

---

## **Regarding the change of names mentioned in the document, such as Hitachi Electric and Hitachi XX, to Renesas Technology Corp.**

---

The semiconductor operations of Mitsubishi Electric and Hitachi were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Hitachi, Hitachi, Ltd., Hitachi Semiconductors, and other Hitachi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Renesas Technology Home Page: <http://www.renesas.com>

Renesas Technology Corp.  
Customer Support Dept.  
April 1, 2003

## Cautions

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.

Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.

The information described here may contain technical inaccuracies or typographical errors.

Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.

Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (<http://www.renesas.com>).

4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination. Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.

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.

## Hitachi 16-Bit Single-Chip Microcomputer

# H8S/2612 Series

H8S/2612

HD6432612

H8S/2611

HD6432611

H8S/2612F-ZTAT<sup>TM</sup>

HD64F2612

Hardware Manual



ADE-602-220C

Rev. 4.0

3/18/03

Hitachi, Ltd.

## Cautions

1. Hitachi neither warrants nor grants licenses of any rights of Hitachi's or any third party's patent, copyright, trademark, or other intellectual property rights for information contained in this document. Hitachi bears no responsibility for problems that may arise with third party's rights, including intellectual property rights, in connection with use of the information contained in this document.
2. Products and product specifications may be subject to change without notice. Confirm that you have received the latest product standards or specifications before final design, purchase or use.
3. Hitachi makes every attempt to ensure that its products are of high quality and reliability. However, contact Hitachi's sales office before using the product in an application that demands especially high quality and reliability or where its failure or malfunction may directly threaten human life or cause risk of bodily injury, such as aerospace, aeronautics, nuclear power, combustion control, transportation, traffic, safety equipment or medical equipment for life support.
4. Design your application so that the product is used within the ranges guaranteed by Hitachi particularly for maximum rating, operating supply voltage range, heat radiation characteristics, installation conditions and other characteristics. Hitachi bears no responsibility for failure or damage when used beyond the guaranteed ranges. Even within the guaranteed ranges, consider normally foreseeable failure rates or failure modes in semiconductor devices and employ systemic measures such as fail-safes, so that the equipment incorporating Hitachi product does not cause bodily injury, fire or other consequential damage due to operation of the Hitachi product.
5. This product is not designed to be radiation resistant.
6. No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without written approval from Hitachi.
7. Contact Hitachi's sales office for any questions regarding this document or Hitachi semiconductor products.

## General Precautions on Handling of Product

### 1. Treatment of NC Pins

**Note:** Do not connect anything to the NC pins.

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

### 2. Treatment of Unused Input Pins

**Note:** Fix all unused input pins to high or low level.

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

### 3. Processing before Initialization

**Note:** When power is first supplied, the product's state is undefined.

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

### 4. Prohibition of Access to Undefined or Reserved Addresses

**Note:** Access to undefined or reserved addresses is prohibited.

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

# Configuration of This Manual

This manual comprises the following items:

1. General Precautions on Handling of Product
2. Configuration of This Manual
3. Preface
4. Contents
5. Overview
6. Description of Functional Modules

- CPU and System-Control Modules
- On-Chip Peripheral Modules

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

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

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

7. Electrical Characteristics
8. Appendix
9. Index

# Preface

The H8S/2612 Series are single-chip microcomputers made up of the high-speed H8S/2600 CPU as its core, and the peripheral functions required to configure a system. The H8S/2600 CPU has an instruction set that is compatible with the H8/300 and H8/300H CPUs.

This LSI is equipped with a data transfer controller (DTC), ROM and RAM memory, a PC break controller (PBC), a 16-bit timer pulse unit (TPU), a motor management timer (MMT), a programmable pulse generator (PPG), a watchdog timer (WDT), a serial communication interface (SCI), a Hitachi controller area network (HCAN), an A/D converter, and I/O ports as on-chip peripheral modules required for system configuration. This LSI is suitable for use as an embedded microcomputer for high-level control systems. A single-power flash memory (F-ZTAT™) version is available for this LSI's ROM. This provides flexibility as it 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 with specifications that will most probably change.

Note: F-ZTAT™ is a trademark of Hitachi, Ltd.

**Target Users:** This manual was written for users who will be using the H8S/2612 Series 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 hardware functions and electrical characteristics of the H8S/2612 Series to the target users.  
Refer to the H8S/2600 Series, H8S/2000 Series Programming 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 parts 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 Programming Manual.
  - In order to understand the details of a register when its name is 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 Appendix A, On-Chip I/O Register.  
Examples:     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.
- 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 you require.  
                                         <http://www.hitachisemiconductor.com/>

H8S/2612 Series manuals:

| Manual Title                                        | ADE No.     |
|-----------------------------------------------------|-------------|
| H8S/2612 Series Hardware Manual                     | This manual |
| H8S/2600 Series, H8S/2000 Series Programming Manual | ADE-602-083 |

User's manuals for development tools:

| Manual Title                                                                          | ADE No.     |
|---------------------------------------------------------------------------------------|-------------|
| H8S, H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual | ADE-702-247 |
| H8S, H8/300 Series Simulator/Debugger User's Manual                                   | ADE-702-282 |
| H8S, H8/300 Series Hitachi Embedded Workshop, Hitachi Debugging Interface Tutorial    | ADE-702-231 |
| Hitachi Embedded Workshop User's Manual                                               | ADE-702-201 |

# Main Revisions and Additions in this Edition

| Item                                                                                                | Page      | Revisions (See Manual for Details)                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                    |       |         |                |           |      |  |                    |
|-----------------------------------------------------------------------------------------------------|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|-------|---------|----------------|-----------|------|--|--------------------|
| All                                                                                                 | —         | Package code amended<br>(Before) FP-80A → (After) FP-80Q                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                    |       |         |                |           |      |  |                    |
| 1.1 Overview                                                                                        | 1         | Amended                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                    |       |         |                |           |      |  |                    |
| • On-chip memory                                                                                    |           | <table><tr><th>ROM</th><th>Model</th><th>Remarks</th></tr><tr><td>F-ZTAT Version</td><td>HD64F2612</td><td></td></tr></table>                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | ROM                | Model | Remarks | F-ZTAT Version | HD64F2612 |      |  |                    |
| ROM                                                                                                 | Model     | Remarks                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                    |       |         |                |           |      |  |                    |
| F-ZTAT Version                                                                                      | HD64F2612 |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                    |       |         |                |           |      |  |                    |
| 5.3 Register Descriptions                                                                           | 63        | Description added<br><br>The interrupt controller has the following registers. For detail on system control register (SYSCR), refer to section 3.2.2, System Control Register (SYSCR). For details on register addresses and ...                                                                                                                                                                                                                                                                                                                                                                    |                    |       |         |                |           |      |  |                    |
| 8.5.7 Number of DTC Execution States                                                                | 113       | Formula amended<br><br>Number of execution states = $I \cdot (1 + S_I) + \Sigma (J \cdot S_J + K \cdot S_K + L \cdot S_L) + M \cdot S_M$                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                    |       |         |                |           |      |  |                    |
| 13.3.1 Watchdog Timer Mode                                                                          | 287       | 4th line<br><br>Description amended<br><br>When the WDT is used as a watchdog timer, and if TCNT overflows without being rewritten because of a system malfunction or other error, an internal reset occurs and the internal chip states can be reset.<br><br>TCNT does not overflow while the system is operating normally. Software must prevent TCNT overflows by rewriting the TCNT value (normally be writing H'00) before overflows occurs.<br><br>In this case, select power-on reset by setting the RSTS bit of the RSTCSR to 0.<br><br>The internal reset signal is output for 518 states. |                    |       |         |                |           |      |  |                    |
| 13.5.1 Notes on Register Access<br>Figure 13-2 Writing to TCNT, TCSR, and RSTCSR (Example for WDT0) | 289       | TCSR write<br>Writing 0 to WOVSF bit<br><br>Address: H'FF74<br>H'FF76 <table><tr><td>15</td><td>8</td><td>7</td><td>0</td></tr><tr><td></td><td>H'A5</td><td></td><td>Write data or H'00</td></tr></table>                                                                                                                                                                                                                                                                                                                                                                                          | 15                 | 8     | 7       | 0              |           | H'A5 |  | Write data or H'00 |
| 15                                                                                                  | 8         | 7                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | 0                  |       |         |                |           |      |  |                    |
|                                                                                                     | H'A5      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Write data or H'00 |       |         |                |           |      |  |                    |
| 14.3.7 Serial Status Register (SSR)                                                                 | 306       | TEND bit description amended<br>[Setting conditions]<br><ul style="list-style-type: none"><li>When the TE bit in SCR . . .</li><li>When the ERS bit is 0 and the TDRE bit is 1 . . .</li></ul>                                                                                                                                                                                                                                                                                                                                                                                                      |                    |       |         |                |           |      |  |                    |
| 15.3.7 Transmit Acknowledge Register (TXACK)                                                        | 366, 367  | Item of R/W amended<br>(Before) R/W → (After) R/(W)*                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                    |       |         |                |           |      |  |                    |
| 15.3.8 Abort Acknowledge Register (ABACK)                                                           |           | Note added<br><br>Note: * Only 1 can be written to clear the flag.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                    |       |         |                |           |      |  |                    |

| Item                                                                               | Page       | Revisions (See Manual for Details)                                                                                                                                                                        |
|------------------------------------------------------------------------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15.3.9 Receive Complete Register (RXPR)                                            | 368 to 373 | Item of R/W amended<br>(Before) R/W → (After) R(W)*                                                                                                                                                       |
| 15.3.10 Remote Request Register (RFPR)                                             |            | Note added<br>Note: * Only 1 can be written to clear the flag.                                                                                                                                            |
| 15.3.11 Interrupt Register (IRR)                                                   |            |                                                                                                                                                                                                           |
| 15.3.16 Unread Message Status Register (UMSR)                                      | 377        | R/W in bits 2 to 0 amended<br>(Before) R/W → (After) R(W)*<br>Note added<br>Note: * Only 1 can be written to clear the flag.                                                                              |
| 15.4.2 Initialization after Hardware Reset                                         | 386        | Notes amended<br>Notes: *1 SJW is stipulated in the CAN specifications: $3 \geq \text{SJW} \geq 0$<br>*2 The minimum value of TSEG2 is stipulated in the CAN stipulations: $\text{TSEG2} \geq \text{SJW}$ |
| Table 15-2 Limits for Settable Value                                               |            |                                                                                                                                                                                                           |
| 15.5 Interrupt                                                                     | 398        | ERS0/OVR0 Description amended                                                                                                                                                                             |
| Table 15-4 HCAN Interrupt Sources                                                  |            | (Before) Overload frame transmission interrupt/bus off recovery interrupt (11 recessive bits × 128 times) → (After) Overload frame transmission interrupt                                                 |
| 15.8.9 Usage of Bit Manipulation Instructions                                      | 401, 402   | Newly added                                                                                                                                                                                               |
| 15.8.10 HCAN TXCR Operation                                                        |            |                                                                                                                                                                                                           |
| 20.4.3 Setting Oscillation Stabilization Time after Clearing Software Standby Mode | 462        | Note * added<br>16 states*                                                                                                                                                                                |
| Table 20-3 Oscillation Stabilization Time Settings                                 |            | Note: * Setting prohibited.                                                                                                                                                                               |
| 21.3.3 Timing of On-Chip Supporting Modules                                        | 477        | $t_{\text{MTOD}}$ max value amended<br>(Before) — → (After) 50                                                                                                                                            |
| Table 21-6 Timing of On-Chip Supporting Modules                                    |            |                                                                                                                                                                                                           |
| C. Product Code Lineup                                                             | 526        | Note * added<br>Mask ROM version*<br>Note: * In planning                                                                                                                                                  |
| D. Package Dimensions                                                              | 526        | Replaced                                                                                                                                                                                                  |
| Figure D-1 FP-80Q Dimensions                                                       |            |                                                                                                                                                                                                           |

# Contents

|           |                                                                         |    |
|-----------|-------------------------------------------------------------------------|----|
| Section 1 | Overview.....                                                           | 1  |
| 1.1       | Overview.....                                                           | 1  |
| 1.2       | Internal Block Diagram.....                                             | 2  |
| 1.3       | Pin Arrangement .....                                                   | 3  |
| 1.4       | Pin Functions .....                                                     | 4  |
| Section 2 | CPU.....                                                                | 9  |
| 2.1       | Features.....                                                           | 9  |
| 2.1.1     | Differences between H8S/2600 CPU and H8S/2000 CPU .....                 | 10 |
| 2.1.2     | Differences from H8/300 CPU .....                                       | 11 |
| 2.1.3     | Differences from H8/300H CPU.....                                       | 11 |
| 2.2       | CPU Operating Modes .....                                               | 12 |
| 2.2.1     | Normal Mode .....                                                       | 12 |
| 2.2.2     | Advanced Mode.....                                                      | 13 |
| 2.3       | Address Space.....                                                      | 16 |
| 2.4       | Register Configuration.....                                             | 17 |
| 2.4.1     | General Registers .....                                                 | 18 |
| 2.4.2     | Program Counter (PC) .....                                              | 19 |
| 2.4.3     | Extended Control Register (EXR) .....                                   | 19 |
| 2.4.4     | Condition-Code Register (CCR).....                                      | 20 |
| 2.4.5     | Multiply-Accumulate Register (MAC) .....                                | 22 |
| 2.4.6     | Initial Values of CPU Registers.....                                    | 22 |
| 2.5       | Data Formats .....                                                      | 23 |
| 2.5.1     | General Register Data Formats .....                                     | 23 |
| 2.5.2     | Memory Data Formats .....                                               | 25 |
| 2.6       | Instruction Set .....                                                   | 26 |
| 2.6.1     | Table of Instructions Classified by Function .....                      | 27 |
| 2.6.2     | Basic Instruction Formats .....                                         | 36 |
| 2.7       | Addressing Modes and Effective Address Calculation.....                 | 38 |
| 2.7.1     | Register Direct—Rn.....                                                 | 38 |
| 2.7.2     | Register Indirect—@ERn .....                                            | 38 |
| 2.7.3     | Register Indirect with Displacement—@(d:16, ERn) or @(d:32, ERn).....   | 38 |
| 2.7.4     | Register Indirect with Post-Increment or Pre-Decrement—@ERn+ or @-ERn.. | 39 |
| 2.7.5     | Absolute Address—@aa:8, @aa:16, @aa:24, or @aa:32.....                  | 39 |
| 2.7.6     | Immediate—#xx:8, #xx:16, or #xx:32 .....                                | 40 |
| 2.7.7     | Program-Counter Relative—@(d:8, PC) or @(d:16, PC).....                 | 40 |
| 2.7.8     | Memory Indirect—@@aa:8 .....                                            | 40 |
| 2.7.9     | Effective Address Calculation .....                                     | 41 |
| 2.8       | Processing States.....                                                  | 44 |

|                  |                                                                                        |           |
|------------------|----------------------------------------------------------------------------------------|-----------|
| 2.9              | Usage Notes .....                                                                      | 45        |
| 2.9.1            | Usage Notes on Bit Manipulation Instructions .....                                     | 45        |
| <b>Section 3</b> | <b>MCU Operating Modes .....</b>                                                       | <b>47</b> |
| 3.1              | Operating Mode Selection.....                                                          | 47        |
| 3.2              | Register Descriptions .....                                                            | 47        |
| 3.2.1            | Mode Control Register(MDCR) .....                                                      | 47        |
| 3.2.2            | System Control Register(SYSCR) .....                                                   | 48        |
| 3.3              | Pin Functions in Each Operating Mode .....                                             | 49        |
| 3.3.1            | Pin Functions .....                                                                    | 49        |
| 3.4              | Address Map .....                                                                      | 50        |
| <b>Section 4</b> | <b>Exception Handling .....</b>                                                        | <b>51</b> |
| 4.1              | Exception Handling Types and Priority .....                                            | 51        |
| 4.2              | Exception Sources and Exception Vector Table .....                                     | 51        |
| 4.3              | Reset .....                                                                            | 53        |
| 4.3.1            | Reset Exception Handling.....                                                          | 53        |
| 4.3.2            | Interrupts after Reset.....                                                            | 55        |
| 4.3.3            | State of On-Chip Supporting Modules after Reset Release .....                          | 55        |
| 4.4              | Traces.....                                                                            | 56        |
| 4.5              | Interrupts.....                                                                        | 56        |
| 4.6              | Trap Instruction.....                                                                  | 57        |
| 4.7              | Stack Status after Exception Handling.....                                             | 58        |
| 4.8              | Usage Note.....                                                                        | 59        |
| <b>Section 5</b> | <b>Interrupt Controller.....</b>                                                       | <b>61</b> |
| 5.1              | Features .....                                                                         | 61        |
| 5.2              | Input/Output Pins .....                                                                | 63        |
| 5.3              | Register Descriptions .....                                                            | 63        |
| 5.3.1            | Interrupt Priority Registers A to H, J, K, M<br>(IPRA to IPRH, IPRJ, IPRK, IPRM) ..... | 64        |
| 5.3.2            | IRQ Enable Register (IER) .....                                                        | 65        |
| 5.3.3            | IRQ Sense Control Registers H and L (ISCRH, ISCRL).....                                | 65        |
| 5.3.4            | IRQ Status Register (ISR).....                                                         | 67        |
| 5.4              | Interrupt .....                                                                        | 69        |
| 5.4.1            | External Interrupts .....                                                              | 69        |
| 5.4.2            | Internal Interrupts.....                                                               | 70        |
| 5.5              | Interrupt Exception Handling Vector Table.....                                         | 70        |
| 5.6              | Interrupt Control Modes and Interrupt Operation .....                                  | 73        |
| 5.6.1            | Interrupt Control Mode 0 .....                                                         | 73        |
| 5.6.2            | Interrupt Control Mode 2 .....                                                         | 75        |
| 5.6.3            | Interrupt Exception Handling Sequence .....                                            | 76        |
| 5.6.4            | Interrupt Response Times .....                                                         | 78        |

|                                                  |                                                                                                                |           |
|--------------------------------------------------|----------------------------------------------------------------------------------------------------------------|-----------|
| 5.6.5                                            | DTC Activation by Interrupt.....                                                                               | 79        |
| 5.7                                              | Usage Notes .....                                                                                              | 79        |
| 5.7.1                                            | Contention between Interrupt Generation and Disabling.....                                                     | 79        |
| 5.7.2                                            | Instructions that Disable Interrupts .....                                                                     | 80        |
| 5.7.3                                            | When Interrupts are Disabled .....                                                                             | 80        |
| 5.7.4                                            | Interrupts during Execution of EEPMOV Instruction.....                                                         | 81        |
| <b>Section 6 PC Break Controller (PBC) .....</b> |                                                                                                                | <b>83</b> |
| 6.1                                              | Features.....                                                                                                  | 83        |
| 6.2                                              | Register Descriptions .....                                                                                    | 84        |
| 6.2.1                                            | Break Address Register A (BARA) .....                                                                          | 84        |
| 6.2.2                                            | Break Address Register B (BARB) .....                                                                          | 85        |
| 6.2.3                                            | Break Control Register A (BCRA) .....                                                                          | 85        |
| 6.2.4                                            | Break Control Register B (BCRB).....                                                                           | 86        |
| 6.3                                              | Operation .....                                                                                                | 86        |
| 6.3.1                                            | PC Break Interrupt Due to Instruction Fetch .....                                                              | 86        |
| 6.3.2                                            | PC Break Interrupt Due to Data Access.....                                                                     | 86        |
| 6.3.3                                            | Notes on PC Break Interrupt Handling .....                                                                     | 87        |
| 6.3.4                                            | Operation in Transitions to Power-Down Modes .....                                                             | 87        |
| 6.3.5                                            | When Instruction Execution is Delayed by One State .....                                                       | 88        |
| 6.4                                              | Usage Notes .....                                                                                              | 89        |
| 6.4.1                                            | Module Stop Mode Setting .....                                                                                 | 89        |
| 6.4.2                                            | PC Break Interrupts.....                                                                                       | 89        |
| 6.4.3                                            | CMFA and CMFB .....                                                                                            | 89        |
| 6.4.4                                            | PC Break Interrupt when DTC is Bus Master.....                                                                 | 89        |
| 6.4.5                                            | PC Break Set for Instruction Fetch at Address Following<br>BSR, JSR, JMP, TRAPA, RTE, or RTS Instruction ..... | 89        |
| 6.4.6                                            | I Bit Set by LDC, ANDC, ORC, or XORC Instruction .....                                                         | 89        |
| 6.4.7                                            | PC Break Set for Instruction Fetch at Address Following Bcc Instruction.....                                   | 90        |
| 6.4.8                                            | PC Break Set for Instruction Fetch at Branch Destination Address of<br>Bcc Instruction .....                   | 90        |
| <b>Section 7 Bus Controller.....</b>             |                                                                                                                | <b>91</b> |
| 7.1                                              | Basic Timing.....                                                                                              | 91        |
| 7.1.1                                            | On-Chip Memory Access Timing (ROM, RAM) .....                                                                  | 91        |
| 7.1.2                                            | On-Chip Support Module Access Timing.....                                                                      | 92        |
| 7.1.3                                            | On-Chip HCAN Module Access Timing.....                                                                         | 93        |
| 7.1.4                                            | On-Chip MMT Module Access Timing.....                                                                          | 93        |
| 7.2                                              | Bus Arbitration.....                                                                                           | 94        |
| 7.2.1                                            | Order of Priority of the Bus Masters.....                                                                      | 94        |
| 7.2.2                                            | Bus Transfer Timing.....                                                                                       | 94        |

|           |                                                       |     |
|-----------|-------------------------------------------------------|-----|
| Section 8 | Data Transfer Controller (DTC)                        | 95  |
| 8.1       | Features                                              | 95  |
| 8.2       | Register Configuration                                | 97  |
| 8.2.1     | DTC Mode Register A (MRA)                             | 98  |
| 8.2.2     | DTC Mode Register B (MRB)                             | 99  |
| 8.2.3     | DTC Source Address Register (SAR)                     | 99  |
| 8.2.4     | DTC Destination Address Register (DAR)                | 99  |
| 8.2.5     | DTC Transfer Count Register A (CRA)                   | 99  |
| 8.2.6     | DTC Transfer Count Register B (CRB)                   | 100 |
| 8.2.7     | DTC Enable Registers (DTCER)                          | 100 |
| 8.2.8     | DTC Vector Register (DTVECR)                          | 100 |
| 8.3       | Activation Sources                                    | 101 |
| 8.4       | Location of Register Information and DTC Vector Table | 102 |
| 8.5       | Operation                                             | 105 |
| 8.5.1     | Normal Mode                                           | 107 |
| 8.5.2     | Repeat Mode                                           | 108 |
| 8.5.3     | Block Transfer Mode                                   | 109 |
| 8.5.4     | Chain Transfer                                        | 110 |
| 8.5.5     | Interrupts                                            | 111 |
| 8.5.6     | Operation Timing                                      | 111 |
| 8.5.7     | Number of DTC Execution States                        | 112 |
| 8.6       | Procedures for Using DTC                              | 114 |
| 8.6.1     | Activation by Interrupt                               | 114 |
| 8.6.2     | Activation by Software                                | 114 |
| 8.7       | Examples of Use of the DTC                            | 114 |
| 8.7.1     | Normal Mode                                           | 114 |
| 8.7.2     | Chain Transfer                                        | 115 |
| 8.7.3     | Software Activation                                   | 116 |
| 8.8       | Usage Notes                                           | 116 |
| 8.8.1     | Module Stop Mode Setting                              | 116 |
| 8.8.2     | On-Chip RAM                                           | 116 |
| 8.8.3     | DTCE Bit Setting                                      | 117 |
| Section 9 | I/O Ports                                             | 119 |
| 9.1       | Port 1                                                | 122 |
| 9.1.1     | Port 1 Data Direction Register (P1DDR)                | 122 |
| 9.1.2     | Port 1 Data Register (P1DR)                           | 123 |
| 9.1.3     | Port 1 Register (PORT1)                               | 123 |
| 9.1.4     | Pin Functions                                         | 124 |
| 9.2       | Port 4                                                | 126 |
| 9.2.1     | Port 4 Register (PORT4)                               | 126 |
| 9.3       | Port 9                                                | 127 |
| 9.3.1     | Port 9 Register (PORT9)                               | 127 |

|       |                                                   |     |
|-------|---------------------------------------------------|-----|
| 9.4   | Port A .....                                      | 128 |
| 9.4.1 | Port A Data Direction Register (PADDR) .....      | 128 |
| 9.4.2 | Port A Data Register (PADR) .....                 | 129 |
| 9.4.3 | Port A Register (PORTA) .....                     | 129 |
| 9.4.4 | Port A Pull-Up MOS Control Register (PAPCR) ..... | 130 |
| 9.4.5 | Port A Open-Drain Control Register (PAODR) .....  | 130 |
| 9.4.6 | Pin Functions .....                               | 131 |
| 9.5   | Port B .....                                      | 132 |
| 9.5.1 | Port B Data Direction Register (PBDDR) .....      | 132 |
| 9.5.2 | Port B Data Register (PBDR) .....                 | 133 |
| 9.5.3 | Port B Register (PORTB) .....                     | 133 |
| 9.5.4 | Port B Pull-Up MOS Control Register (PBPCR) ..... | 134 |
| 9.5.5 | Port B Open-Drain Control Register (PBODR) .....  | 134 |
| 9.5.6 | Pin Functions .....                               | 135 |
| 9.6   | Port C .....                                      | 137 |
| 9.6.1 | Port C Data Direction Register (PCDDR) .....      | 137 |
| 9.6.2 | Port C Data Register (PCDR) .....                 | 138 |
| 9.6.3 | Port C Register (PORTC) .....                     | 139 |
| 9.6.4 | Port C Pull-Up MOS Control Register (PCPCR) ..... | 139 |
| 9.6.5 | Port C Open-Drain Control Register (PCODR) .....  | 140 |
| 9.6.6 | Pin Functions .....                               | 140 |
| 9.7   | Port D .....                                      | 142 |
| 9.7.1 | Port D Data Direction Register (PDDDR) .....      | 142 |
| 9.7.2 | Port D Data Register (PDDR) .....                 | 143 |
| 9.7.3 | Port D Register (PORTD) .....                     | 143 |
| 9.7.4 | Port D Pull-up MOS Control Register (PDPCR) ..... | 143 |
| 9.8   | Port F .....                                      | 144 |
| 9.8.1 | Port F Data Direction Register (PFDDR) .....      | 144 |
| 9.8.2 | Port F Data Register (PFDR) .....                 | 145 |
| 9.8.3 | Port F Register (PORTF) .....                     | 145 |
| 9.8.4 | Pin Functions .....                               | 146 |

## Section 10 16-Bit Timer Pulse Unit (TPU).....149

|        |                                              |     |
|--------|----------------------------------------------|-----|
| 10.1   | Features .....                               | 149 |
| 10.2   | Input/Output Pins .....                      | 153 |
| 10.3   | Register Descriptions .....                  | 154 |
| 10.3.1 | Timer Control Register (TCR) .....           | 155 |
| 10.3.2 | Timer Mode Register (TMDR) .....             | 160 |
| 10.3.3 | Timer I/O Control Register (TIOR) .....      | 163 |
| 10.3.4 | Timer Interrupt Enable Register (TIER) ..... | 179 |
| 10.3.5 | Timer Status Register (TSR) .....            | 181 |
| 10.3.6 | Timer Counter (TCNT) .....                   | 184 |
| 10.3.7 | Timer General Register (TGR) .....           | 185 |

|                                               |                                                                  |     |
|-----------------------------------------------|------------------------------------------------------------------|-----|
| 10.3.8                                        | Timer Start Register (TSTR) .....                                | 185 |
| 10.3.9                                        | Timer Synchro Register (TSYR) .....                              | 186 |
| 10.4                                          | Operation .....                                                  | 186 |
| 10.4.1                                        | Basic Functions .....                                            | 186 |
| 10.4.2                                        | Synchronous Operation .....                                      | 193 |
| 10.4.3                                        | Buffer Operation .....                                           | 194 |
| 10.4.4                                        | Cascaded Operation .....                                         | 198 |
| 10.4.5                                        | PWM Modes .....                                                  | 199 |
| 10.4.6                                        | Phase Counting Mode .....                                        | 204 |
| 10.5                                          | Interrupts .....                                                 | 211 |
| 10.6                                          | DTC Activation .....                                             | 213 |
| 10.7                                          | A/D Converter Activation .....                                   | 213 |
| 10.8                                          | Operation Timing .....                                           | 214 |
| 10.8.1                                        | Input/Output Timing .....                                        | 214 |
| 10.8.2                                        | Interrupt Signal Timing .....                                    | 218 |
| 10.9                                          | Usage Notes .....                                                | 221 |
| 10.9.1                                        | Module Stop Mode Setting .....                                   | 221 |
| 10.9.2                                        | Input Clock Restrictions .....                                   | 221 |
| 10.9.3                                        | Caution on Period Setting .....                                  | 221 |
| 10.9.4                                        | Contention between TCNT Write and Clear Operations .....         | 222 |
| 10.9.5                                        | Contention between TCNT Write and Increment Operations .....     | 223 |
| 10.9.6                                        | Contention between TGR Write and Compare Match .....             | 224 |
| 10.9.7                                        | Contention between Buffer Register Write and Compare Match ..... | 225 |
| 10.9.8                                        | Contention between TGR Read and Input Capture .....              | 226 |
| 10.9.9                                        | Contention between TGR Write and Input Capture .....             | 227 |
| 10.9.10                                       | Contention between Buffer Register Write and Input Capture ..... | 228 |
| 10.9.11                                       | Contention between Overflow/Underflow and Counter Clearing ..... | 229 |
| 10.9.12                                       | Contention between TCNT Write and Overflow/Underflow .....       | 230 |
| 10.9.13                                       | Multiplexing of I/O Pins .....                                   | 230 |
| 10.9.14                                       | Interrupts in Module Stop Mode .....                             | 230 |
| Section 11 Motor Management Timer (MMT) ..... |                                                                  | 231 |
| 11.1                                          | Features .....                                                   | 231 |
| 11.2                                          | Input/Output Pins .....                                          | 233 |
| 11.3                                          | Register Descriptions .....                                      | 233 |
| 11.3.1                                        | Timer Mode Register (TMDR) .....                                 | 234 |
| 11.3.2                                        | Timer Control Register (TCNR) .....                              | 235 |
| 11.3.3                                        | Timer Status Register (TSR) .....                                | 236 |
| 11.3.4                                        | Timer Counter (TCNT) .....                                       | 236 |
| 11.3.5                                        | Timer Buffer Registers (TBR) .....                               | 237 |
| 11.3.6                                        | Timer General Registers (TGR) .....                              | 237 |
| 11.3.7                                        | Timer Dead Time Counters (TDCNT) .....                           | 237 |
| 11.3.8                                        | Timer Dead Time Data Register (TDDR) .....                       | 237 |

|                                                     |                                                                                                               |     |
|-----------------------------------------------------|---------------------------------------------------------------------------------------------------------------|-----|
| 11.3.9                                              | Timer Period Buffer Register (TPBR) .....                                                                     | 237 |
| 11.3.10                                             | Timer Period Data Register (TPDR).....                                                                        | 237 |
| 11.3.11                                             | MMT Pin Control Register (MMTPC) .....                                                                        | 238 |
| 11.4                                                | Operation .....                                                                                               | 238 |
| 11.4.1                                              | Sample Setting Procedure .....                                                                                | 239 |
| 11.4.2                                              | Output Protection Functions .....                                                                             | 247 |
| 11.5                                                | Interrupts .....                                                                                              | 247 |
| 11.6                                                | Operation Timing .....                                                                                        | 248 |
| 11.6.1                                              | Input/Output Timing .....                                                                                     | 248 |
| 11.6.2                                              | Interrupt Signal Timing.....                                                                                  | 252 |
| 11.7                                                | Usage Notes .....                                                                                             | 253 |
| 11.7.1                                              | Module Stop Mode Setting .....                                                                                | 253 |
| 11.7.2                                              | Notes for MMT Operation .....                                                                                 | 253 |
| 11.8                                                | Port Output Enable (POE).....                                                                                 | 256 |
| 11.8.1                                              | Features .....                                                                                                | 256 |
| 11.8.2                                              | Input/Output Pins .....                                                                                       | 257 |
| 11.8.3                                              | Register Descriptions .....                                                                                   | 257 |
| 11.8.4                                              | Operation .....                                                                                               | 261 |
| Section 12 Programmable Pulse Generator (PPG) ..... |                                                                                                               | 263 |
| 12.1                                                | Features .....                                                                                                | 263 |
| 12.2                                                | Input/Output Pins .....                                                                                       | 265 |
| 12.3                                                | Register Descriptions .....                                                                                   | 265 |
| 12.3.1                                              | Next Data Enable Registers H,L (NDERH, NDERL).....                                                            | 266 |
| 12.3.2                                              | Output Data Registers H,L (PODRH, PODRL).....                                                                 | 267 |
| 12.3.3                                              | Next Data Registers H,L (NDRH, NDRL) .....                                                                    | 268 |
| 12.3.4                                              | PPG Output Control Register (PCR) .....                                                                       | 270 |
| 12.3.5                                              | PPG Output Mode Register (PMR) .....                                                                          | 271 |
| 12.4                                                | Operation .....                                                                                               | 272 |
| 12.4.1                                              | Overview .....                                                                                                | 272 |
| 12.4.2                                              | Output Timing.....                                                                                            | 273 |
| 12.4.3                                              | Sample Setup Procedure for Normal Pulse Output .....                                                          | 274 |
| 12.4.4                                              | Example of Normal Pulse Output (Example of Five-Phase Pulse Output) .....                                     | 275 |
| 12.4.5                                              | Non-Overlapping Pulse Output.....                                                                             | 276 |
| 12.4.6                                              | Sample Setup Procedure for Non-Overlapping Pulse Output .....                                                 | 278 |
| 12.4.7                                              | Example of Non-Overlapping Pulse Output<br>(Example of Four-Phase Complementary Non-Overlapping Output) ..... | 279 |
| 12.4.8                                              | Inverted Pulse Output .....                                                                                   | 281 |
| 12.4.9                                              | Pulse Output Triggered by Input Capture .....                                                                 | 282 |
| 12.5                                                | Usage Notes .....                                                                                             | 282 |
| 12.5.1                                              | Module Stop Mode Setting .....                                                                                | 282 |
| 12.5.2                                              | Operation of Pulse Output Pins.....                                                                           | 282 |

