAT version, Masked ROM version):
 $V_{CC} = 2.7\text{ V to }3.6\text{ V}^*$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}^*$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2\text{ V to }3.6\text{ V}^*$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}^*$,
 $V_{ref} = 2.2\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ to }6.25\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Condition A | | | Condition B | | | Condition C | | | Unit |
|-------------------------------------|-------------|-----|-----------|-------------|-----|-----------|-------------|-----|-----------|---------------|
| | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | |
| Resolution | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | 10 | bits |
| Conversion time | 13.1 | — | — | 9.6 | — | — | 20.9 | — | — | μs |
| Analog input capacitance | — | — | 20 | — | — | 20 | — | — | 20 | pF |
| Permissible signal-source impedance | — | — | 5 | — | — | 5 | — | — | 5 | k Ω |
| Nonlinearity error | — | — | ± 6.0 | — | — | ± 6.0 | — | — | ± 6.0 | LSB |
| Offset error | — | — | ± 4.0 | — | — | ± 4.0 | — | — | ± 4.0 | LSB |
| Full-scale error | — | — | ± 4.0 | — | — | ± 4.0 | — | — | ± 4.0 | LSB |
| Quantization error | — | — | ± 0.5 | — | — | ± 0.5 | — | — | ± 0.5 | LSB |
| Absolute accuracy | — | — | ± 8.0 | — | — | ± 8.0 | — | — | ± 8.0 | LSB |

Note: * AN0 and AN1 can be used only when $V_{CC} = AV_{CC}$.

27.6.5 D/A Conversion Characteristics

Table 27.58 lists the D/A conversion characteristics.

Table 27.58 D/A Conversion Characteristics

Condition A (ZTAT version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ to }10\text{ MHz}$, T_a
 $= -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

Condition B (Masked ROM version): $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.7\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$, $\phi = 2\text{ to }13.5\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications)

Condition C (Masked ROM version): $V_{CC} = 2.2\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.2\text{ V to }3.6\text{ V}$,
 $V_{ref} = 2.2\text{ V to }AV_{CC}$, $V_{SS} = AV_{SS} = 0\text{ V}$,
 $\phi = 2\text{ to }6.25\text{ MHz}$,
 $T_a = -20^\circ\text{C to }+75^\circ\text{C}$ (regular specifications),
 $T_a = -40^\circ\text{C to }+85^\circ\text{C}$ (wide-range specifications)

| Item | Condition A | | | Condition B | | | Condition C | | | Unit | Test Condition |
|--------------------|-------------|-----------|-----------|-------------|-----------|-----------|-------------|-----------|-----------|---------------|--------------------------------|
| | Min | Typ | Max | Min | Typ | Max | Min | Typ | Max | | |
| Resolution | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 | bits | |
| Conversion time | — | — | 10 | — | — | 10 | — | — | 10 | μs | Load capacitance = 20 pF |
| Absolute accuracy* | — | ± 2.0 | ± 3.0 | — | ± 2.0 | ± 3.0 | — | ± 3.0 | ± 4.0 | LSB | Load resistance = 2 M Ω |
| | — | — | ± 2.0 | — | — | ± 2.0 | — | — | ± 3.0 | LSB | Load resistance = 4 M Ω |

Note: * Does not apply in module stop mode, software standby mode, watch mode, subactive mode, or subsleep mode.

27.6.6 Flash Memory Characteristics

Table 27.59 lists the flash memory characteristics.

Table 27.59 Flash Memory Characteristics

Conditions: $V_{CC} = 2.7\text{ V to }3.6\text{ V}$, $AV_{CC} = 2.7\text{ V to }3.6\text{ V}$, $V_{ref} = 2.7\text{ V to }AV_{CC}$,
 $V_{SS} = AV_{SS} = 0\text{ V}$, $V_{CC} = 3.0\text{ V to }3.6\text{ V}$ (Programming/erasing operating voltage range), $T_a = -20^\circ\text{C to }+50^\circ\text{C}$ (Programming/erasing operating temperature range; regular specifications)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|------------------------------------|--|-------------------|---------------------|------|-------------------|------------------------------------|
| Programming time ^{*1*2*4} | t_p | — | 10 | 200 | ms/128 bytes | |
| Erase time ^{*1*3*5} | t_E | — | 100 | 1200 | ms/block | |
| Reprogramming count | N_{WEC} | 100 ^{*6} | 10000 ^{*7} | — | Times | |
| Data holding time ^{*8} | t_{DRP} | 10 | — | — | year | |
| Programming | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs |
| | Wait time after PSU1 bit setting ^{*1} | t_{spsu} | 50 | 50 | — | μs |
| | Wait time after P1 bit setting ^{*1*4} | t_{sp10} | 8 | 10 | 12 | μs |
| | | t_{sp30} | 28 | 30 | 32 | μs $1 \leq n \leq 6$ |
| | | t_{sp200} | 198 | 200 | 202 | μs $7 \leq n \leq 1000$ |
| | Wait time after P1 bit clear ^{*1} | t_{cp} | 5 | 5 | — | μs |
| | Wait time after PSU1 bit clear ^{*1} | t_{cpsu} | 5 | 5 | — | μs |
| | Wait time after PV1 bit setting ^{*1} | t_{spv} | 4 | 4 | — | μs |
| | Wait time after H'FF dummy write ^{*1} | t_{spvr} | 2 | 2 | — | μs |
| | Wait time after PV1 bit clear ^{*1} | t_{cpv} | 2 | 2 | — | μs |
| | Wait time after SWE1 bit clear | t_{cswe} | 100 | 100 | — | μs |
| | Maximum programming count ^{*1*4} | N1 | — | — | 6 ^{*4} | Times |
| | | N2 | — | — | 994 ^{*4} | |

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|-------|--|------------|-----|-----|-----|---------|-----------------|
| Erase | Wait time after SWE1 bit setting ^{*1} | t_{sswe} | 1 | 1 | — | μs | |
| | Wait time after ESU1 bit setting ^{*1} | t_{sesu} | 100 | 100 | — | μs | |
| | Wait time after E1 bit setting ^{*1*5} | t_{se} | 10 | 10 | 100 | ms | |
| | Wait time after E1 bit clear ^{*1} | t_{ce} | 10 | 10 | — | μs | |
| | Wait time after ESU1 bit clear ^{*1} | t_{cesu} | 10 | 10 | — | μs | |
| | Wait time after EV1 bit setting ^{*1} | t_{sev} | 20 | 20 | — | μs | |
| | Wait time after H'FF dummy write ^{*1} | t_{sevr} | 2 | 2 | — | μs | |
| | Wait time after EV1 bit clear ^{*1} | t_{cev} | 4 | 4 | — | μs | |
| | Wait time after SWE1 bit clear | t_{cswe} | 100 | 100 | — | μs | |
| | Maximum erase count ^{*1*5} | N | — | — | 100 | Times | |

- Notes: 1. Make each time setting in accordance with the program/program-verify flowchart or erase/erase-verify flowchart.
2. Programming time per 128 bytes (Shows the total period for which the P1 bit in the flash memory control register 1 (FLMCR1) is set. It does not include the program verification time.)
3. Block erase time (Shows the total period for which the E1 bit in FLMCR1 is set. It does not include the erase verification time.)
4. Maximum programming time value
 $t_p(\text{max}) = \text{Wait time after P1 bit setting } (t_{sp}) \times \text{maximum program count (N)}$
 $(t_{sp30} + t_{sp10}) \times 6 + (t_{sp200}) \times 994$
5. Relationship among the maximum erase time ($t_E(\text{max})$), the wait time after E1 bit setting (t_{se}), and the maximum erase count (N) is shown below.
 $t_E(\text{max}) = \text{Wait time after E1 bit setting } (t_{se}) \times \text{maximum erase count (N)}$
6. The minimum times that all characteristics after reprogramming are guaranteed. (The range between 1 and a minimum value is guaranteed.)
7. Reference value at 25°C. (Normally, it is a reference that rewriting is enabled up to this value.)
8. Data hold characteristics are when reprogramming is performed within the range of specifications including a minimum value.

27.7 Operating Timing

27.7.1 Clock Timing

The clock timing is shown below.

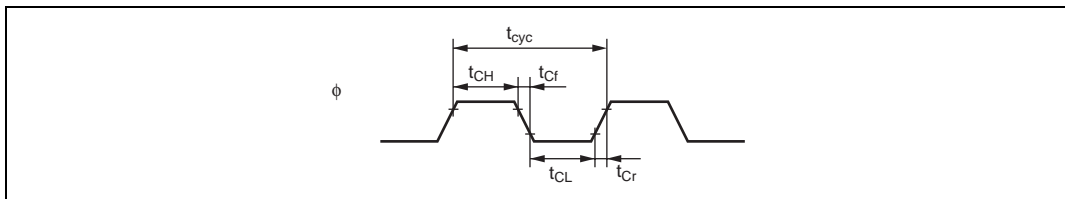


Figure 27.10 System Clock Timing

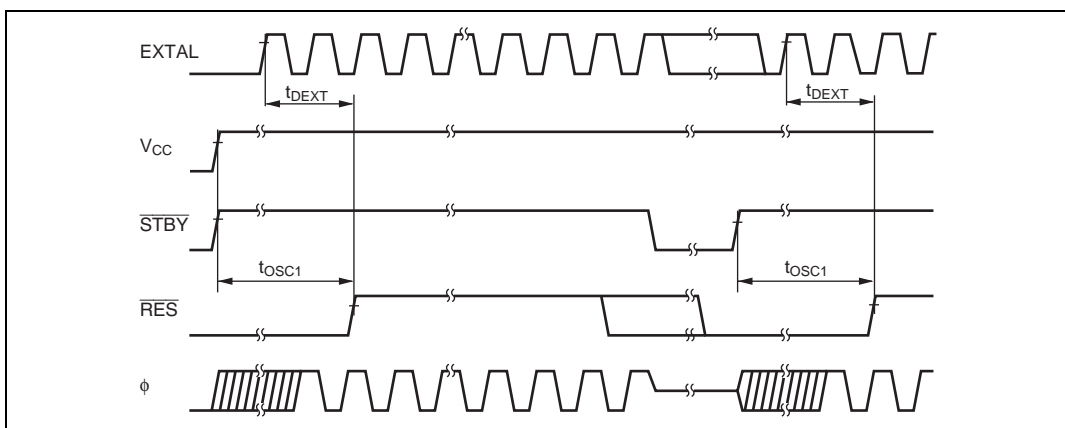


Figure 27.11 Oscillation Stabilization Timing

27.7.2 Control Signal Timing

The control signal timing is shown below.