|            |                                                                                 |     |
|------------|---------------------------------------------------------------------------------|-----|
| Section 13 | Watchdog Timer .....                                                            | 283 |
| 13.1       | Features .....                                                                  | 283 |
| 13.2       | Register Descriptions .....                                                     | 284 |
| 13.2.1     | Timer Counter (TCNT) .....                                                      | 284 |
| 13.2.2     | Timer Control/Status Register (TCSR) .....                                      | 284 |
| 13.2.3     | Reset Control/Status Register (RSTCSR) .....                                    | 286 |
| 13.3       | Operation .....                                                                 | 287 |
| 13.3.1     | Watchdog Timer Mode .....                                                       | 287 |
| 13.3.2     | Interval Timer Mode .....                                                       | 287 |
| 13.4       | Interrupts .....                                                                | 288 |
| 13.5       | Usage Notes .....                                                               | 288 |
| 13.5.1     | Notes on Register Access .....                                                  | 288 |
| 13.5.2     | Contention between Timer Counter (TCNT) Write and Increment .....               | 289 |
| 13.5.3     | Changing Value of CKS2 to CKS0 .....                                            | 290 |
| 13.5.4     | Switching between Watchdog Timer Mode and Interval Timer Mode .....             | 290 |
| 13.5.5     | Internal Reset in Watchdog Timer Mode .....                                     | 290 |
| 13.5.6     | OVF Flag Clearing in Interval Timer Mode .....                                  | 290 |
| Section 14 | Serial Communication Interface (SCI) .....                                      | 291 |
| 14.1       | Features .....                                                                  | 291 |
| 14.2       | Input/Output Pins .....                                                         | 293 |
| 14.3       | Register Descriptions .....                                                     | 293 |
| 14.3.1     | Receive Shift Register (RSR) .....                                              | 294 |
| 14.3.2     | Receive Data Register (RDR) .....                                               | 294 |
| 14.3.3     | Transmit Data Register (TDR) .....                                              | 294 |
| 14.3.4     | Transmit Shift Register (TSR) .....                                             | 294 |
| 14.3.5     | Serial Mode Register (SMR) .....                                                | 295 |
| 14.3.6     | Serial Control Register (SCR) .....                                             | 299 |
| 14.3.7     | Serial Status Register (SSR) .....                                              | 302 |
| 14.3.8     | Smart Card Mode Register (SCMR) .....                                           | 307 |
| 14.3.9     | Bit Rate Register (BRR) .....                                                   | 308 |
| 14.4       | Operation in Asynchronous Mode .....                                            | 315 |
| 14.4.1     | Data Transfer Format .....                                                      | 315 |
| 14.4.2     | Receive Data Sampling Timing and Reception Margin in<br>Asynchronous Mode ..... | 317 |
| 14.4.3     | Clock .....                                                                     | 318 |
| 14.4.4     | SCI Initialization (Asynchronous Mode) .....                                    | 319 |
| 14.4.5     | Data Transmission (Asynchronous Mode) .....                                     | 320 |
| 14.4.6     | Serial Data Reception (Asynchronous Mode) .....                                 | 322 |
| 14.5       | Multiprocessor Communication Function .....                                     | 326 |
| 14.5.1     | Multiprocessor Serial Data Transmission .....                                   | 328 |
| 14.5.2     | Multiprocessor Serial Data Reception .....                                      | 329 |
| 14.6       | Operation in Clocked Synchronous Mode .....                                     | 332 |

|                                                        |                                                                                         |     |
|--------------------------------------------------------|-----------------------------------------------------------------------------------------|-----|
| 14.6.1                                                 | Clock.....                                                                              | 332 |
| 14.6.2                                                 | SCI Initialization (Clocked Synchronous Mode).....                                      | 333 |
| 14.6.3                                                 | Serial Data Transmission (Clocked Synchronous Mode) .....                               | 334 |
| 14.6.4                                                 | Serial Data Reception (Clocked Synchronous Mode).....                                   | 337 |
| 14.6.5                                                 | Simultaneous Serial Data Transmission and Reception<br>(Clocked Synchronous Mode).....  | 339 |
| 14.7                                                   | Operation in Smart Card Interface .....                                                 | 341 |
| 14.7.1                                                 | Pin Connection Example.....                                                             | 341 |
| 14.7.2                                                 | Data Format (Except for Block Transfer Mode).....                                       | 342 |
| 14.7.3                                                 | Block Transfer Mode .....                                                               | 343 |
| 14.7.4                                                 | Receive Data Sampling Timing and Reception Margin in Smart Card<br>Interface Mode ..... | 344 |
| 14.7.5                                                 | Initialization .....                                                                    | 345 |
| 14.7.6                                                 | Data Transmission (Except for Block Transfer Mode).....                                 | 345 |
| 14.7.7                                                 | Serial Data Reception (Except for Block Transfer Mode).....                             | 349 |
| 14.7.8                                                 | Clock Output Control.....                                                               | 350 |
| 14.8                                                   | Interrupts .....                                                                        | 352 |
| 14.8.1                                                 | Interrupts in Normal Serial Communication Interface Mode .....                          | 352 |
| 14.8.2                                                 | Interrupts in Smart Card Interface Mode .....                                           | 353 |
| 14.9                                                   | Usage Notes .....                                                                       | 354 |
| 14.9.1                                                 | Module Stop Mode Setting .....                                                          | 354 |
| 14.9.2                                                 | Break Detection and Processing .....                                                    | 354 |
| 14.9.3                                                 | Mark State and Break Detection .....                                                    | 354 |
| 14.9.4                                                 | Receive Error Flags and Transmit Operations<br>(Clocked Synchronous Mode Only).....     | 354 |
| Section 15 Hitachi Controller Area Network (HCAN)..... |                                                                                         | 355 |
| 15.1                                                   | Features .....                                                                          | 355 |
| 15.2                                                   | Input/Output Pins .....                                                                 | 357 |
| 15.3                                                   | Register Descriptions .....                                                             | 357 |
| 15.3.1                                                 | Master Control Register (MCR) .....                                                     | 358 |
| 15.3.2                                                 | General Status Register (GSR) .....                                                     | 359 |
| 15.3.3                                                 | Bit Configuration Register (BCR) .....                                                  | 361 |
| 15.3.4                                                 | Mailbox Configuration Register (MBCR) .....                                             | 363 |
| 15.3.5                                                 | Transmit Wait Register (TXPR) .....                                                     | 364 |
| 15.3.6                                                 | Transmit Wait Cancel Register (TXCR).....                                               | 365 |
| 15.3.7                                                 | Transmit Acknowledge Register (TXACK) .....                                             | 366 |
| 15.3.8                                                 | Abort Acknowledge Register (ABACK) .....                                                | 367 |
| 15.3.9                                                 | Receive Complete Register (RXPR).....                                                   | 368 |
| 15.3.10                                                | Remote Request Register (RFPR).....                                                     | 369 |
| 15.3.11                                                | Interrupt Register (IRR).....                                                           | 370 |
| 15.3.12                                                | Mailbox Interrupt Mask Register (MBIMR).....                                            | 374 |
| 15.3.13                                                | Interrupt Mask Register (IMR) .....                                                     | 375 |

|            |                                                   |     |
|------------|---------------------------------------------------|-----|
| 15.3.14    | Receive Error Counter (REC).....                  | 376 |
| 15.3.15    | Transmit Error Counter (TEC).....                 | 376 |
| 15.3.16    | Unread Message Status Register (UMSR).....        | 377 |
| 15.3.17    | Local Acceptance Filter Masks (LAFML, LAFMH)..... | 378 |
| 15.3.18    | Message Control (MC0 to MC15).....                | 380 |
| 15.3.19    | Message Data (MD0 to MD15).....                   | 382 |
| 15.3.20    | HCAN Monitor Register (HCANMON).....              | 382 |
| 15.4       | Operation .....                                   | 383 |
| 15.4.1     | Hardware and Software Resets .....                | 383 |
| 15.4.2     | Initialization after Hardware Reset .....         | 383 |
| 15.4.3     | Message Transmission .....                        | 389 |
| 15.4.4     | Message Reception .....                           | 392 |
| 15.4.5     | HCAN Sleep Mode.....                              | 395 |
| 15.4.6     | HCAN Halt Mode.....                               | 397 |
| 15.5       | Interrupts.....                                   | 398 |
| 15.6       | DTC Interface .....                               | 399 |
| 15.7       | CAN Bus Interface.....                            | 400 |
| 15.8       | Usage Notes .....                                 | 400 |
| 15.8.1     | Module Stop Mode Setting .....                    | 400 |
| 15.8.2     | Reset .....                                       | 400 |
| 15.8.3     | HCAN Sleep Mode.....                              | 401 |
| 15.8.4     | Interrupts.....                                   | 401 |
| 15.8.5     | Error Counters.....                               | 401 |
| 15.8.6     | Register Access.....                              | 401 |
| 15.8.7     | HCAN Medium-Speed Mode .....                      | 401 |
| 15.8.8     | Register Hold in Standby Modes .....              | 401 |
| 15.8.9     | Usage of Bit Manipulation Instructions .....      | 401 |
| 15.8.10    | HCAN TXCR Operation.....                          | 402 |
| Section 16 | A/D Converter .....                               | 403 |
| 16.1       | Features.....                                     | 403 |
| 16.2       | Input/Output Pins .....                           | 405 |
| 16.3       | Register Description.....                         | 406 |
| 16.3.1     | A/D Data Registers A to D (ADDRA to ADDR D) ..... | 406 |
| 16.3.2     | A/D Control/Status Register (ADCSR) .....         | 407 |
| 16.3.3     | A/D Control Register (ADCR) .....                 | 409 |
| 16.4       | Operation .....                                   | 410 |
| 16.4.1     | Single Mode.....                                  | 410 |
| 16.4.2     | Scan Mode .....                                   | 410 |
| 16.4.3     | Input Sampling and A/D Conversion Time .....      | 411 |
| 16.4.4     | External Trigger Input Timing.....                | 413 |
| 16.5       | Interrupts.....                                   | 413 |
| 16.6       | A/D Conversion Precision Definitions.....         | 414 |

|                                        |                                                               |     |
|----------------------------------------|---------------------------------------------------------------|-----|
| 16.7                                   | Usage Notes .....                                             | 416 |
| 16.7.1                                 | Module Stop Mode Setting .....                                | 416 |
| 16.7.2                                 | Permissible Signal Source Impedance .....                     | 416 |
| 16.7.3                                 | Influences on Absolute Precision.....                         | 416 |
| 16.7.4                                 | Range of Analog Power Supply and Other Pin Settings .....     | 417 |
| 16.7.5                                 | Notes on Board Design .....                                   | 417 |
| 16.7.6                                 | Notes on Noise Countermeasures .....                          | 417 |
| Section 17 RAM .....                   |                                                               | 419 |
| Section 18 ROM .....                   |                                                               | 421 |
| 18.1                                   | Features .....                                                | 421 |
| 18.2                                   | Mode Transitions .....                                        | 422 |
| 18.3                                   | Block Configuration.....                                      | 426 |
| 18.4                                   | Input/Output Pins .....                                       | 427 |
| 18.5                                   | Register Descriptions .....                                   | 427 |
| 18.5.1                                 | Flash Memory Control Register 1 (FLMCR1).....                 | 428 |
| 18.5.2                                 | Flash Memory Control Register 2 (FLMCR2).....                 | 429 |
| 18.5.3                                 | Erase Block Register 1 (EBR1) .....                           | 429 |
| 18.5.4                                 | Erase Block Register 2 (EBR2) .....                           | 430 |
| 18.5.5                                 | RAM Emulation Register (RAMER).....                           | 430 |
| 18.6                                   | On-Board Programming Modes .....                              | 431 |
| 18.6.1                                 | Boot Mode .....                                               | 432 |
| 18.6.2                                 | Programming/Erasing in User Program Mode.....                 | 434 |
| 18.7                                   | Flash Memory Emulation in RAM .....                           | 435 |
| 18.8                                   | Flash Memory Programming/Erasing .....                        | 437 |
| 18.8.1                                 | Program/Program-Verify .....                                  | 437 |
| 18.8.2                                 | Erase/Erase-Verify .....                                      | 439 |
| 18.8.3                                 | Interrupt Handling when Programming/Erasing Flash Memory..... | 439 |
| 18.9                                   | Program/Erase Protection .....                                | 441 |
| 18.9.1                                 | Hardware Protection .....                                     | 441 |
| 18.9.2                                 | Software Protection.....                                      | 441 |
| 18.9.3                                 | Error Protection.....                                         | 441 |
| 18.10                                  | Programmer Mode .....                                         | 442 |
| 18.11                                  | Power-Down States for Flash Memory .....                      | 442 |
| Section 19 Clock Pulse Generator ..... |                                                               | 443 |
| 19.1                                   | Register Descriptions .....                                   | 444 |
| 19.1.1                                 | System Clock Control Register (SCKCR) .....                   | 444 |
| 19.1.2                                 | Low-Power Control Register (LPWRCR) .....                     | 445 |
| 19.2                                   | Oscillator.....                                               | 446 |
| 19.2.1                                 | Connecting a Crystal Resonator.....                           | 446 |
| 19.2.2                                 | External Clock Input.....                                     | 447 |

|                                             |                                                                                   |     |
|---------------------------------------------|-----------------------------------------------------------------------------------|-----|
| 19.3                                        | PLL Circuit .....                                                                 | 449 |
| 19.4                                        | Medium-Speed Clock Divider .....                                                  | 449 |
| 19.5                                        | Bus Master Clock Selection Circuit .....                                          | 449 |
| 19.6                                        | Usage Notes .....                                                                 | 450 |
| 19.6.1                                      | Note on Crystal Resonator .....                                                   | 450 |
| 19.6.2                                      | Note on Board Design .....                                                        | 450 |
| Section 20 Power-Down Modes .....           |                                                                                   | 453 |
| 20.1                                        | Register Descriptions .....                                                       | 456 |
| 20.1.1                                      | Standby Control Register (SBYCR) .....                                            | 456 |
| 20.1.2                                      | Module Stop Control Registers A to C (MSTPCRA to MSTPCRC) .....                   | 458 |
| 20.2                                        | Medium-Speed Mode .....                                                           | 459 |
| 20.3                                        | Sleep Mode .....                                                                  | 460 |
| 20.3.1                                      | Transition to Sleep Mode .....                                                    | 460 |
| 20.3.2                                      | Clearing Sleep Mode .....                                                         | 460 |
| 20.4                                        | Software Standby Mode .....                                                       | 461 |
| 20.4.1                                      | Transition to Software Standby Mode .....                                         | 461 |
| 20.4.2                                      | Clearing Software Standby Mode .....                                              | 461 |
| 20.4.3                                      | Setting Oscillation Stabilization Time after Clearing Software Standby Mode ..... | 462 |
| 20.4.4                                      | Software Standby Mode Application Example .....                                   | 463 |
| 20.5                                        | Hardware Standby Mode .....                                                       | 464 |
| 20.5.1                                      | Transition to Hardware Standby Mode .....                                         | 464 |
| 20.5.2                                      | Clearing Hardware Standby Mode .....                                              | 464 |
| 20.5.3                                      | Hardware Standby Mode Timings .....                                               | 464 |
| 20.6                                        | Module Stop Mode .....                                                            | 465 |
| 20.6.1                                      | Module Stop Mode .....                                                            | 465 |
| 20.7                                        | ø Clock Output Disabling Function .....                                           | 465 |
| 20.8                                        | Usage Notes .....                                                                 | 467 |
| 20.8.1                                      | I/O Port Status .....                                                             | 467 |
| 20.8.2                                      | Current Dissipation during Oscillation Stabilization Wait Period .....            | 467 |
| 20.8.3                                      | DTC Module Stop .....                                                             | 467 |
| 20.8.4                                      | On-Chip Supporting Module Interrupt .....                                         | 467 |
| 20.8.5                                      | Writing to MSTPCR .....                                                           | 467 |
| Section 21 Electrical Characteristics ..... |                                                                                   | 469 |
| 21.1                                        | Absolute Maximum Ratings .....                                                    | 469 |
| 21.2                                        | DC Characteristics .....                                                          | 470 |
| 21.3                                        | AC Characteristics .....                                                          | 472 |
| 21.3.1                                      | Clock Timing .....                                                                | 473 |
| 21.3.2                                      | Control Signal Timing .....                                                       | 474 |
| 21.3.3                                      | Timing of On-Chip Supporting Modules .....                                        | 476 |
| 21.4                                        | A/D Conversion Characteristics .....                                              | 481 |
| 21.5                                        | Flash Memory Characteristics .....                                                | 482 |

Appendix .....485

A. On-chip I/O Register.....485

    A.1 Register Addresses.....485

    A.2 Register Bits.....501

    A.3 Register States in Each Operating Mode .....515

B. I/O Port States in Each Pin State.....525

C. Product Code Lineup .....526

D. Package Dimensions .....526

Index .....527

# Figures

## Section 1 Overview

|            |                             |   |
|------------|-----------------------------|---|
| Figure 1-1 | Internal Block Diagram..... | 2 |
| Figure 1-2 | Pin Arrangement.....        | 3 |

## Section 2 CPU

|             |                                                            |    |
|-------------|------------------------------------------------------------|----|
| Figure 2-1  | Exception Vector Table (Normal Mode) .....                 | 13 |
| Figure 2-2  | Stack Structure in Normal Mode .....                       | 13 |
| Figure 2-3  | Exception Vector Table (Advanced Mode) .....               | 14 |
| Figure 2-4  | Stack Structure in Advanced Mode .....                     | 15 |
| Figure 2-5  | Memory Map .....                                           | 16 |
| Figure 2-6  | CPU Registers.....                                         | 17 |
| Figure 2-7  | Usage of General Registers.....                            | 18 |
| Figure 2-8  | Stack .....                                                | 19 |
| Figure 2-9  | General Register Data Formats (1) .....                    | 23 |
| Figure 2-9  | General Register Data Formats (2) .....                    | 24 |
| Figure 2-10 | Memory Data Formats .....                                  | 25 |
| Figure 2-11 | Instruction Formats (Examples).....                        | 37 |
| Figure 2-12 | Branch Address Specification in Memory Indirect Mode ..... | 41 |
| Figure 2-13 | State Transitions .....                                    | 45 |

## Section 3 MCU Operating Modes

|            |                   |    |
|------------|-------------------|----|
| Figure 3-1 | Address Map ..... | 50 |
|------------|-------------------|----|

## Section 4 Exception Handling

|            |                                                                                              |    |
|------------|----------------------------------------------------------------------------------------------|----|
| Figure 4-1 | Reset Sequence (Advanced Mode with On-Chip ROM Enabled) .....                                | 54 |
| Figure 4-2 | Reset Sequence (Advanced Mode with On-Chip ROM Disabled:<br>Cannot be Used in this LSI)..... | 55 |
| Figure 4-3 | Stack Status after Exception Handling.....                                                   | 58 |
| Figure 4-4 | Operation when SP Value is Odd .....                                                         | 59 |

## Section 5 Interrupt Controller

|            |                                                                                  |    |
|------------|----------------------------------------------------------------------------------|----|
| Figure 5-1 | Block Diagram of Interrupt Controller .....                                      | 62 |
| Figure 5-2 | Block Diagram of Interrupts IRQ0 to IRQ5 .....                                   | 69 |
| Figure 5-3 | Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 0 .. | 74 |
| Figure 5-4 | Flowchart of Procedure Up to Interrupt Acceptance in Control Mode 2 .....        | 76 |
| Figure 5-5 | Interrupt Exception Handling .....                                               | 77 |
| Figure 5-6 | Contention between Interrupt Generation and Disabling.....                       | 80 |

## Section 6 PC Break Controller (PBC)

|            |                                                |    |
|------------|------------------------------------------------|----|
| Figure 6-1 | Block Diagram of PC Break Controller.....      | 84 |
| Figure 6-2 | Operation in Power-Down Mode Transitions ..... | 87 |

## Section 7 Bus Controller

|            |                                                               |    |
|------------|---------------------------------------------------------------|----|
| Figure 7-1 | On-Chip Memory Access Cycle .....                             | 91 |
| Figure 7-2 | On-Chip Support Module Access Cycle .....                     | 92 |
| Figure 7-3 | On-Chip HCAN Module Access Cycle (Wait States Inserted) ..... | 93 |
| Figure 7-4 | On-Chip MMT Module Access Cycle .....                         | 93 |

## Section 8 Data Transfer Controller (DTC)

|             |                                                                                      |     |
|-------------|--------------------------------------------------------------------------------------|-----|
| Figure 8-1  | Block Diagram of DTC .....                                                           | 96  |
| Figure 8-2  | Block Diagram of DTC Activation Source Control .....                                 | 102 |
| Figure 8-3  | Correspondence between DTC Vector Address and Register Information .....             | 103 |
| Figure 8-4  | Flowchart of DTC Operation .....                                                     | 106 |
| Figure 8-5  | Memory Mapping in Normal Mode .....                                                  | 107 |
| Figure 8-6  | Memory Mapping in Repeat Mode .....                                                  | 108 |
| Figure 8-7  | Memory Mapping in Block Transfer Mode .....                                          | 109 |
| Figure 8-8  | Chain Transfer Operation .....                                                       | 110 |
| Figure 8-9  | DTC Operation Timing (Example in Normal Mode or Repeat Mode) .....                   | 111 |
| Figure 8-10 | DTC Operation Timing<br>(Example of Block Transfer Mode, with Block Size of 2) ..... | 112 |
| Figure 8-11 | DTC Operation Timing (Example of Chain Transfer) .....                               | 112 |

## Section 10 16-Bit Timer Pulse Unit (TPU)

|              |                                                                         |     |
|--------------|-------------------------------------------------------------------------|-----|
| Figure 10-1  | Block Diagram of TPU .....                                              | 152 |
| Figure 10-2  | Example of Counter Operation Setting Procedure .....                    | 187 |
| Figure 10-3  | Free-Running Counter Operation .....                                    | 188 |
| Figure 10-4  | Periodic Counter Operation .....                                        | 189 |
| Figure 10-5  | Example of Setting Procedure for Waveform Output by Compare Match ..... | 189 |
| Figure 10-6  | Example of 0 Output/1 Output Operation .....                            | 190 |
| Figure 10-7  | Example of Toggle Output Operation .....                                | 190 |
| Figure 10-8  | Example of Input Capture Operation Setting Procedure .....              | 191 |
| Figure 10-9  | Example of Input Capture Operation .....                                | 192 |
| Figure 10-10 | Example of Synchronous Operation Setting Procedure .....                | 193 |
| Figure 10-11 | Example of Synchronous Operation .....                                  | 194 |
| Figure 10-12 | Compare Match Buffer Operation .....                                    | 195 |
| Figure 10-13 | Input Capture Buffer Operation .....                                    | 195 |
| Figure 10-14 | Example of Buffer Operation Setting Procedure .....                     | 196 |
| Figure 10-15 | Example of Buffer Operation (1) .....                                   | 196 |
| Figure 10-16 | Example of Buffer Operation (2) .....                                   | 197 |
| Figure 10-17 | Cascaded Operation Setting Procedure .....                              | 198 |
| Figure 10-18 | Example of Cascaded Operation (1) .....                                 | 199 |
| Figure 10-19 | Example of Cascaded Operation (2) .....                                 | 199 |
| Figure 10-20 | Example of PWM Mode Setting Procedure .....                             | 201 |
| Figure 10-21 | Example of PWM Mode Operation (1) .....                                 | 201 |
| Figure 10-22 | Example of PWM Mode Operation (2) .....                                 | 202 |
| Figure 10-23 | Example of PWM Mode Operation (3) .....                                 | 203 |

|              |                                                                        |     |
|--------------|------------------------------------------------------------------------|-----|
| Figure 10-24 | Example of Phase Counting Mode Setting Procedure .....                 | 204 |
| Figure 10-25 | Example of Phase Counting Mode 1 Operation.....                        | 205 |
| Figure 10-26 | Example of Phase Counting Mode 2 Operation.....                        | 206 |
| Figure 10-27 | Example of Phase Counting Mode 3 Operation.....                        | 207 |
| Figure 10-28 | Example of Phase Counting Mode 4 Operation.....                        | 208 |
| Figure 10-29 | Phase Counting Mode Application Example .....                          | 210 |
| Figure 10-30 | Count Timing in Internal Clock Operation.....                          | 214 |
| Figure 10-31 | Count Timing in External Clock Operation .....                         | 214 |
| Figure 10-32 | Output Compare Output Timing .....                                     | 215 |
| Figure 10-33 | Input Capture Input Signal Timing .....                                | 215 |
| Figure 10-34 | Counter Clear Timing (Compare Match).....                              | 216 |
| Figure 10-35 | Counter Clear Timing (Input Capture).....                              | 216 |
| Figure 10-36 | Buffer Operation Timing (Compare Match).....                           | 217 |
| Figure 10-37 | Buffer Operation Timing (Input Capture).....                           | 217 |
| Figure 10-38 | TGI Interrupt Timing (Compare Match).....                              | 218 |
| Figure 10-39 | TGI Interrupt Timing (Input Capture) .....                             | 218 |
| Figure 10-40 | TCIV Interrupt Setting Timing .....                                    | 219 |
| Figure 10-41 | TCIU Interrupt Setting Timing .....                                    | 219 |
| Figure 10-42 | Timing for Status Flag Clearing by CPU.....                            | 220 |
| Figure 10-43 | Timing for Status Flag Clearing by DTC Activation.....                 | 220 |
| Figure 10-44 | Phase Difference, Overlap, and Pulse Width in Phase Counting Mode..... | 221 |
| Figure 10-45 | Contention between TCNT Write and Clear Operations .....               | 222 |
| Figure 10-46 | Contention between TCNT Write and Increment Operations.....            | 223 |
| Figure 10-47 | Contention between TGR Write and Compare Match .....                   | 224 |
| Figure 10-48 | Contention between Buffer Register Write and Compare Match .....       | 225 |
| Figure 10-49 | Contention between TGR Read and Input Capture.....                     | 226 |
| Figure 10-50 | Contention between TGR Write and Input Capture.....                    | 227 |
| Figure 10-51 | Contention between Buffer Register Write and Input Capture .....       | 228 |
| Figure 10-52 | Contention between Overflow and Counter Clearing .....                 | 229 |
| Figure 10-53 | Contention between TCNT Write and Overflow .....                       | 230 |

## **Section 11 Motor Management Timer (MMT)**

|              |                                                                     |     |
|--------------|---------------------------------------------------------------------|-----|
| Figure 11-1  | Block Diagram of MMT .....                                          | 232 |
| Figure 11-2  | Sample Operating Mode Setting Procedure.....                        | 239 |
| Figure 11-3  | MMT Canceling Procedure .....                                       | 240 |
| Figure 11-4  | Example of TCNT Count Operation.....                                | 241 |
| Figure 11-5  | Examples of Counter and Register Operations .....                   | 242 |
| Figure 11-6  | Example of PWM Waveform Generation.....                             | 245 |
| Figure 11-7  | Example of TCNT Counter Clearing .....                              | 246 |
| Figure 11-8  | Example of Toggle Output Waveform Synchronized with PWM Period..... | 246 |
| Figure 11-9  | Count Timing.....                                                   | 248 |
| Figure 11-10 | TCNT Counter Clearing Timing.....                                   | 248 |
| Figure 11-11 | TDCNT Operation Timing .....                                        | 249 |

|              |                                                                   |     |
|--------------|-------------------------------------------------------------------|-----|
| Figure 11-12 | Dead Time Generation Timing .....                                 | 250 |
| Figure 11-13 | Buffer Operation Timing .....                                     | 251 |
| Figure 11-14 | TGI Interrupt Timing .....                                        | 252 |
| Figure 11-15 | Timing of Status Flag Clearing by CPU .....                       | 252 |
| Figure 11-16 | Timing of Status Flag Clearing by DTC Controller.....             | 253 |
| Figure 11-17 | Contention between Buffer Register Write and Compare Match .....  | 254 |
| Figure 11-18 | Contention between Compare Register Write and Compare Match ..... | 255 |
| Figure 11-19 | Block Diagram of POE .....                                        | 256 |
| Figure 11-20 | Low Level Detection Operation.....                                | 261 |

## Section 12 Programmable Pulse Generator (PPG)

|              |                                                                       |     |
|--------------|-----------------------------------------------------------------------|-----|
| Figure 12-1  | Block Diagram of PPG .....                                            | 264 |
| Figure 12-2  | PPG Output Operation .....                                            | 272 |
| Figure 12-3  | Timing of Transfer and Output of NDR Contents (Example).....          | 273 |
| Figure 12-4  | Setup Procedure for Normal Pulse Output (Example).....                | 274 |
| Figure 12-5  | Normal Pulse Output Example (Five-Phase Pulse Output).....            | 275 |
| Figure 12-6  | Non-Overlapping Pulse Output.....                                     | 276 |
| Figure 12-7  | Non-Overlapping Operation and NDR Write Timing.....                   | 277 |
| Figure 12-8  | Setup Procedure for Non-Overlapping Pulse Output (Example) .....      | 278 |
| Figure 12-9  | Non-Overlapping Pulse Output Example (Four-Phase Complementary) ..... | 279 |
| Figure 12-10 | Inverted Pulse Output (Example).....                                  | 281 |
| Figure 12-11 | Pulse Output Triggered by Input Capture (Example) .....               | 282 |

## Section 13 Watchdog Timer

|             |                                                            |     |
|-------------|------------------------------------------------------------|-----|
| Figure 13-1 | Block Diagram of WDT.....                                  | 283 |
| Figure 13-2 | Writing to TCNT, TCSR, and RSTCSR (example for WDT0) ..... | 289 |
| Figure 13-3 | Contention between TCNT Write and Increment .....          | 289 |

## Section 14 Serial Communication Interface (SCI)

|             |                                                                                                                   |     |
|-------------|-------------------------------------------------------------------------------------------------------------------|-----|
| Figure 14-1 | Block Diagram of SCI .....                                                                                        | 292 |
| Figure 14-2 | Data Format in Asynchronous Communication<br>(Example with 8-Bit Data, Parity, Two Stop Bits) .....               | 315 |
| Figure 14-3 | Receive Data Sampling Timing in Asynchronous Mode.....                                                            | 317 |
| Figure 14-4 | Relationship between Output Clock and Transfer Data Phase<br>(Asynchronous Mode) .....                            | 318 |
| Figure 14-5 | Sample SCI Initialization Flowchart.....                                                                          | 319 |
| Figure 14-6 | Example of Operation in Transmission in Asynchronous Mode<br>(Example with 8-Bit Data, Parity, One Stop Bit)..... | 320 |
| Figure 14-7 | Sample Serial Transmission Flowchart.....                                                                         | 321 |
| Figure 14-8 | Example of SCI Operation in Reception<br>(Example with 8-Bit Data, Parity, One Stop Bit).....                     | 322 |
| Figure 14-9 | Sample Serial Reception Data Flowchart (1).....                                                                   | 324 |
| Figure 14-9 | Sample Serial Reception Data Flowchart (2).....                                                                   | 325 |

|                                                                                                                              |     |
|------------------------------------------------------------------------------------------------------------------------------|-----|
| Figure 14-10 Example of Communication Using Multiprocessor Format<br>(Transmission of Data H'AA to Receiving Station A)..... | 327 |
| Figure 14-11 Sample Multiprocessor Serial Transmission Flowchart.....                                                        | 328 |
| Figure 14-12 Example of SCI Operation in Reception<br>(Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit) .....      | 329 |
| Figure 14-13 Sample Multiprocessor Serial Reception Flowchart (1) .....                                                      | 330 |
| Figure 14-13 Sample Multiprocessor Serial Reception Flowchart (2) .....                                                      | 331 |
| Figure 14-14 Data Format in Synchronous Communication (For LSB-First).....                                                   | 332 |
| Figure 14-15 Sample SCI Initialization Flowchart.....                                                                        | 333 |
| Figure 14-16 Sample SCI Transmission Operation in Clocked Synchronous Mode.....                                              | 335 |
| Figure 14-17 Sample Serial Transmission Flowchart.....                                                                       | 336 |
| Figure 14-18 Example of SCI Operation in Reception.....                                                                      | 337 |
| Figure 14-19 Sample Serial Reception Flowchart.....                                                                          | 338 |
| Figure 14-20 Sample Flowchart of Simultaneous Serial Transmit and Receive Operations.....                                    | 340 |
| Figure 14-21 Schematic Diagram of Smart Card Interface Pin Connections .....                                                 | 341 |
| Figure 14-22 Normal Smart Card Interface Data Format.....                                                                    | 342 |
| Figure 14-23 Direct Convention (SDIR = SINV = $O/\overline{E}$ = 0).....                                                     | 342 |
| Figure 14-24 Inverse Convention (SDIR = SINV = $O/\overline{E}$ = 1).....                                                    | 343 |
| Figure 14-25 Receive Data Sampling Timing in Smart Card Mode<br>(Using Clock of 372 Times the Transfer Rate).....            | 344 |
| Figure 14-26 Retransfer Operation in SCI Transmit Mode .....                                                                 | 346 |
| Figure 14-27 TEND Flag Generation Timing in Transmission Operation.....                                                      | 347 |
| Figure 14-28 Example of Transmission Processing Flow .....                                                                   | 348 |
| Figure 14-29 Retransfer Operation in SCI Receive Mode .....                                                                  | 349 |
| Figure 14-30 Example of Reception Processing Flow .....                                                                      | 350 |
| Figure 14-31 Timing for Fixing Clock Output Level .....                                                                      | 350 |
| Figure 14-32 Clock Halt and Restart Procedure.....                                                                           | 351 |

## Section 15 Hitachi Controller Area Network (HCAN)

|                                                           |     |
|-----------------------------------------------------------|-----|
| Figure 15-1 HCAN Block Diagram.....                       | 356 |
| Figure 15-2 Message Control Register Configuration.....   | 380 |
| Figure 15-3 Standard Format.....                          | 380 |
| Figure 15-4 Extended Format.....                          | 380 |
| Figure 15-5 Message Data Configuration.....               | 382 |
| Figure 15-6 Hardware Reset Flowchart.....                 | 384 |
| Figure 15-7 Software Reset Flowchart.....                 | 385 |
| Figure 15-8 Detailed Description of One Bit .....         | 386 |
| Figure 15-9 Transmission Flowchart.....                   | 389 |
| Figure 15-10 Transmit Message Cancellation Flowchart..... | 391 |
| Figure 15-11 Reception Flowchart.....                     | 392 |
| Figure 15-12 Unread Message Overwrite Flowchart .....     | 394 |
| Figure 15-13 HCAN Sleep Mode Flowchart.....               | 395 |
| Figure 15-14 HCAN Halt Mode Flowchart.....                | 397 |

|                                                         |     |
|---------------------------------------------------------|-----|
| Figure 15-15 DTC Transfer Flowchart.....                | 399 |
| Figure 15-16 High-Speed Interface Using PCA82C250 ..... | 400 |

## Section 16 A/D Converter

|                                                             |     |
|-------------------------------------------------------------|-----|
| Figure 16-1 Block Diagram of A/D Converter.....             | 404 |
| Figure 16-2 A/D Conversion Timing .....                     | 411 |
| Figure 16-3 External Trigger Input Timing.....              | 413 |
| Figure 16-4 A/D Conversion Precision Definitions.....       | 415 |
| Figure 16-5 A/D Conversion Precision Definitions.....       | 415 |
| Figure 16-6 Example of Analog Input Circuit.....            | 416 |
| Figure 16-7 Example of Analog Input Protection Circuit..... | 418 |
| Figure 16-8 Analog Input Pin Equivalent Circuit.....        | 418 |

## Section 18 ROM

|                                                                             |     |
|-----------------------------------------------------------------------------|-----|
| Figure 18-1 Block Diagram of Flash Memory .....                             | 422 |
| Figure 18-2 Flash Memory State Transitions .....                            | 423 |
| Figure 18-3 Boot Mode .....                                                 | 424 |
| Figure 18-4 User Program Mode.....                                          | 425 |
| Figure 18-5 Flash Memory Block Configuration .....                          | 426 |
| Figure 18-6 Programming/Erasing Flowchart Example in User Program Mode..... | 434 |
| Figure 18-7 Flowchart for Flash Memory Emulation in RAM.....                | 435 |
| Figure 18-8 Example of RAM Overlap Operation .....                          | 436 |
| Figure 18-9 Program/Program-Verify Flowchart .....                          | 438 |
| Figure 18-10 Erase/Erase-Verify Flowchart.....                              | 440 |

## Section 19 Clock Pulse Generator

|                                                                 |     |
|-----------------------------------------------------------------|-----|
| Figure 19-1 Block Diagram of Clock Pulse Generator.....         | 443 |
| Figure 19-2 Connection of Crystal Resonator (Example) .....     | 446 |
| Figure 19-3 Crystal Resonator Equivalent Circuit.....           | 446 |
| Figure 19-4 External Clock Input (Examples).....                | 447 |
| Figure 19-5 External Clock Input Timing .....                   | 448 |
| Figure 19-6 Note on Board Design of Oscillator Circuit.....     | 450 |
| Figure 19-7 External Circuitry Recommended for PLL Circuit..... | 451 |

## Section 20 Power-Down Modes

|                                                                    |     |
|--------------------------------------------------------------------|-----|
| Figure 20-1 Mode Transition Diagram.....                           | 454 |
| Figure 20-2 Medium-Speed Mode Transition and Clearance Timing..... | 460 |
| Figure 20-3 Software Standby Mode Application Example.....         | 463 |
| Figure 20-4 Timing of Transition to Hardware Standby Mode .....    | 464 |
| Figure 20-5 Timing of Recovery from Hardware Standby Mode .....    | 465 |

## Section 21 Electrical Characteristics

|                                                    |     |
|----------------------------------------------------|-----|
| Figure 21-1 Output Load Circuit.....               | 472 |
| Figure 21-2 System Clock Timing .....              | 473 |
| Figure 21-3 Oscillation Stabilization Timing ..... | 474 |
| Figure 21-4 Reset Input Timing .....               | 475 |

|              |                                                       |     |
|--------------|-------------------------------------------------------|-----|
| Figure 21-5  | Interrupt Input Timing .....                          | 475 |
| Figure 21-6  | I/O Port Input/Output Timing .....                    | 477 |
| Figure 21-7  | TPU Input/Output Timing.....                          | 478 |
| Figure 21-8  | TPU Clock Input Timing .....                          | 478 |
| Figure 21-9  | SCK Clock Input Timing.....                           | 478 |
| Figure 21-10 | SCI Input/Output Timing (Clock Synchronous Mode)..... | 479 |
| Figure 21-11 | A/D Converter External Trigger Input Timing .....     | 479 |
| Figure 21-12 | HCAN Input/Output Timing.....                         | 479 |
| Figure 21-13 | PPG Output Timing .....                               | 479 |
| Figure 21-14 | MMT Input/Output Timing.....                          | 480 |
| Figure 21-15 | POE Input/Output Timing.....                          | 480 |

## Appendix

|            |                                 |     |
|------------|---------------------------------|-----|
| Figure D.1 | FP-80Q Package Dimensions ..... | 526 |
|------------|---------------------------------|-----|

# Tables

## Section 2 CPU

|            |                                         |    |
|------------|-----------------------------------------|----|
| Table 2-1  | Instruction Classification.....         | 26 |
| Table 2-2  | Operation Notation.....                 | 27 |
| Table 2-3  | Data Transfer Instructions.....         | 28 |
| Table 2-4  | Arithmetic Operations Instructions..... | 29 |
| Table 2-5  | Logic Operations Instructions.....      | 31 |
| Table 2-6  | Shift Instructions.....                 | 31 |
| Table 2-7  | Bit Manipulation Instructions.....      | 32 |
| Table 2-8  | Branch Instructions.....                | 34 |
| Table 2-9  | System Control Instructions.....        | 35 |
| Table 2-10 | Block Data Transfer Instructions.....   | 36 |
| Table 2-11 | Addressing Modes.....                   | 38 |
| Table 2-12 | Absolute Address Access Ranges.....     | 39 |
| Table 2-13 | Effective Address Calculation.....      | 42 |

## Section 3 MCU Operating Modes

|           |                                   |    |
|-----------|-----------------------------------|----|
| Table 3-1 | MCU Operating Mode Selection..... | 47 |
| Table 3-2 | Pin Functions in Each Mode.....   | 49 |

## Section 4 Exception Handling

|           |                                                                      |    |
|-----------|----------------------------------------------------------------------|----|
| Table 4-1 | Exception Types and Priority.....                                    | 51 |
| Table 4-2 | Exception Handling Vector Table.....                                 | 52 |
| Table 4-3 | Status of CCR and EXR after Trace Exception Handling.....            | 56 |
| Table 4-4 | Status of CCR and EXR after Trap Instruction Exception Handling..... | 57 |

## Section 5 Interrupt Controller

|           |                                                                      |    |
|-----------|----------------------------------------------------------------------|----|
| Table 5-1 | Pin Configuration.....                                               | 63 |
| Table 5-2 | Interrupt Sources, Vector Addresses, and Interrupt Priorities.....   | 71 |
| Table 5-3 | Interrupt Control Modes.....                                         | 73 |
| Table 5-4 | Interrupt Response Times.....                                        | 78 |
| Table 5-5 | Number of States in Interrupt Handling Routine Execution Status..... | 79 |

## Section 8 Data Transfer Controller (DTC)

|           |                                                                       |     |
|-----------|-----------------------------------------------------------------------|-----|
| Table 8-1 | Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs..... | 104 |
| Table 8-2 | Register Information in Normal Mode.....                              | 107 |
| Table 8-3 | Register Information in Repeat Mode.....                              | 108 |
| Table 8-4 | Register Information in Block Transfer Mode.....                      | 109 |
| Table 8-5 | DTC Execution Status.....                                             | 113 |
| Table 8-6 | Number of States Required for Each Execution Status.....              | 113 |

## Section 9 I/O Ports

|           |                       |     |
|-----------|-----------------------|-----|
| Table 9-1 | Port Functions.....   | 120 |
| Table 9-2 | P17 Pin Function..... | 124 |

|            |                        |     |
|------------|------------------------|-----|
| Table 9-3  | P16 Pin Function ..... | 124 |
| Table 9-4  | P15 Pin Function ..... | 124 |
| Table 9-5  | P14 Pin Function ..... | 125 |
| Table 9-6  | P13 Pin Function ..... | 125 |
| Table 9-7  | P12 Pin Function ..... | 125 |
| Table 9-8  | P11 Pin Function ..... | 126 |
| Table 9-9  | P10 Pin Function ..... | 126 |
| Table 9-10 | PA3 Pin Function ..... | 131 |
| Table 9-11 | PA2 Pin Function ..... | 131 |
| Table 9-12 | PA1 Pin Function ..... | 131 |
| Table 9-13 | PA0 Pin Function ..... | 131 |
| Table 9-14 | PB7 Pin Function ..... | 135 |
| Table 9-15 | PB6 Pin Function ..... | 135 |
| Table 9-16 | PB5 Pin Function ..... | 135 |
| Table 9-17 | PB4 Pin Function ..... | 136 |
| Table 9-18 | PB3 Pin Function ..... | 136 |
| Table 9-19 | PB2 Pin Function ..... | 136 |
| Table 9-20 | PB1 Pin Function ..... | 137 |
| Table 9-21 | PB0 Pin Function ..... | 137 |

## **Section 10 16-Bit Timer Pulse Unit (TPU)**

|             |                                               |     |
|-------------|-----------------------------------------------|-----|
| Table 10-1  | TPU Functions .....                           | 150 |
| Table 10-2  | TPU Pins .....                                | 153 |
| Table 10-3  | CCLR0 to CCLR2 (channels 0 and 3).....        | 157 |
| Table 10-4  | CCLR0 to CCLR2 (channels 1, 2, 4, and 5)..... | 157 |
| Table 10-5  | TPSC0 to TPSC2 (channel 0).....               | 158 |
| Table 10-6  | TPSC0 to TPSC2 (channel 1).....               | 158 |
| Table 10-7  | TPSC0 to TPSC2 (channel 2).....               | 159 |
| Table 10-8  | TPSC0 to TPSC2 (channel 3).....               | 159 |
| Table 10-9  | TPSC0 to TPSC2 (channel 4).....               | 160 |
| Table 10-10 | TPSC0 to TPSC2 (channel 5).....               | 160 |
| Table 10-11 | MD0 to MD3 .....                              | 162 |
| Table 10-12 | TIORH_0 (channel 0).....                      | 164 |
| Table 10-13 | TIORL_0 (channel 0) .....                     | 165 |
| Table 10-14 | TIOR_1 (channel 1) .....                      | 166 |
| Table 10-15 | TIOR_2 (channel 2) .....                      | 167 |
| Table 10-16 | TIORH_3 (channel 3).....                      | 168 |
| Table 10-17 | TIORL_3 (channel 3).....                      | 169 |
| Table 10-18 | TIOR_4 (channel 4) .....                      | 170 |
| Table 10-19 | TIOR_5 (channel 5) .....                      | 171 |
| Table 10-20 | TIORH_0 (channel 0).....                      | 172 |
| Table 10-21 | TIORL_0 (channel 0) .....                     | 173 |
| Table 10-22 | TIOR_1 (channel 1) .....                      | 174 |

|             |                                                        |     |
|-------------|--------------------------------------------------------|-----|
| Table 10-23 | TIOR_2 (channel 2).....                                | 175 |
| Table 10-24 | TIORH_3 (channel 3).....                               | 176 |
| Table 10-25 | TIORL_3 (channel 3).....                               | 177 |
| Table 10-26 | TIOR_4 (channel 4).....                                | 178 |
| Table 10-27 | TIOR_5 (channel 5).....                                | 179 |
| Table 10-28 | Register Combinations in Buffer Operation.....         | 195 |
| Table 10-29 | Cascaded Combinations.....                             | 198 |
| Table 10-30 | PWM Output Registers and Output Pins.....              | 200 |
| Table 10-31 | Phase Counting Mode Clock Input Pins.....              | 204 |
| Table 10-33 | Up/Down-Count Conditions in Phase Counting Mode 2..... | 206 |
| Table 10-34 | Up/Down-Count Conditions in Phase Counting Mode 3..... | 207 |
| Table 10-35 | Up/Down-Count Conditions in Phase Counting Mode 4..... | 208 |
| Table 10-36 | TPU Interrupts.....                                    | 212 |

## **Section 11 Motor Management Timer (MMT)**

|            |                                                        |     |
|------------|--------------------------------------------------------|-----|
| Table 11-1 | Pin Configuration.....                                 | 233 |
| Table 11-2 | Initial Values of TBRU to TBRW and Initial Output..... | 243 |
| Table 11-3 | MMT Interrupt Sources.....                             | 247 |
| Table 11-4 | Pin Configuration.....                                 | 257 |

## **Section 12 Programmable Pulse Generator (PPG)**

|            |                   |     |
|------------|-------------------|-----|
| Table 12-1 | PPG I/O Pins..... | 265 |
|------------|-------------------|-----|

## **Section 13 Watchdog Timer**

|            |                           |     |
|------------|---------------------------|-----|
| Table 13-1 | WDT Interrupt Source..... | 288 |
|------------|---------------------------|-----|

## **Section 14 Serial Communication Interface (SCI)**

|             |                                                                                                            |     |
|-------------|------------------------------------------------------------------------------------------------------------|-----|
| Table 14-1  | Pin Configuration.....                                                                                     | 293 |
| Table 14-2  | The Relationships between The N Setting in BRR and Bit Rate B.....                                         | 308 |
| Table 14-3  | BRR Settings for Various Bit Rates (Asynchronous Mode).....                                                | 309 |
| Table 14-4  | Maximum Bit Rate for Each Frequency (Asynchronous Mode).....                                               | 311 |
| Table 14-5  | Maximum Bit Rate with External Clock Input (Asynchronous Mode).....                                        | 312 |
| Table 14-6  | BRR Settings for Various Bit Rates (Clocked Synchronous Mode).....                                         | 313 |
| Table 14-7  | Maximum Bit Rate with External Clock Input (Clocked Synchronous Mode).....                                 | 313 |
| Table 14-8  | Examples of Bit Rate for Various BRR Settings<br>(Smart Card Interface Mode) (When n = 0 and S = 372)..... | 314 |
| Table 14-9  | Maximum Bit Rate at Various Frequencies<br>(Smart Card Interface Mode) (When S = 372).....                 | 314 |
| Table 14-10 | Serial Transfer Formats (Asynchronous Mode).....                                                           | 316 |
| Table 14-11 | SSR Status Flags and Receive Data Handling.....                                                            | 323 |
| Table 14-12 | SCI Interrupt Sources.....                                                                                 | 352 |
| Table 14-13 | SCI Interrupt Sources.....                                                                                 | 353 |

## **Section 15 Hitachi Controller Area Network (HCAN)**

|            |                                |     |
|------------|--------------------------------|-----|
| Table 15-1 | HCAN Pins.....                 | 357 |
| Table 15-2 | Limits for Settable Value..... | 386 |

|                                              |                                                                                             |     |
|----------------------------------------------|---------------------------------------------------------------------------------------------|-----|
| Table 15-3                                   | Setting Range for TSEG1 and TSEG2 in BCR.....                                               | 387 |
| Table 15-4                                   | HCAN Interrupt Sources .....                                                                | 398 |
| <b>Section 16 A/D Converter</b>              |                                                                                             |     |
| Table 16-1                                   | Pin Configuration .....                                                                     | 405 |
| Table 16-2                                   | Analog Input Channels and Corresponding ADDR Registers.....                                 | 407 |
| Table 16-3                                   | A/D Conversion Time (Single Mode).....                                                      | 412 |
| Table 16-4                                   | A/D Conversion Time (Scan Mode) .....                                                       | 412 |
| Table 16-5                                   | A/D Converter Interrupt Source .....                                                        | 413 |
| Table 16-6                                   | Analog Pin Specifications .....                                                             | 418 |
| <b>Section 18 ROM</b>                        |                                                                                             |     |
| Table 18-1                                   | Differences between Boot Mode and User Program Mode.....                                    | 423 |
| Table 18-2                                   | Pin Configuration .....                                                                     | 427 |
| Table 18-3                                   | Setting On-Board Programming Modes .....                                                    | 431 |
| Table 18-4                                   | Boot Mode Operation.....                                                                    | 433 |
| Table 18-5                                   | System Clock Frequencies for which Automatic Adjustment of LSI Bit Rate<br>is Possible..... | 433 |
| Table 18-6                                   | Flash Memory Operating States .....                                                         | 442 |
| <b>Section 19 Clock Pulse Generator</b>      |                                                                                             |     |
| Table 19-1                                   | Damping Resistance Value .....                                                              | 446 |
| Table 19-2                                   | Crystal Resonator Characteristics .....                                                     | 446 |
| Table 19-3                                   | External Clock Input Conditions.....                                                        | 448 |
| <b>Section 20 Power-Down Modes</b>           |                                                                                             |     |
| Table 20-1                                   | Low Power Dissipation Mode Transition Conditions .....                                      | 453 |
| Table 20-2                                   | LSI Internal States in Each Mode.....                                                       | 455 |
| Table 20-3                                   | Oscillation Stabilization Time Settings .....                                               | 462 |
| Table 20-4                                   | ø Pin State in Each Processing State .....                                                  | 466 |
| <b>Section 21 Electrical Characteristics</b> |                                                                                             |     |
| Table 21-1                                   | Absolute Maximum Ratings.....                                                               | 469 |
| Table 21-2                                   | DC Characteristics.....                                                                     | 470 |
| Table 21-3                                   | Permissible Output Currents .....                                                           | 472 |
| Table 21-4                                   | Clock Timing .....                                                                          | 473 |
| Table 21-5                                   | Control Signal Timing.....                                                                  | 474 |
| Table 21-6                                   | Timing of On-Chip Supporting Modules .....                                                  | 476 |
| Table 21-7                                   | A/D Conversion Characteristics.....                                                         | 481 |
| Table 21-8                                   | Flash Memory Characteristics.....                                                           | 482 |

# Section 1 Overview

## 1.1 Overview

- High-speed H8S/2600 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
  - 69 basic instructions
- Various peripheral functions
  - PC break controller
  - Data transfer controller
  - 16-bit timer-pulse unit (TPU)
  - Motor management timer (MMT)
  - Programmable pulse generator (PPG)
  - Watchdog timer
  - Asynchronous or clocked synchronous serial communication interface (SCI)
  - Hitachi controller area network (HCAN)
  - 10-bit A/D converter
  - Clock pulse generator
- On-chip memory

| ROM            | Model     | ROM   | RAM | Remarks     |
|----------------|-----------|-------|-----|-------------|
| F-ZTAT Version | HD64F2612 | 128kB | 4kB |             |
| Mask ROM       | HD6432612 | 128kB | 4kB | In planning |
| Version        | HD6432611 | 64kB  | 4kB |             |

- General I/O ports
  - I/O pins: 43
  - Input-only pins: 13
- Supports various power-down states
- Compact package

| Package | (Code) | Body Size      | Pin Pitch |
|---------|--------|----------------|-----------|
| QFP-80  | FP-80Q | 14.0 × 14.0 mm | 0.65 mm   |

## 1.2 Internal Block Diagram

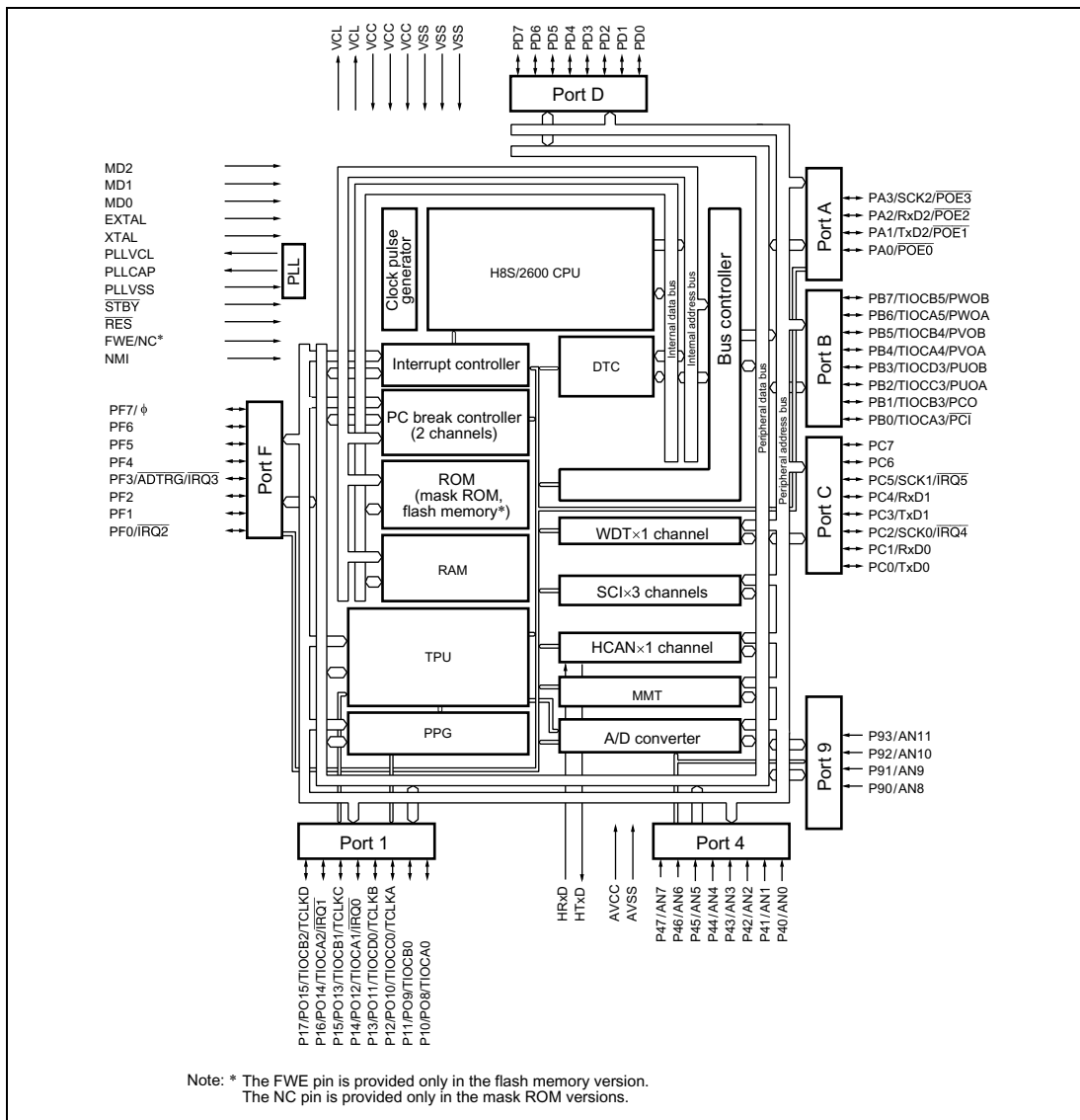
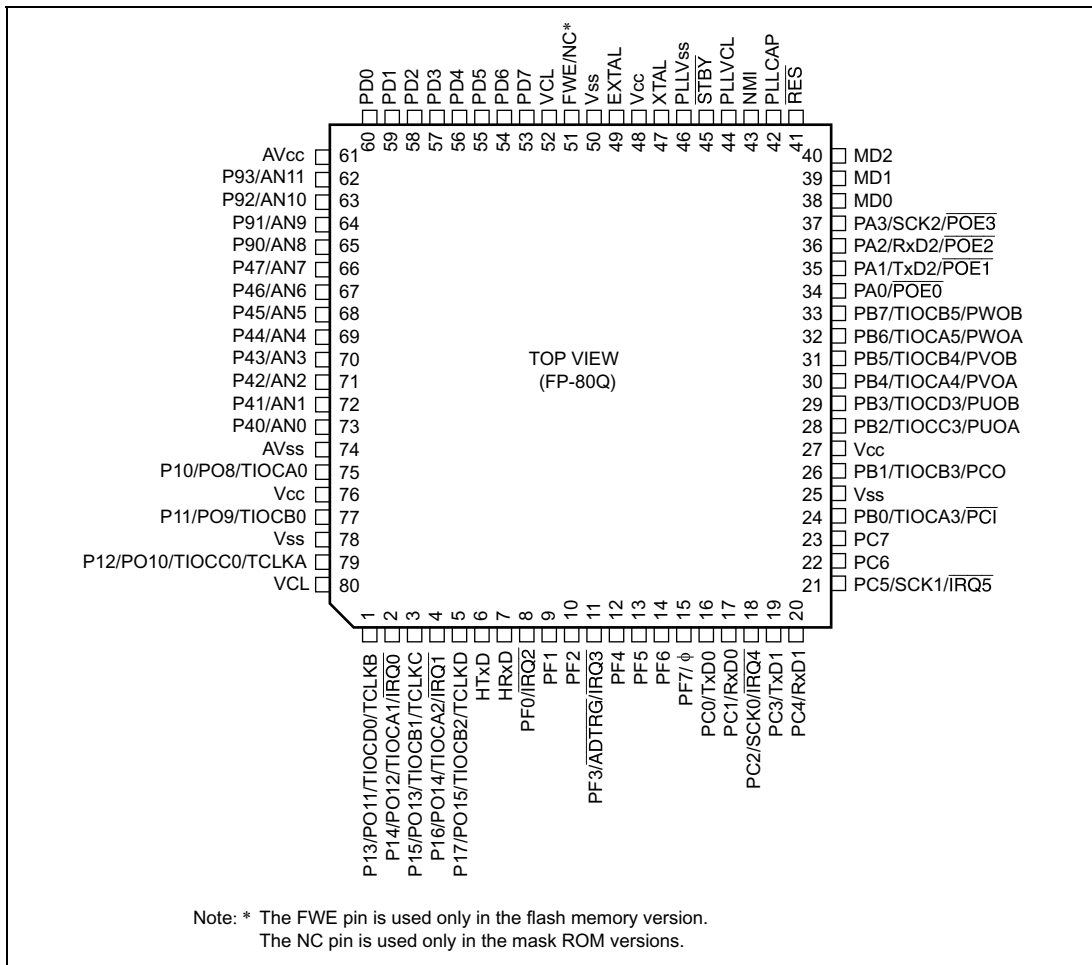


Figure 1-1 Internal Block Diagram

## 1.3 Pin Arrangement



**Figure 1-2 Pin Arrangement**

## 1.4 Pin Functions

| Type                   | Symbol | Pin NO. | I/O    | Function                                                                                                                                                                                                     |
|------------------------|--------|---------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Power Supply           | VCC    | 27      | Input  | Power supply pins. Connect all these pins to the system power supply.                                                                                                                                        |
|                        |        | 48      |        |                                                                                                                                                                                                              |
|                        |        | 76      |        |                                                                                                                                                                                                              |
|                        | VSS    | 25      | Input  | Ground pins. Connect all these pins to the system power supply (0V).                                                                                                                                         |
|                        |        | 50      |        |                                                                                                                                                                                                              |
|                        |        | 78      |        |                                                                                                                                                                                                              |
|                        | VCL    | 52      | Output | External capacitance pin for internal power-down power supply. Connect this pin to VSS via a 0.1- $\mu$ F capacitor (placed close to the pins).                                                              |
|                        |        | 80      |        |                                                                                                                                                                                                              |
|                        |        |         |        |                                                                                                                                                                                                              |
| Clock                  | PLLVCL | 44      | Output | External capacitance pin for internal power-down power supply for an on-chip PLL oscillator. Connect this pin to PLLVSS via a 0.1- $\mu$ F capacitor (placed close to the pins).                             |
|                        | PLLVSS | 46      | Input  | On-chip PLL oscillator ground pin.                                                                                                                                                                           |
|                        | PLLCAP | 42      | Output | External capacitance pin for an on-chip PLL oscillator.                                                                                                                                                      |
|                        | XTAL   | 47      | Input  | For connection to a crystal resonator. For examples of crystal resonator connection and external clock input, see section 19, Clock Pulse Generator.                                                         |
|                        | EXTAL  | 49      | Input  | For connection to a crystal resonator.( An external clock can be supplied from the EXTAL pin.) For examples of crystal resonator connection and external clock input, see section 19, Clock Pulse Generator. |
|                        | $\phi$ | 15      | Output | Supplies the system clock to external devices.                                                                                                                                                               |
| Operating mode control | MD2    | 40      | Input  | Set the operating mode. Inputs at these pins should not be changed during operation.                                                                                                                         |
|                        | MD1    | 39      |        |                                                                                                                                                                                                              |
|                        | MD0    | 38      |        |                                                                                                                                                                                                              |
| System control         | RES    | 41      | Input  | Reset input pin. When this pin is low, the chip is reset.                                                                                                                                                    |
|                        | STBY   | 45      | Input  | When this pin is low, a transition is made to hardware standby mode.                                                                                                                                         |
|                        | FWE    | 51      | Input  | Pin for use by flash memory. This pin is only used in the flash memory version.                                                                                                                              |

| Type                                | Symbol                   | Pin NO. | I/O          | Function                                                                      |
|-------------------------------------|--------------------------|---------|--------------|-------------------------------------------------------------------------------|
| Interrupts                          | NMI                      | 43      | Input        | Nonmaskable interrupt pin. If this pin is not used, it should be fixed-high.  |
|                                     | $\overline{\text{IRQ5}}$ | 21      | Input        | These pins request a maskable interrupt.                                      |
|                                     | $\overline{\text{IRQ4}}$ | 18      |              |                                                                               |
|                                     | $\overline{\text{IRQ3}}$ | 11      |              |                                                                               |
|                                     | $\overline{\text{IRQ2}}$ | 8       |              |                                                                               |
|                                     | $\overline{\text{IRQ1}}$ | 4       |              |                                                                               |
|                                     | $\overline{\text{IRQ0}}$ | 2       |              |                                                                               |
| 16-bit timer-pulse unit             | TCLKA                    | 79      | Input        | These pins input an external clock.                                           |
|                                     | TCLKB                    | 1       |              |                                                                               |
|                                     | TCLKC                    | 3       |              |                                                                               |
|                                     | TCLKD                    | 5       |              |                                                                               |
|                                     | TIOCA0                   | 75      | Input/Output | TGRA_0 to TGRD_0 input capture input/output compare output/PWM output pins.   |
|                                     | TIOCB0                   | 77      |              |                                                                               |
|                                     | TIOCC0                   | 79      |              |                                                                               |
|                                     | TIOCD0                   | 1       |              |                                                                               |
|                                     | TIOCA1                   | 2       | Input/Output | TGRA_1 to TGRB_1 input capture input/output compare output/PWM output pins.   |
|                                     | TIOCB1                   | 3       |              |                                                                               |
|                                     | TIOCA2                   | 4       | Input/Output | TGRA_2 to TGRB_2 input capture input/output compare output/PWM output pins.   |
|                                     | TIOCB2                   | 5       |              |                                                                               |
|                                     | TIOCA3                   | 24      | Input/Output | TGRA_3 to TGRD_3 input capture input/output compare output/PWM output pins.   |
|                                     | TIOCB3                   | 26      |              |                                                                               |
|                                     | TIOCC3                   | 28      |              |                                                                               |
|                                     | TIOCD3                   | 29      |              |                                                                               |
|                                     | TIOCA4                   | 30      | Input/Output | TGRA_4 to TGRB_4 input capture input/output compare output/PWM output pins.   |
|                                     | TIOCB4                   | 31      |              |                                                                               |
|                                     | TIOCA5                   | 32      | Input/Output | TGRA_5 to TGRB_5 input capture input/output compare output/PWM output pins.   |
|                                     | TIOCB5                   | 33      |              |                                                                               |
| Program-mable pulse generator (PPG) | PO15                     | 5       | Output       | Pulse output pins.                                                            |
|                                     | PO14                     | 4       |              |                                                                               |
|                                     | PO13                     | 3       |              |                                                                               |
|                                     | PO12                     | 2       |              |                                                                               |
|                                     | PO11                     | 1       |              |                                                                               |
|                                     | PO10                     | 79      |              |                                                                               |
|                                     | PO9                      | 77      |              |                                                                               |
| Motor management timer (MMT)        | PO8                      | 75      |              |                                                                               |
|                                     | PUOA                     | 28      | Output       | U-phase output pin for 6-phase non-overlap PWM waveforms.                     |
|                                     | PUOB                     | 29      | Output       | $\overline{\text{U}}$ -phase output pin for 6-phase non-overlap PWM waveforms |
|                                     | PVOA                     | 30      | Output       | V-phase output pin for 6-phase non-overlap PWM waveforms.                     |

| Type                                                       | Symbol       | Pin NO. | I/O          | Function                                                                                                                       |
|------------------------------------------------------------|--------------|---------|--------------|--------------------------------------------------------------------------------------------------------------------------------|
| Motor management timer (MMT)                               | PVOB         | 31      | Output       | $\bar{V}$ -phase output pin for 6-phase non-overlap PWM waveforms.                                                             |
|                                                            | PWOA         | 32      | Output       | W-phase output pin for 6-phase non-overlap PWM waveforms.                                                                      |
|                                                            | PWOB         | 33      | Output       | $\bar{W}$ -phase output pin for 6-phase non-overlap PWM waveforms.                                                             |
|                                                            | PC $\bar{I}$ | 24      | Input        | Counter clear input pin by external input.                                                                                     |
|                                                            | PCO          | 26      | Output       | Output pin for toggle synchronized with 6-phase non-overlap PWM waveforms.                                                     |
|                                                            | POE3         | 37      | Input        | Input pin for signal requesting to set the MMT waveform output pin to the high-impedance state.                                |
|                                                            | POE2         | 36      |              |                                                                                                                                |
|                                                            | POE1         | 35      |              |                                                                                                                                |
|                                                            | POE0         | 34      |              |                                                                                                                                |
| Serial communication interface (SCI)/ smart card interface | TxD2         | 35      | Output       | Data output pins                                                                                                               |
|                                                            | TxD1         | 19      |              |                                                                                                                                |
|                                                            | TxD0         | 16      |              |                                                                                                                                |
|                                                            | RxD2         | 36      | Input        | Data input pins                                                                                                                |
|                                                            | RxD1         | 20      |              |                                                                                                                                |
|                                                            | RxD0         | 17      |              |                                                                                                                                |
|                                                            | SCK2         | 37      | Input/Output | Clock input/output pins                                                                                                        |
|                                                            | SCK1         | 21      |              |                                                                                                                                |
|                                                            | SCK0         | 18      |              |                                                                                                                                |
| HCAN                                                       | HTxD         | 6       | Output       | CAN bus transmission pin                                                                                                       |
|                                                            | HRxD         | 7       | Input        | CAN bus reception pin                                                                                                          |
| A/D converter                                              | AN11         | 62      | Input        | Analog input pins                                                                                                              |
|                                                            | AN10         | 63      |              |                                                                                                                                |
|                                                            | AN9          | 64      |              |                                                                                                                                |
|                                                            | AN8          | 65      |              |                                                                                                                                |
|                                                            | AN7          | 66      |              |                                                                                                                                |
|                                                            | AN6          | 67      |              |                                                                                                                                |
|                                                            | AN5          | 68      |              |                                                                                                                                |
|                                                            | AN4          | 69      |              |                                                                                                                                |
|                                                            | AN3          | 70      |              |                                                                                                                                |
|                                                            | AN2          | 71      |              |                                                                                                                                |
|                                                            | AN1          | 72      |              |                                                                                                                                |
|                                                            | AN0          | 73      |              |                                                                                                                                |
|                                                            | ADTRG        | 11      | Input        | Pin for input of an external trigger to start A/D conversion                                                                   |
|                                                            | AVCC         | 61      | Input        | Power supply pin for the A/D converter. When the A/D converter is not used, connect this pin to the system power supply (+5V). |
|                                                            | AVSS         | 74      | Input        | The ground pin for the A/D converter. Connect this pin to the system power supply (0V).                                        |

| Type      | Symbol | Pin NO. | I/O              | Function                |
|-----------|--------|---------|------------------|-------------------------|
| I/O ports | P17    | 5       | Input/<br>Output | Eight input/output pins |
|           | P16    | 4       |                  |                         |
|           | P15    | 3       |                  |                         |
|           | P14    | 2       |                  |                         |
|           | P13    | 1       |                  |                         |
|           | P12    | 79      |                  |                         |
|           | P11    | 77      |                  |                         |
|           | P10    | 75      |                  |                         |
|           | P47    | 66      | Input            | Eight input pins        |
|           | P46    | 67      |                  |                         |
|           | P45    | 68      |                  |                         |
|           | P44    | 69      |                  |                         |
|           | P43    | 70      |                  |                         |
|           | P42    | 71      |                  |                         |
|           | P41    | 72      |                  |                         |
|           | P40    | 73      |                  |                         |
|           | P93    | 62      | Input            | Four input pins         |
|           | P92    | 63      |                  |                         |
|           | P91    | 64      |                  |                         |
|           | P90    | 65      |                  |                         |
|           | PA3    | 37      | Input/<br>Output | Four input/output pins  |
|           | PA2    | 36      |                  |                         |
|           | PA1    | 35      |                  |                         |
|           | PA0    | 34      |                  |                         |
|           | PB7    | 33      | Input/<br>Output | Eight input/output pins |
|           | PB6    | 32      |                  |                         |
|           | PB5    | 31      |                  |                         |
|           | PB4    | 30      |                  |                         |
|           | PB3    | 29      |                  |                         |
|           | PB2    | 28      |                  |                         |
|           | PB1    | 26      |                  |                         |
|           | PB0    | 24      |                  |                         |
|           | PC7    | 23      | Input/<br>Output | Eight input/output pins |
|           | PC6    | 22      |                  |                         |
|           | PC5    | 21      |                  |                         |
|           | PC4    | 20      |                  |                         |
|           | PC3    | 19      |                  |                         |
|           | PC2    | 18      |                  |                         |
|           | PC1    | 17      |                  |                         |
|           | PC0    | 16      |                  |                         |

| Type      | Symbol | Pin NO. | I/O              | Function                |
|-----------|--------|---------|------------------|-------------------------|
| I/O ports | PD7    | 53      | Input/<br>Output | Eight input/output pins |
|           | PD6    | 54      |                  |                         |
|           | PD5    | 55      |                  |                         |
|           | PD4    | 56      |                  |                         |
|           | PD3    | 57      |                  |                         |
|           | PD2    | 58      |                  |                         |
|           | PD1    | 59      |                  |                         |
|           | PD0    | 60      |                  |                         |
|           | PF7    | 15      | Input/<br>Output | Eight input/output pins |
|           | PF6    | 14      |                  |                         |
|           | PF5    | 13      |                  |                         |
|           | PF4    | 12      |                  |                         |
|           | PF3    | 11      |                  |                         |
|           | PF2    | 10      |                  |                         |
|           | PF1    | 9       |                  |                         |
|           | PF0    | 8       |                  |                         |

# Section 2 CPU

The H8S/2600 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/2600 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/2600 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 CPUs
  - Can execute H8/300 and H8/300H CPUs object programs
- General-register architecture
  - Sixteen 16-bit general registers also usable as sixteen 8-bit registers or eight 32-bit registers
- Sixty-nine basic instructions
  - 8/16/32-bit arithmetic and logic instructions
  - Multiply and divide instructions
  - Powerful bit-manipulation instructions
  - Multiply-and-accumulate instruction
- 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
  - $8 \times 8$ -bit register-register multiply : 3 states
  - $16 \div 8$ -bit register-register divide : 12 states
  - $16 \times 16$ -bit register-register multiply : 4 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 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/2600 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.
  - A multiply-and-accumulate instruction has 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/2600 CPU has the following enhancements:

- Additional control register
  - One 8-bit and two 32-bit control registers have been added.
- Enhanced instructions
  - Addressing modes of bit-manipulation instructions have been enhanced.
  - A multiply-and-accumulate instruction has 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.