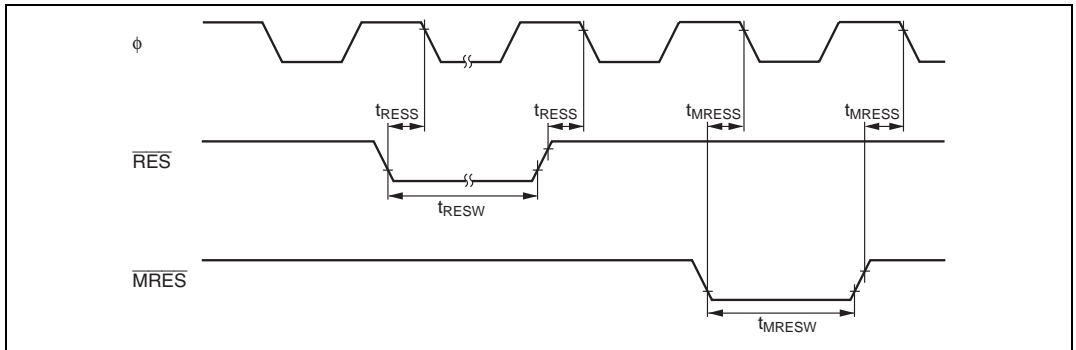


Figure 27.12 Reset Input Timing

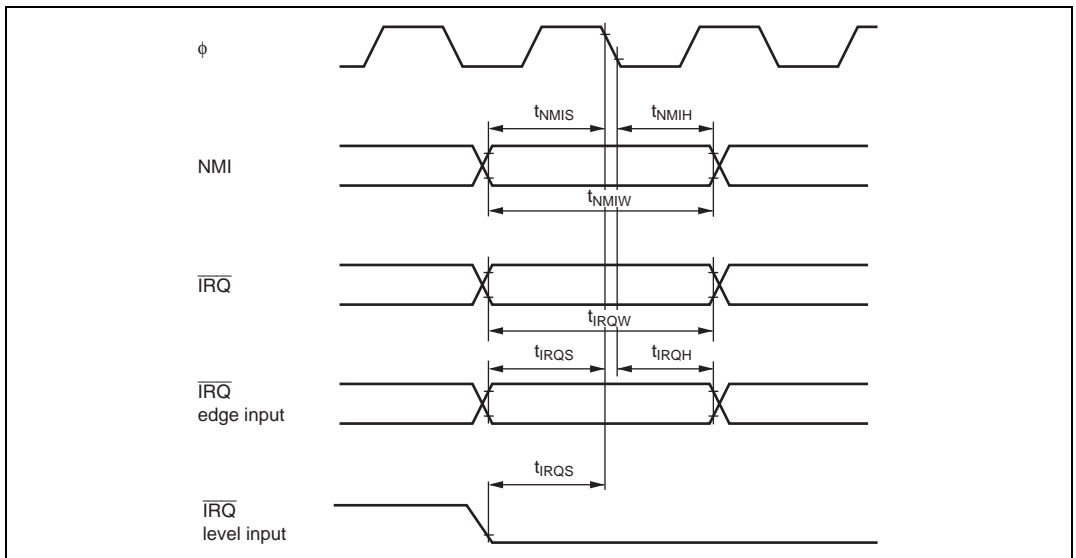


Figure 27.13 Interrupt Input Timing

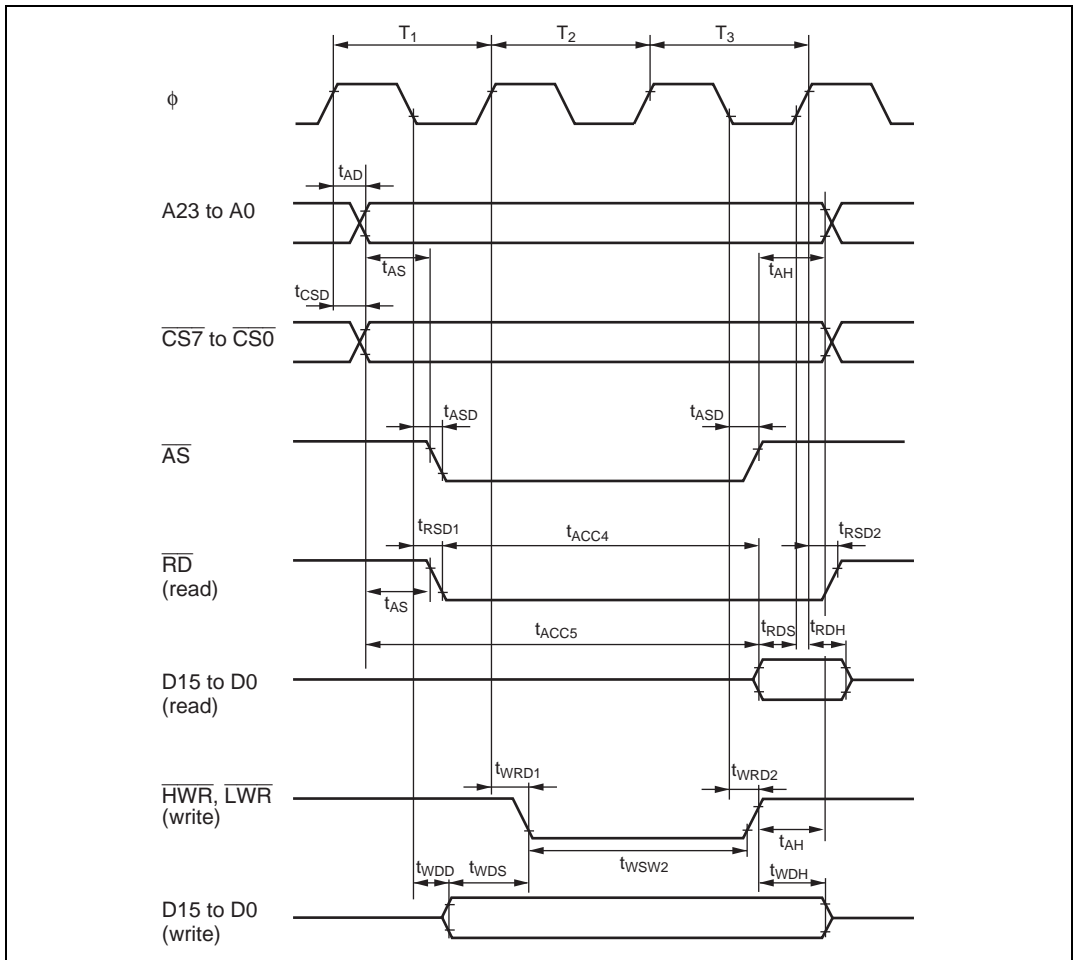


Figure 27.15 Basic Bus Timing (Three-State Access)

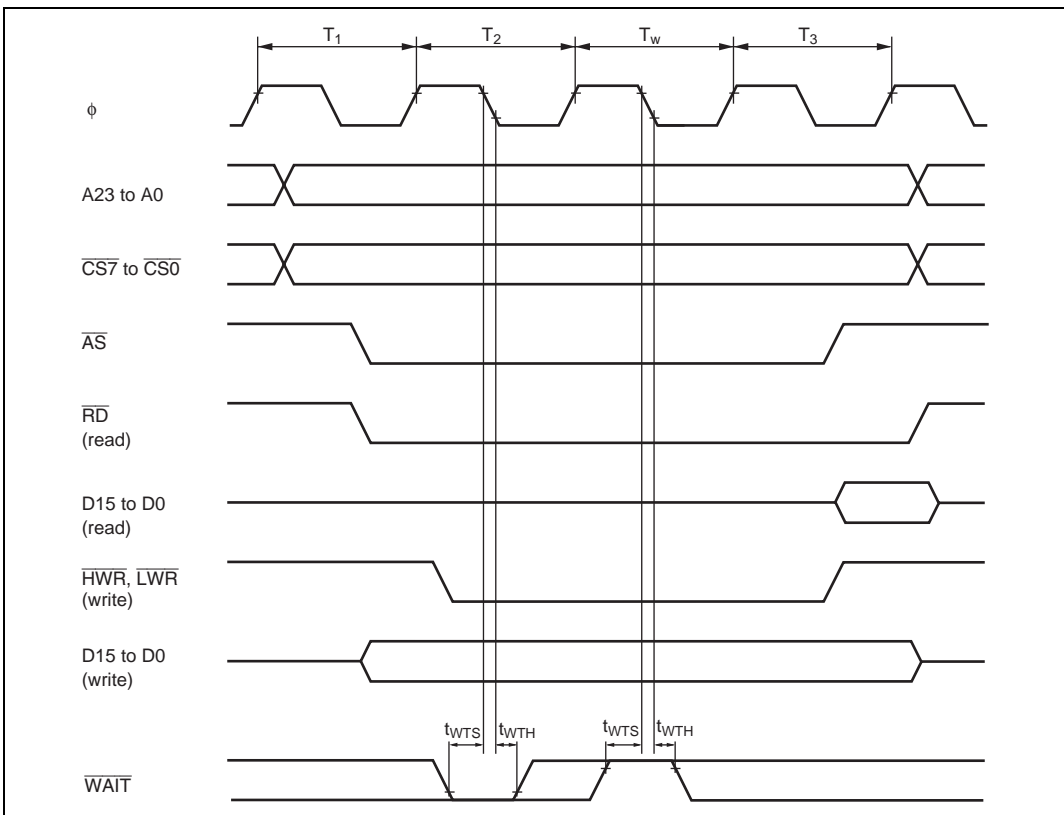


Figure 27.16 Basic Bus Timing (Three-State Access with One Wait State)

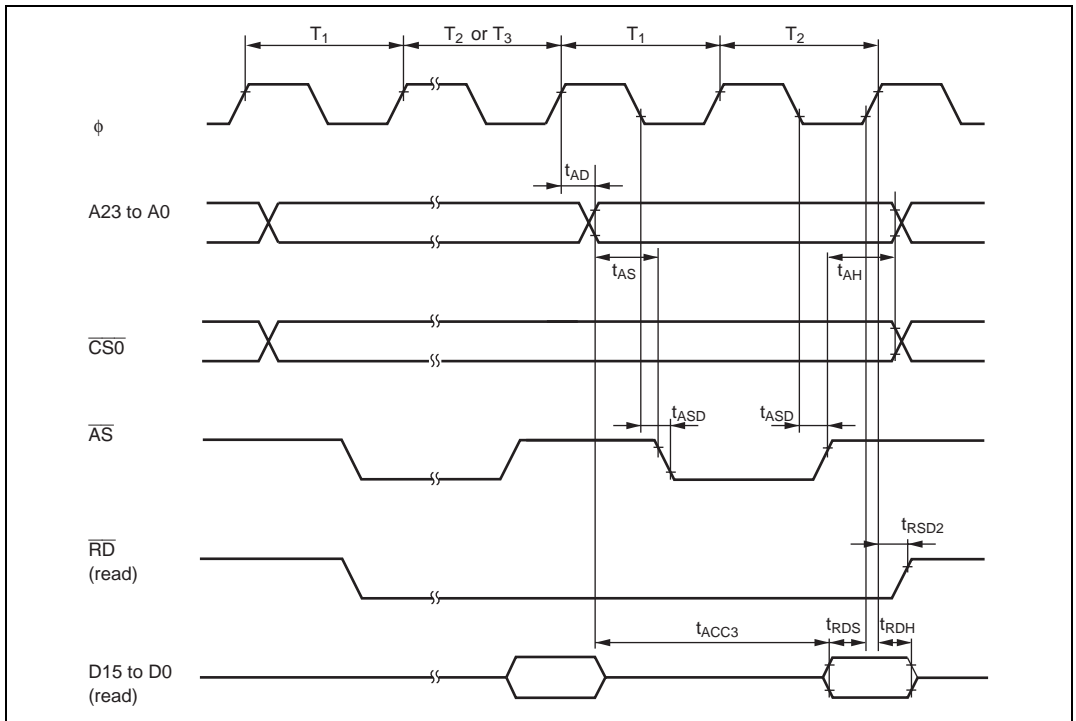


Figure 27.17 Burst ROM Access Timing (Two-State Access)

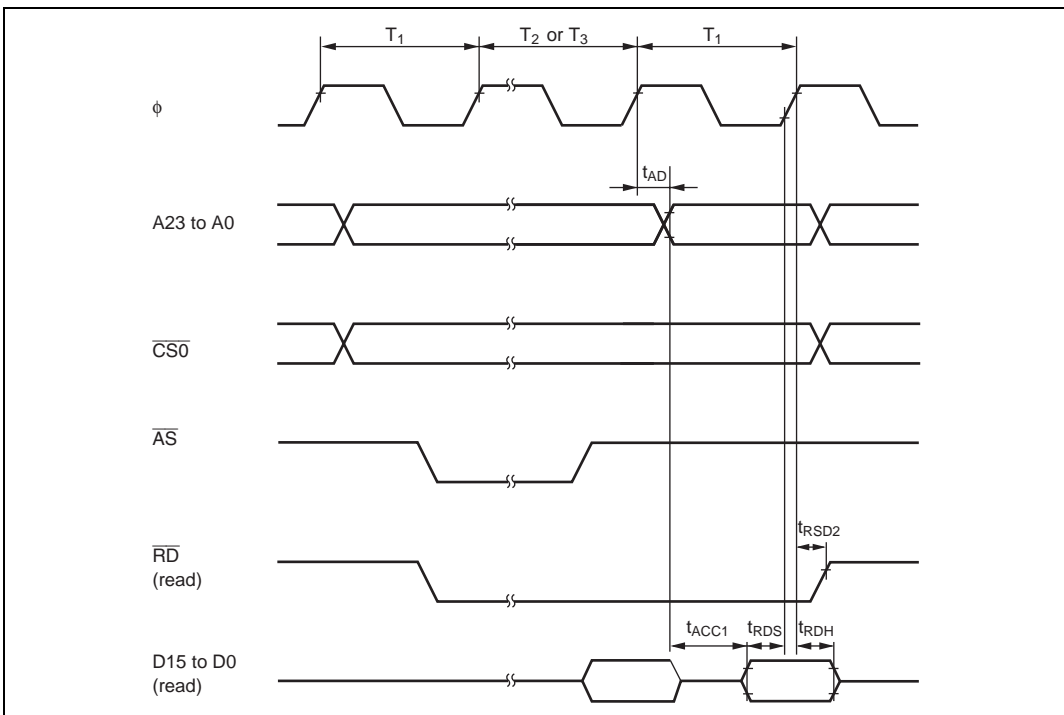


Figure 27.18 Burst ROM Access Timing (One-State Access)

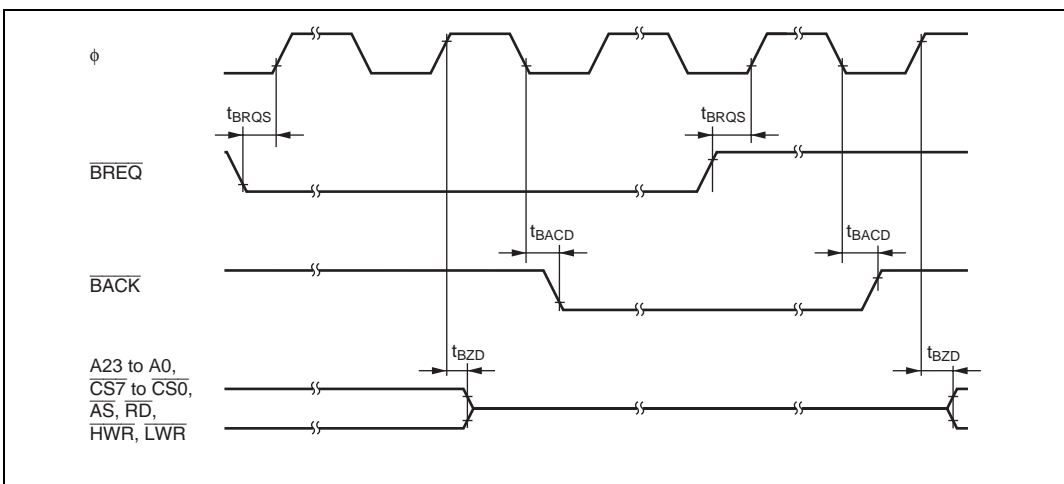


Figure 27.19 External Bus Release Timing

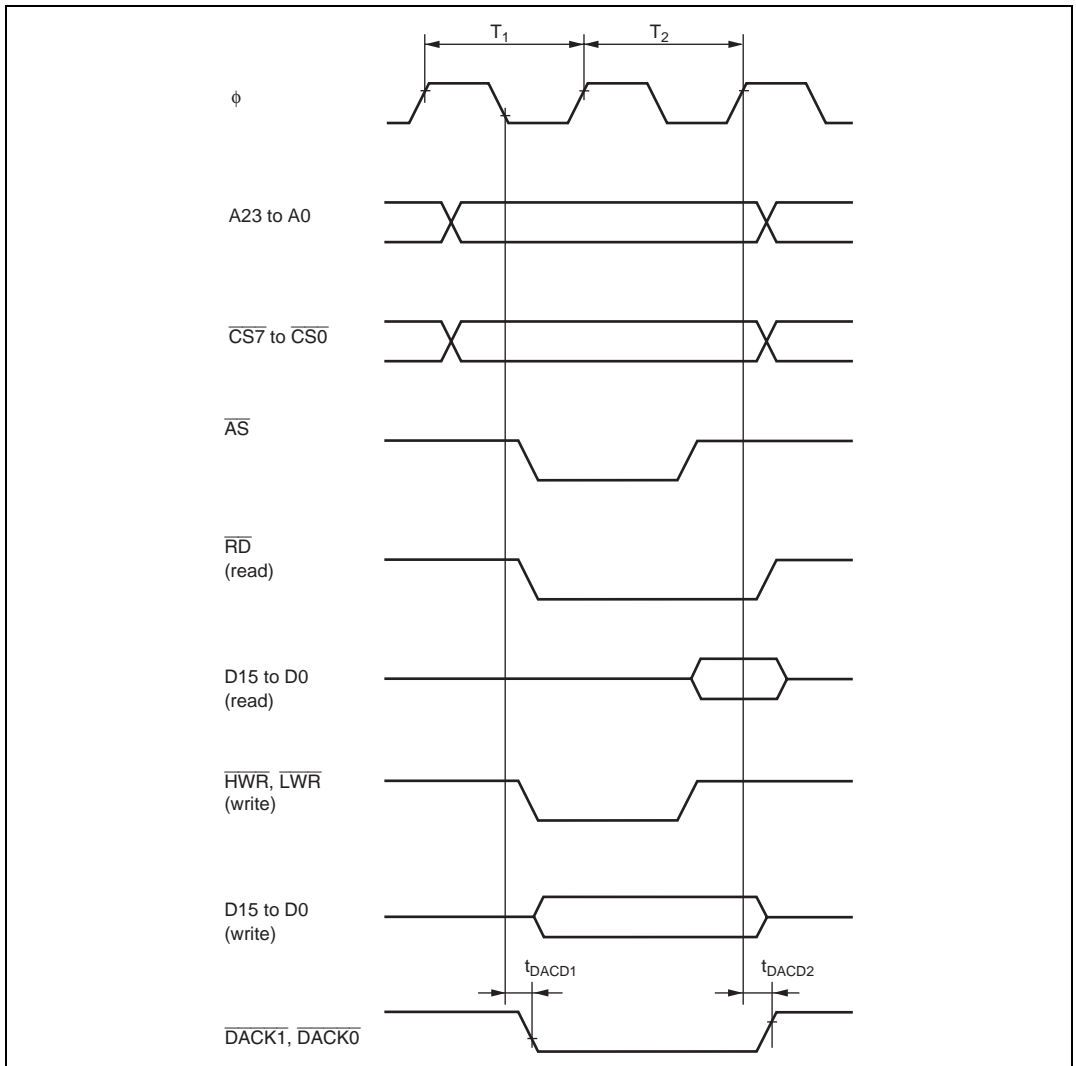


Figure 27.20 DMAC Single Address Transfer Timing (Two-State Access)

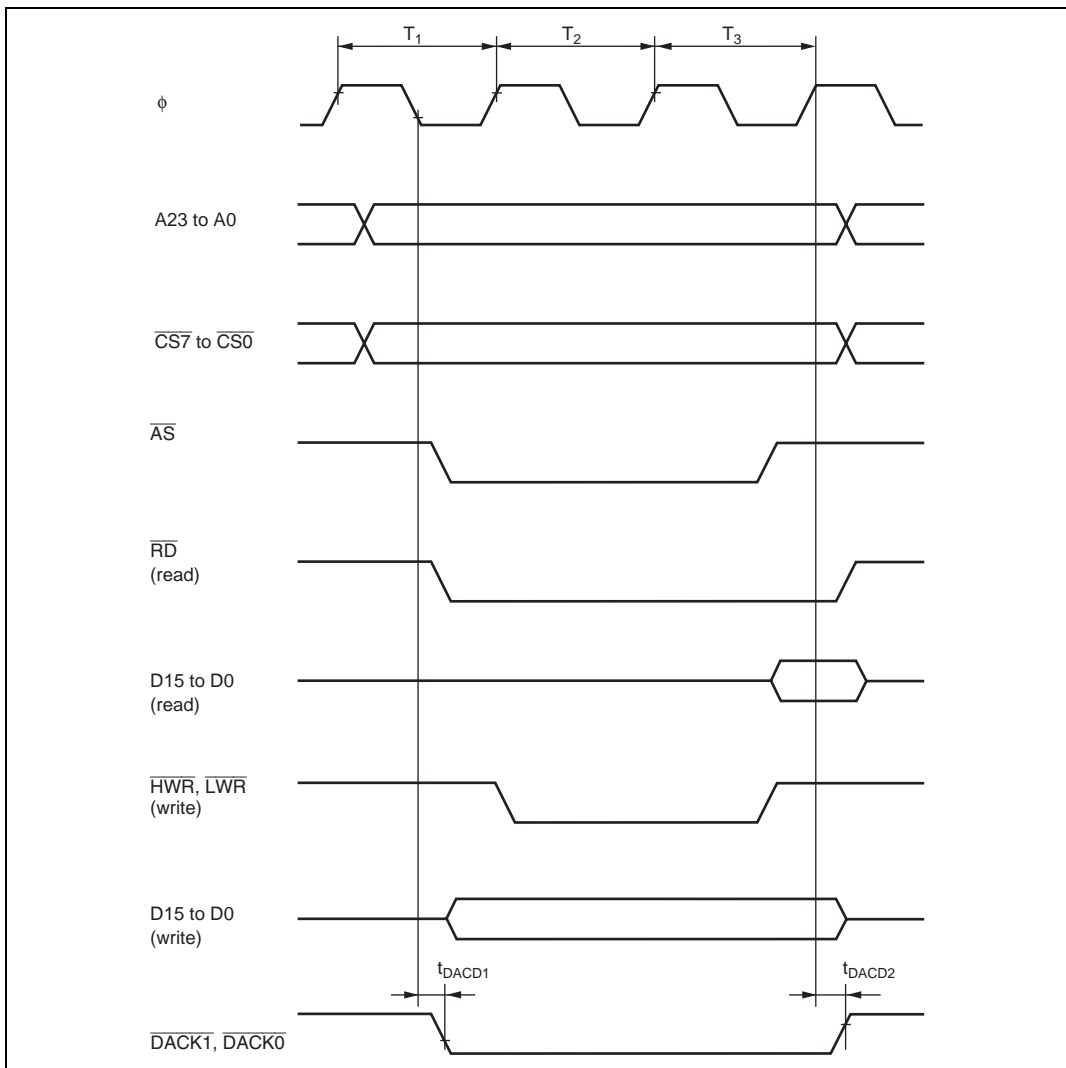


Figure 27.21 DMAC Single Address Transfer Timing (Three-State Access)

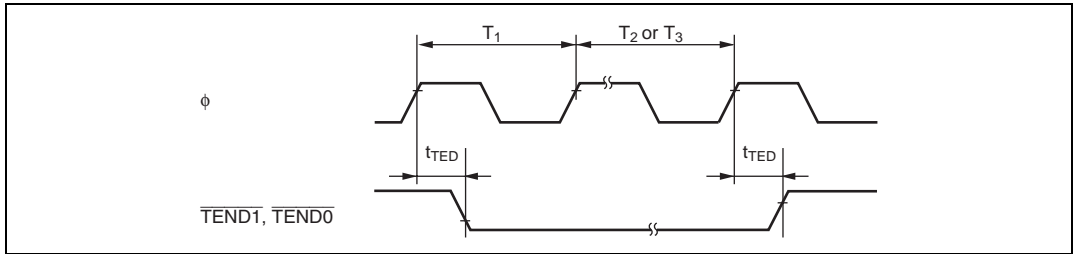


Figure 27.22 DMAC TEND Output Timing

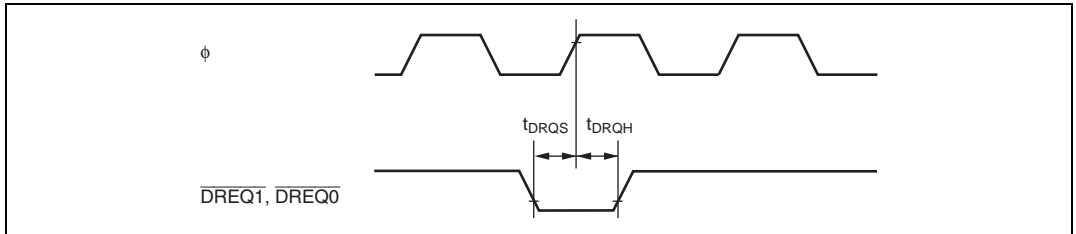


Figure 27.23 DMAC DREQ Input Timing

27.7.4 Timing of On-Chip Peripheral Modules

Figures 27.24 to 27.34 show the timing of on-chip peripheral modules.