## 2.2 CPU Operating Modes

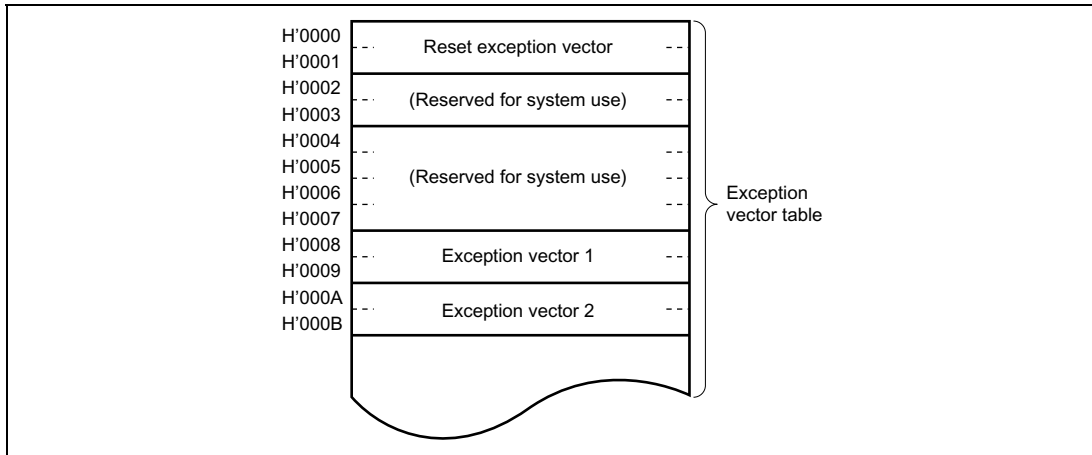
The H8S/2600 CPU has two operating modes: normal and advanced. Normal mode supports a maximum 64-kbyte address space. Advanced mode supports a maximum 16-Mbyte total address space. The mode is selected by the mode pins.

### 2.2.1 Normal Mode

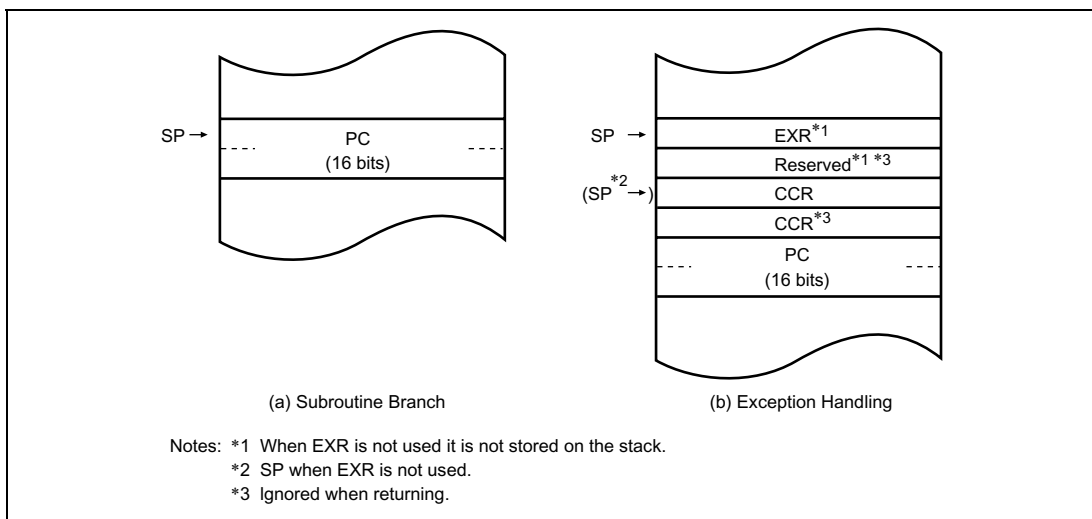
The exception vector table and stack have the same structure as in the H8/300 CPU.

- **Address Space**  
A maximum address space of 64 kbytes can be accessed.
- **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. The exception vector table differs depending on the microcontroller. 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**  
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 16-Mbyte maximum address space is provided.

- Extended Registers (En)

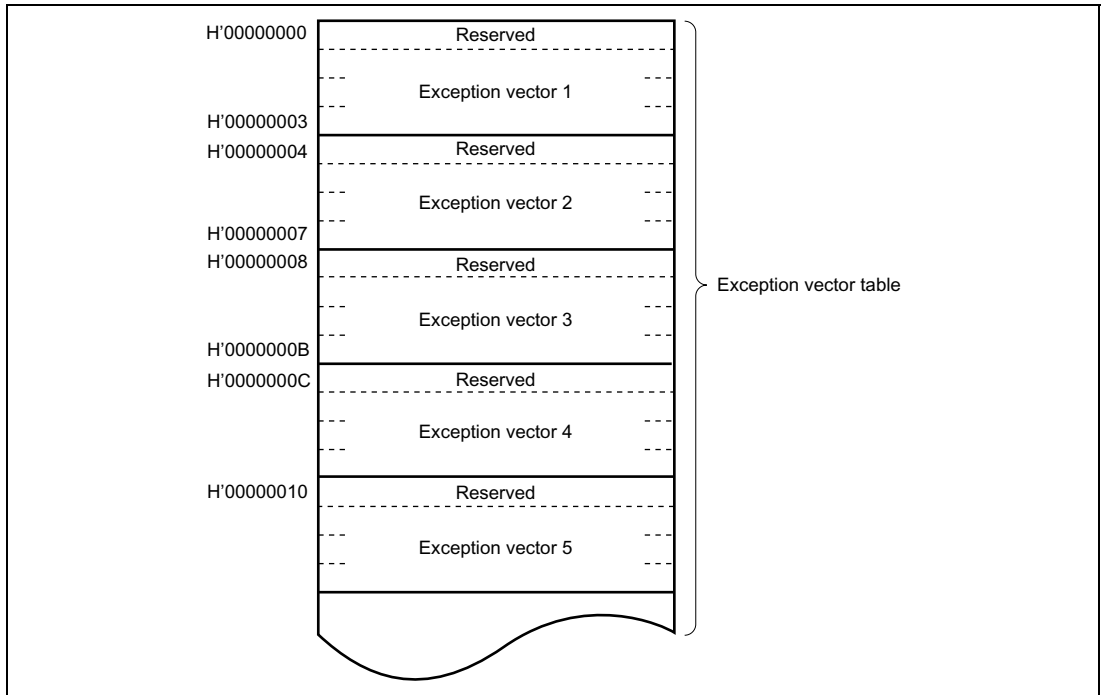
The extended registers (E0 to E7) can be used as 16-bit registers, or as the upper 16-bit segments of 32-bit registers or address registers.

- 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.

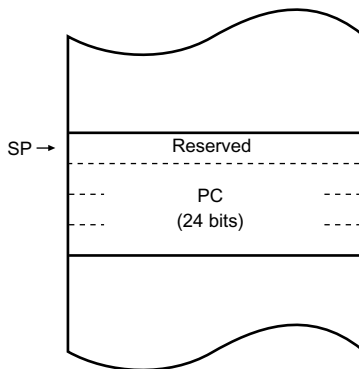


**Figure 2-3 Exception Vector Table (Advanced Mode)**

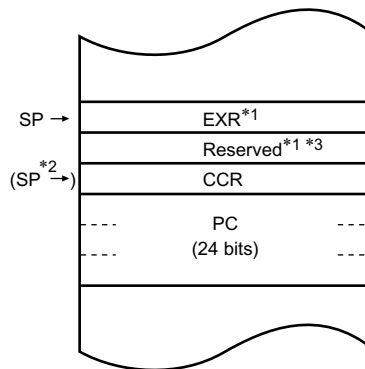
The memory indirect addressing mode (@@aa:8) employed in the JMP and JSR instructions uses an 8-bit absolute address included in the instruction code to specify a memory operand that contains a branch address. In advanced mode the operand is a 32-bit longword operand, providing a 32-bit branch address. 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.



(a) Subroutine Branch



(b) Exception Handling

Notes: \*1 When EXR is not used it is not stored on the stack.

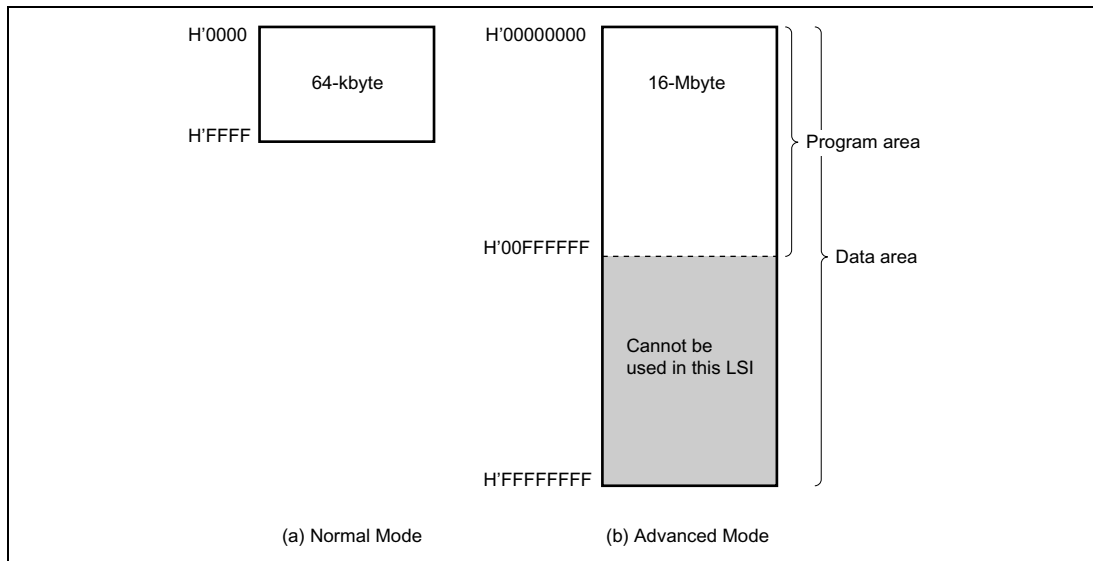
\*2 SP when EXR is not used.

\*3 Ignored when returning.

**Figure 2-4 Stack Structure in Advanced Mode**

## 2.3 Address Space

Figure 2-5 shows a memory map for the H8S/2600 CPU. The H8S/2600 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**

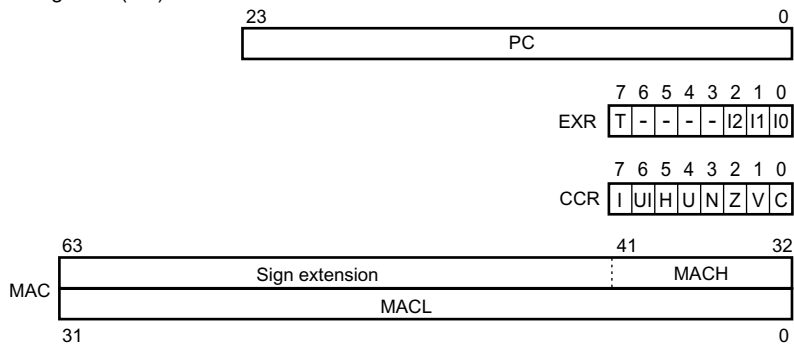
## 2.4 Register Configuration

The H8S/2600 CPU has the internal registers shown in figure 2-6. There are two types of registers; general registers and control registers. The control registers are a 24-bit program counter (PC), an 8-bit extended control register (EXR), an 8-bit condition code register (CCR), and a 64-bit multiply-accumulate register (MAC).

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                  | H   | :Half-carry flag              |
| PC       | :Program counter                | U   | :User bit                     |
| EXR      | :Extended control register      | N   | :Negative flag                |
| T        | :Trace bit                      | Z   | :Zero flag                    |
| I2 to I0 | :Interrupt mask bits            | V   | :Overflow flag                |
| CCR      | :Condition-code register        | C   | :Carry flag                   |
| I        | :Interrupt mask bit             | MAC | :Multiply-accumulate register |
| UI       | :User bit or interrupt mask bit |     |                               |

Figure 2-6 CPU Registers

## 2.4.1 General Registers

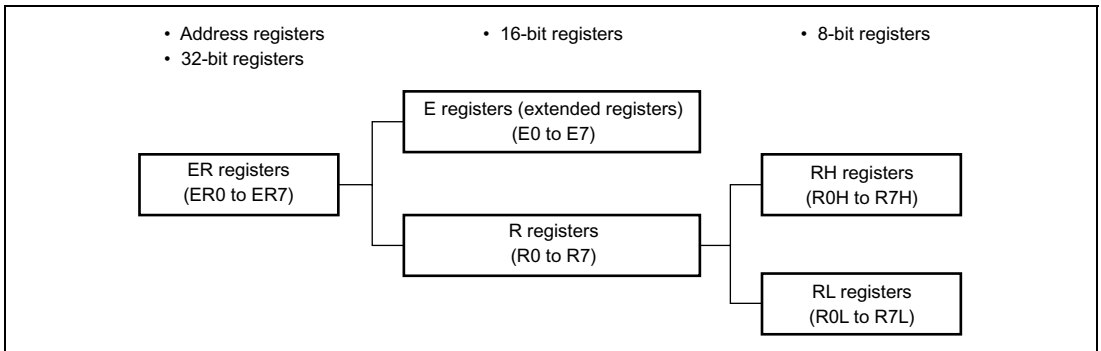
The H8S/2600 CPU has eight 32-bit general registers. These general registers are all functionally identical and can be used as both address registers and data registers. When a general register is used as a data register, it can be accessed as a 32-bit, 16-bit, or 8-bit register. Figure 2-7 illustrates 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 of sixteen 16-bit registers. The E registers (E0 to E7) are also referred to as extended registers.

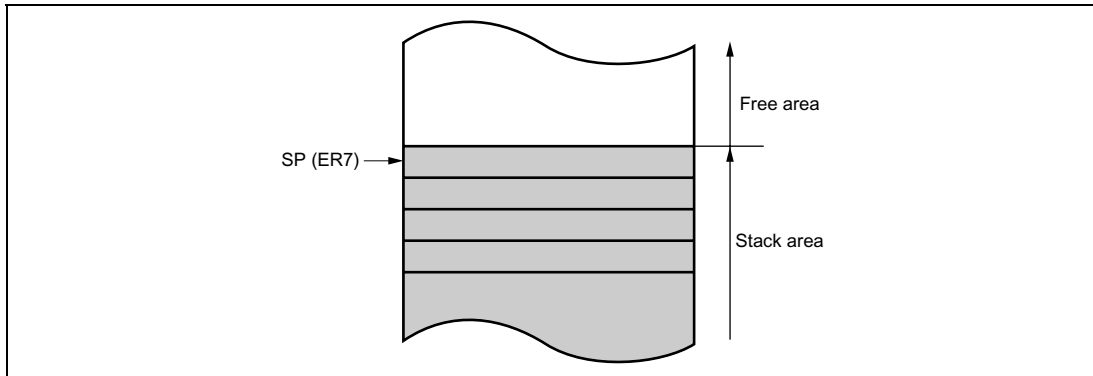
The R registers divide into 8-bit general registers designated by the letters RH (R0H to R7H) and RL (R0L to R7L). These registers are functionally equivalent, providing a maximum of sixteen 8-bit registers.

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**

### 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, are 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<br><br>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   | —        | 1             | —   | Reserved                                                                                                                                                                               |
| 5   |          |               |     | They are always read as 1.                                                                                                                                                             |
| 4   |          |               |     |                                                                                                                                                                                        |
| 3   |          |               |     |                                                                                                                                                                                        |
| 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 | <p>Interrupt Mask Bit</p> <p>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.</p>                                                                                                                                                                                                                                                       |
| 6   | UI       | undefined     | R/W | <p>User Bit or Interrupt Mask Bit</p> <p>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.</p>                                                                                                                                                                                                                                                                                                                            |
| 5   | H        | undefined     | R/W | <p>Half-Carry Flag</p> <p>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.</p> |
| 4   | U        | undefined     | R/W | <p>User Bit</p> <p>Can be written and read by software using the LDC, STC, ANDC, ORC, and XORC instructions.</p>                                                                                                                                                                                                                                                                                                                                                                                                                |
| 3   | N        | undefined     | R/W | <p>Negative Flag</p> <p>Stores the value of the most significant bit of data as a sign bit.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 2   | Z        | undefined     | R/W | <p>Zero Flag</p> <p>Set to 1 to indicate zero data, and cleared to 0 to indicate non-zero data.</p>                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 1   | V        | undefined     | R/W | <p>Overflow Flag</p> <p>Set to 1 when an arithmetic overflow occurs, and cleared to 0 at other times.</p>                                                                                                                                                                                                                                                                                                                                                                                                                       |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                          |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0   | C        | undefined     | R/W | <p>Carry Flag</p> <p>Set to 1 when a carry occurs, and cleared to 0 otherwise. Used by:</p> <ul style="list-style-type: none"> <li>• Add instructions, to indicate a carry</li> <li>• Subtract instructions, to indicate a borrow</li> <li>• Shift and rotate instructions, to indicate a carry</li> </ul> <p>The carry flag is also used as a bit accumulator by bit manipulation instructions.</p> |

### 2.4.5 Multiply-Accumulate Register (MAC)

This 64-bit register stores the results of multiply-and-accumulate operations. It consists of two 32-bit registers denoted MACH and MACL. The lower 10 bits of MACH are valid; the upper bits are a sign extension.

### 2.4.6 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

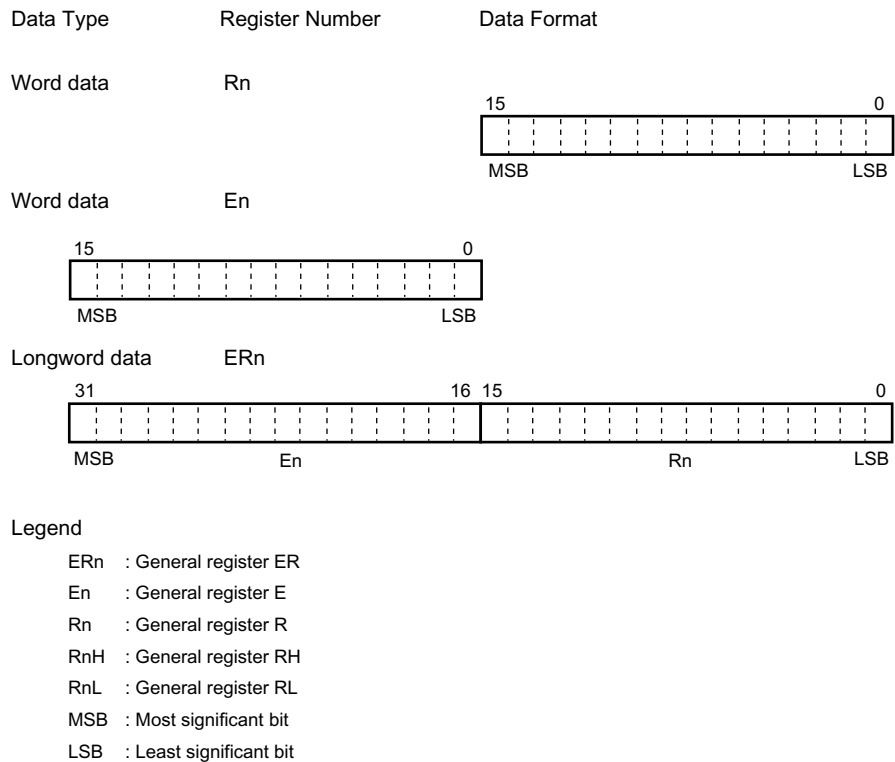
The H8S/2600 CPU can process 1-bit, 4-bit (BCD), 8-bit (byte), 16-bit (word), and 32-bit (longword) data. Bit-manipulation instructions operate on 1-bit data by accessing bit n (n = 0, 1, 2, ..., 7) of byte operand data. The DAA and DAS decimal-adjust instructions treat byte data as two digits of 4-bit BCD data.

## 2.5.1 General Register Data Formats

Figure 2-9 shows the data formats in general registers.

| Data Type      | Register Number | Data Format                                                                     |
|----------------|-----------------|---------------------------------------------------------------------------------|
| 1-bit data     | RnH             | <div><div>70</div><div><div>76543210</div><div>Don't care</div></div></div>     |
| 1-bit data     | RnL             | <div><div>Don't care</div><div><div>76543210</div></div></div>                  |
| 4-bit BCD data | RnH             | <div><div>7430</div><div><div>UpperLower</div><div>Don't care</div></div></div> |
| 4-bit BCD data | RnL             | <div><div>Don't care</div><div><div>7430</div><div>UpperLower</div></div></div> |
| Byte data      | RnH             | <div><div>70</div><div><div>MSBLSB</div><div>Don't care</div></div></div>       |
| Byte data      | RnL             | <div><div>Don't care</div><div><div>70</div><div>MSBLSB</div></div></div>       |

Figure 2-9 General Register Data Formats (1)

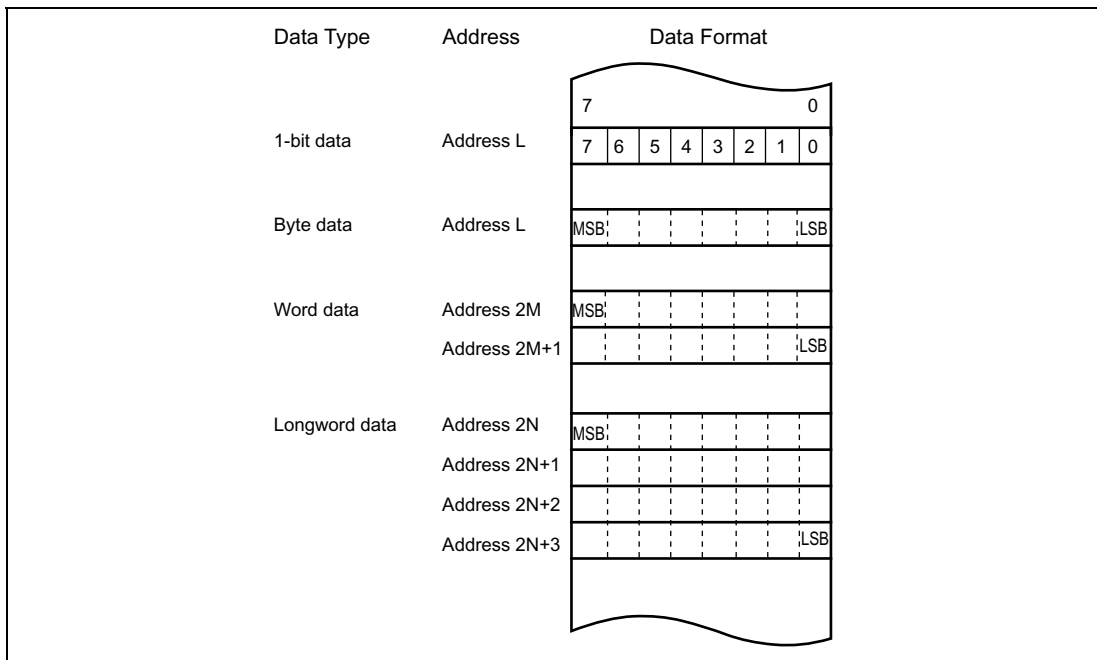


**Figure 2-9 General Register Data Formats (2)**

### 2.5.2 Memory Data Formats

Figure 2-10 shows the data formats in memory. The H8S/2600 CPU can access word data and longword data in memory, however 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, an address error does not occur, however the least significant bit of the address is regarded as 0, so access begins 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/2600 CPU has 69 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* <sup>1</sup> , PUSH* <sup>1</sup>                                            | W/L   |       |
|                       | LDM, STM                                                                          | L     |       |
|                       | MOVFP* <sup>3</sup> , MOVTP* <sup>3</sup>                                         | B     |       |
| Arithmetic operations | ADD, SUB, CMP, NEG                                                                | B/W/L | 23    |
|                       | ADDX, SUBX, DAA, DAS                                                              | B     |       |
|                       | INC, DEC                                                                          | B/W/L |       |
|                       | ADDS, SUBS                                                                        | L     |       |
|                       | MULXU, DIVXU, MULXS, DIVXS                                                        | B/W   |       |
|                       | EXTU, EXTS                                                                        | W/L   |       |
|                       | TAS* <sup>4</sup>                                                                 | B     |       |
|                       | MAC, LDMAC, STMAC, CLRMAC                                                         | —     |       |
| 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* <sup>2</sup> , JMP, BSR, JSR, RTS                                            | —     | 5     |
| System control        | TRAPA, RTE, SLEEP, LDC, STC, ANDC, ORC, XORC, NOP                                 | —     | 9     |
| Block data transfer   | EPMOV                                                                             | —     | 1     |

Total: 69

Notes: B-byte; W-word; L-longword.

\*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.

## 2.6.1 Table of Instructions Classified by Function

Tables 2-3 to 2-10 summarizes 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)             |
| MAC            | Multiply-accumulate 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**

| <b>Instruction</b> | <b>Size*</b> | <b>Function</b>                                                                                                                                             |
|--------------------|--------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MOV                | B/W/L        | (EAs) → Rd, Rs → (EAd)<br>Moves data between two general registers or between a general register and memory, or moves immediate data to a general register. |
| MOVFPE             | B            | Cannot be used in this LSI.                                                                                                                                 |
| MOVTPE             | B            | Cannot be used in this LSI.                                                                                                                                 |
| POP                | W/L          | @SP+ → Rn<br>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<br>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                | L            | @SP+ → Rn (register list)<br>Pops two or more general registers from the stack.                                                                             |
| STM                | L            | Rn (register list) → @-SP<br>Pushes two or more general registers onto the stack.                                                                           |

Note: \* Refers to the operand size.

B: Byte

W: Word

L: Longword

**Table 2-4 Arithmetic Operations Instructions**

| <b>Instruction</b> | <b>Size*</b> | <b>Function</b>                                                                                                                                                                                                                                                                                        |
|--------------------|--------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ADD<br>SUB         | B/W/L        | $Rd \pm Rs \rightarrow Rd$ , $Rd \pm \#IMM \rightarrow Rd$<br>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<br>SUBX       | B            | $Rd \pm Rs \pm C \rightarrow Rd$ , $Rd \pm \#IMM \pm C \rightarrow Rd$<br>Performs addition or subtraction with carry on byte data in two general registers, or on immediate data and data in a general register.                                                                                      |
| INC<br>DEC         | B/W/L        | $Rd \pm 1 \rightarrow Rd$ , $Rd \pm 2 \rightarrow Rd$<br>Increments or decrements a general register by 1 or 2. (Byte operands can be incremented or decremented by 1 only.)                                                                                                                           |
| ADDS<br>SUBS       | L            | $Rd \pm 1 \rightarrow Rd$ , $Rd \pm 2 \rightarrow Rd$ , $Rd \pm 4 \rightarrow Rd$<br>Adds or subtracts the value 1, 2, or 4 to or from data in a 32-bit register.                                                                                                                                      |
| DAA<br>DAS         | B            | $Rd \text{ decimal adjust} \rightarrow Rd$<br>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          | $Rd \times Rs \rightarrow Rd$<br>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          | $Rd \times Rs \rightarrow Rd$<br>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          | $Rd \div Rs \rightarrow Rd$<br>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 <sup>*1</sup> | Function                                                                                                                                                                                                                                                                                                                                                |
|-------------------|--------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DIVXS             | B/W                | $Rd \div Rs \rightarrow Rd$<br>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              | $Rd - Rs, Rd - \#IMM$<br>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              | $0 - Rd \rightarrow Rd$<br>Takes the two's complement (arithmetic complement) of data in a general register.                                                                                                                                                                                                                                            |
| EXTU              | W/L                | $Rd$ (zero extension) $\rightarrow Rd$<br>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                | $Rd$ (sign extension) $\rightarrow Rd$<br>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 <sup>*2</sup> | B                  | $@ERd - 0, 1 \rightarrow (<bit\ 7> \text{ of } @ERd)$<br>Tests memory contents, and sets the most significant bit (bit 7) to 1.                                                                                                                                                                                                                         |
| MAC               | —                  | $(EAs) \times (EAd) + MAC \rightarrow MAC$<br>Performs signed multiplication on memory contents and adds the result to the multiply-accumulate register. The following operations can be performed:<br>16 bits $\times$ 16 bits + 32 bits $\rightarrow$ 32 bits, saturating<br>16 bits $\times$ 16 bits + 42 bits $\rightarrow$ 42 bits, non-saturating |
| CLRMAC            | —                  | $0 \rightarrow MAC$<br>Clears the multiply-accumulate register to zero.                                                                                                                                                                                                                                                                                 |
| LDMAC<br>STMAC    | L                  | $Rs \rightarrow MAC, MAC \rightarrow Rd$<br>Transfers data between a general register and a multiply-accumulate register.                                                                                                                                                                                                                               |

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 | $Rd \wedge Rs \rightarrow Rd$ , $Rd \wedge \#IMM \rightarrow Rd$<br>Performs a logical AND operation on a general register and another general register or immediate data.          |
| OR          | B/W/L | $Rd \vee Rs \rightarrow Rd$ , $Rd \vee \#IMM \rightarrow Rd$<br>Performs a logical OR operation on a general register and another general register or immediate data.               |
| XOR         | B/W/L | $Rd \oplus Rs \rightarrow Rd$ , $Rd \oplus \#IMM \rightarrow Rd$<br>Performs a logical exclusive OR operation on a general register and another general register or immediate data. |
| NOT         | B/W/L | $\neg Rd \rightarrow Rd$<br>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<br>SHAR   | B/W/L | $Rd \text{ (shift)} \rightarrow Rd$<br>Performs an arithmetic shift on general register contents.<br>1-bit or 2-bit shifts are possible.    |
| SHLL<br>SHLR   | B/W/L | $Rd \text{ (shift)} \rightarrow Rd$<br>Performs a logical shift on general register contents.<br>1-bit or 2-bit shifts are possible.        |
| ROTL<br>ROTR   | B/W/L | $Rd \text{ (rotate)} \rightarrow Rd$<br>Rotates general register contents.<br>1-bit or 2-bit rotations are possible.                        |
| ROTXL<br>ROTXR | B/W/L | $Rd \text{ (rotate)} \rightarrow Rd$<br>Rotates general register contents through the carry flag.<br>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**

| <b>Instruction</b> | <b>Size*</b> | <b>Function</b>                                                                                                                                                                                                                                                 |
|--------------------|--------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BSET               | B            | $1 \rightarrow (\text{<bit-No.> of <EAd>})$<br>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            | $0 \rightarrow (\text{<bit-No.> of <EAd>})$<br>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            | $\neg (\text{<bit-No.> of <EAd>}) \rightarrow (\text{<bit-No.> of <EAd>})$<br>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            | $\neg (\text{<bit-No.> of <EAd>}) \rightarrow Z$<br>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            | $C \wedge (\text{<bit-No.> of <EAd>}) \rightarrow C$<br>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            | $C \wedge \neg (\text{<bit-No.> of <EAd>}) \rightarrow C$<br>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            | $C \vee (\text{<bit-No.> of <EAd>}) \rightarrow C$<br>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            | $C \vee \neg (\text{<bit-No.> of <EAd>}) \rightarrow C$<br>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     | $C \oplus (<\text{bit-No.}> \text{ of } <\text{EAd}>) \rightarrow C$<br>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     | $C \oplus \neg (<\text{bit-No.}> \text{ of } <\text{EAd}>) \rightarrow C$<br>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.<br>The bit number is specified by 3-bit immediate data. |
| BLD         | B     | $(<\text{bit-No.}> \text{ of } <\text{EAd}>) \rightarrow C$<br>Transfers a specified bit in a general register or memory operand to the carry flag.                                                                                                                           |
| BILD        | B     | $\neg (<\text{bit-No.}> \text{ of } <\text{EAd}>) \rightarrow C$<br>Transfers the inverse of a specified bit in a general register or memory operand to the carry flag.<br>The bit number is specified by 3-bit immediate data.                                               |
| BST         | B     | $C \rightarrow (<\text{bit-No.}> \text{ of } <\text{EAd}>)$<br>Transfers the carry flag value to a specified bit in a general register or memory operand.                                                                                                                     |
| BIST        | B     | $\neg C \rightarrow (<\text{bit-No.}> \text{ of } <\text{EAd}>)$<br>Transfers the inverse of the carry flag value to a specified bit in a general register or memory operand.<br>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.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |          |             |           |         |               |        |         |               |       |     |      |                |     |             |                |          |                               |         |          |                 |         |     |           |         |     |       |         |     |                |         |     |              |         |     |      |         |     |       |         |     |                  |                  |     |           |                  |     |              |                           |     |               |                           |
|             |                               | <table> <tr> <th>Mnemonic</th><th>Description</th><th>Condition</th></tr> <tr> <td>BRA(BT)</td><td>Always (true)</td><td>Always</td></tr> <tr> <td>BRN(BF)</td><td>Never (false)</td><td>Never</td></tr> <tr> <td>BHI</td><td>High</td><td><math>C \vee Z = 0</math></td></tr> <tr> <td>BLS</td><td>Low or same</td><td><math>C \vee Z = 1</math></td></tr> <tr> <td>BCC(BHS)</td><td>Carry clear<br/>(high or same)</td><td><math>C = 0</math></td></tr> <tr> <td>BCS(BLO)</td><td>Carry set (low)</td><td><math>C = 1</math></td></tr> <tr> <td>BNE</td><td>Not equal</td><td><math>Z = 0</math></td></tr> <tr> <td>BEQ</td><td>Equal</td><td><math>Z = 1</math></td></tr> <tr> <td>BVC</td><td>Overflow clear</td><td><math>V = 0</math></td></tr> <tr> <td>BVS</td><td>Overflow set</td><td><math>V = 1</math></td></tr> <tr> <td>BPL</td><td>Plus</td><td><math>N = 0</math></td></tr> <tr> <td>BMI</td><td>Minus</td><td><math>N = 1</math></td></tr> <tr> <td>BGE</td><td>Greater or equal</td><td><math>N \oplus V = 0</math></td></tr> <tr> <td>BLT</td><td>Less than</td><td><math>N \oplus V = 1</math></td></tr> <tr> <td>BGT</td><td>Greater than</td><td><math>Z \vee (N \oplus V) = 0</math></td></tr> <tr> <td>BLE</td><td>Less or equal</td><td><math>Z \vee (N \oplus V) = 1</math></td></tr> </table> | 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<br>(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 | $Z \vee (N \oplus V) = 1$ |
| 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<br>(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<br>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)<br>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<br>Logically ANDs the CCR or EXR contents with immediate data.                                                                                                                               |
| ORC         | B     | CCR ∨ #IMM → CCR, EXR ∨ #IMM → EXR<br>Logically ORs the CCR or EXR contents with immediate data.                                                                                                                                |
| XORC        | B     | CCR ⊕ #IMM → CCR, EXR ⊕ #IMM → EXR<br>Logically XORs the CCR or EXR contents with immediate data.                                                                                                                               |
| NOP         | —     | PC + 2 → PC<br>Only increments the program counter.                                                                                                                                                                             |

Note: \* Refers to the operand size.

B: Byte

W: Word

L: Longword

**Table 2-10 Block Data Transfer Instructions**

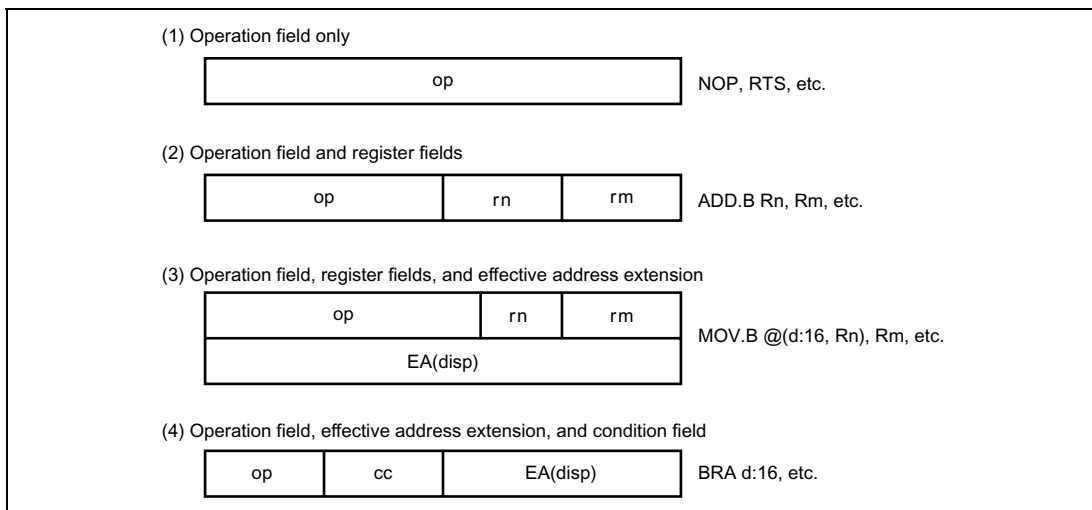
| Instruction | Size | Function                                                                                                                                                                                                                                                                                                                                                              |
|-------------|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| EEPMOV.B    | —    | if R4L $\neq$ 0 then<br>Repeat @ER5+ $\rightarrow$ @ER6+<br>R4L-1 $\rightarrow$ R4L<br>Until R4L = 0<br>else next;                                                                                                                                                                                                                                                    |
| EEPMOV.W    | —    | if R4 $\neq$ 0 then<br>Repeat @ER5+ $\rightarrow$ @ER6+<br>R4-1 $\rightarrow$ R4<br>Until R4 = 0<br>else next;<br><br>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.<br><br>Execution of the next instruction begins as soon as the transfer is completed. |

## 2.6.2 Basic Instruction Formats

This LSI instructions consist of 2-byte (1-word) units. An instruction consists of an operation field (op field), a register field (r field), an effective address extension (EA field), 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/2600 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<br>Register indirect with pre-decrement | @ERn+<br>@-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 or Pre-Decrement—@ERn+ or @-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,<br>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.

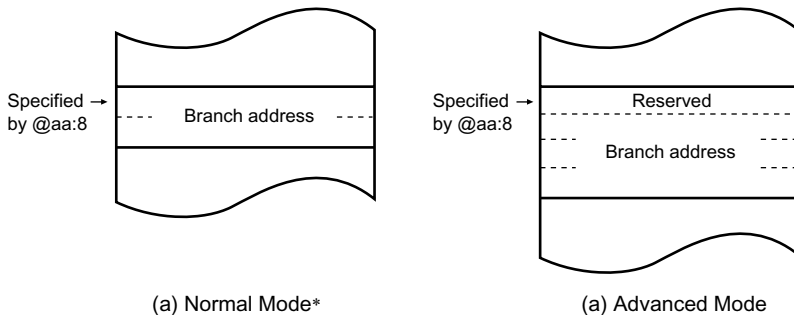
## **2.7.8 Memory Indirect—@@aa:8**

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.



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

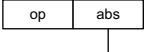
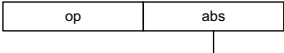
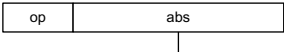
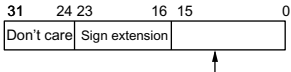
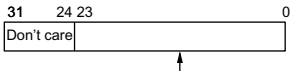
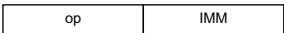
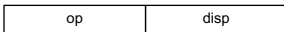
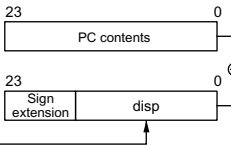
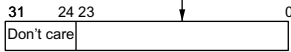
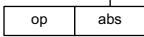
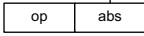
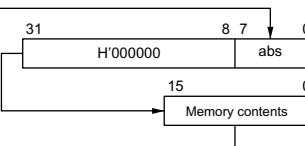
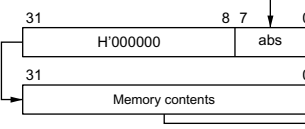
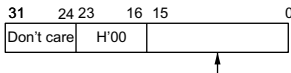
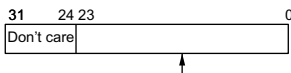
Table 2-13 indicates how effective addresses are calculated in each addressing mode. In normal mode the upper 8 bits of the effective address are ignored in order to generate a 16-bit address.

Note : Normal mode is not available in this LSI.

Table 2-13 Effective Address Calculation

| No | Addressing Mode and Instruction Format                                                                                                                                                                                                                        | Effective Address Calculation                                                                                                                                                                        | Effective Address (EA)                                                                                                                                                                        |
|----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | Register direct(Rn)<br><div><div>op</div><div>rm</div><div>rn</div></div>                                                                                                                                                                                     |                                                                                                                                                                                                      | Operand is general register contents.                                                                                                                                                         |
| 2  | Register indirect(@ERn)<br><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<br>@(d:16,ERn) or @(d:32,ERn)<br><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<br>• Register indirect with post-increment @ERn+<br><div><div>op</div><div>r</div><div></div></div><br>• Register indirect with pre-decrement @-ERn<br><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  | <p>Absolute address</p> <p>@aa:8</p>  <p>@aa:16</p>  <p>@aa:24</p>  <p>@aa:32</p>  |                                                                                                                                                                        |     |
| 6  | <p>Immediate</p> <p>#xx:8/#xx:16/#xx:32</p>                                                                                                                                                                                                                                                                                           |                                                                                                                                                                        | Operand is immediate data.                                                                                                                                                                                                                                                                                                                 |
| 7  | <p>Program-counter relative</p> <p>@(d:8,PC)/@(d:16,PC)</p>                                                                                                                                                                                                                                                                           |                                                                                       |                                                                                                                                                                                                                                                          |
| 8  | <p>Memory indirect @aa:8</p> <p>• Normal mode*</p>  <p>• Advanced mode</p>                                                                                                                                                                         |   |                                                                                                                                                                     |

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

## 2.8 Processing States

The H8S/2600 CPU has five main processing states: the reset state, exception handling state, program execution state, bus-released state, and power-down state. Figure 2-14 shows a diagram of the processing states. 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.

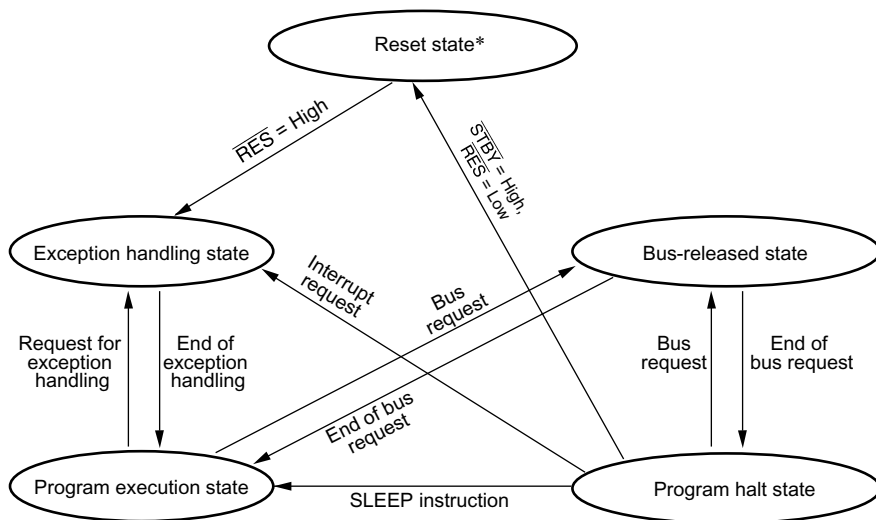
- Bus-Released State

In a product which has a bus master other than the CPU, such as a 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.

- Program stop 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 20, Power-Down Modes.



Notes: From any state, a transition to hardware standby mode occurs when  $\overline{\text{STBY}}$  goes low.

\* From any state except hardware standby mode, a transition to the reset state occurs whenever  $\overline{\text{RES}}$  goes low. A transition can also be made to the reset state when the watchdog timer overflows.

**Figure 2-13 State Transitions**

## 2.9 Usage Notes

### 2.9.1 Usage Notes on Bit Manipulation Instructions

The BSET, BCLR, BNOT, BST, and BIST instructions are used to read data in bytes, then, after bit manipulation, they write data in bytes again. Therefore, special care is necessary to use these instructions for the registers and the ports that include write-only bit.

The BCLR instruction can be used to clear the flags in the internal I/O registers to 0. In this time, if it is obvious that the flag has been set to 1 in the interrupt processing routine or other processing, there is no need to read the flag beforehand.



# Section 3 MCU Operating Modes

## 3.1 Operating Mode Selection

This LSI supports only operating mode 7, that is, the advanced single-chip mode. The operating mode is determined by the setting of the mode pins (MD2 to MD0). Only mode 7 can be used in this LSI. Therefore, all mode pins must be fixed high, as shown in table 3-1. Do not change the mode pin settings during operation.

**Table 3-1 MCU Operating Mode Selection**

| MCU<br>Operating<br>Mode | MD2 | MD1 | MD0 | CPU<br>Operating<br>Mode | Description      | On-Chip<br>ROM | External Data Bus |              |
|--------------------------|-----|-----|-----|--------------------------|------------------|----------------|-------------------|--------------|
|                          |     |     |     |                          |                  |                | Initial<br>Width  | Max<br>Width |
| 7                        | 1   | 1   | 1   | Advanced<br>mode         | Single-chip mode | Enabled        | —                 | —            |

## 3.2 Register Descriptions

The following registers are related to the operating mode. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

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

### 3.2.1 Mode Control Register(MDCR)

| Bit | Bit Name | Initial Value | R/W | Descriptions                                                                                                                                                                                                                                                                                                                                                         |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | 1             | R/W | Reserved<br>Only 1 should be written to this bit.                                                                                                                                                                                                                                                                                                                    |
| 6   | —        | 0             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                             |
| 5   | —        | 0             | —   | These bits are always read as 0 and cannot be modified.                                                                                                                                                                                                                                                                                                              |
| 4   | —        | 0             | —   |                                                                                                                                                                                                                                                                                                                                                                      |
| 3   | —        | 0             | —   |                                                                                                                                                                                                                                                                                                                                                                      |
| 2   | MDS2     | —             | 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. These latches are canceled by a reset. These latches are canceled by a reset. |
| 1   | MDS1     | —             | R   |                                                                                                                                                                                                                                                                                                                                                                      |
| 0   | MDS0     | —             | R   |                                                                                                                                                                                                                                                                                                                                                                      |

### 3.2.2 System Control Register(SYSCR)

SYSCR is an 8-bit readable/writable register that selects saturating or non-saturating calculation for the MAC instruction, selects the interrupt control mode and the detected edge for NMI, and enables or disables on-chip RAM.

| Bit | Bit Name | Initial Value | R/W | Descriptions                                                                                                                                                                                                                                                                                      |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | MACS     | 0             | —   | MAC Saturation<br>Selects either saturating or non-saturating calculation for the MAC instruction.<br>0: Non-saturating calculation for the MAC instruction<br>1: Saturating calculation for the MAC instruction                                                                                  |
| 6   | —        | 0             | —   | Reserved<br>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 5.6, Interrupt Control Modes and Interrupt Operation.<br>00: Interrupt control mode 0<br>01: Setting prohibited<br>10: Interrupt control mode 2<br>11: Setting prohibited |
| 4   | INTM0    | 0             | R/W |                                                                                                                                                                                                                                                                                                   |
| 3   | NMIEG    | 0             | R/W | NMI Edge Select<br>Selects the valid edge of the NMI interrupt input.<br>0: An interrupt is requested at the falling edge of NMI input<br>1: An interrupt is requested at the rising edge of NMI input                                                                                            |
| 2   | —        | 0             | —   | Reserved                                                                                                                                                                                                                                                                                          |
| 1   | —        | 0             | —   | These bits are always read as 0 and cannot be modified.                                                                                                                                                                                                                                           |
| 0   | RAME     | 1             | R/W | RAM Enable<br>Enables or disables on-chip RAM. The RAME bit is initialized when the reset status is released.<br>0: On-chip RAM is disabled<br>1: On-chip RAM is enabled                                                                                                                          |

### 3.3 Pin Functions in Each Operating Mode

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

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

#### 3.3.1 Pin Functions

Table 3-2 shows their functions in mode 7.

**Table 3-2 Pin Functions in Each Mode**

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

**Legend**

- P: I/O port
- A: Address bus output
- D: Data bus I/O
- C: Control signals, clock I/O
- \*: After reset

3.4 Address Map

Figure 3-1 shows the address map in each operating mode.

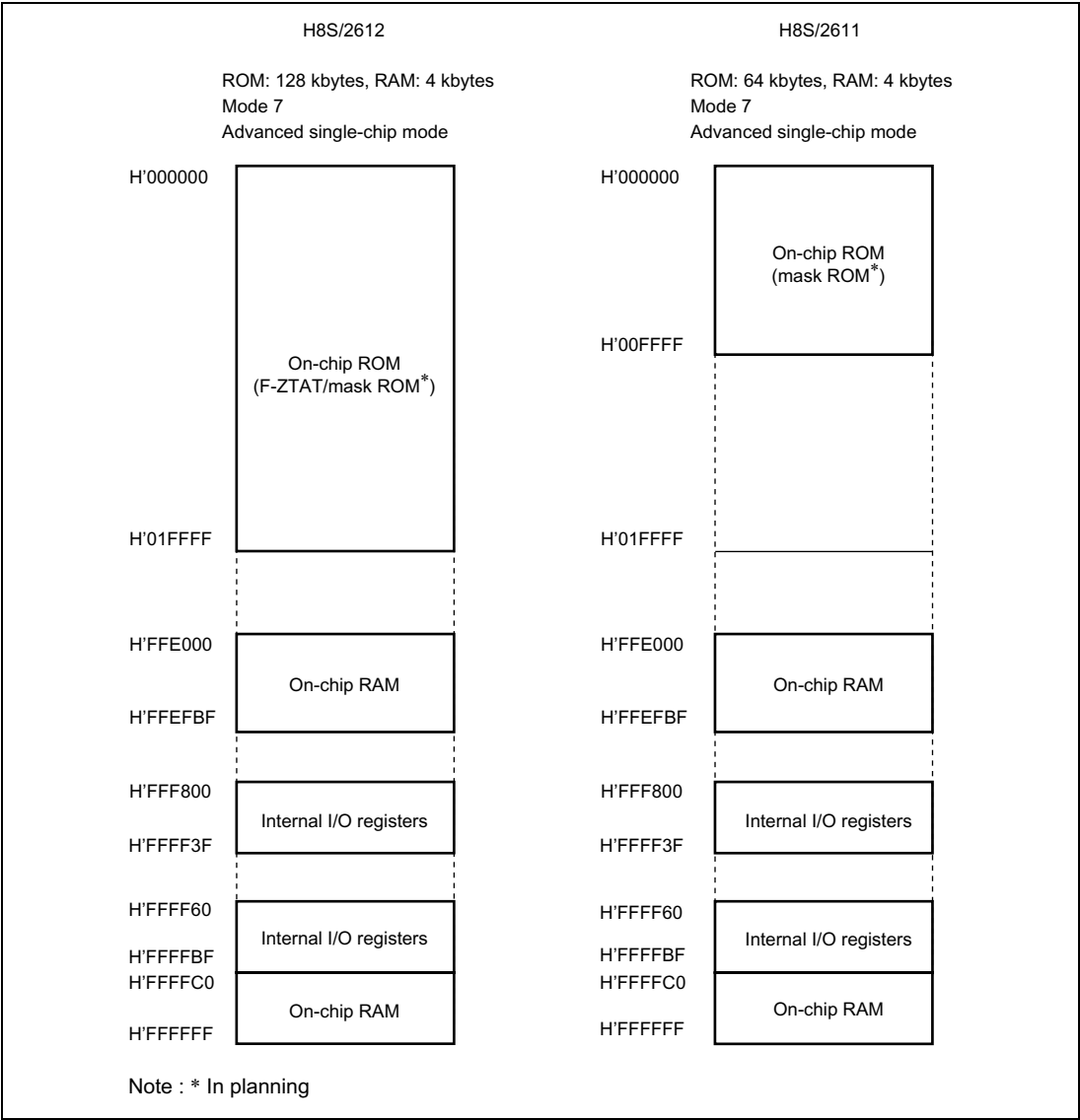


Figure 3-1 Address Map

# Section 4 Exception Handling

## 4.1 Exception Handling Types and Priority

As table 4-1 indicates, exception handling may be caused by a reset, 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. Exception sources, the stack structure, and operation of the CPU vary depending on the interrupt control mode. For details on the interrupt control mode, refer to section 5, Interrupt Controller.

**Table 4-1 Exception Types and Priority**

| Priority  | Exception Type                 | Start of Exception Handling                                                                                                                                                                             |
|-----------|--------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| High<br>↑ | Reset                          | Starts immediately after a low-to-high transition at the $\overline{\text{RES}}$ pin, or when the watchdog timer overflows. The CPU enters the reset state when the $\overline{\text{RES}}$ pin is low. |
|           | Trace <sup>*1</sup>            | Starts when execution of the current instruction or exception handling ends, if the trace (T) bit in the EXR is set to 1.                                                                               |
|           | Direct transition              | Starts when a direction transition occurs as the result of SLEEP instruction execution.                                                                                                                 |
|           | Interrupt                      | Starts when execution of the current instruction or exception handling ends, if an interrupt request has been issued <sup>*2</sup> .                                                                    |
| Low       | Trap instruction <sup>*3</sup> | Started by execution of a trap instruction (TRAPA).                                                                                                                                                     |

Notes: \*1 Traces are enabled only in interrupt control mode 2. Trace exception handling is not executed after execution of an RTE instruction.

\*2 Interrupt detection is not performed on completion of ANDC, ORC, XORC, or LDC instruction execution, or on completion of reset exception handling.

\*3 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. Since the usable modes differ depending on the product, for details on each product, refer to section 3, MCU Operating Modes.

**Table 4-2 Exception Handling Vector Table**

| Exception Source                             |      | Vector Number | Vector Address <sup>*1</sup> |                  |
|----------------------------------------------|------|---------------|------------------------------|------------------|
|                                              |      |               | Normal Mode <sup>*2</sup>    | Advanced Mode    |
| Power-on reset                               |      | 0             | H'0000 to H'0001             | H'0000 to H'0003 |
| Manual reset <sup>*2</sup>                   |      | 1             | H'0002 to H'0003             | H'0004 to H'0007 |
| Reserved for system use                      |      | 2             | H'0004 to H'0005             | H'0008 to H'000B |
|                                              |      | 3             | H'0006 to H'0007             | H'000C to H'000F |
|                                              |      | 4             | H'0008 to H'0019             | H'0010 to H'0013 |
| Trace                                        |      | 5             | H'000A to H'000B             | H'0014 to H'0017 |
| Interrupt (direct transitions) <sup>*2</sup> |      | 6             | H'000C to H'000D             | H'0018 to H'001B |
| Interrupt (NMI)                              |      | 7             | H'000E to H'000F             | H'001C to H'001F |
| Trap instruction (#0)                        |      | 8             | H'0010 to H'0011             | H'0020 to H'0023 |
| (#1)                                         |      | 9             | H'0012 to H'0013             | H'0024 to H'0027 |
| (#2)                                         |      | 10            | H'0014 to H'0015             | H'0028 to H'002B |
| (#3)                                         |      | 11            | H'0016 to H'0017             | H'002C to H'002F |
| Reserved for system use                      |      | 12            | H'0018 to H'0019             | H'0030 to H'0033 |
|                                              |      | 13            | H'001A to H'001B             | H'0034 to H'0037 |
|                                              |      | 14            | H'001C to H'001D             | H'0038 to H'003B |
|                                              |      | 15            | H'001E to H'001F             | H'003C to H'003F |
| External interrupt                           | IRQ0 | 16            | H'0020 to H'0021             | H'0040 to H'0043 |
|                                              | IRQ1 | 17            | H'0022 to H'0023             | H'0044 to H'0047 |
|                                              | IRQ2 | 18            | H'0024 to H'0025             | H'0048 to H'004B |
|                                              | IRQ3 | 19            | H'0026 to H'0027             | H'004C to H'004F |
|                                              | IRQ4 | 20            | H'0028 to H'0029             | H'0050 to H'0053 |
|                                              | IRQ5 | 21            | H'002A to H'002B             | H'0054 to H'0057 |
| Reserved for system use                      |      | 22            | H'002C to H'002D             | H'0058 to H'005B |
|                                              |      | 23            | H'002E to H'002F             | H'005C to H'005F |
| Internal interrupt <sup>*3</sup>             |      | 24            | H'0030 to H'0031             | H'0060 to H'0063 |
|                                              |      | 127           | H'00FE to H'00FF             | H'01FC to H'01FF |

Notes: <sup>\*1</sup> Lower 16 bits of the address.

<sup>\*2</sup> Not available in this LSI.

<sup>\*3</sup> For details of internal interrupt vectors, see section 5.5, Interrupt Exception Handling Vector Table.

## 4.3 Reset

A reset has the highest exception priority.

When the  $\overline{\text{RES}}$  pin goes low, all processing halts and 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  $\overline{\text{RES}}$  pin low for at least 20 states. A reset initializes the internal state of the CPU and the registers of on-chip supporting modules.

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

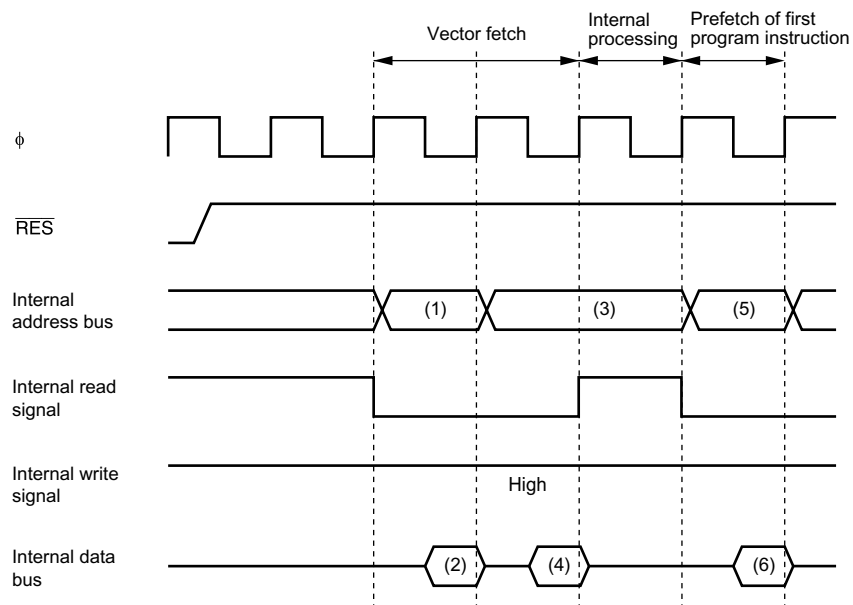
The interrupt control mode is 0 immediately after reset.

### 4.3.1 Reset Exception Handling

When the  $\overline{\text{RES}}$  pin goes high after being held low for the necessary time, this LSI starts reset exception handling as follows:

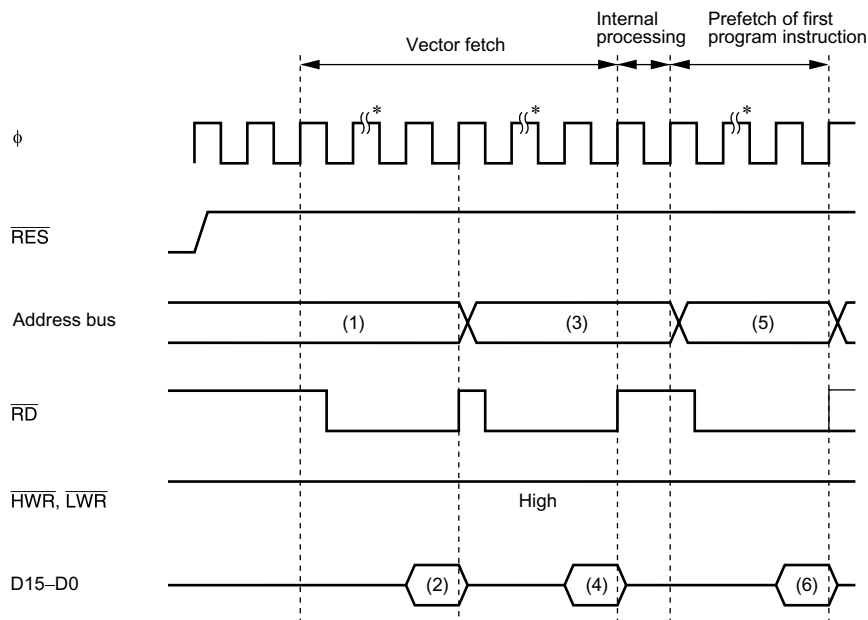
1. The internal state of the CPU and the registers of the on-chip supporting modules are initialized, the T bit is cleared to 0 in EXR, and the I bit is set to 1 in EXR and CCR.
2. The reset exception handling vector address is read and transferred to the PC, and program execution starts from the address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

Figures 4-1 and 4-2 show examples of the reset sequence.



- (1)(3) Reset exception handling vector address (when reset, (1)=H'000000, (3)=H'000002)  
 (2)(4) Start address (contents of reset exception handling vector address)  
 (5) Start address ((5)=(2)(4))  
 (6) First program instruction

**Figure 4-1 Reset Sequence (Advanced Mode with On-Chip ROM Enabled)**



- (1)(3) Reset exception handling vector address (when reset, (1)=H'000000, (3)=H'000002)  
 (2)(4) Start address (contents of reset exception handling vector address)  
 (5) Start address ((5)=(2)(4))  
 (6) First program instruction

Note: \* Three program wait states are inserted.

**Figure 4-2 Reset Sequence (Advanced Mode with On-Chip ROM Disabled: Cannot be Used in this LSI)**

### 4.3.2 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: 32, SP`).

### 4.3.3 State of On-Chip Supporting Modules after Reset Release

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

## 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 4-3 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. 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.

Interrupts are accepted even within the trace exception handling routine.

**Table 4-3 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 controller has two interrupt control modes and can assign interrupts other than NMI to eight priority/mask levels to enable multiplexed interrupt control. The source to start interrupt exception handling and the vector address differ depending on the product. 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.
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 4-4 shows the status of CCR and EXR after execution of trap instruction exception handling.

**Table 4-4 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 4-3 shows the stack after completion of trap instruction exception handling and interrupt exception handling.

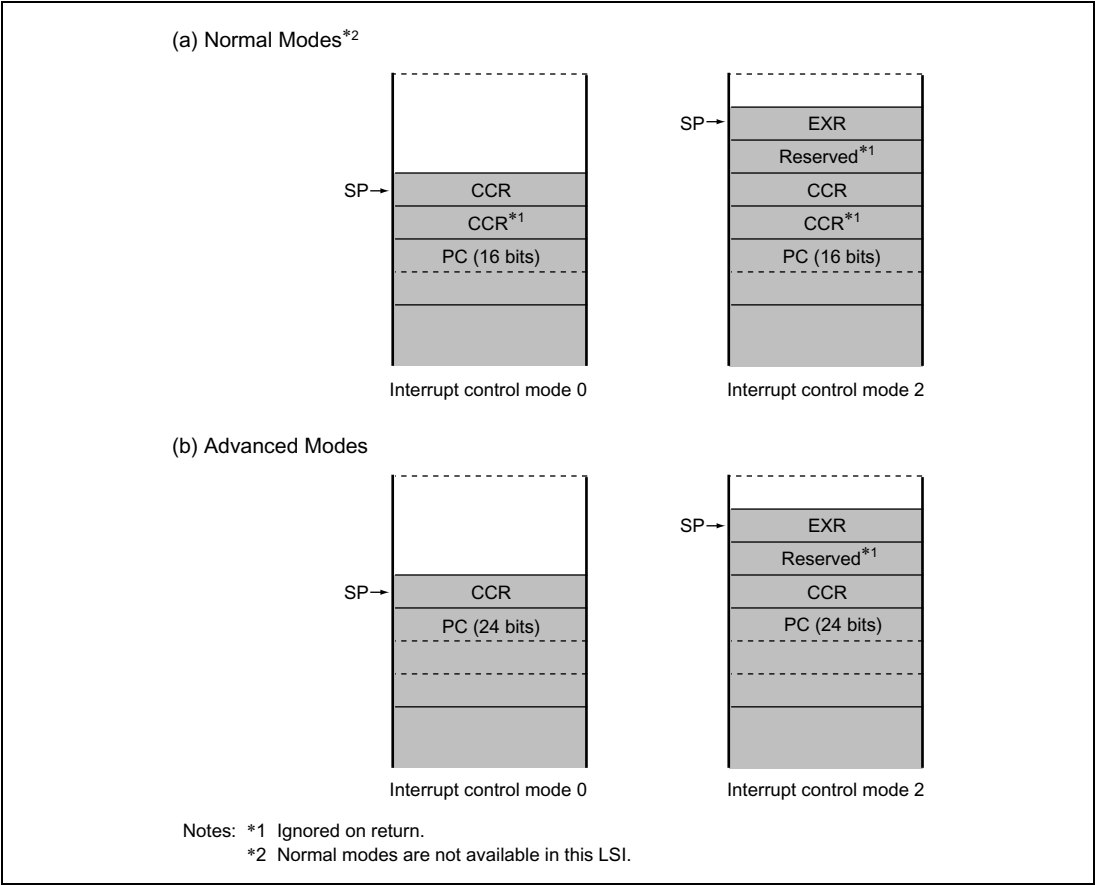


Figure 4-3 Stack Status after Exception Handling

## 4.8 Usage Note

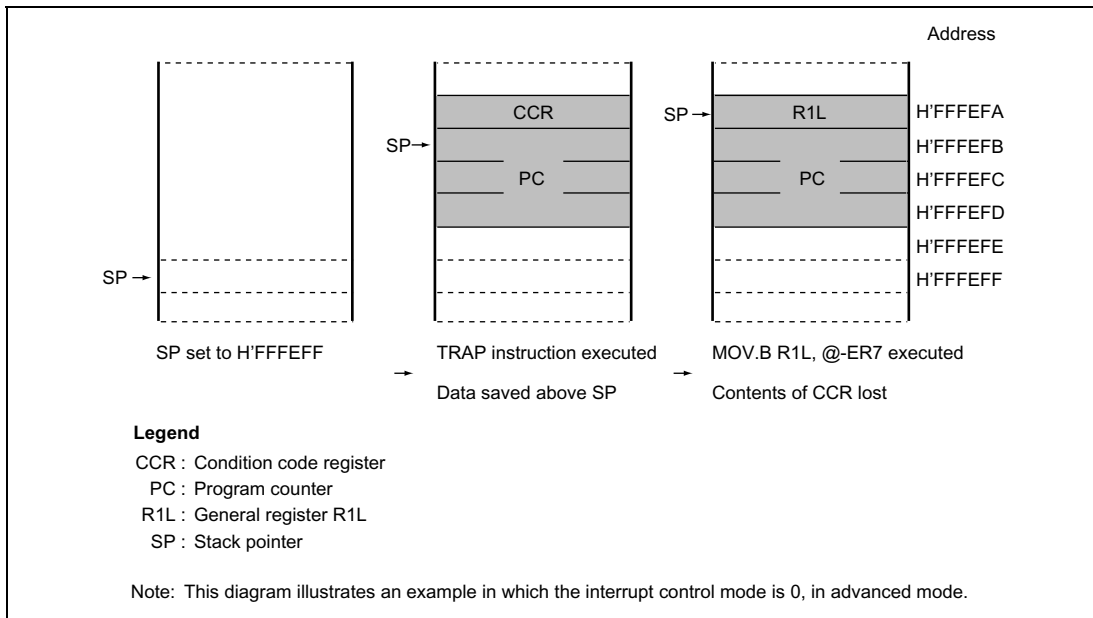
When accessing word data or longword data, this LSI assumes that the lowest address bit is 0. The stack should always be accessed by word transfer instruction or longword transfer instruction, and the value of the stack pointer (SP, ER7) should always be kept even. Use the following instructions to save 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 4-4 shows an example of what happens when the SP value is odd.