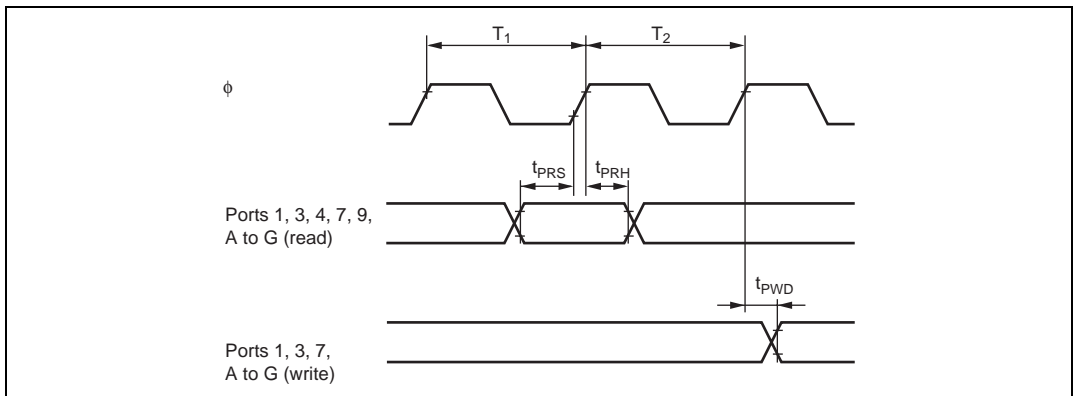


Figure 27.24 I/O Port Input/Output Timing

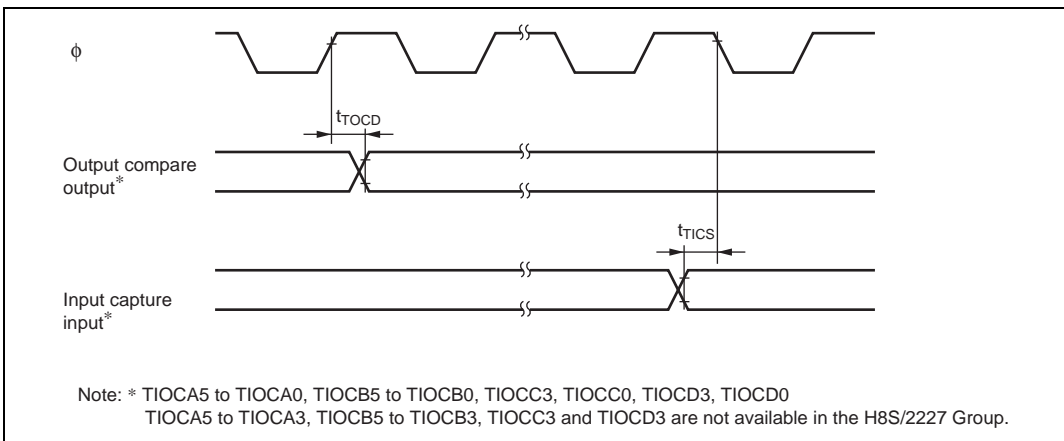


Figure 27.25 TPU Input/Output Timing

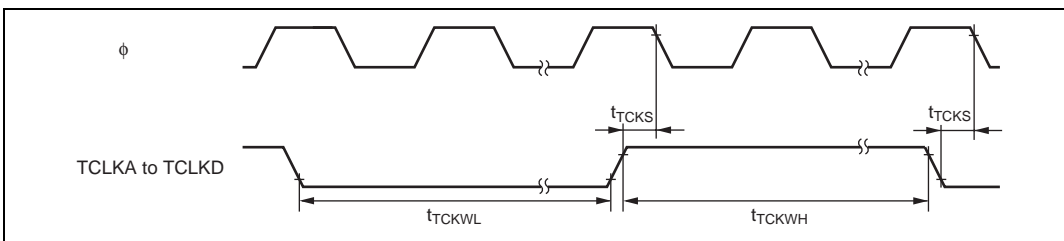


Figure 27.26 TPU Clock Input Timing

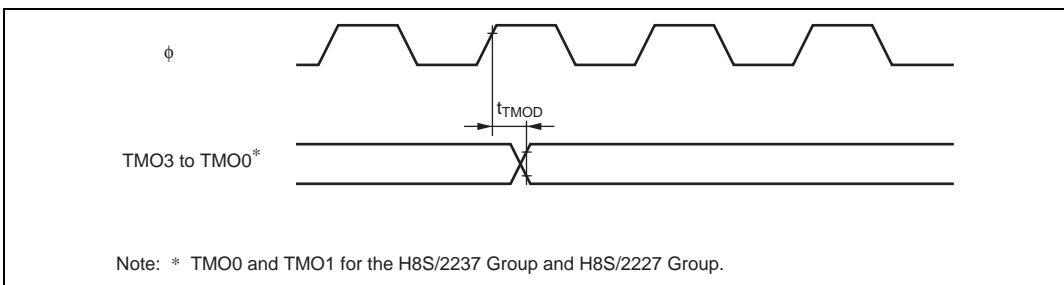


Figure 27.27 8-Bit Timer Output Timing

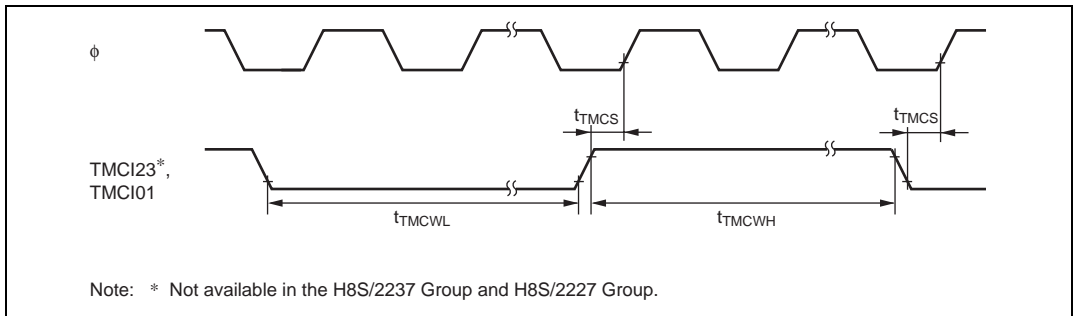


Figure 27.28 8-Bit Timer Clock Input Timing

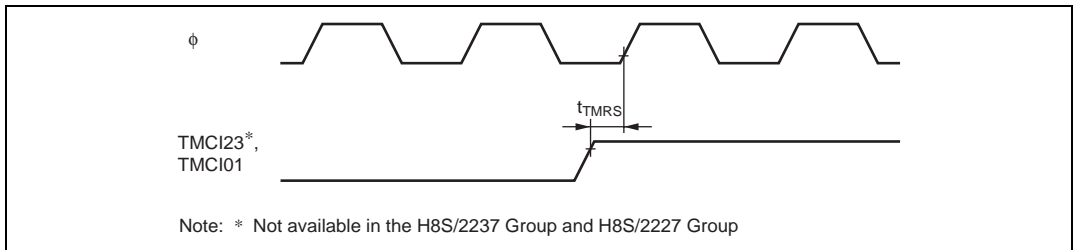


Figure 27.29 8-Bit Timer Reset Input Timing

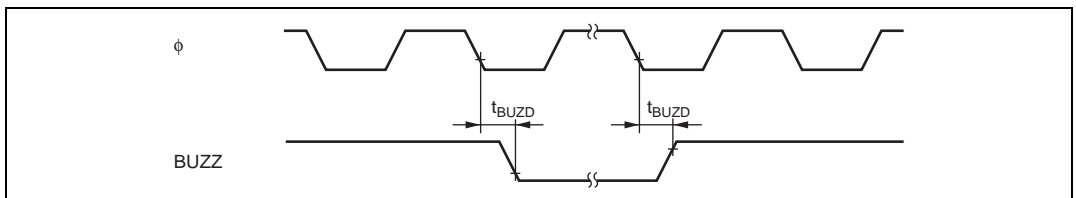


Figure 27.30 WDT_1 Output Timing

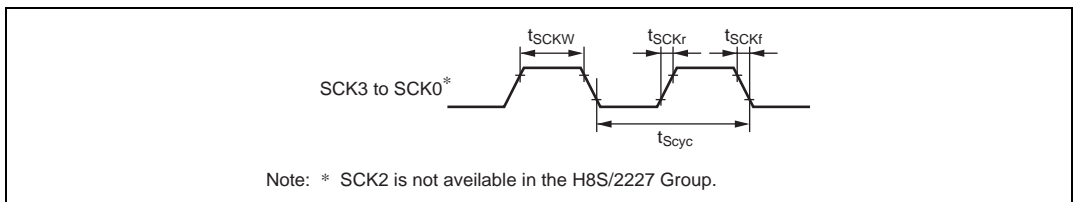
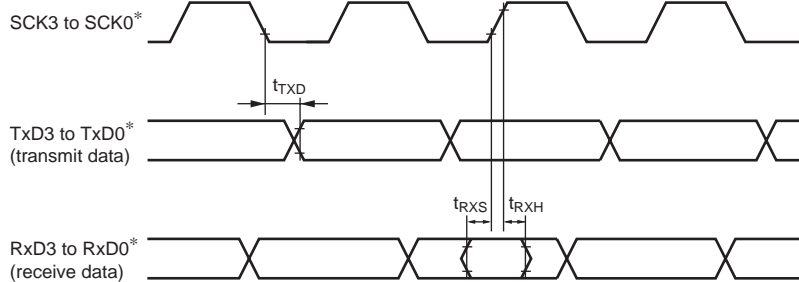


Figure 27.31 SCK Clock Input Timing



Note: * SCK2, TxD2, and RxD2 are not available in the H8S/2227 Group.

Figure 27.32 SCI Input/Output Timing (Clocked Synchronous Mode)

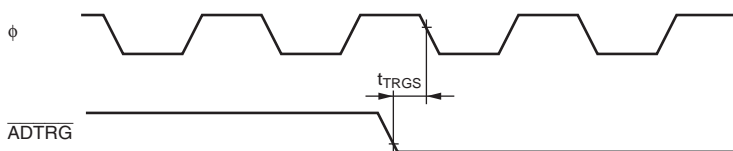
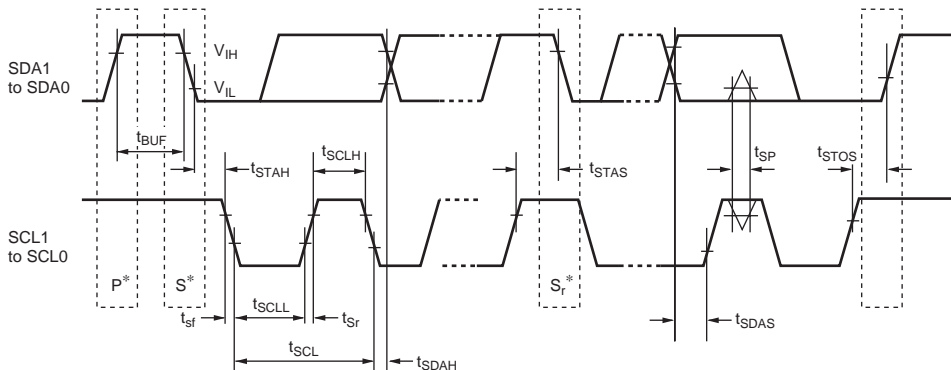


Figure 27.33 A/D Converter External Trigger Input Timing



Note: * S, P, and S_r indicate the following conditions.

S: Start condition

P: Stop condition

S_r: Retransmission start condition

Figure 27.34 I²C Bus Interface Input/Output Timing (Optional)

27.8 Usage Note

Though the F-ZTAT version and the masked ROM version satisfy electrical characteristics described in this manual, the actual value of electrical characteristics, operating margin, and noise margin may differ due to the differences of production process, on-chip ROM, and layout patterning.

When the system has been evaluated with the F-ZTAT version, the equivalent evaluation should be implemented to the masked ROM version when shifted to the masked ROM version.

Appendix A I/O Port States in Each Pin State

A.1 I/O Port State in Each Pin State

| Port Name Pin Name | MCU Operating Mode | Power-On Reset | Manual Reset | Hardware Standby Mode | Software Standby Mode, Watch Mode | Bus Mastership Release State | Program Execution State, Sleep Mode, Subsleep Mode |
|--|--------------------------|-------------------|-----------------|-----------------------------|--|---------------------------------------|---|
| P17 to P14 | 4 to 7 | T | keep | T | keep | keep | I/O port |
| P13/TIOCD0/ TCLKB/A23 | 7 | T | keep | T | keep | keep | I/O port |
| P12/TIOCC0/ TCLKA/A22 | | | | | | | |
| P11/TIOCB0/ A21 | | | | | | | |
| When the address output is selected by the AEn bit | 4 to 6 | T | keep | T | [OPE = 0] T [OPE = 1] keep | T | Address output |
| When a port is selected | 4 to 6 | T | keep | T | keep | keep | I/O port |
| P10/TIOCA0/ DACK0* ³ /A20 | 7 | T | keep | T | keep | keep | I/O port |
| When the address output is selected by the AEn bit | 4, 5 6 | L T | keep | T | [OPE = 0] T [OPE = 1] keep | T | Address output |
| When a port is selected | 4 to 6 | T* ¹ | keep | T | keep | keep | I/O port |

| Port Name Pin Name | MCU Operating Mode | Power-On Reset | Manual Reset | Hardware Standby Mode | Software Standby Mode, Watch Mode | Bus Mastership Release State | Program Execution State, Sleep Mode, Subsleep Mode |
|--|-----------------------|-------------------|-----------------|-----------------------------|--|---------------------------------------|---|
| Port 3 | 4 to 7 | T | keep | T | keep | keep | I/O port |
| Port 4 | 4 to 7 | T | T | T | T | T | Input port |
| P77 to P74 | 4 to 7 | T | keep | T | keep | keep | I/O port |
| P73/TMO1/ TEND1* ³ /CS7 | 7 | T | keep | T | keep | keep | I/O port |
| P72/TMO0/ TEND0* ³ /CS6 | 4 to 6 | T | keep | T | [DDR = 0] OPE = 0] T | T | [DDR = 0] Input port |
| P71/TMRI23* ² / TMCI23* ² / DREQ1* ³ /CS5 | | | | | [DDR = 0] OPE = 1] H | | [DDR = 1] CS7 to CS4 |
| P70/TMRI01/ TMCI01/ DREQ0* ³ /CS4 | | | | | | | |
| P97/DA1* ⁴ P96/DA0* ⁴ | 4 to 7 | T | T | T | [DAOEn = 1] keep [DAOEn = 0] T | keep | Input port |
| Port A | 7 | T | keep | T | keep | keep | I/O port |
| When the address output is selected by the AEn bit | 4, 5 <hr/> 6 | L <hr/> T | keep | T | [OPE = 0] T [OPE = 1] keep | T | Address output |
| When a port is selected | 4 to 6 | T* ¹ | keep | T | keep | keep | I/O port |

| Port Name | MCU Operating Mode | Power-On Reset | Manual Reset | Hardware Standby Mode | Software Standby Mode, Watch Mode | Bus Mastership Release State | Program Execution State, Sleep Mode, Subsleep Mode |
|---|--------------------------|-------------------|-----------------|-----------------------------|---|---------------------------------------|---|
| Port B | 7 | T | keep | T | keep | keep | I/O port |
| When the address output is selected by the AEn bit | 4, 5 <hr/> 6 | L <hr/> T | keep | T | [OPE = 0] T [OPE = 1] keep | T | Address output |
| When a port is selected | 4 to 6 | T* ¹ | keep | T | keep | keep | I/O port |
| Port C | 4, 5 | L | keep | T | [OPE = 0] T [OPE = 1] keep | T | Address output |
| | 6 | T | keep | T | [DDR · OPE = 0] T [DDR · OPE = 1] keep | T | [DDR = 0] Input port [DDR = 1] Address output |
| | 7 | T | keep | T | keep | keep | I/O port |
| Port D | 4 to 6 | T | T | T | T | T | Data bus |
| | 7 | T | keep | T | keep | keep | I/O port |
| Port E 8-bit bus | 4 to 6 | T | keep | T | keep | keep | I/O port |
| 16-bit bus | 4 to 6 | T | T | T | T | T | Data bus |
| | 7 | T | keep | T | keep | keep | I/O port |

| Port Name Pin Name | MCU Operating Mode | Power-On Reset | Manual Reset | Hardware Standby Mode | Software Standby Mode, Watch Mode | Bus Mastership Release State | Program Execution State, Sleep Mode, Subsleep Mode |
|--|-----------------------|--------------------------|---|-----------------------------|--|--|---|
| PF7/ ϕ | 4 to 6 | Clock output | [DDR = 0] Input port [DDR = 1] Clock output | T | [DDR = 0] Input port [DDR = 1] H | [DDR = 0] Input port [DDR = 1] Clock output | [DDR = 0] Input port [DDR = 1] Clock output |
| | 7 | T | keep | T | [DDR = 0] Input port [DDR = 1] H | [DDR = 0] Input port [DDR = 1] Clock output | [DDR = 0] Input port [DDR = 1] Clock output |
| PF6/ $\overline{\text{AS}}$ | 4 to 6 | H | H | T | [OPE = 0] T | T | $\overline{\text{AS}}$, $\overline{\text{RD}}$, $\overline{\text{HWR}}$ |
| PF5/ $\overline{\text{RD}}$ | | | | | [OPE = 1] H | | |
| PF4/ $\overline{\text{HWR}}$ | 7 | T | keep | T | keep | keep | I/O port |
| PF3/ $\overline{\text{LWR}}$ / $\overline{\text{ADTRG}}/\overline{\text{IRQ3}}$ | 7 | T | keep | T | keep | keep | I/O port |
| 8-bit bus | 4 to 6 | (Mode 4) | keep | T | keep | keep | I/O port |
| 16-bit bus | 4 to 6 | H (Mode 5, 6) T | H | T | [OPE = 0] T [OPE = 1] H | T | $\overline{\text{LWR}}$ |
| PF2/ $\overline{\text{WAIT}}$ | 4 to 6 | T | keep | T | [WAITE = 0] keep [WAITE = 1] T | [WAITE = 0] keep [WAITE = 1] T | [WAITE = 0] I/O port [WAITE = 1] $\overline{\text{WAIT}}$ |
| | 7 | T | keep | T | keep | keep | I/O port |
| PF1/ $\overline{\text{BACK}}$ / BUZZ | 4 to 6 | T | keep | T | [BRLE = 0] keep [BRLE = 1] H | L | [BRLE = 0] I/O port [BRLE = 1] $\overline{\text{BACK}}$ |
| | 7 | T | keep | T | keep | keep | I/O port |

| Port Name Pin Name | MCU Operating Mode | Power-On Reset | Manual Reset | Hardware Standby Mode | Software Standby Mode, Watch Mode | Bus Mastership Release State | Program Execution State, Sleep Mode, Subsleep Mode |
|---|--------------------------|-------------------|-----------------|-----------------------------|--|---------------------------------------|---|
| PF0/ $\overline{\text{BREQ}}$ / $\overline{\text{IRQ2}}$ | 4 to 6 | T | keep | T | [BRLE = 0] keep | T | [BRLE = 0] I/O port |
| | | | | | [BRLE = 1] T | | [BRLE = 1] $\overline{\text{BREQ}}$ |
| | 7 | T | keep | T | keep | keep | I/O port |
| PG4/ $\overline{\text{CS0}}$ | 4, 5 | H | keep | T | [DDR · OPE = 0] T | T | [DDR = 0] I/O port |
| | 6 | T | | | [DDR · OPE = 1] H | | [DDR = 1] $\overline{\text{CS0}}$ (H in sleep mode and subsleep mode.) |
| | 7 | T | keep | T | keep | keep | I/O port |
| PG3/ $\overline{\text{CS1}}$ | 4 to 6 | T | keep | T | [DDR · OPE = 0] T | T | [DDR = 0] Input port |
| PG2/ $\overline{\text{CS2}}$ | | | | | | | |
| PG1/ $\overline{\text{CS3}}$ / $\overline{\text{IRQ7}}$ | | | | | [DDR · OPE = 1] H | | [DDR = 1] $\overline{\text{CS1}}$ to $\overline{\text{CS3}}$ |
| | 7 | T | keep | T | keep | keep | I/O port |
| PG0/ $\overline{\text{IRQ6}}$ | 4 to 7 | T | keep | T | keep | keep | I/O port |

Legend:

H: High level

L: Low level

T: High-impedance

keep: The input port becomes high-impedance, and the output port retains its state

DDR: Data direction register

OPE: Output port enable

WAITE: Wait input enable

BRLE: Bus release enable

Notes: 1. The port state is L (address input) in modes 4 and 5.

2. Not available in the H8S/2237 Group and H8S/2227 Group.

3. Supported only by the H8S/2239 Group.

4. Not available in the H8S/2227 Group.

Appendix B Product Codes

Table B.1 Product Codes of H8S/2258 Group

| Product Type | | | Product Code | Mark Code | Package (Package Code) |
|--------------|----------------------------|--|--------------|-------------------|---------------------------|
| H8S/2258 | Flash memory version | Standard product | HD64F2258 | HD64F2258TE13 | 100-pin TQFP (TFP-100B) |
| | | | | HD64F2258F13 | 100-pin QFP (FP-100A) |
| | | | | HD64F2258FA13 | 100-pin QFP (FP-100B) |
| | Masked ROM version | Standard product | HD6432258 | HD6432258(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432258(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432258(***)FA | 100-pin QFP (FP-100B) |
| | | | HD6432256 | HD6432256(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432256(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432256(***)FA | 100-pin QFP (FP-100B) |
| | | On-chip I ² C bus interface product | HD6432258W | HD6432258W(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432258W(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432258W(***)FA | 100-pin QFP (FP-100B) |
| | | | HD6432256W | HD6432256W(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432256W(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432256W(***)FA | 100-pin QFP (FP-100B) |

Legend:

(***) : ROM code

Note: A standard product of F-ZTAT version includes an I²C bus interface.

Please contact Renesas Technology agency to confirm the current status of each product.

Table B.2 Product Codes of H8S/2239 Group

| Product Type | | | Product Code | Mark Code | Package (Package Code) |
|--------------|----------------------------|--|--------------|-------------------|---------------------------|
| H8S/2239 | Flash memory version | Standard product | HD64F2239 | HD64F2239TE20 | 100-pin TQFP (TFP-100B) |
| | | | | HD64F2239TF20 | 100-pin TQFP (TFP-100G) |
| | | | | HD64F2239FA20 | 100-pin QFP (FP-100B) |
| | | | | HD64F2239BQ20 | 112-pin TFBGA (TBP-112A) |
| | | | | HD64F2239TE16 | 100-pin TQFP (TFP-100B) |
| | | | | HD64F2239TF16 | 100-pin TQFP (TFP-100G) |
| | | | | HD64F2239FA16 | 100-pin QFP (FP-100B) |
| | | | | HD64F2239BQ16 | 112-pin TFBGA (TBP-112A) |
| | Masked ROM version | Standard product | HD6432239 | HD6432239(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432239(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432239(***)FA | 100-pin QFP (FP-100B) |
| | | On-chip I ² C bus interface product | HD6432239W | HD6432239W(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432239W(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432239W(***)FA | 100-pin QFP (FP-100B) |

Legend:

(***) : ROM code

Note: A standard product of F-ZTAT version includes an I²C bus interface.