**Figure 4-4 Operation when SP Value is Odd**

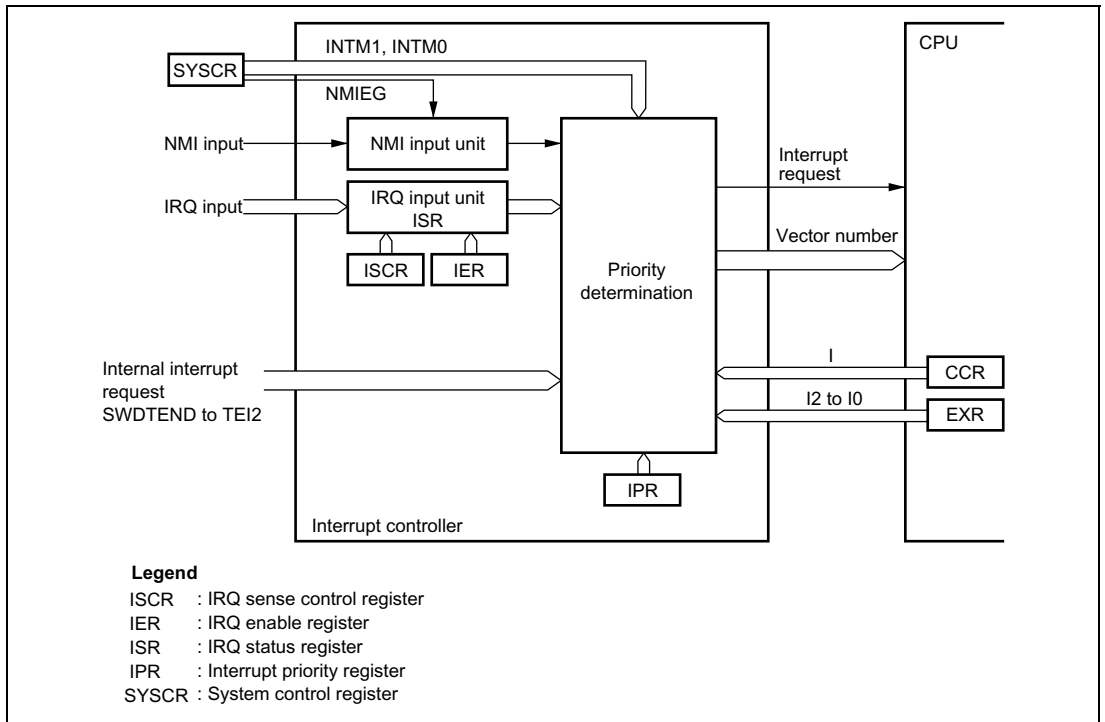


# Section 5 Interrupt Controller

## 5.1 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, and can be accepted at all times.
- 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.
- Seven 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 selected for IRQ0 to IRQ5.
- DTC control
  - The DTC can be activated by an interrupt request.

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<br>Rising or falling edge can be selected                          |
| IRQ5 | Input | Maskable external interrupts<br>Rising, falling, or both edges, or level sensing, can be selected |
| IRQ4 | Input |                                                                                                   |
| IRQ3 | Input |                                                                                                   |
| IRQ2 | Input |                                                                                                   |
| IRQ1 | Input |                                                                                                   |
| IRQ0 | Input |                                                                                                   |

## 5.3 Register Descriptions

The interrupt controller has the following registers. For details on system control register (SYSCR), refer to section 3.2.2, System Control Register (SYSCR). For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register.

- 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 J (IPRJ)
- Interrupt priority register K (IPRK)
- Interrupt priority register M (IPRM)

### 5.3.1 Interrupt Priority Registers A to H, J, K, M (IPRA to IPRH, IPRJ, IPRK, IPRM)

The IPR registers are eleven 8-bit readable/writable registers that set priorities (levels 7 to 0) for interrupts 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<br>These bits are always read as 0.                                                                                                                                                                                                                                        |
| 6   | IPR6     | 1             | R/W | Sets the priority of the corresponding interrupt source.<br>000: Priority level 0 (Lowest)<br>001: Priority level 1<br>010: Priority level 2<br>011: Priority level 3<br>100: Priority level 4<br>101: Priority level 5<br>110: Priority level 6<br>111: Priority level 7 (Highest) |
| 5   | IPR5     | 1             | R/W |                                                                                                                                                                                                                                                                                     |
| 4   | IPR4     | 1             | R/W |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
| 3   | —        | 0             | —   | Reserved<br>These bits are always read as 0.                                                                                                                                                                                                                                        |
| 2   | IPR2     | 1             | R/W | Sets the priority of the corresponding interrupt source.<br>000: Priority level 0 (Lowest)<br>001: Priority level 1<br>010: Priority level 2<br>011: Priority level 3<br>100: Priority level 4<br>101: Priority level 5<br>110: Priority level 6<br>111: Priority level 7 (Highest) |
| 1   | IPR1     | 1             | R/W |                                                                                                                                                                                                                                                                                     |
| 0   | IPR0     | 1             | R/W |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |
|     |          |               |     |                                                                                                                                                                                                                                                                                     |

### 5.3.2 IRQ Enable Register (IER)

IER is an 8-bit readable/writable register that controls the enabling and disabling of interrupt requests IRQ0 to IRQ5.

| Bit | Bit Name | Initial Value | R/W | Description                                                              |
|-----|----------|---------------|-----|--------------------------------------------------------------------------|
| 7   | —        | 0             | R/W | Reserved                                                                 |
| 6   | —        | 0             | R/W | Only 0 should be written to these bits.                                  |
| 5   | IRQ5E    | 0             | R/W | IRQ5 Enable<br>The IRQ5 interrupt request is enabled when this bit is 1. |
| 4   | IRQ4E    | 0             | R/W | IRQ4 Enable<br>The IRQ4 interrupt request is enabled when this bit is 1. |
| 3   | IRQ3E    | 0             | R/W | IRQ3 Enable<br>The IRQ3 interrupt request is enabled when this bit is 1. |
| 2   | IRQ2E    | 0             | R/W | IRQ2 Enable<br>The IRQ2 interrupt request is enabled when this bit is 1. |
| 1   | IRQ1E    | 0             | R/W | IRQ1 Enable<br>The IRQ1 interrupt request is enabled when this bit is 1. |
| 0   | IRQ0E    | 0             | R/W | IRQ0 Enable<br>The IRQ0 interrupt request is enabled when this bit is 1. |

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

The ISCR registers are 16-bit readable/writable registers that select the source that generates an interrupt request at pins  $\overline{\text{IRQ0}}$  to  $\overline{\text{IRQ5}}$ .

| Bit | Bit Name | Initial Value | R/W | Description                                                                                        |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------|
| 15  | —        | 0             | R/W | Reserved                                                                                           |
| 14  | —        | 0             | R/W | Only 0 should be written to these bits.                                                            |
| 13  | —        | 0             | R/W |                                                                                                    |
| 12  | —        | 0             | R/W |                                                                                                    |
| 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 |
| 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 |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                        |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------|
| 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 is an 8-bit readable/writable register that indicates the status of IRQ0 to IRQ5 interrupt requests.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | 0             | R/W | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 6   | —        | 0             | R/W | Only 0 should be written to these bits.                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 5   | IRQ5F    | 0             | R/W | [Setting condition]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 4   | IRQ4F    | 0             | R/W | When the interrupt source selected by the ISCR registers occurs                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 3   | IRQ3F    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 2   | IRQ2F    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 1   | IRQ1F    | 0             | R/W | [Clearing conditions]                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 0   | IRQ0F    | 0             | R/W | <ul style="list-style-type: none"> <li>• Cleared by reading IRQnF flag when IRQnF = 1, then writing 0 to IRQnF flag</li> <li>• When interrupt exception handling is executed when low-level detection is set and <math>\overline{\text{IRQn}}</math> input is high</li> <li>• When IRQn interrupt exception handling is executed when falling, rising, or both-edge detection is set</li> <li>• When the DTC is activated by an IRQn interrupt, and the DISEL bit in MRB of the DTC is cleared to 0</li> </ul> |

(n=5 to 0)

## 5.4 Interrupt

### 5.4.1 External Interrupts

There are seven external interrupts: NMI and IRQ0 to IRQ5. These interrupts can be used to restore this LSI from software standby mode.

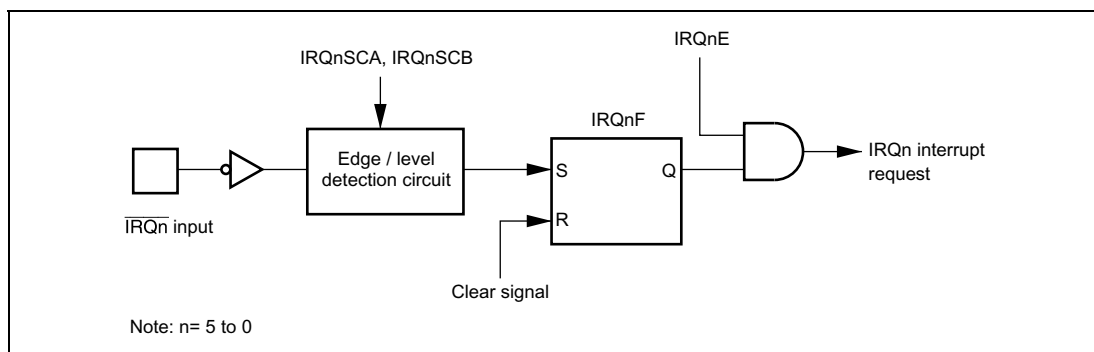
**NMI Interrupt:** NMI is the highest-priority interrupt, and is always accepted by the CPU regardless of the interrupt control mode or the status of the CPU interrupt mask bits. The NMIEG bit in SYSCR can be used to select whether an interrupt is requested at a rising edge or a falling edge on the NMI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

**IRQ0 to IRQ5 Interrupts:** Interrupts IRQ0 to IRQ5 are requested by an input signal at pins  $\overline{\text{IRQ0}}$  to  $\overline{\text{IRQ5}}$ . Interrupts IRQ0 to IRQ5 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 pins  $\overline{\text{IRQ0}}$  to  $\overline{\text{IRQ5}}$ .
- Enabling or disabling of interrupt requests IRQ0 to IRQ5 can be selected with IER.
- The interrupt priority level can be set with IPR.
- The status of interrupt requests IRQ0 to IRQ5 is indicated in ISR. ISR flags can be cleared to 0 by software.

The detection of IRQ0 to IRQ5 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.

A block diagram of interrupts IRQ0 to IRQ5 is shown in figure 5-2.



**Figure 5-2 Block Diagram of Interrupts IRQ0 to IRQ5**

### 5.4.2 Internal Interrupts

The sources for internal interrupts from on-chip supporting modules have the following features:

- For each on-chip supporting module there are flags that indicate the interrupt request status, and enable bits that select enabling or disabling of these interrupts. If both of these are set to 1 for a particular interrupt source, an interrupt request is issued to the interrupt controller.
- The interrupt priority level can be set by means of IPR.
- The DTC can be activated by a TPU, SCI, or other interrupt request.
- When the DTC is activated by an interrupt request, it is not affected by the interrupt control mode or CPU interrupt mask bit.

## 5.5 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.

**Table 5-2 Interrupt Sources, Vector Addresses, and Interrupt Priorities**

| Interrupt Source | Origin of Interrupt Source | Vector Number | Vector Address* | IPR            | Priority                                    |
|------------------|----------------------------|---------------|-----------------|----------------|---------------------------------------------|
|                  |                            |               | Advanced Mode   |                |                                             |
| External pin     | NMI                        | 7             | H'001C          |                | <div>High</div> <div>↑</div> <div>Low</div> |
|                  | IRQ0                       | 16            | H'0040          | IPRA6 to IPRA4 |                                             |
|                  | IRQ1                       | 17            | H'0044          | IPRA2 to IPRA0 |                                             |
|                  | IRQ2                       | 18            | H'0048          | IPRB6 to IPRB4 |                                             |
|                  | IRQ3                       | 19            | H'004C          |                |                                             |
|                  | IRQ4                       | 20            | H'0050          | IPRB2 to IPRB0 |                                             |
|                  | IRQ5                       | 21            | H'0054          |                |                                             |
| —                | Reserved for system use    | 22            | H'0058          |                |                                             |
|                  | Reserved for system use    | 23            | H'005C          |                |                                             |
| DTC              | SWDTEND                    | 24            | H'0060          | IPRC2 to IPRC0 |                                             |
| Watchdog timer 0 | WOVI0                      | 25            | H'0064          | IPRD6 to IPRD4 |                                             |
| PC break         | PC break                   | 27            | H'006C          | IPRE6 to IPRE4 |                                             |
| A/D              | ADI                        | 28            | H'0070          | IPRE2 to IPRE0 |                                             |
| TPU channel 0    | TGIA_0                     | 32            | H'0080          | IPRF6 to IPRF4 |                                             |
|                  | TGIB_0                     | 33            | H'0084          |                |                                             |
|                  | TGIC_0                     | 34            | H'0088          |                |                                             |
|                  | TGID_0                     | 35            | H'008C          |                |                                             |
|                  | TCIV_0                     | 36            | H'0090          |                |                                             |
| TPU channel 1    | TGIA_1                     | 40            | H'00A0          | IPRF2 to IPRF0 |                                             |
|                  | TGIB_1                     | 41            | H'00A4          |                |                                             |
|                  | TCIV_1                     | 42            | H'00A8          |                |                                             |
|                  | TCIU_1                     | 43            | H'00AC          |                |                                             |
| TPU channel 2    | TGIA_2                     | 44            | H'00B0          | IPRG6 to IPRG4 |                                             |
|                  | TGIB_2                     | 45            | H'00B4          |                |                                             |
|                  | TCIV_2                     | 46            | H'00B8          |                |                                             |
|                  | TCIU_2                     | 47            | H'00BC          |                |                                             |

| Interrupt Source | Origin of Interrupt Source | Vector Number | Vector Address* | IPR            | Priority  |
|------------------|----------------------------|---------------|-----------------|----------------|-----------|
|                  |                            |               | Advanced Mode   |                |           |
| TPU<br>channel 3 | TGIA_3                     | 48            | H'00C0          | IPRG2 to IPRG0 | High<br>↑ |
|                  | TGIB_3                     | 49            | H'00C4          |                |           |
|                  | TGIC_3                     | 50            | H'00C8          |                |           |
|                  | TGID_3                     | 51            | H'00CC          |                |           |
|                  | TCIV_3                     | 52            | H'00D0          |                |           |
| TPU<br>channel 4 | TGIA_4                     | 56            | H'00E0          | IPRH6 to IPRH4 |           |
|                  | TGIB_4                     | 57            | H'00E4          |                |           |
|                  | TCIV_4                     | 58            | H'00E8          |                |           |
|                  | TCIU_4                     | 59            | H'00EC          |                |           |
| TPU<br>channel 5 | TGIA_5                     | 60            | H'00F0          | IPRH2 to IPRH0 |           |
|                  | TGIB_5                     | 61            | H'00F4          |                |           |
|                  | TCIV_5                     | 62            | H'00F8          |                |           |
|                  | TCIU_5                     | 63            | H'00FC          |                |           |
| SCI<br>channel 0 | ERI_0                      | 80            | H'0140          | IPRJ2 to IPRJ0 |           |
|                  | RXI_0                      | 81            | H'0144          |                |           |
|                  | TXI_0                      | 82            | H'0148          |                |           |
|                  | TEI_0                      | 83            | H'014C          |                |           |
| SCI<br>channel 1 | ERI_1                      | 84            | H'0150          | IPRK6 to IPRK4 |           |
|                  | RXI_1                      | 85            | H'0154          |                |           |
|                  | TXI_1                      | 86            | H'0158          |                |           |
|                  | TEI_1                      | 87            | H'015C          |                |           |
| SCI<br>channel 2 | ERI_2                      | 88            | H'0160          | IPRK2 to IPRK0 |           |
|                  | RXI_2                      | 89            | H'0164          |                |           |
|                  | TXI_2                      | 90            | H'0168          |                |           |
|                  | TEI_2                      | 91            | H'016C          |                |           |
| HCAN             | ERS0, OVR0                 | 104           | H'01A0          | IPRM6 to IPRM4 |           |
|                  | RM0                        | 105           | H'01A4          |                |           |
|                  | RM1                        | 106           | H'01A8          |                |           |
|                  | SLE0                       | 107           | H'01AC          |                |           |
| MMT              | TGIMN                      | 108           | H'01B0          | IPRM2 to IPRM0 |           |
|                  | TGINN                      | 109           | H'01B4          |                |           |
|                  | POEIN                      | 110           | H'01B8          |                |           |
| —                | Reserved for system use    | 111           | H'01BC          |                | Low       |

Note: \* Lower 16 bits of the start address.

## 5.6 Interrupt Control Modes and Interrupt Operation

The interrupt controller has two modes: interrupt control mode 0 and interrupt control mode 2. Interrupt operations differ depending on the interrupt control mode. The interrupt control mode is selected by SYSCR. Table 5-3 shows the differences between interrupt control mode 0 and interrupt control mode 2.

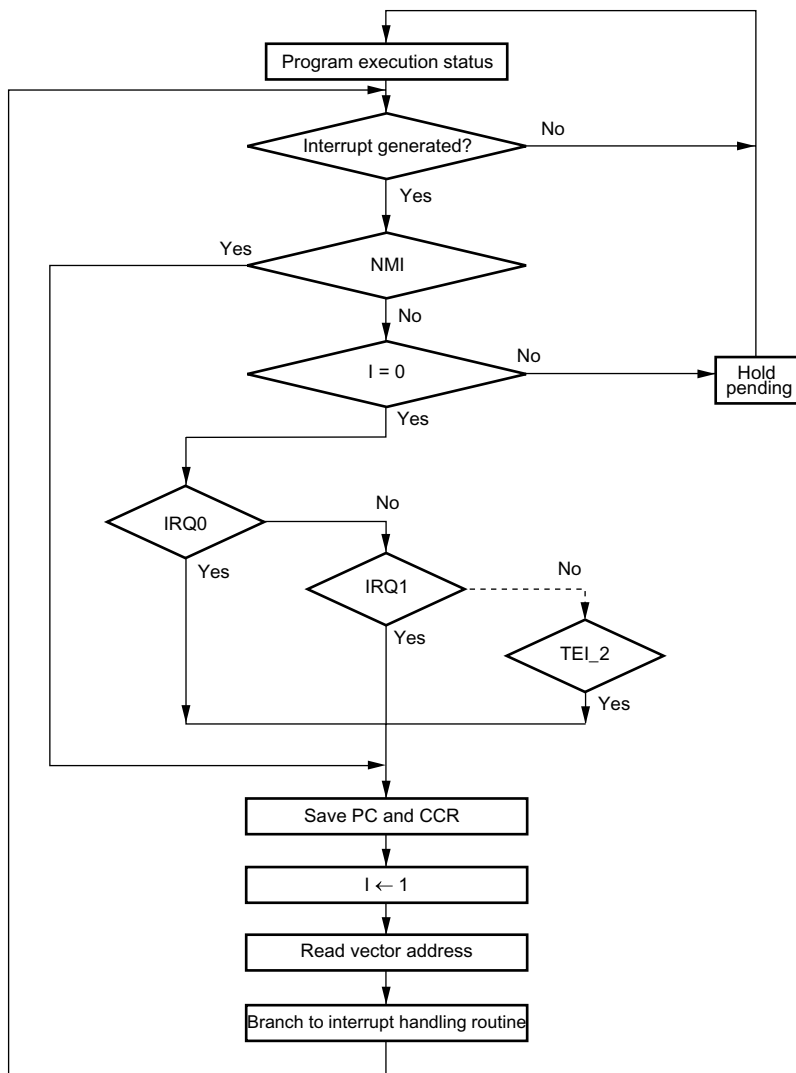
**Table 5-3 Interrupt Control Modes**

| <b>Interrupt Control Mode</b> | <b>Priority Setting Registers</b> | <b>Interrupt Mask Bits</b> | <b>Description</b>                                                                                                                    |
|-------------------------------|-----------------------------------|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| 0                             | Default                           | I                          | The priorities of interrupt sources are fixed at the default settings.<br>Interrupt sources, except for NMI, are masked by the I bit. |
| 2                             | IPR                               | I2 to I0                   | 8 priority levels other than NMI can be set with IPR.<br>8-level interrupt mask control is performed by bits I2 to I0.                |

### 5.6.1 Interrupt Control Mode 0

In interrupt control mode 0, interrupt requests other than for NMI are masked by the I bit of the CCR in the CPU. Figure 5-3 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.

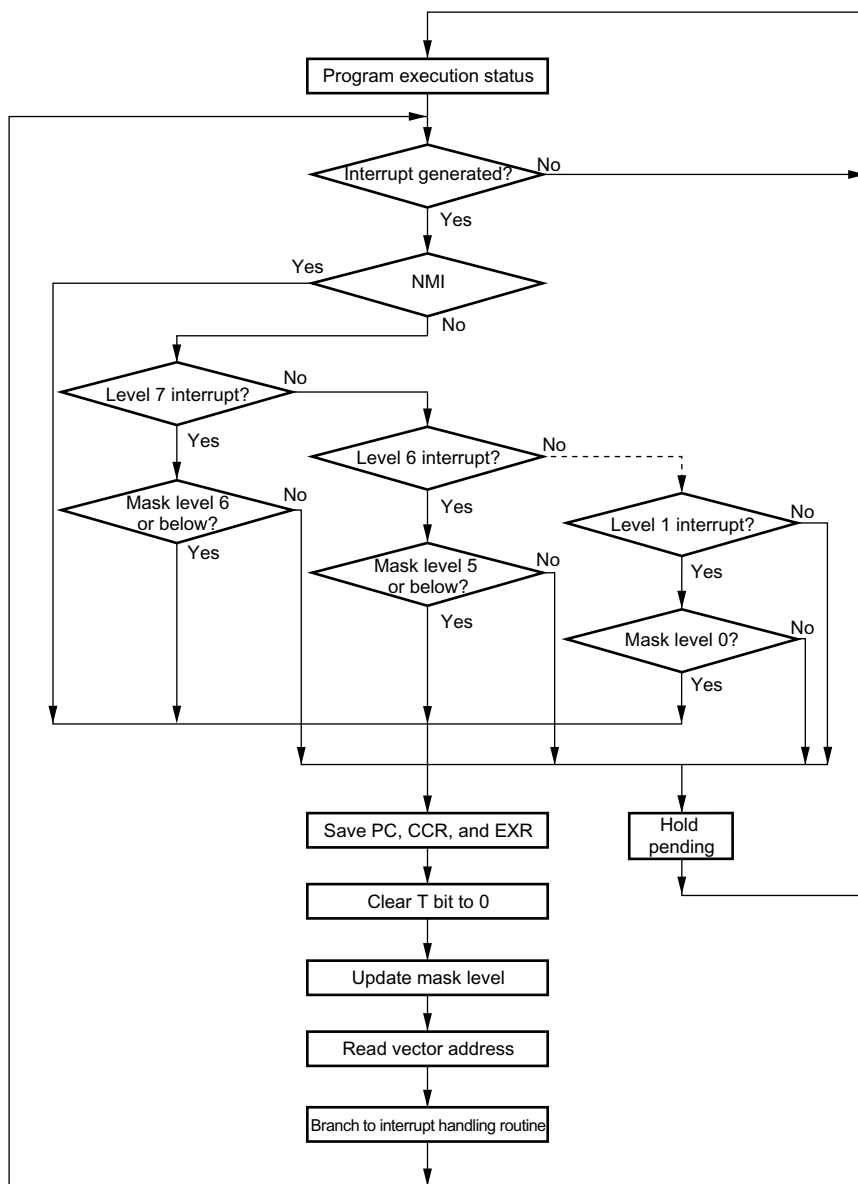


**Figure 5-3 Flowchart of Procedure Up to Interrupt Acceptance in Interrupt Control Mode 0**

### 5.6.2 Interrupt Control Mode 2

In interrupt control mode 2, mask control is applied to eight levels for interrupt requests other than NMI by comparing the EXR interrupt mask level (I2 to I0 bits) in the CPU and the IPR setting. Figure 5-4 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 5-4 Flowchart of Procedure Up to Interrupt Acceptance in Control Mode 2**

### 5.6.3 Interrupt Exception Handling Sequence

Figure 5-5 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.

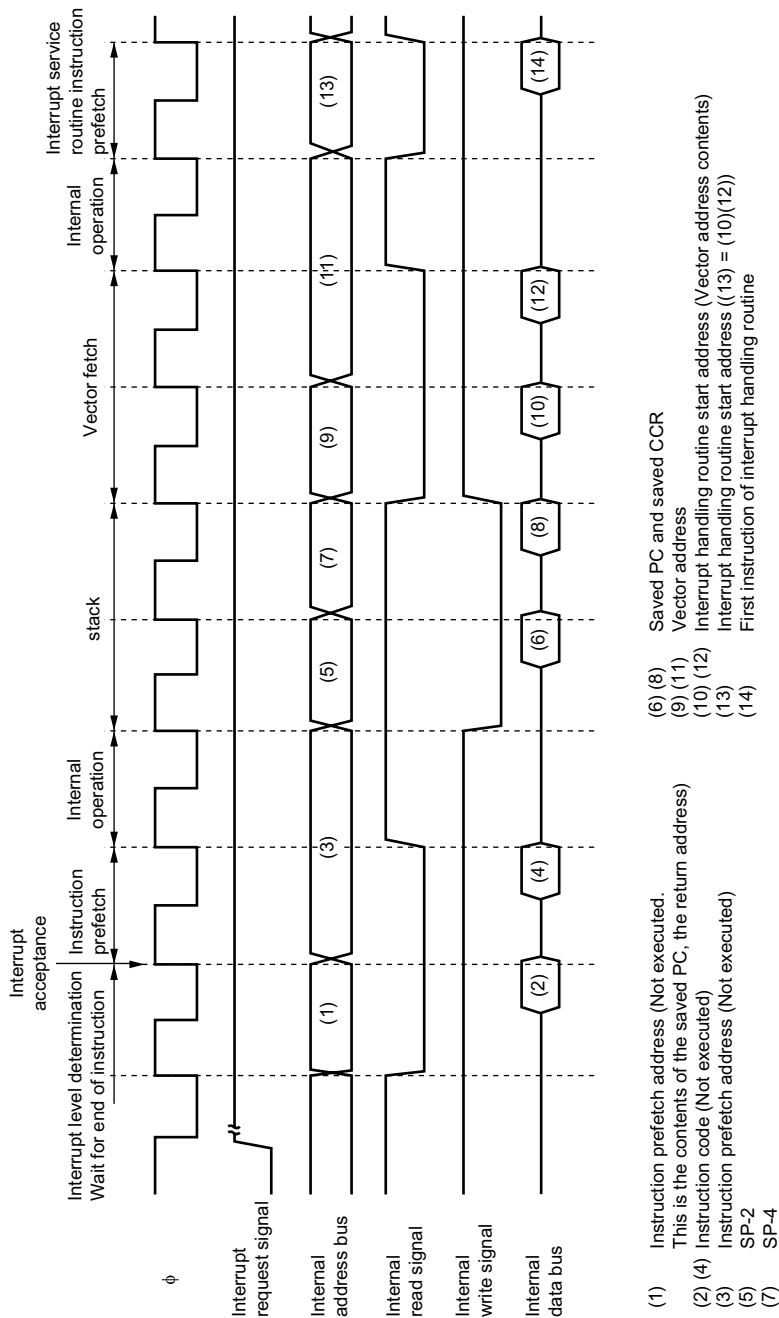


Figure 5-5 Interrupt Exception Handling

### 5.6.4 Interrupt Response Times

Table 5-4 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 5-4 are explained in table 5-5.

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 5-4 Interrupt Response Times**

| No.                          | Execution Status                                                     | Normal Mode <sup>*5</sup>  |                            | Advanced Mode              |                            |
|------------------------------|----------------------------------------------------------------------|----------------------------|----------------------------|----------------------------|----------------------------|
|                              |                                                                      | Interrupt control mode 0   | Interrupt control mode 2   | Interrupt control mode 0   | Interrupt control mode 2   |
| 1                            | Interrupt priority determination <sup>*1</sup>                       | 3                          | 3                          | 3                          | 3                          |
| 2                            | Number of wait states until executing instruction ends <sup>*2</sup> | 1 to 19 + 2·S <sub>I</sub> | 1 to 19 + 2·S <sub>I</sub> | 1 to 19 + 2·S <sub>I</sub> | 1 to 19 + 2·S <sub>I</sub> |
| 3                            | PC, CCR, EXR stack save                                              | 2·S <sub>K</sub>           | 3·S <sub>K</sub>           | 2·S <sub>K</sub>           | 3·S <sub>K</sub>           |
| 4                            | Vector fetch                                                         | S <sub>I</sub>             | S <sub>I</sub>             | 2·S <sub>I</sub>           | 2·S <sub>I</sub>           |
| 5                            | Instruction fetch <sup>*3</sup>                                      | 2·S <sub>I</sub>           | 2·S <sub>I</sub>           | 2·S <sub>I</sub>           | 2·S <sub>I</sub>           |
| 6                            | Internal processing <sup>*4</sup>                                    | 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 5-5 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 <sub>I</sub> | 1               | 4                 | 6+2m           | 2              | 3+m            |
| Branch address read | S <sub>J</sub> |                 |                   |                |                |                |
| Stack manipulation  | S <sub>K</sub> |                 |                   |                |                |                |

**Legend**

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

Note:\* Cannot be used in this LSI.

**5.6.5 DTC Activation by Interrupt**

The DTC can be activated by an interrupt. For details, see section 8, Data Transfer Controller (DTC).

**5.7 Usage Notes****5.7.1 Contention between Interrupt Generation and Disabling**

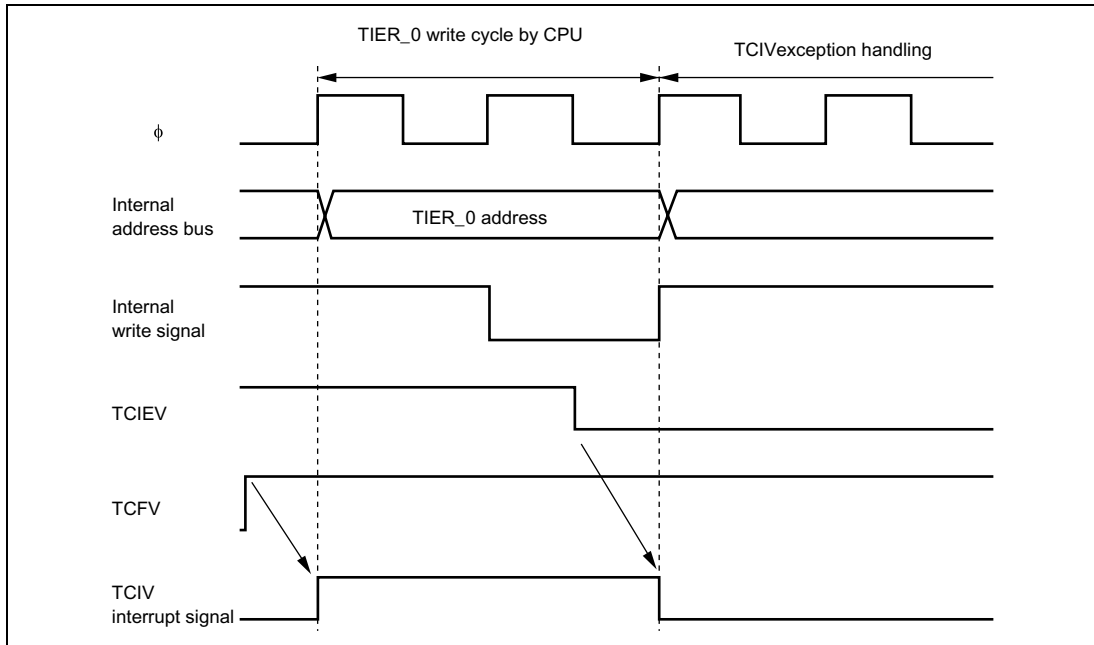
When an interrupt enable bit is cleared to 0 to disable interrupts, the disabling becomes effective after execution of the instruction.