Please contact Renesas Technology agency to confirm the current status of each product.

Table B.3 Product Codes of H8S/2238 Group

| Product Type | | | Product Code | Mark Code | Package (Package Code) |
|--------------|----------------------------|---|--------------|--------------------|---------------------------|
| H8S/2238B | Flash memory version | 5-V version | HD64F2238B | HD64F2238BTE13 | 100-pin TQFP (TFP-100B) |
| | | | | HD64F2238BTF13 | 100-pin TQFP (TFP-100G) |
| | | | | HD64F2238BF13 | 100-pin QFP (FP-100A) |
| | | | | HD64F2238BFA13 | 100-pin QFP (FP-100B) |
| | Masked ROM version | 5-V version | HD6432238B | HD6432238B(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432238B(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432238B(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432238B(***)FA | 100-pin QFP (FP-100B) |
| | | On-chip I ² C bus interface product (5-V version) | HD6432238BW | HD6432238BW(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432238BW(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432238BW(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432238BW(***)FA | 100-pin QFP (FP-100B) |
| H8S/2238R | Flash memory version | 3-V version | HD64F2238R | HD64F2238RTE13 | 100-pin TQFP (TFP-100B) |
| | | | | HD64F2238RTE13 | 100-pin TQFP (TFP-100G) |
| | | | | HD64F2238RFA13 | 100-pin QFP (FP-100B) |
| | | | | HD64F2238RBQ13 | 112-pin TFBGA (TBP-112A) |
| | | | | HD64F2238RBR13 | 112-pin LFBGA (BP-112) |
| | | 2.2-V version | HD64F2238R | HD64F2238RTE6 | 100-pin TQFP (TFP-100B) |
| | | | | HD64F2238RTE6 | 100-pin TQFP (TFP-100G) |
| | | | | HD64F2238RFA6 | 100-pin QFP (FP-100B) |
| | | | | HD64F2238RBQ6 | 112-pin TFBGA (TBP-112A) |
| | | | | HD64F2238RBR6 | 112-pin LFBGA (BP-112) |
| | | 3-V version, 2.2-V version | HD6432238R | HD6432238R(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432238R(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432238R(***)FA | 100-pin QFP (FP-100B) |
| | | | | HD6432238RW(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432238RW(***)TF | 100-pin TQFP (TFP-100G) |
| | | On-chip I ² C bus interface product (3-V version) | HD6432238RW | HD6432238RW(***)FA | 100-pin QFP (FP-100B) |

| Product Type | | | Product Code | Mark Code | Package (Package Code) |
|--------------|---|-------------------------------|--------------|--------------------|---------------------------|
| H8S/2236B | Masked ROM version | 5-V version | HD6432236B | HD6432236B(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432236B(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432236B(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432236B(***)FA | 100-pin QFP (FP-100B) |
| | On-chip I ² C bus interface product (5-V version) | HD6432236BW | | HD6432236BW(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432236BW(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432236BW(***)F | 100-pin QFP (FP-100A) |
| | | | | HD6432236BW(***)FA | 100-pin QFP (FP-100B) |
| H8S/2236R | Masked ROM version | 3-V version, 2.2-V version | HD6432236R | HD6432236R(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432236R(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432236R(***)FA | 100-pin QFP (FP-100B) |
| | On-chip I ² C bus interface product (3-V version) | HD6432236RW | | HD6432236RW(***)TE | 100-pin TQFP (TFP-100B) |
| | | | | HD6432236RW(***)TF | 100-pin TQFP (TFP-100G) |
| | | | | HD6432236RW(***)FA | 100-pin QFP (FP-100B) |

Legend:

(***) : ROM code

Note: Please contact Renesas Technology agency to confirm the current status of each product.

Table B.4 Product Codes of H8S/2237 Group and H8S/2227 Group

| Product Type | | Product Code | Mark Code | Package (Package Code) |
|--------------|----------------------|--------------|------------------|-------------------------|
| H8S/2237 | Flash memory version | HD6472237 | HD6472237TE10 | 100-pin TQFP (TFP-100B) |
| | | | HD6472237TF10 | 100-pin TQFP (TFP-100G) |
| | | | HD6472237F10 | 100-pin QFP (FP-100A) |
| | | | HD6472237FA10 | 100-pin QFP (FP-100B) |
| | Masked ROM version | HD6432237 | HD6432237(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432237(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432237(***)F | 100-pin QFP (FP-100A) |
| | | | HD6432237(***)FA | 100-pin QFP (FP-100B) |
| H8S/2235 | Masked ROM version | HD6432235 | HD6432235(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432235(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432235(***)F | 100-pin QFP (FP-100A) |
| | | | HD6432235(***)FA | 100-pin QFP (FP-100B) |
| H8S/2233 | Masked ROM version | HD6432233 | HD6432233(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432233(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432233(***)F | 100-pin QFP (FP-100A) |
| | | | HD6432233(***)FA | 100-pin QFP (FP-100B) |
| H8S/2227 | Flash memory version | HD64F2227 | HD64F2227TE13 | 100-pin TQFP (TFP-100B) |
| | | | HD64F2227TF13 | 100-pin TQFP (TFP-100G) |
| | Masked ROM version | HD6432227 | HD6432227(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432227(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432227(***)F | 100-pin QFP (FP-100A) |
| | | | HD6432227(***)FA | 100-pin QFP (FP-100B) |
| H8S/2225* | Masked ROM version | HD6432225 | HD6432225(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432225(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432225(***)FA | 100-pin QFP (FP-100B) |
| H8S/2224* | Masked ROM version | HD6432224 | HD6432224(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432224(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432224(***)FA | 100-pin QFP (FP-100B) |
| H8S/2223* | Masked ROM version | HD6432223 | HD6432223(***)TE | 100-pin TQFP (TFP-100B) |
| | | | HD6432223(***)TF | 100-pin TQFP (TFP-100G) |
| | | | HD6432223(***)FA | 100-pin QFP (FP-100B) |

Legend:

(***): ROM code

Note: * The 100-pin QFP (FP-100A) is not available for the HD6432225, HD6432224, and HD6432223. When the 100-pin QFP (FP-100A) is necessary, choose HD6432235(***)F, HD6432233(***)F, or HD6432227(***)F.

Appendix C Package Dimensions

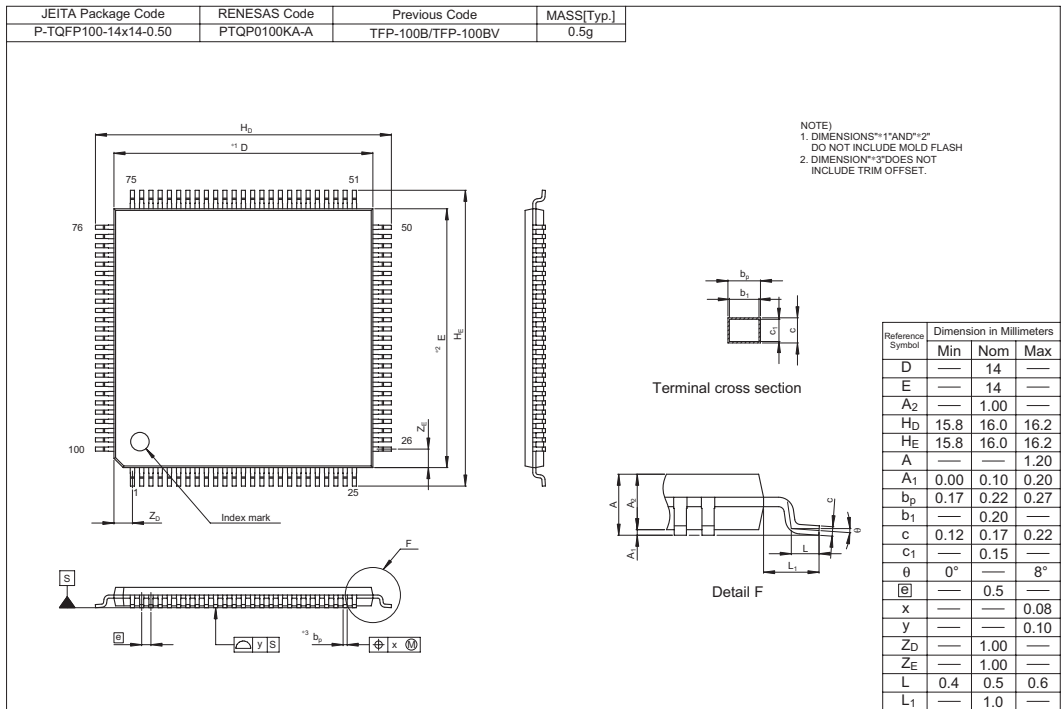


Figure C.1 TFP-100B Package Dimensions

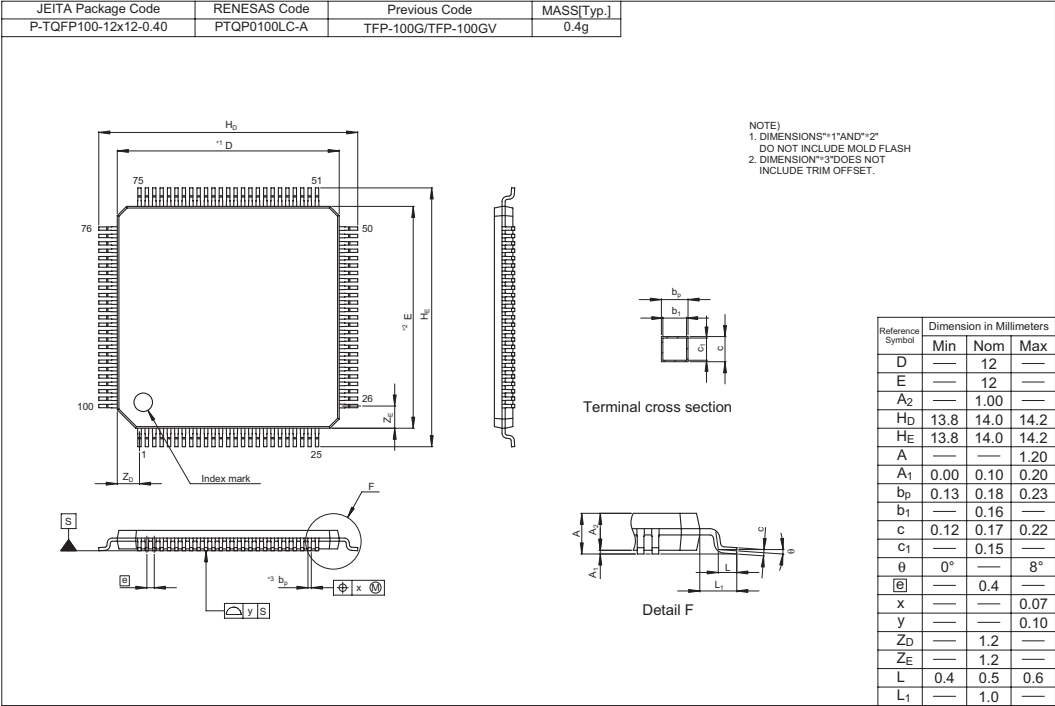
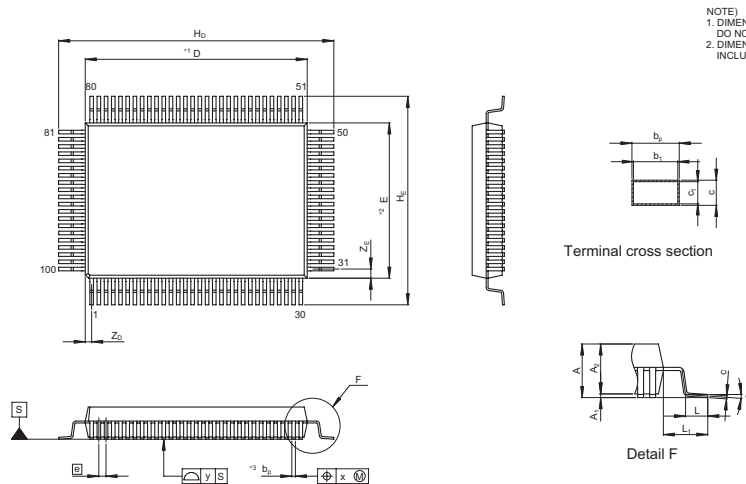


Figure C.2 TFP-100G Package Dimensions

| JEITA Package Code | RENESAS Code | Previous Code | MASS[Typ.] |
|---------------------|--------------|------------------|------------|
| P-QFP100-14x20-0.65 | PRQP0100JE-B | FP-100A/FP-100AV | 1.7g |



| Reference Symbol | Dimension in Millimeters | | |
|------------------|--------------------------|------|------|
| | Min | Nom | Max |
| D | — | 20 | — |
| E | — | 14 | — |
| A ₂ | — | 2.70 | — |
| H _D | 24.4 | 24.8 | 25.2 |
| H _E | 18.4 | 18.8 | 19.2 |
| A | — | — | 3.10 |
| A ₁ | 0.00 | 0.20 | 0.30 |
| b _p | 0.24 | 0.32 | 0.40 |
| b ₁ | — | 0.30 | — |
| c | 0.12 | 0.17 | 0.22 |
| c ₁ | — | 0.15 | — |
| θ | 0° | — | 10° |
| Ⓢ | — | 0.65 | — |
| x | — | — | 0.13 |
| y | — | — | 0.15 |
| Z _D | — | 0.58 | — |
| Z _E | — | 0.83 | — |
| L | 1.0 | 1.2 | 1.4 |
| L ₁ | — | 2.4 | — |