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 5-6 shows an example in which the TGIEA bit in the TPU's TIER\_0 register 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 5-6 Contention between Interrupt Generation and Disabling**

### 5.7.2 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.

### 5.7.3 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.

#### 5.7.4 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
```

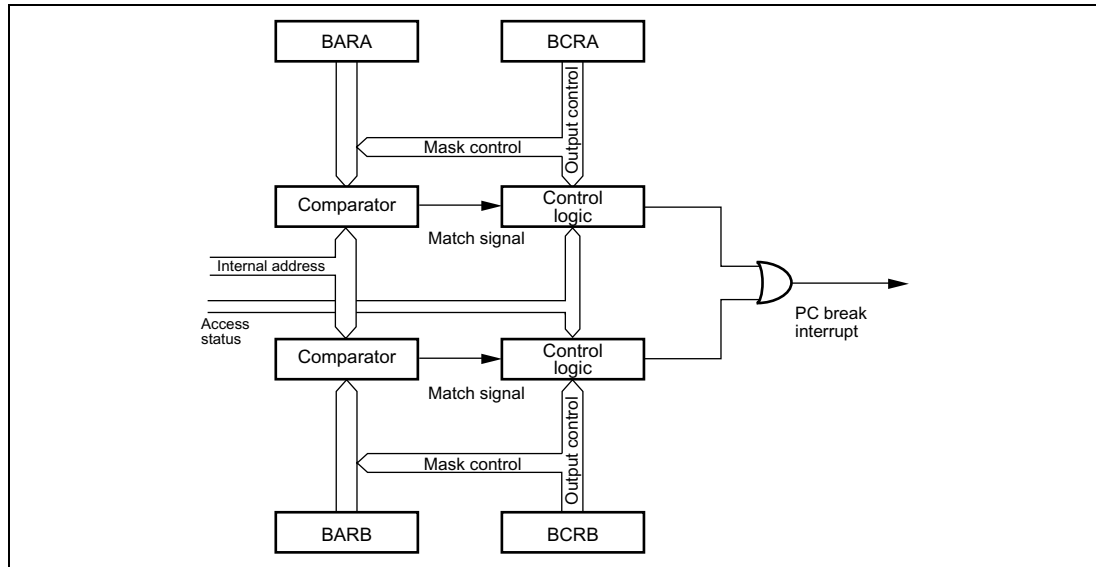


## Section 6 PC Break Controller (PBC)

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. For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register.

- 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<br>These bits are read as an undefined value and cannot be modified. |
| 23 to 0  | BAA23 to BAA0 | H'000000      | R/W | 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. BCRA also contains a condition match flag.

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

### 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 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 BCR.

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 3 to 5 (BAMA2–0). Set bits 1 and 2 (CSELA1–0) 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 3 to 5 (BAMA2–0). Set bits 1 and 2 (CSELA1–0) 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.

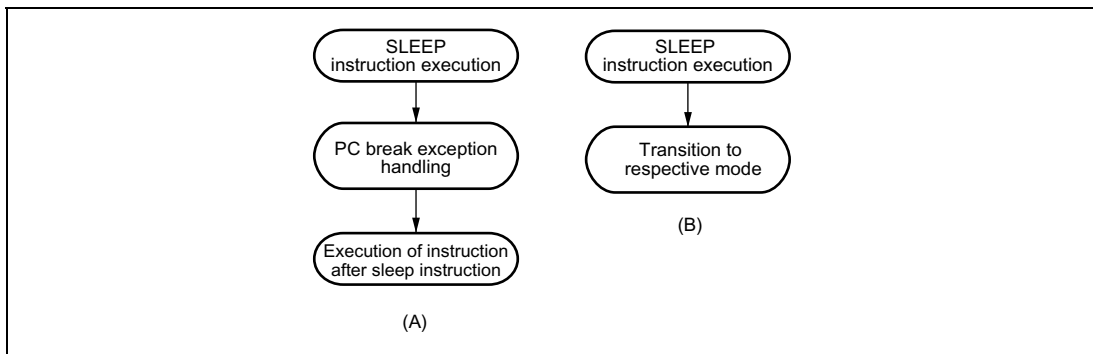
### 6.3.3 Notes on PC Break Interrupt Handling

- 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:  
After execution of the SLEEP instruction, a transition is not made to sleep 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 to software standby 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 (B)).



**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 20, 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 is Bus Master**

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

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

When a PC break is set for an instruction fetch at an address following a BSR, JSR, JMP, TRAPA, RTE, or 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, or XORC Instruction**

When the I bit is set by an LDC, ANDC, ORC, or 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**

When a PC break is set for an instruction fetch at the branch destination address of a 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

The H8S/2600 CPU is driven by a system clock, denoted by the symbol  $\phi$ .

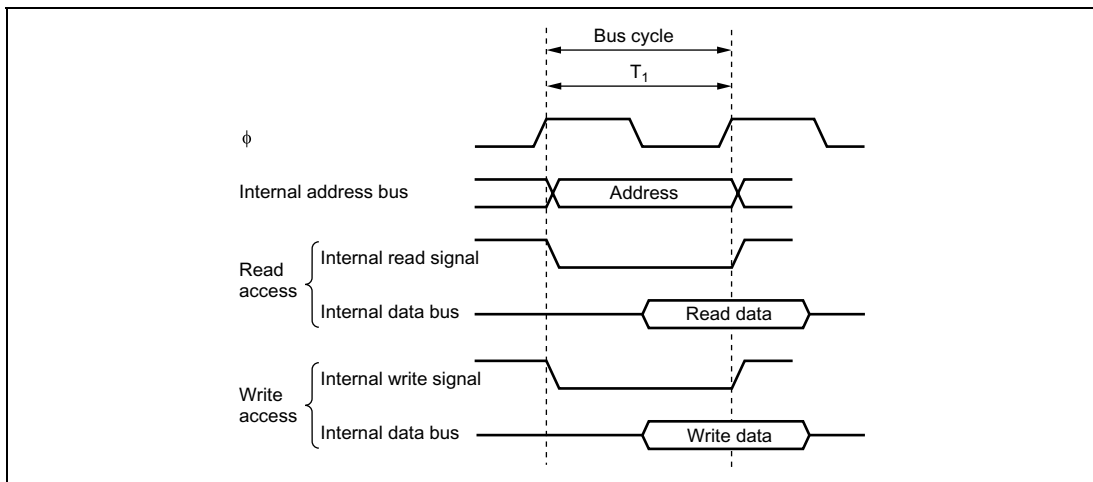
The bus controller controls a memory cycle and a bus cycle. Different methods are used to access on-chip memory and on-chip support modules. The bus controller also has a bus arbitration function, and controls the operation of the internal bus masters: the CPU and data transfer controller (DTC).

## 7.1 Basic Timing

The period from one rising edge of  $\phi$  to the next is referred to as a "state." The memory cycle or bus cycle consists of one, two, three, or four states. Different methods are used to access on-chip memory, on-chip support modules, and the external address space.

### 7.1.1 On-Chip Memory Access Timing (ROM, RAM)

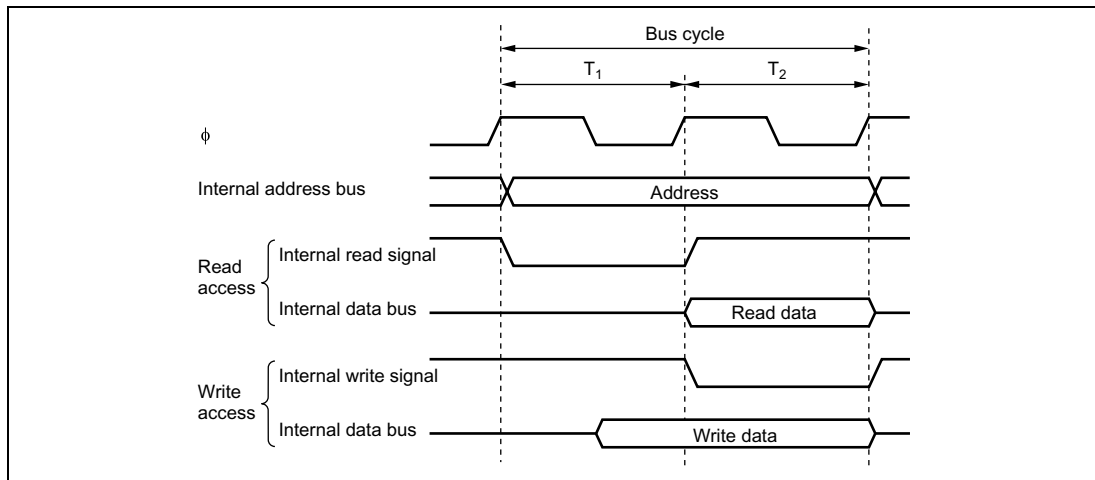
On-chip memory is accessed in one state. The data bus is 16 bits wide, permitting both byte and word transfer instruction. Figure 7-1 shows the on-chip memory access cycle.



**Figure 7-1 On-Chip Memory Access Cycle**

### 7.1.2 On-Chip Support Module Access Timing

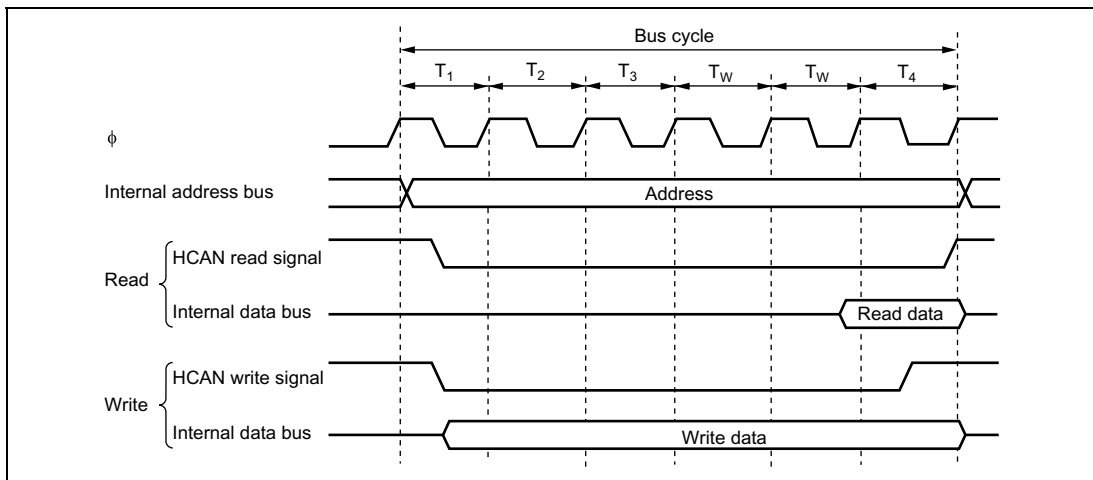
The on-chip support modules, except for HCAN, MMT, and POE, are accessed in two states. The data bus is either 8 bits or 16 bits wide, depending on the particular internal I/O register being accessed. For details, refer to Appendix A, On-Chip I/O Register. Figure 7-2 shows access timing for the on-chip supporting modules.



**Figure 7-2 On-Chip Support Module Access Cycle**

### 7.1.3 On-Chip HCAN Module Access Timing

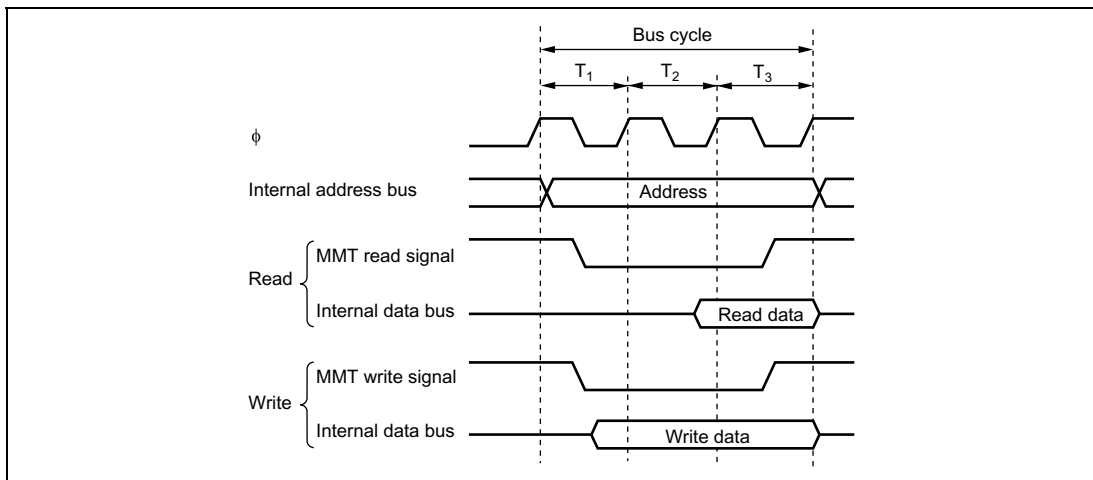
On-chip HCAN module access is performed in four states. The data bus width is 16 bits. Wait states can be inserted by means of a wait request from the HCAN. On-chip HCAN module access timing is shown in figures 7-3.



**Figure 7-3 On-Chip HCAN Module Access Cycle (Wait States Inserted)**

### 7.1.4 On-Chip MMT Module Access Timing

On-chip MMT module and POE access are performed in three states. The data width is 16 bits. On-chip MMT module access timings are shown in figure 7-4.



**Figure 7-4 On-Chip MMT Module Access Cycle**

## 7.2 Bus Arbitration

The Bus Controller has a bus arbiter that arbitrates bus master operations.

There are two bus masters, the CPU and DTC, which perform read/write operations when they control the bus.

### 7.2.1 Order of Priority of the Bus Masters

Each bus master requests the bus by means of a bus request signal. 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) DTC > CPU (Low)

### 7.2.2 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. The CPU is the lowest-priority bus master, and if a bus request is received from the 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 such operations. For details, refer to section 2.7, Bus States During Instruction Execution, in the H8S/2600 Series, H8S/2000 Series Programming Manual.
- If the CPU is in sleep mode, it transfers the bus immediately.

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).

## Section 8 Data Transfer Controller (DTC)

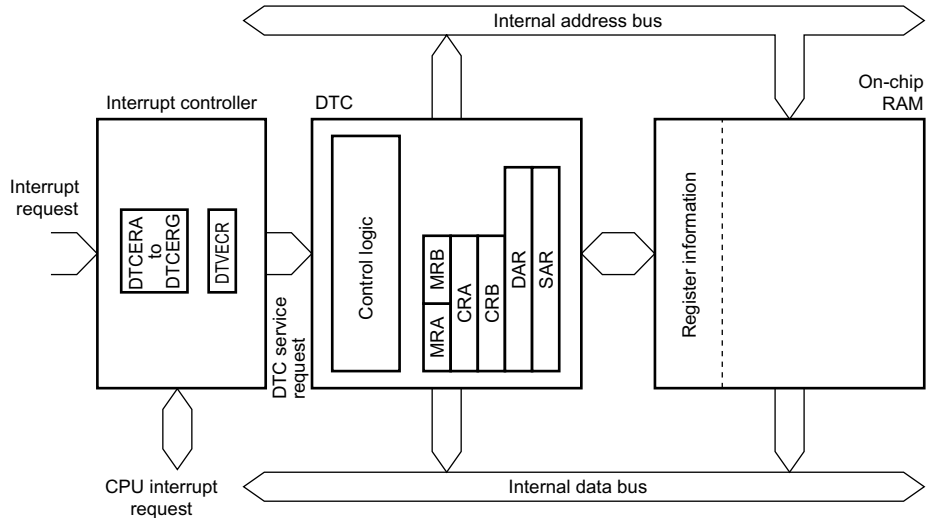
This LSI includes a data transfer controller (DTC). The DTC can be activated by an interrupt or software, to transfer data.

Figure 8-1 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.

### 8.1 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



#### Legend

|                  |                                        |
|------------------|----------------------------------------|
| MRA, MRB         | : DTC mode registers A and B           |
| CRA, CRB         | : DTC transfer count registers A and B |
| SAR              | : DTC source address register          |
| DAR              | : DTC destination address register     |
| DTCERA to DTCERG | : DTC enable registers A to G          |
| DTVECR           | : DTC vector register                  |

**Figure 8-1 Block Diagram of DTC**

## 8.2 Register Configuration

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 (DTCER)
- DTC vector register (DTVECR)

For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register.

## 8.2.1 DTC Mode Register A (MRA)

MRA is an 8-bit register that 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.<br>0X: SAR is fixed<br>10: SAR is incremented after a transfer<br>(by +1 when Sz = 0; by +2 when Sz = 1)<br>11: SAR is decremented after a transfer<br>(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.<br>0X: DAR is fixed<br>10: DAR is incremented after a transfer<br>(by +1 when Sz = 0; by +2 when Sz = 1)<br>11: DAR is decremented after a transfer<br>(by -1 when Sz = 0; by -2 when Sz = 1)                   |
| 3   | MD1      | Undefined     | —   | DTC Mode                                                                                                                                                                                                                                                                  |
| 2   | MD0      | Undefined     | —   | These bits specify the DTC transfer mode.<br>00: Normal mode<br>01: Repeat mode<br>10: Block transfer mode<br>11: Setting prohibited                                                                                                                                      |
| 1   | DTS      | Undefined     | —   | DTC Transfer Mode Select<br>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.<br>0: Destination side is repeat area or block area<br>1: Source side is repeat area or block area |
| 0   | Sz       | Undefined     | —   | DTC Data Transfer Size<br>Specifies the size of data to be transferred.<br>0: Byte-size transfer<br>1: Word-size transfer                                                                                                                                                 |

### Legend

X : Don't care

## 8.2.2 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<br><br>When this bit is set to 1, a chain transfer will be performed. For details, refer to section 8.5.4, Chain Transfer.<br><br>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. |
| 6   | DISEL    | Undefined     | —   | DTC Interrupt Select<br><br>When this bit is set to 1, a CPU interrupt request is generated every time after the end of a data transfer. When this bit is set to 0, a CPU interrupt request is generated at the time when the specified number of data transfer ends.                                                                            |
| 5   | —        | Undefined     | —   | Reserved                                                                                                                                                                                                                                                                                                                                         |
| 4   | —        | Undefined     | —   | These bits have no effect on DTC operation. Only 0 should be written to these bits.                                                                                                                                                                                                                                                              |
| 3   | —        | Undefined     | —   |                                                                                                                                                                                                                                                                                                                                                  |
| 2   | —        | Undefined     | —   |                                                                                                                                                                                                                                                                                                                                                  |
| 1   | —        | Undefined     | —   |                                                                                                                                                                                                                                                                                                                                                  |
| 0   | —        | Undefined     | —   |                                                                                                                                                                                                                                                                                                                                                  |

## 8.2.3 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.

## 8.2.4 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.

## 8.2.5 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). CRAH holds the number of transfers while CRAL functions as an 8-bit transfer 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.

## 8.2.6 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.

## 8.2.7 DTC Enable Registers (DTCER)

DTCER is comprised of seven registers; DTCERA to DTCERG, and is a register that specifies DTC activation interrupt sources. The correspondence between interrupt sources and DTCE bits is shown in table 8-1. For DTCE bit setting, use bit manipulation instructions such as BSET and BCLR for reading and writing. 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.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                          |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | DTCE7    | 0             | R/W | DTC Activation Enable                                                                                                                                                |
| 6   | DTCE6    | 0             | R/W | Setting this bit to 1 specifies a relevant interrupt source as a DTC activation source.                                                                              |
| 5   | DTCE5    | 0             | R/W |                                                                                                                                                                      |
| 4   | DTCE4    | 0             | R/W |                                                                                                                                                                      |
| 3   | DTCE3    | 0             | R/W |                                                                                                                                                                      |
| 2   | DTCE2    | 0             | R/W | [Clearing conditions]                                                                                                                                                |
| 1   | DTCE1    | 0             | R/W | <ul style="list-style-type: none"> <li>When the DISEL bit is 1 and the data transfer has ended</li> <li>When the specified number of transfers have ended</li> </ul> |
| 0   | DTCE0    | 0             | R/W |                                                                                                                                                                      |
|     |          |               |     | These bits are not cleared when the DISEL bit is 0 and the specified number of transfers have not been completed                                                     |

## 8.2.8 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.

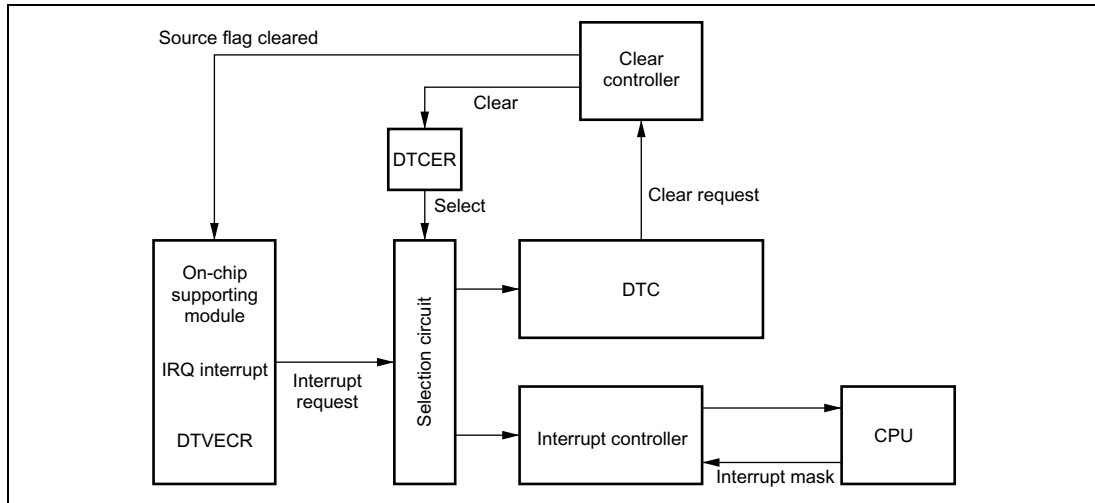
| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | SWDTE    | 0             | R/W | <p>DTC Software Activation Enable</p> <p>Setting this bit to 1 activates DTC. Only 1 can be written to this bit.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• When the DISEL bit is 0 and the specified number of transfers have not ended</li> <li>• When 0 s written to the DISEL bit after a software-activated data transfer end interrupt (SWDTEND) request has been sent to the CPU.</li> </ul> <p>When the DISEL bit is 1 and data transfer has ended or when the specified number of transfers have ended, this bit will not be cleared.</p> |
| 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 + (vector number × 2). For example, when DTVEC6 to DTVEC0 = H'10, the vector address is H'0420. When the bit SWDTE is 0, these bits can be written.                                                                                                                                                                                                                                                                                                                                                                                     |
| 2   | DTVEC2   | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 1   | DTVEC1   | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 0   | DTVEC0   | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |

### 8.3 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 RXI\_0, for example, is the RDRF flag of SCI\_0.

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 8-2 shows a block diagram of activation source control. For details, see section 5, Interrupt Controller.



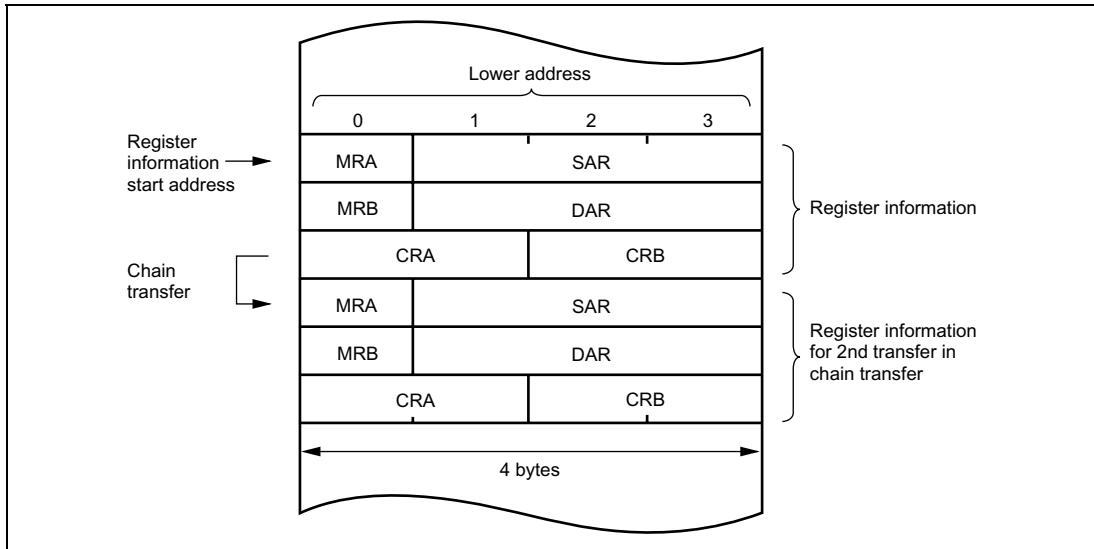
**Figure 8-2 Block Diagram of DTC Activation Source Control**

## 8.4 Location of Register Information and DTC Vector Table

Locate the register information in the on-chip RAM (addresses: H'FFEB0 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 8-3. 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 and the register information start address should be located at the vector address corresponding to the interrupt source. 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.



**Figure 8-3 Correspondence between DTC Vector Address and Register Information**

**Table 8-1 Interrupt Sources, DTC Vector Addresses, and Corresponding DTCEs**

| Interrupt Source | Origin of Interrupt Source | Vector Number | DTC Vector Address           | DTCE*  | 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 |          |
|                  | Reserved for               | 22            | H'042C                       | DTCEA1 |          |
|                  | System use                 | 23            | H'042E                       | DTCEA0 |          |
| A/D              | ADI (A/D conversion end)   | 28            | H'0438                       | DTCEB6 | ↑        |
| TPU channel 0    | TGIA_0                     | 32            | H'0440                       | DTCEB5 |          |
|                  | TGIB_0                     | 33            | H'0442                       | DTCEB4 |          |
|                  | TGIC_0                     | 34            | H'0444                       | DTCEB3 |          |
|                  | TGID_0                     | 35            | H'0446                       | DTCEB2 |          |
| TPU channel 1    | TGIA_1                     | 40            | H'0450                       | DTCEB1 |          |
|                  | TGIB_1                     | 41            | H'0452                       | DTCEB0 |          |
| TPU channel 2    | TGIA_2                     | 44            | H'0458                       | DTCEC7 |          |
|                  | TGIB_2                     | 45            | H'045A                       | DTCEC6 |          |
| TPU channel 3    | TGIA_3                     | 48            | H'0460                       | DTCEC5 |          |
|                  | TGIB_3                     | 49            | H'0462                       | DTCEC4 |          |
|                  | TGIC_3                     | 50            | H'0464                       | DTCEC3 |          |
|                  | TGID_3                     | 51            | H'0466                       | DTCEC2 |          |
| TPU channel 4    | TGIA_4                     | 56            | H'0470                       | DTCEC1 |          |
|                  | TGIB_4                     | 57            | H'0472                       | DTCEC0 |          |
| TPU channel 5    | TGIA_5                     | 60            | H'0478                       | DTCED5 |          |
|                  | TGIB_5                     | 61            | H'047A                       | DTCED4 | Low      |

| Interrupt Source | Origin of Interrupt Source | Vector Number | DTC Vector Address | DTCE*  | Priority  |
|------------------|----------------------------|---------------|--------------------|--------|-----------|
|                  | Reserved for system use    | 64            | H'0480             | DTCED3 | High<br>↑ |
|                  |                            | 65            | H'0482             | DTCED2 |           |
|                  |                            | 68            | H'0488             | DTCED1 |           |
|                  |                            | 69            | H'048A             | DTCED0 |           |
|                  |                            | 72            | H'0490             | DTCEE7 |           |
|                  |                            | 73            | H'0492             | DTCEE6 |           |
|                  |                            | 74            | H'0494             | DTCEE5 |           |
|                  |                            | 75            | H'0496             | DTCEE4 |           |
| SCI channel 0    | RXI_0                      | 81            | H'04A2             | DTCEE3 | ↑         |
|                  | TXI_0                      | 82            | H'04A4             | DTCEE2 |           |
| SCI channel 1    | RXI_1                      | 85            | H'04AA             | DTCEE1 |           |
|                  | TXI_1                      | 86            | H'04AC             | DTCEE0 |           |
| SCI channel 2    | RXI_2                      | 89            | H'04B2             | DTCEF7 |           |
|                  | TXI_2                      | 90            | H'04B4             | DTCEF6 |           |
| HCAN             | Reserved for system use    | 104           | H'04D0             | DTCEG7 |           |
|                  | RM0                        | 105           | H'04D2             | DTCEG6 |           |
|                  | Reserved for system use    | 106           | H'04D4             | DTCEG5 |           |
|                  | Reserved for system use    | 107           | H'04D6             | DTCEG4 |           |
| MMT              | TGIMN                      | 108           | H'04D8             | DTCEG3 |           |
|                  | TGINN                      | 109           | H'04DA             | DTCEG2 |           |
| —                | Reserved for system use    | 110           | H'04DC             | DTCEG1 |           |
|                  | Reserved for system use    | 111           | H'04DE             | DTCEG0 | Low       |

Note: \*DTCE bits with no corresponding interrupt are reserved, and should be written with 0.

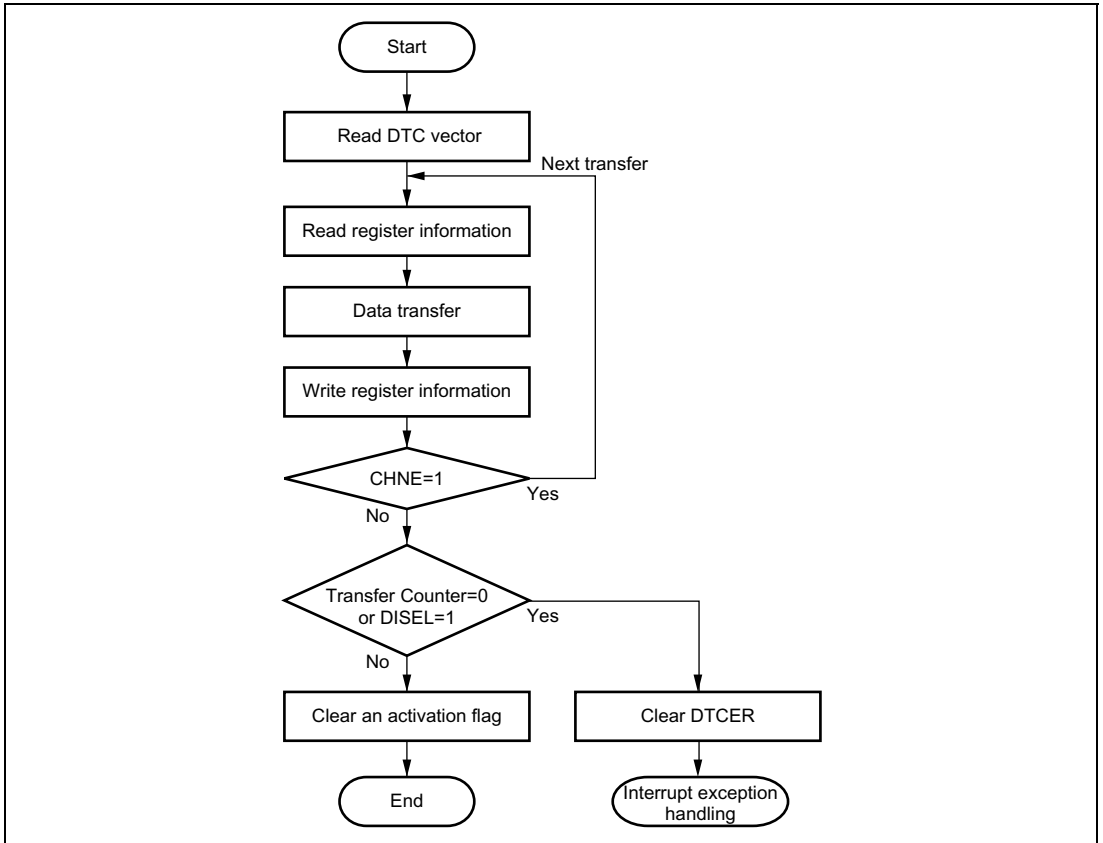
## 8.5 Operation

Register information is stored in on-chip memory. When activated, the DTC reads register information in on-chip memory 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 8-4 Flowchart of DTC Operation**

### 8.5.1 Normal Mode

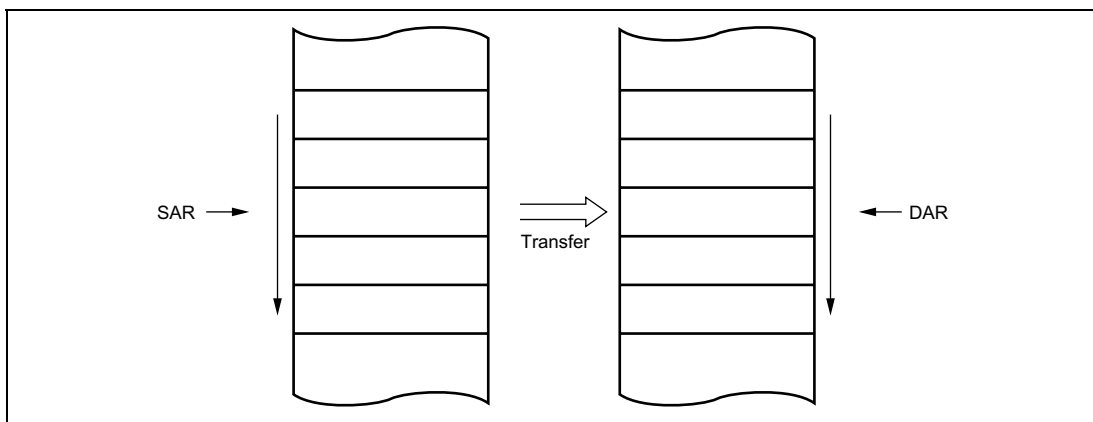
In normal mode, one operation transfers one byte or one word of data.

Table 8-2 lists the register information in normal mode.

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 8-2 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 8-5 Memory Mapping in Normal Mode**

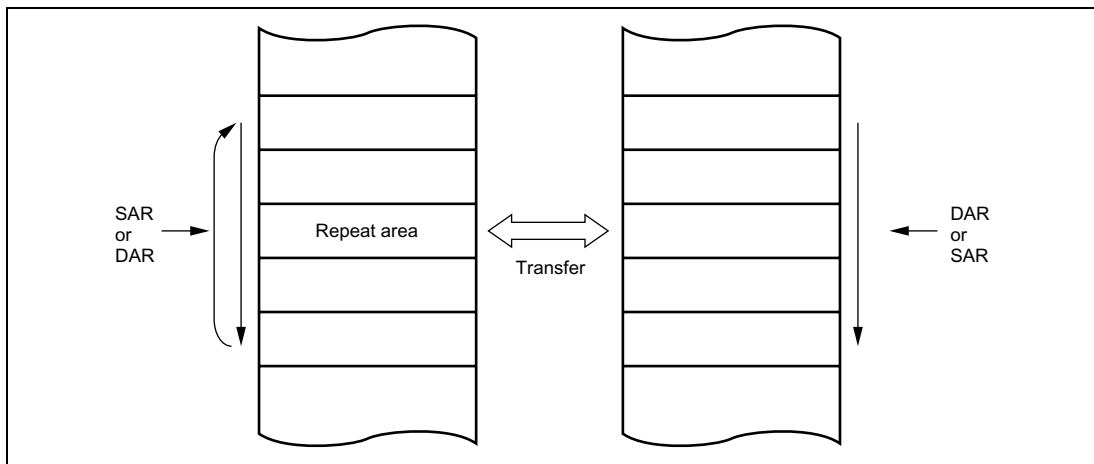
## 8.5.2 Repeat Mode

In repeat mode, one operation transfers one byte or one word of data. Table 8-3 lists the register information in repeat mode.

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 8-3 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 8-6 Memory Mapping in Repeat Mode**

### 8.5.3 Block Transfer Mode

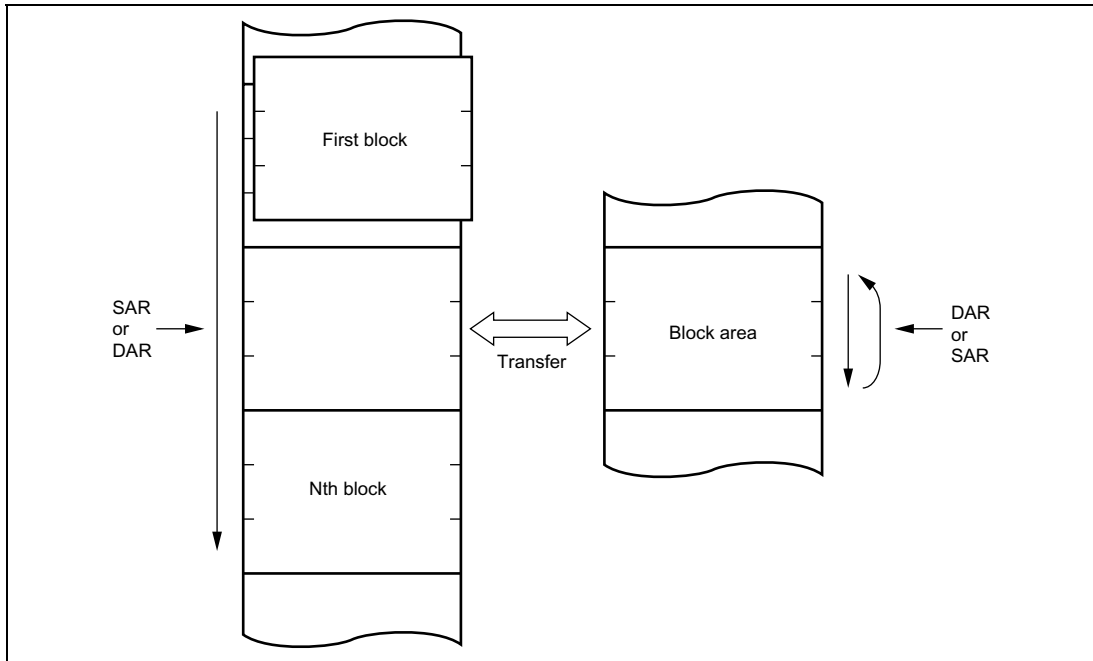
In block transfer mode, one operation transfers one block of data. Either the transfer source or the transfer destination is designated as a block area. Table 8-4 lists the register information in block transfer mode.

The block size can be between 1 and 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 8-4 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 8-7 Memory Mapping in Block Transfer Mode**

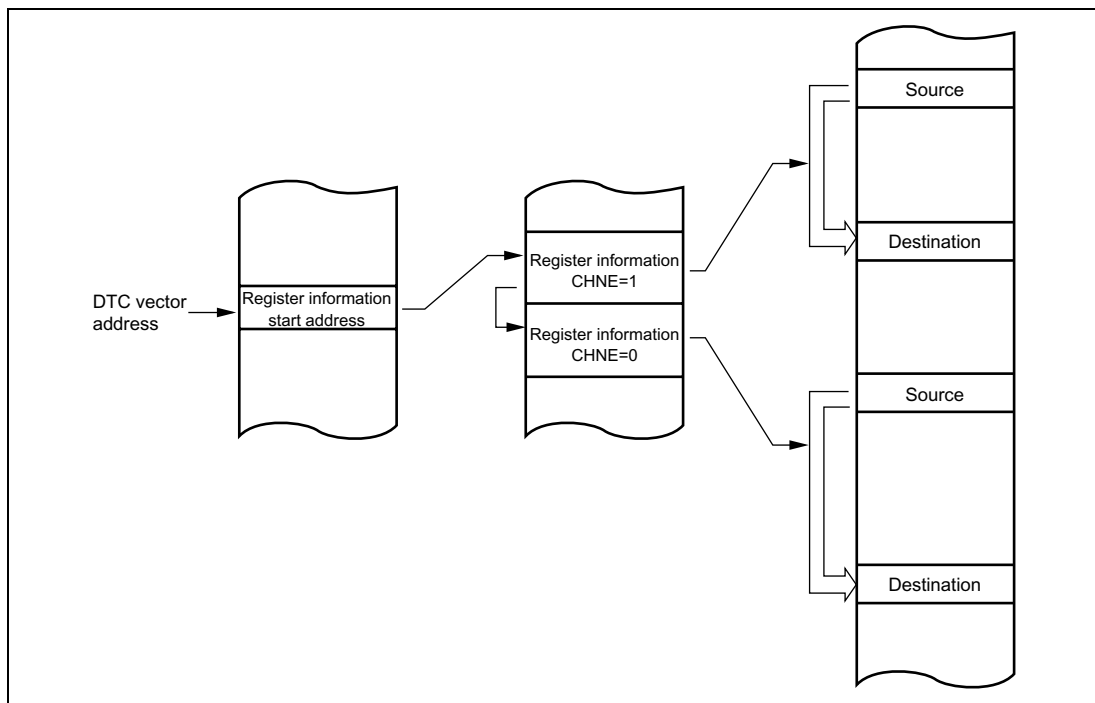
## 8.5.4 Chain Transfer

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 8-8 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 8-8 Chain Transfer Operation**

## 8.5.5 Interrupts

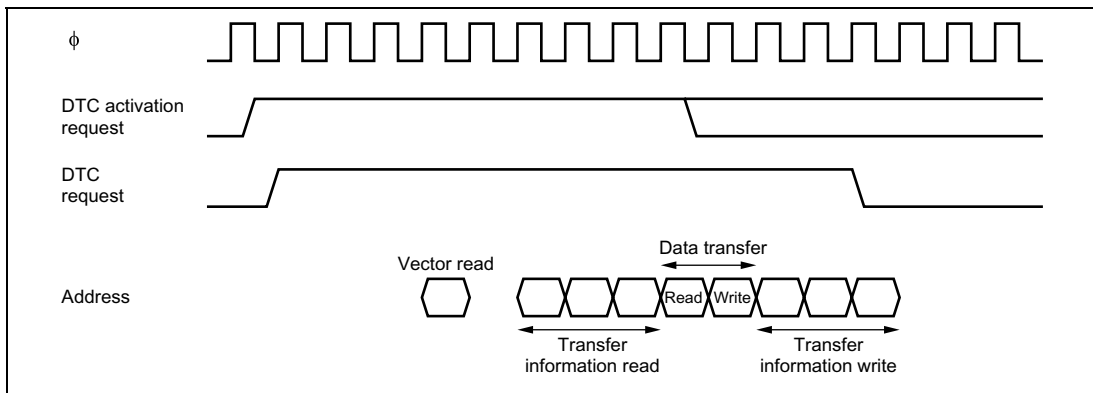
An interrupt request is issued to the CPU when the DTC has completed the specified number of data transfers, or a data transfer for which the DISEL bit was set to 1. In the case of interrupt activation, the interrupt set as the activation source is generated. These interrupts to the CPU are subject to 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.

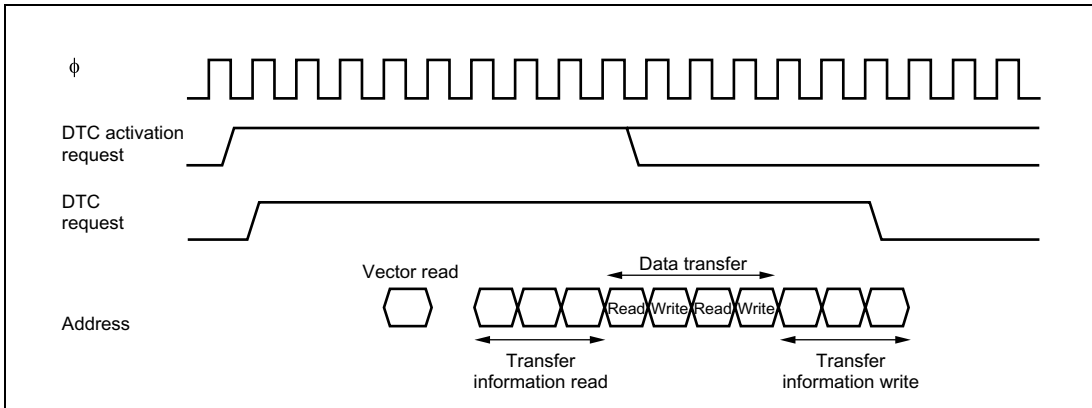
When the DISEL bit is 1 and one data transfer has been completed, or the specified number of transfers have been completed, after data transfer ends the SWDTE bit is held at 1 and an SWDTEND interrupt is generated. The interrupt handling routine will then clear the SWDTE bit to 0.

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.

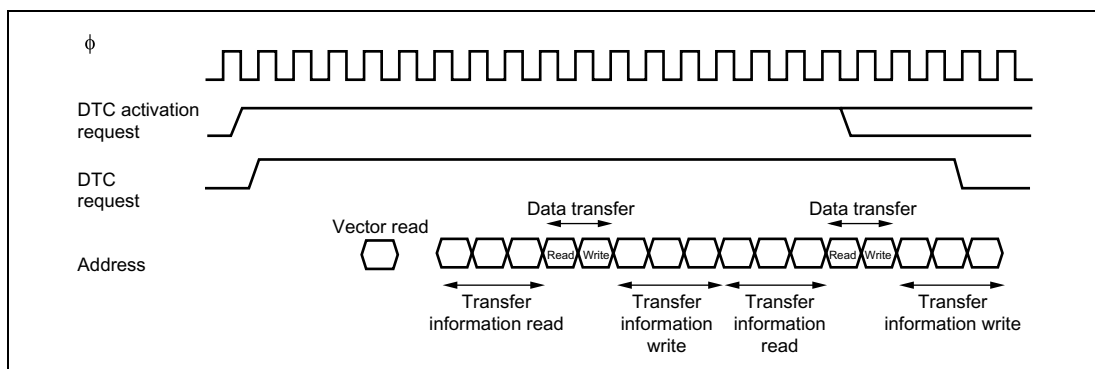
## 8.5.6 Operation Timing



**Figure 8-9 DTC Operation Timing (Example in Normal Mode or Repeat Mode)**



**Figure 8-10 DTC Operation Timing (Example of Block Transfer Mode, with Block Size of 2)**



**Figure 8-11 DTC Operation Timing (Example of Chain Transfer)**

### 8.5.7 Number of DTC Execution States

Table 8-5 lists execution status for a single DTC data transfer, and table 8-6 shows the number of states required for each execution status.

**Table 8-5 DTC Execution Status**

| Mode           | Vector Read<br>I | Register Information<br>Read/Write<br>J | Data Read<br>K | Data Write<br>L | Internal<br>Operations<br>M |
|----------------|------------------|-----------------------------------------|----------------|-----------------|-----------------------------|
| 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 8-6 Number of States Required for Each Execution Status**

| Object to be Accessed    |                                          | On-<br>Chip<br>RAM | On-<br>Chip<br>ROM | On-Chip I/O<br>Registers |    | External Devices* |      |   |     |
|--------------------------|------------------------------------------|--------------------|--------------------|--------------------------|----|-------------------|------|---|-----|
| Bus width                |                                          | 32                 | 16                 | 8                        | 16 | 8                 | 16   |   |     |
| Access states            |                                          | 1                  | 1                  | 2                        | 2  | 2                 | 3    | 2 | 3   |
| Execution<br>status      | Vector read $S_I$                        | —                  | 1                  | —                        | —  | 4                 | 6+2m | 2 | 3+m |
|                          | Register information<br>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                  |                    |                          |    |                   |      |   |     |

Note:\* Cannot be used in this LSI.

The number of execution states is calculated from using the formula below. Note that  $\Sigma$  is the sum of all transfers activated by one activation event (the number in which the CHNE bit is set to 1, plus 1).

$$\text{Number of execution states} = I \cdot (1 + S_I) + \Sigma (J \cdot S_J + K \cdot S_K + L \cdot S_L) + M \cdot S_M$$

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.

## 8.6 Procedures for Using DTC

### 8.6.1 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.

### 8.6.2 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.

## 8.7 Examples of Use of the DTC

### 8.7.1 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.

### 8.7.2 Chain Transfer

An example of DTC chain transfer is shown in which pulse output is performed using the PPG. Chain transfer can be used to perform pulse output data transfer and PPG output trigger cycle updating. Repeat mode transfer to the PPG's NDR is performed in the first half of the chain transfer, and normal mode transfer to the TPU's TGR in the second half. This is because clearing of the activation source and interrupt generation at the end of the specified number of transfers are restricted to the second half of the chain transfer (transfer when CHNE = 0).

1. Perform settings for transfer to the PPG's NDR. Set MRA to incrementing source address (SM1 = 1, SM0 = 0), a fixed destination address (DM1 = DM0 = 0), repeat mode (MD1 = 0, MD0 = 1), and word size (Sz = 1). Set the source side as a repeat area (DTS = 1). Set MRB to chain mode (CHNE = 1, DISEL = 0). Set the data table start address in SAR, the NDRH address in DAR, and the data table size in CRAH and CRAL. CRB can be set to any value.
2. Perform settings for transfer to the TPU's TGR. Set MRA to incrementing source address (SM1 = 1, SM0 = 0), a fixed destination address (DM1 = DM0 = 0), normal mode (MD1 = MD0 = 0), and word size (Sz = 1). Set the data table start address in SAR, the TGRA address in DAR, and the data table size in CRA. CRB can be set to any value.
3. Locate the TPU transfer register information consecutively after the NDR transfer register information.
4. Set the start address of the NDR transfer register information to the DTC vector address.
5. Set the bit corresponding to TGIA in DTCER to 1.
6. Set TGRA as an output compare register (output disabled) with TIOR, and enable the TGIA interrupt with TIER.
7. Set the initial output value in PODR, and the next output value in NDR. Set bits in DDR and NDER for which output is to be performed to 1. Using PCR, select the TPU compare match to be used as the output trigger.
8. Set the CST bit in TSTR to 1, and start the TCNT count operation.

9. Each time a TGRA compare match occurs, the next output value is transferred to NDR and the set value of the next output trigger period is transferred to TGRA. The activation source TGFA flag is cleared.
10. When the specified number of transfers are completed (the TPU transfer CRA value is 0), the TGFA flag is held at 1, the DTCE bit is cleared to 0, and a TGIA interrupt request is sent to the CPU. Termination processing should be performed in the interrupt handling routine.

### 8.7.3 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.

## 8.8 Usage Notes

### 8.8.1 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 halted. Register access is enabled by clearing module stop mode. For details, refer to section 20, Power-Down Modes.

### 8.8.2 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 must not be cleared to 0.

### 8.8.3 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 9 I/O Ports

Table 9-1 summarizes the port functions. The pins of each port also have other functions such as input/output or interrupt input pins of on-chip supporting 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 a DR or DDR register.

Ports A to D have a built-in input pull-up MOS function and a MOS input pull-up control register (PCR) to control the on/off state of MOS input pull-up.

Ports A to C include an open-drain control register (ODR) that controls the on/off state of the output buffer PMOS.

All the I/O ports can drive a single TTL load and a 30 pF capacitive load.

**Table 9-1 Port Functions**

| Port   | Description                                                                                  | Port and Other Functions Name | Input/Output and Output Type              |
|--------|----------------------------------------------------------------------------------------------|-------------------------------|-------------------------------------------|
| Port 1 | General I/O port also functioning as TPU I/O pins, PPG output pins, and interrupt input pins | P17/PO15/TIOCB2/TCLKD         |                                           |
|        |                                                                                              | P16/PO14/TIOCA2/IRQ1          |                                           |
|        |                                                                                              | P15/PO13/TIOCB1/TCLKC         |                                           |
|        |                                                                                              | P14/PO12/TIOCA1/IRQ0          |                                           |
|        |                                                                                              | P13/PO11/TIOCD0/TCLKB         |                                           |
|        |                                                                                              | P12/PO10/TIOCC0/TCLKA         |                                           |
|        |                                                                                              | P11/PO9/TIOCB0                |                                           |
|        |                                                                                              | P10/PO8/TIOCA0                |                                           |
| Port 4 | General input port also functioning as A/D converter analog inputs                           | P47/AN7                       |                                           |
|        |                                                                                              | P46/AN6                       |                                           |
|        |                                                                                              | P45/AN5                       |                                           |
|        |                                                                                              | P44/AN4                       |                                           |
|        |                                                                                              | P43/AN3                       |                                           |
|        |                                                                                              | P42/AN2                       |                                           |
|        |                                                                                              | P41/AN1                       |                                           |
|        |                                                                                              | P40/AN0                       |                                           |
| Port 9 | General input port also functioning as A/D converter analog inputs                           | P93/AN11                      |                                           |
|        |                                                                                              | P92/AN10                      |                                           |
|        |                                                                                              | P91/AN9                       |                                           |
|        |                                                                                              | P90/AN8                       |                                           |
| Port A | General I/O port also functioning as SCI_2 I/O pins, POE input pins                          | PA3/SCK2/POE3                 | Built-in MOS input pull-up                |
|        |                                                                                              | PA2/RxD2/POE2                 | Push-pull or open-drain output selectable |
|        |                                                                                              | PA1/TxD2/POE1                 |                                           |
|        |                                                                                              | PA0/POE0                      |                                           |
| Port B | General I/O port also functioning as TPU_5, TPU_4, and TPU_3 I/O pins, and MMT I/O pins      | PB7/TIOCB5/PWOB               | Built-in MOS input pull-up                |
|        |                                                                                              | PB6/TIOCA5/PWOA               | Push-pull or open-drain output selectable |
|        |                                                                                              | PB5/TIOCB4/PVOB               |                                           |
|        |                                                                                              | PB4/TIOCA4/PVOA               |                                           |
|        |                                                                                              | PB3/TIOCD3/PUOB               |                                           |
|        |                                                                                              | PB2/TIOCC3/PUOA               |                                           |
|        |                                                                                              | PB1/TIOCB3/PCO                |                                           |
|        |                                                                                              | PB0/TIOCA3/PCI                |                                           |

| Port   | Description                                                                                                                                   | Port and Other Functions Name | Input/Output and Output Type              |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|-------------------------------------------|
| Port C | General I/O port also functioning as SCL_1 and SCL_0 I/O pins, and interrupt input pins                                                       | PC7                           | Built-in MOS input pull-up                |
|        |                                                                                                                                               | PC6                           | Push-pull or open-drain output selectable |
|        |                                                                                                                                               | PC5/SCK1/IRQ5                 |                                           |