Figure C.3 FP-100A Package Dimensions

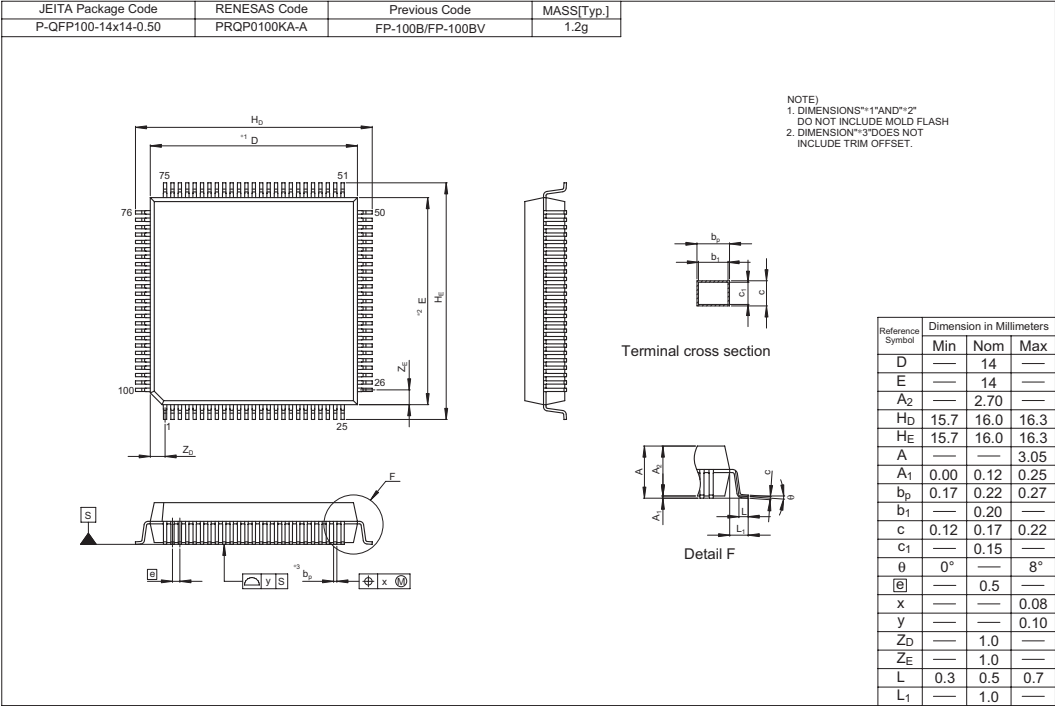


Figure C.4 FP-100B Package Dimensions

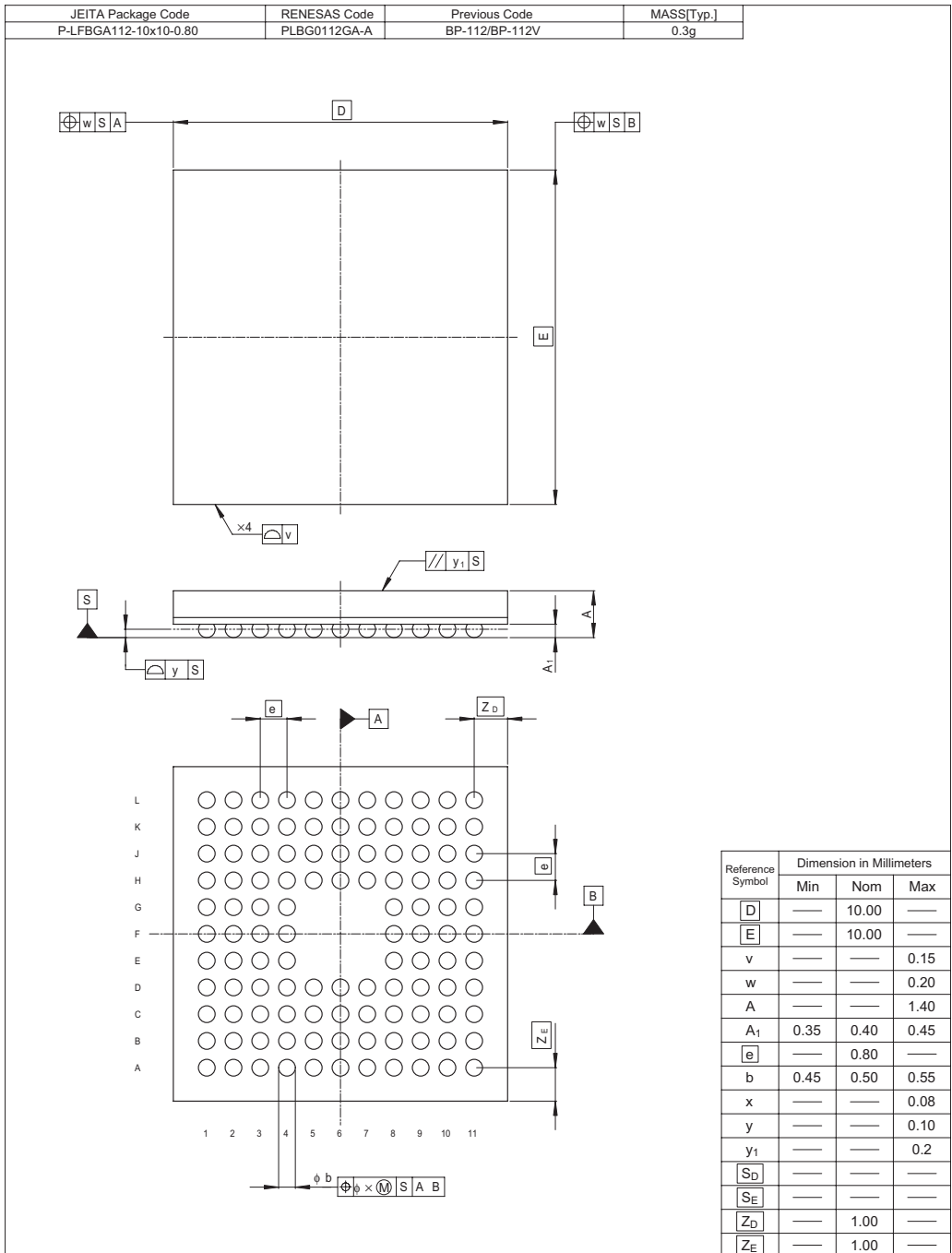


Figure C.5 BP-112 Package Dimensions

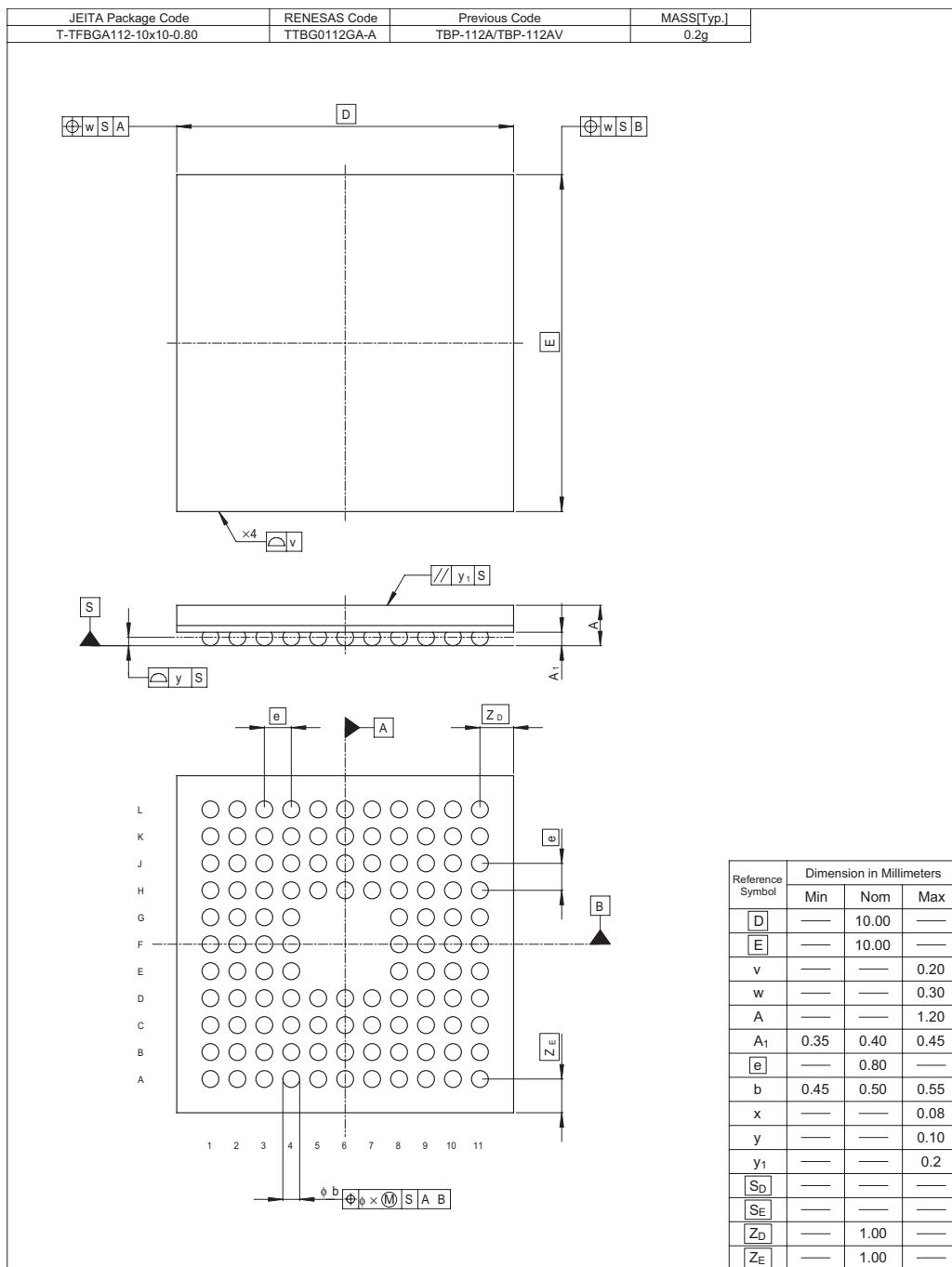


Figure C.6 TBP-112A, TBP-112AV Package Dimensions

Index

| | | | |
|---|--------------------|--------------------------------------|--------------------|
| 16-Bit Timer Pulse Unit..... | 359 | CMIA..... | 458 |
| A/D Conversion Time..... | 699 | CMIB..... | 458 |
| A/D Converter..... | 689 | Condition Field..... | 89 |
| A/D Converter Activation..... | 426 | Condition-Code Register..... | 74 |
| Absolute Address..... | 91 | CPU..... | 63 |
| ABWCR..... | 812, 823, 834 | CRA..... | 285, 808, 818, 830 |
| Activation by Software..... | 301 | CRB..... | 285, 808, 818, 830 |
| ADCR..... | 695, 816, 828, 838 | D/A Converter..... | 707 |
| ADCSR..... | 693, 816, 828, 838 | DACR..... | 709, 809, 819, 831 |
| ADDR..... | 692, 816, 828, 838 | DADR..... | 708, 809, 819, 831 |
| Address Space..... | 70 | DAR..... | 285, 808, 818, 830 |
| Addressing Modes..... | 90 | Data Transfer Controller..... | 281 |
| ADI..... | 701 | Data Transfer Instructions..... | 81 |
| Advanced Mode..... | 67 | DDCSWR..... | 653, 809, 819, 831 |
| Analog Input Channel..... | 692 | DEND0A..... | 275 |
| Arithmetic Operations Instructions..... | 82 | DEND0B..... | 275 |
| ASTCR..... | 812, 823, 834 | DEND1A..... | 275 |
| Asynchronous Mode..... | 585 | DEND1B..... | 275 |
| BARA..... | 158, 810, 820, 832 | DMABCR..... | 814, 826, 836 |
| BARB..... | 159, 810, 820, 832 | DMACR..... | 814, 826, 836 |
| Bcc..... | 87 | DMATCR..... | 814, 826, 836 |
| BCRA..... | 159, 810, 820, 832 | DMAWER..... | 814, 826, 836 |
| BCRB..... | 160, 810, 820, 832 | DTC Vector Table..... | 289 |
| BCRH..... | 812, 823, 834 | DTCER..... | 286, 810, 820, 832 |
| BCRL..... | 812, 823, 834 | DTVECR..... | 287, 810, 820, 832 |
| Bit Manipulation Instructions..... | 85 | EBR1..... | 817, 828, 838 |
| bit rate..... | 571 | EBR2..... | 817, 828, 838 |
| Bit Rate..... | 571 | Effective Address..... | 94 |
| Block Data Transfer Instructions..... | 89 | Effective Address Extension..... | 89 |
| block transfer mode..... | 295 | ERI..... | 623 |
| Branch Instructions..... | 87 | ETCR..... | 813, 823, 834 |
| Break..... | 625 | Exception Handling..... | 119 |
| break address..... | 157, 160 | Exception Handling Vector Table..... | 120 |
| break conditions..... | 160 | Extended Control Register..... | 73 |
| BRR..... | 571, 815, 827, 837 | External Trigger Input Timing..... | 701 |
| Buffer Operation..... | 405 | FLMCR1..... | 816, 828, 838 |
| Cascaded Operation..... | 409 | FLMCR2..... | 817, 828, 838 |
| Chain Transfer..... | 297 | FLPWCR..... | 817, 828, 838 |

| | | | |
|---------------------------------------|--------------------|------------------------------------|---------------|
| framing error | 592 | overflow error..... | 592 |
| Free-running count operation | 399 | OVI..... | 458 |
| General Registers | 72 | P1DDR | 810, 821, 832 |
| I ² C bus format | 653 | P1DR | 813, 824, 835 |
| I ² C Bus Interface | 633 | P3DDR | 810, 821, 832 |
| ICCR | 644, 815, 827, 836 | P3DR | 813, 824, 835 |
| ICDR | 637, 815, 827, 837 | P3ODR | 811, 821, 832 |
| ICMR | 640, 827, 837 | P7DDR | 810, 821, 832 |
| ICMR/SAR..... | 815 | P7DR..... | 813, 824, 835 |
| ICSR..... | 649, 815, 827, 837 | PADDR | 810, 821, 832 |
| IER | 810, 820, 832 | PADR | 813, 824, 835 |
| Immediate..... | 92 | PAODR | 811, 821, 833 |
| Input Capture Function | 402 | PAPCR | 811, 821, 832 |
| Instruction Set | 79 | parity error..... | 592 |
| Internal Block Diagram..... | 4 | PBDDR | 810, 821, 832 |
| Interrupt Mask Bit..... | 74 | PBDR | 813, 824, 835 |
| Interrupts | 124 | PBPCR | 811, 821, 832 |
| Interval Timer Mode | 474 | PC Break Controller | 157 |
| IOAR..... | 813, 823, 834 | PCDDR | 810, 821, 832 |
| IPR | 812, 822, 834 | PCDR | 813, 824, 835 |
| ISCR..... | 810, 820, 832 | PCPCR | 811, 821, 832 |
| ISR | 810, 820, 832 | PDDDR | 810, 821, 832 |
| List of Registers | 807 | PDDR | 813, 824, 835 |
| Logic Operations Instructions | 84 | PDPCR | 811, 821, 832 |
| LPWRCR | 810, 820, 831 | PEDDR..... | 810, 821, 832 |
| MAR | 812, 823, 834 | PEDR | 813, 824, 835 |
| Mark State | 625 | PEPCR | 811, 821, 832 |
| Mask ROM..... | 753 | Periodic count operation..... | 399 |
| MDCR..... | 104, 810, 820, 831 | PFCR..... | 810, 820, 831 |
| Memory Indirect..... | 93 | PFDDR..... | 811, 821, 832 |
| Memory Map..... | 109 | PFDR..... | 813, 824, 835 |
| MRA | 283, 808, 818, 830 | PGDDR | 811, 821, 832 |
| MRB..... | 284, 808, 818, 830 | PGDR | 813, 824, 835 |
| MSTPCR..... | 810, 820, 831 | Phase Counting Mode | 416 |
| Multiprocessor Communication Function | | Pin Arrangements in Each Mode..... | 20 |
| | 596 | Pin Functions..... | 44 |
| NMI interrupt request..... | 476 | pointer (SP) | 72 |
| Normal Mode | 66, 294, 302 | PORT1 | 817, 828, 838 |
| Operating Mode Selection..... | 103 | PORT3 | 817, 828, 838 |
| Operation Field..... | 89 | PORT4 | 817, 828, 838 |
| overflow | 474 | PORT7 | 817, 828, 838 |

| | | | |
|---|--|----------------------------------|-----------------------------------|
| PORT9 | 817, 828, 838 | Shift Instructions..... | 84 |
| PORTA | 817, 828, 838 | Single Mode..... | 697 |
| PORTB | 817, 828, 838 | Smart Card | 547 |
| PORTC | 817, 828, 838 | Smart Card Interface..... | 610 |
| PORTD | 817, 828, 838 | SMR..... | 553, 815, 826, 836 |
| PORTE..... | 817, 829, 838 | software activation..... | 298, 302 |
| PORTF..... | 817, 829, 838 | SSR | 563, 815, 827, 837 |
| PORTG | 817, 829, 838 | Stack Status..... | 125 |
| Program Counter..... | 73 | Stack Structure | 66, 69 |
| Program-Counter Relative | 92 | SWDTEND..... | 298 |
| PWM Modes | 411 | Synchronous Mode | 602 |
| RAMER | 812, 823, 834 | Synchronous Operation..... | 403 |
| RDR | 552, 815, 827, 837 | SYSCR..... | 105, 809, 820, 831 |
| Register Addresses (in address order)..... | 807 | System Control Instructions..... | 88 |
| Register Bits..... | 818 | TCI1U | 424 |
| Register Configuration..... | 71 | TCI1V | 424 |
| Register Direct..... | 91 | TCI2U | 424 |
| Register Field..... | 89 | TCI2V | 424 |
| Register Indirect..... | 91 | TCI3V | 424 |
| Register Indirect with Displacement..... | 91 | TCI4U | 424 |
| Register Indirect with Post-Increment | 91 | TCI4V | 424 |
| Register Indirect with Pre-Decrement..... | 91 | TCI5U | 424 |
| Register Information | 289 | TCI5V | 424 |
| Register States in Each Operating Mode | 830 | TCNT | 396, 468, 813, 815, 825, 826, |
| repeat mode..... | 294 | | 835, 836 |
| Reset | 121 | TCORA..... | 815, 826, 836 |
| Reset Exception Handling | 122 | TCORB | 815, 826, 836 |
| RSR..... | 552 | TCR | 367, 813, 814, 824, 826, 835, 836 |
| RSTCSR | 472, 815, 826, 836 | TCSR | 468, 815, 826, 836 |
| RXI | 623 | TDR | 552, 815, 827, 837 |
| SAR | 285, 639, 808, 815, 818, 827, 830, 837 | TEI | 623 |
| SARX..... | 639, 815, 827, 837 | TGI0A..... | 424 |
| SBYCR | 809, 820, 831 | TGI0B | 424 |
| Scan Mode | 698 | TGI0C..... | 424 |
| SCKCR | 810, 820, 831 | TGI0D..... | 424 |
| SCMR | 570, 815, 827, 837 | TGI0V..... | 424 |
| SCR..... | 557, 815, 827, 837 | TGI1A..... | 424 |
| SCRX..... | 643, 809, 819, 831 | TGI1B | 424 |
| SEMR | 581, 810, 820, 831 | TGI2A..... | 424 |
| Serial Communication Interface | 547 | TGI2B | 424 |
| serial format | 653 | TGI3A..... | 424 |

| | | | |
|---------------------|--------------------|---------------------------------------|-------------------------|
| TGI3B | 424 | Trap Instruction | 124 |
| TGI3C | 424 | TSR | 393, 553, 813, 825, 835 |
| TGI3D | 424 | TSTR | 396, 812, 822, 833 |
| TGI4A | 424 | TSYR | 397, 812, 822, 833 |
| TGI4B | 424 | TXI | 623 |
| TGI5A | 424 | vector number for the software | |
| TGI5B | 424 | activation interrupt | 287 |
| TGR..... | 396, 813, 825, 835 | Watchdog Timer..... | 465 |
| TIER..... | 391, 813, 824, 835 | Watchdog Timer Mode | 473 |
| TIOR | 373, 813, 824, 835 | Waveform Output by Compare Match | 400 |
| TMDR | 372, 813, 824, 835 | WCRH..... | 812, 823, 834 |
| Toggle output | 401, 461 | WCRL | 812, 823, 834 |
| Traces | 123 | WOVI..... | 476 |

**Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8S/2258, H8S/2239, H8S/2238, H8S/2237,
H8S/2227 Groups**

Publication Date: 1st Edition, September 2002

Rev.6.00, March 18, 2010

Published by: Sales Strategic Planning Div.
Renesas Technology Corp.

Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan



RENESAS SALES OFFICES

<http://www.renesas.com>

Refer to "<http://www.renesas.com/en/network>" for the latest and detailed information.

Renesas Technology America, Inc.

450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited

Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.

Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7858/7898

Renesas Technology Hong Kong Ltd.

7th Floor, North Tower, World Finance Centre, Harbour City, Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2377-3473

Renesas Technology Taiwan Co., Ltd.

10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 3518-3399

Renesas Technology Singapore Pte. Ltd.

1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.

Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology 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: <603> 7955-9390, Fax: <603> 7955-9510

H8S/2258, H8S/2239, H8S/2238,
H8S/2237, H8S/2227 Groups
Hardware Manual



Renesas Electronics Corporation

1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ09B0054-0600

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Renesas Electronics:

[DF2238RBR13WV](#) [DF2239FA20IV](#) [DF2239TE16V](#) [DF2239FA20V](#) [DF2238BFA13IV](#) [DF2239FA16V](#)
[DF2238BFA13V](#) [DF2239TF16V](#) [DF2239TF20V](#) [DF2238RTE13V](#) [DF2239TE20V](#) [DF2238BTF13V](#) [DF2238BTE13V](#)
[DF2238RFA13V](#) [DF2238RBR13V](#) [DF2258FA13IV](#) [DF2258FA13V](#) [DF2238RFA6V](#) [DF2238RTE6V](#) [DF2238RTF6V](#)