|        |                                                                                                                                               | PC4/RxD1                      |                                           |
|        |                                                                                                                                               | PC3/TxD1                      |                                           |
|        |                                                                                                                                               | PC2/SCK0/IRQ4                 |                                           |
|        |                                                                                                                                               | PC1/RxD0                      |                                           |
|        |                                                                                                                                               | PC0/TxD0                      |                                           |
| Port D | General I/O port                                                                                                                              | PD7                           | Built-in MOS input pull-up                |
|        |                                                                                                                                               | PD6                           |                                           |
|        |                                                                                                                                               | PD5                           |                                           |
|        |                                                                                                                                               | PD4                           |                                           |
|        |                                                                                                                                               | PD3                           |                                           |
|        |                                                                                                                                               | PD2                           |                                           |
|        |                                                                                                                                               | PD1                           |                                           |
|        |                                                                                                                                               | PD0                           |                                           |
| Port F | General I/O port also functioning as interrupt input pins, an A/D converter start trigger input pin, and a system clock output pin ( $\phi$ ) | PF7/ $\phi$                   |                                           |
|        |                                                                                                                                               | PF6                           |                                           |
|        |                                                                                                                                               | PF5                           |                                           |
|        |                                                                                                                                               | PF4                           |                                           |
|        |                                                                                                                                               | PF3/ADTRG/IRQ3                |                                           |
|        |                                                                                                                                               | PF2                           |                                           |
|        |                                                                                                                                               | PF1                           |                                           |
|        |                                                                                                                                               | PF0/IRQ2                      |                                           |

## 9.1 Port 1

Port 1 is an 8-bit I/O port and has the following registers. For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register.

- Port 1 data direction register (P1DDR)
- Port 1 data register (P1DR)
- Port 1 register (PORT1)

### 9.1.1 Port 1 Data Direction Register (P1DDR)

P1DDR is an 8-bit write-only register, the individual bits of which specify input or output for the pins of port 1.

P1DDR cannot be read; if it is, an undefined value will be read.

| 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   |                                                                                                                                                                                   |

### 9.1.2 Port 1 Data Register (P1DR)

P1DR is an 8-bit readable/writable register that 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 I/O 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 |                                                                                          |

### 9.1.3 Port 1 Register (PORT1)

PORT1 is an 8-bit read-only register that shows the pin states.

PORT1 cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | P17      | Undefined*    | 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      | Undefined*    | R   |                                                                                                                                                                                  |
| 5   | P15      | Undefined*    | R   |                                                                                                                                                                                  |
| 4   | P14      | Undefined*    | R   |                                                                                                                                                                                  |
| 3   | P13      | Undefined*    | R   |                                                                                                                                                                                  |
| 2   | P12      | Undefined*    | R   |                                                                                                                                                                                  |
| 1   | P11      | Undefined*    | R   |                                                                                                                                                                                  |
| 0   | P10      | Undefined*    | R   |                                                                                                                                                                                  |

Note:\* Determined by the states of pins P17 to P10.

### 9.1.4 Pin Functions

Port 1 pins also function as TPU I/O pins, PPG output pins, and interrupt input pins. The correspondence between the register specification and the pin functions is shown below.

**Table 9-2 P17 Pin Function**

| TPU Channel<br>2 Setting* | Output        | Input or Initial Value                   |            |             |
|---------------------------|---------------|------------------------------------------|------------|-------------|
| P17DDR                    | —             | 0                                        | 1          | 1           |
| NDER15                    | —             | —                                        | 0          | 1           |
| Pin function              | TIOCB2 output | P17 input<br>TIOCB2 input<br>TCLKD input | P17 output | PO15 output |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-3 P16 Pin Function**

| TPU Channel<br>2 Setting* | Output        | Input or Initial Value                  |            |             |
|---------------------------|---------------|-----------------------------------------|------------|-------------|
| P16DDR                    | —             | 0                                       | 1          | 1           |
| NDER14                    | —             | —                                       | 0          | 1           |
| Pin function              | TIOCA2 output | P16 input<br>TIOCA2 input<br>IRQ1 input | P16 output | PO14 output |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-4 P15 Pin Function**

| TPU Channel<br>1 Setting* | Output        | Input or Initial Value                   |            |             |
|---------------------------|---------------|------------------------------------------|------------|-------------|
| P15DDR                    | —             | 0                                        | 1          | 1           |
| NDER13                    | —             | —                                        | 0          | 1           |
| Pin function              | TIOCB1 output | P15 input<br>TIOCB1 input<br>TCLKC input | P15 output | PO13 output |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-5 P14 Pin Function**

| TPU Channel<br>1 Setting* | Output        | Input or Initial Value |            |             |
|---------------------------|---------------|------------------------|------------|-------------|
| P14DDR                    | —             | 0                      | 1          | 1           |
| NDER12                    | —             | —                      | 0          | 1           |
| Pin function              | TIOCA1 output | P14 input              | P14 output | PO12 output |
|                           |               | TIOCA1 input           |            |             |
|                           |               | IRQ0 input             |            |             |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-6 P13 Pin Function**

| TPU Channel<br>0 Setting* | Output        | Input or Initial Value |            |             |
|---------------------------|---------------|------------------------|------------|-------------|
| P13DDR                    | —             | 0                      | 1          | 1           |
| NDER11                    | —             | —                      | 0          | 1           |
| Pin function              | TIOCD0 output | P13 input              | P13 output | PO11 output |
|                           |               | TIOCD0 input           |            |             |
|                           |               | TCLKB input            |            |             |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-7 P12 Pin Function**

| TPU Channel<br>0 Setting* | Output        | Input or Initial Value |            |             |
|---------------------------|---------------|------------------------|------------|-------------|
| P12DDR                    | —             | 0                      | 1          | 1           |
| NDER10                    | —             | —                      | 0          | 1           |
| Pin function              | TIOCC0 output | P12 input              | P12 output | PO10 output |
|                           |               | TIOCC0 input           |            |             |
|                           |               | TCLKA input            |            |             |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-8 P11 Pin Function**

| TPU Channel<br>0 Setting* | Output        | Input or Initial Value    |            |            |
|---------------------------|---------------|---------------------------|------------|------------|
| P11DDR                    | —             | 0                         | 1          | 1          |
| NDER9                     | —             | —                         | 0          | 1          |
| Pin function              | TIOCB0 output | P11 input<br>TIOCB0 input | P11 output | PO9 output |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-9 P10 Pin Function**

| TPU Channel<br>0 Setting* | Output        | Input or Initial Value    |            |            |
|---------------------------|---------------|---------------------------|------------|------------|
| P10DDR                    | —             | 0                         | 1          | 1          |
| NDER8                     | —             | —                         | 0          | 1          |
| Pin function              | TIOCA0 output | P10 input<br>TIOCA0 input | P10 output | PO8 output |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

## 9.2 Port 4

Port 4 is an 8-bit input-only port. Port 4 pins also function as A/D converter analog input pins. For details on register addresses, refer to Appendix A, On-Chip I/O Register.

- Port 4 register (PORT4)

### 9.2.1 Port 4 Register (PORT4)

PORT4 is an 8-bit read-only register that shows port 4 pin states.

PORT4 cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------|
| 7   | P47      | Undefined*    | R   | The pin states are always read when a port 4 read is performed. |
| 6   | P46      | Undefined*    | R   |                                                                 |
| 5   | P45      | Undefined*    | R   |                                                                 |
| 4   | P44      | Undefined*    | R   |                                                                 |
| 3   | P43      | Undefined*    | R   |                                                                 |
| 2   | P42      | Undefined*    | R   |                                                                 |
| 1   | P41      | Undefined*    | R   |                                                                 |
| 0   | P40      | Undefined*    | R   |                                                                 |

Note:\* Determined by the states of pins P47 to P40.

## 9.3 Port 9

Port 9 is a 4-bit input-only port. Port 9 pins also function as A/D converter analog input pins. Port 9 has the following registers. For details on register addresses, refer to Appendix A, On-Chip I/O Register.

- Port 9 register (PORT9)

### 9.3.1 Port 9 Register (PORT9)

PORT9 is an 8-bit read-only register that shows port 9 pin states.

PORT9 cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------|
| 7   | —        | Undefined     | R   | The pin states are always read when a port 9 read is performed. |
| 6   | —        | Undefined     | R   |                                                                 |
| 5   | —        | Undefined     | R   |                                                                 |
| 4   | —        | Undefined     | R   |                                                                 |
| 3   | P93      | Undefined*    | R   |                                                                 |
| 2   | P92      | Undefined*    | R   |                                                                 |
| 1   | P91      | Undefined*    | R   |                                                                 |
| 0   | P90      | Undefined*    | R   |                                                                 |

Note:\* Determined by the states of pins P93 to P90.

## 9.4 Port A

Port A is a 4-bit I/O port that also has other functions. Port A has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O 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)

### 9.4.1 Port A Data Direction Register (PADDR)

PADDR is an 8-bit write-only register, the individual bits of which specify whether the pins of port A are used for input or output. PADDR cannot be read; if it is, an undefined value will be read.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                       |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | Undefined     | —   | Reserved                                                                                                                                                                          |
| 6   | —        | Undefined     | —   |                                                                                                                                                                                   |
| 5   | —        | Undefined     | —   |                                                                                                                                                                                   |
| 4   | —        | Undefined     | —   |                                                                                                                                                                                   |
| 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   |                                                                                                                                                                                   |

### 9.4.2 Port A Data Register (PADR)

PADR is an 8-bit readable/writable register that stores output data for port A pins.

| Bit | Bit Name | Initial Value | R/W | Description                                                                              |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------|
| 7   | —        | Undefined     | —   | Reserved                                                                                 |
| 6   | —        | Undefined     | —   | These bits are read as an undefined value.                                               |
| 5   | —        | Undefined     | —   |                                                                                          |
| 4   | —        | Undefined     | —   |                                                                                          |
| 3   | PA3DR    | 0             | R/W | Output data for a pin is stored when the pin is specified as a general purpose I/O port. |
| 2   | PA2DR    | 0             | R/W |                                                                                          |
| 1   | PA1DR    | 0             | R/W |                                                                                          |
| 0   | PA0DR    | 0             | R/W |                                                                                          |

### 9.4.3 Port A Register (PORTA)

PORTA is an 8-bit read-only register that shows port A pin states.

PORTA cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | Undefined     | —   | Reserved                                                                                                                                                                         |
| 6   | —        | Undefined     | —   | These bits are read as an undefined value.                                                                                                                                       |
| 5   | —        | Undefined     | —   |                                                                                                                                                                                  |
| 4   | —        | Undefined     | —   |                                                                                                                                                                                  |
| 3   | PA3      | Undefined*    | R   | If a port A read is performed while PADDR bits are set to 1, the PADR values are read. If a port A read is performed while PADDR bits are cleared to 0, the pin states are read. |
| 2   | PA2      | Undefined*    | R   |                                                                                                                                                                                  |
| 1   | PA1      | Undefined*    | R   |                                                                                                                                                                                  |
| 0   | PA0      | Undefined*    | R   |                                                                                                                                                                                  |

Note:\* Determined by the states of pins PA3 to PA0.

### 9.4.4 Port A Pull-Up MOS Control Register (PAPCR)

PAPCR is an 8-bit register that controls the MOS input pull-up function.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                               |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | Undefined     | —   | Reserved                                                                                                                  |
| 6   | —        | Undefined     | —   | These bits are read as an undefined value.                                                                                |
| 5   | —        | Undefined     | —   |                                                                                                                           |
| 4   | —        | Undefined     | —   |                                                                                                                           |
| 3   | PA3PCR   | 0             | R/W | When a pin is specified as an input port, setting the corresponding bit to 1 turns on the MOS input pull-up for that pin. |
| 2   | PA2PCR   | 0             | R/W |                                                                                                                           |
| 1   | PA1PCR   | 0             | R/W |                                                                                                                           |
| 0   | PA0PCR   | 0             | R/W |                                                                                                                           |

### 9.4.5 Port A Open-Drain Control Register (PAODR)

PAODR is an 8-bit read/write register that specifies the output type of port A.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                             |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | Undefined     | —   | Reserved                                                                                                                                                                                                                |
| 6   | —        | Undefined     | —   | These bits are read as an undefined value.                                                                                                                                                                              |
| 5   | —        | Undefined     | —   |                                                                                                                                                                                                                         |
| 4   | —        | Undefined     | —   |                                                                                                                                                                                                                         |
| 3   | PA3ODR   | 0             | R/W | When a pin is specified as an output port, setting the corresponding bit to 1 specifies pin output to open-drain and the MOS input pull-up to the off state. Clearing this bit to 0 specifies that to push-pull output. |
| 2   | PA2ODR   | 0             | R/W |                                                                                                                                                                                                                         |
| 1   | PA1ODR   | 0             | R/W |                                                                                                                                                                                                                         |
| 0   | PA0ODR   | 0             | R/W |                                                                                                                                                                                                                         |

### 9.4.6 Pin Functions

Port A pins also function as SCI\_2 I/O, interrupt input, and POE input pins. The correspondence between the register specification and the pin functions is shown below.

**Table 9-10 PA3 Pin Function**

|              |           |            |             |             |            |            |
|--------------|-----------|------------|-------------|-------------|------------|------------|
| POE3E        | 0         |            |             |             | 1          |            |
| CKE1         | 0         |            |             | 1           | —          |            |
| C/A          | 0         |            | 1           | —           | —          |            |
| CKE0         | 0         | 1          | —           | —           | —          |            |
| PA3DDR       | 0         | 1          | —           | —           | —          | —          |
| Pin function | PA3 input | PA3 output | SCK2 output | SCK2 output | SCK2 input | POE3 input |

**Table 9-11 PA2 Pin Function**

|              |           |            |   |            |            |  |
|--------------|-----------|------------|---|------------|------------|--|
| POE2E        | 0         |            |   |            | 1          |  |
| RE           | 0         |            |   | 1          | —          |  |
| PA2DDR       | 0         | 1          | — | —          | —          |  |
| Pin function | PA2 input | PA2 output |   | RxD2 input | POE2 input |  |

**Table 9-12 PA1 Pin Function**

|              |           |            |   |             |            |  |
|--------------|-----------|------------|---|-------------|------------|--|
| POE1E        | 0         |            |   |             | 1          |  |
| TE           | 0         |            |   | 1           | —          |  |
| PA1DDR       | 0         | 1          | — | —           | —          |  |
| Pin function | PA1 input | PA1 output |   | TxD2 output | POE1 input |  |

**Table 9-13 PA0 Pin Function**

|              |           |            |   |   |            |  |
|--------------|-----------|------------|---|---|------------|--|
| POE0E        | 0         |            |   |   | 1          |  |
| PA0DDR       | 0         | 1          | — | — | —          |  |
| Pin function | PA0 input | PA0 output |   |   | POE0 input |  |

## 9.5 Port B

Port B is an 8-bit I/O port that also has other functions. Port B has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

- Port B data direction register (PBDDR)
- Port B data register (PBDR)
- Port B register (PORTB)
- Port B pull-up MOS control register (PBPCR)
- Port B open-drain control register (PBODR)

### 9.5.1 Port B Data Direction Register (PBDDR)

PBDDR is an 8-bit write-only register, the individual bits of which specify whether the pins of port B are used for input or output.

PBDDR cannot be read; if it is, an undefined value will be read.

| 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 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   | 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   |                                                                                                                                                                                   |

### 9.5.2 Port B Data Register (PBDR)

PBDR is an 8-bit readable/writable register that stores output data for the 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 I/O 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 |                                                                                          |

### 9.5.3 Port B Register (PORTB)

PORTB is an 8-bit read-only register that shows port B pin states.

PORTB cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PB7      | Undefined*    | R   | If a port B read is performed while PBDDR bits are set to 1, the PBDR values are read. If a port B read is performed while PBDDR bits are cleared to 0, the pin states are read. |
| 6   | PB6      | Undefined*    | R   |                                                                                                                                                                                  |
| 5   | PB5      | Undefined*    | R   |                                                                                                                                                                                  |
| 4   | PB4      | Undefined*    | R   |                                                                                                                                                                                  |
| 3   | PB3      | Undefined*    | R   |                                                                                                                                                                                  |
| 2   | PB2      | Undefined*    | R   |                                                                                                                                                                                  |
| 1   | PB1      | Undefined*    | R   |                                                                                                                                                                                  |
| 0   | PB0      | Undefined*    | R   |                                                                                                                                                                                  |

Note:\* Determined by the states of pins PB7 to PB0.

### 9.5.4 Port B Pull-Up MOS Control Register (PBPCR)

PBPCR is an 8-bit read/write register that controls the on/off state of MOS input pull-up of port B.

| 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 MOS input pull-up 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 |                                                                                                                           |

### 9.5.5 Port B Open-Drain Control Register (PBODR)

PBODR is an 8-bit read/write register that specifies the output type of port B.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                               |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PB7ODR   | 0             | R/W | When a pin function is specified as an output port, setting the corresponding bit to 1 specifies pin output as open-drain and the PMOS input pull-up to the off state. Clearing this bit to 0 specifies push-pull output. |
| 6   | PB6ODR   | 0             | R/W |                                                                                                                                                                                                                           |
| 5   | PB5ODR   | 0             | R/W |                                                                                                                                                                                                                           |
| 4   | PB4ODR   | 0             | R/W |                                                                                                                                                                                                                           |
| 3   | PB3ODR   | 0             | R/W |                                                                                                                                                                                                                           |
| 2   | PB2ODR   | 0             | R/W |                                                                                                                                                                                                                           |
| 1   | PB1ODR   | 0             | R/W |                                                                                                                                                                                                                           |
| 0   | PB0ODR   | 0             | R/W |                                                                                                                                                                                                                           |

### 9.5.6 Pin Functions

Port B pins also function as TPU and MMT I/O pins. The correspondence between the register specification and the pin functions is shown below.

**Table 9-14 PB7 Pin Function**

|                        |               |                        |            |             |   |
|------------------------|---------------|------------------------|------------|-------------|---|
| PWOBE                  | 0             |                        |            |             | 1 |
| TPU channel 5 setting* | Output        | Input or Initial Value |            |             | — |
| PB7DDR                 | —             | 0                      | 1          | —           |   |
| Pin function           | TIOCB5 output | PB7 input              | PB7 output | PWOB output |   |
|                        |               | TIOCB5 input           |            |             |   |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-15 PB6 Pin Function**

|                        |               |                        |            |             |   |
|------------------------|---------------|------------------------|------------|-------------|---|
| PWOAE                  | 0             |                        |            |             | 1 |
| TPU channel 5 setting* | Output        | Input or Initial Value |            |             | — |
| PB6DDR                 | —             | 0                      | 1          | —           |   |
| Pin function           | TIOCA5 output | PB6 input              | PB6 output | PWOA output |   |
|                        |               | TIOCA5 input           |            |             |   |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-16 PB5 Pin Function**

|                        |               |                        |            |             |   |
|------------------------|---------------|------------------------|------------|-------------|---|
| PVOBE                  | 0             |                        |            |             | 1 |
| TPU channel 4 setting* | Output        | Input or Initial Value |            |             | — |
| PB5DDR                 | —             | 0                      | 1          | —           |   |
| Pin function           | TIOCB4 output | PB5 input              | PB5 output | PVOB output |   |
|                        |               | TIOCB4 input           |            |             |   |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-17 PB4 Pin Function**

|                        |               |                        |            |             |
|------------------------|---------------|------------------------|------------|-------------|
| PVOAE                  | 0             |                        |            | 1           |
| TPU channel 4 setting* | Output        | Input or Initial Value |            | —           |
| PB4DDR                 | —             | 0                      | 1          | —           |
| Pin function           | TIOCA4 output | PB4 input              | PB4 output | PVOA output |
|                        |               | TIOCA4 input           |            |             |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-18 PB3 Pin Function**

|                        |               |                        |            |             |
|------------------------|---------------|------------------------|------------|-------------|
| PUOBE                  | 0             |                        |            | 1           |
| TPU channel 3 setting* | Output        | Input or Initial Value |            | —           |
| PB3DDR                 | —             | 0                      | 1          | —           |
| Pin function           | TIOCD3 output | PB3 input              | PB3 output | PUOB output |
|                        |               | TIOCD3 input           |            |             |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-19 PB2 Pin Function**

|                        |               |                        |            |             |
|------------------------|---------------|------------------------|------------|-------------|
| PUOAE                  | 0             |                        |            | 1           |
| TPU channel 3 setting* | Output        | Input or Initial Value |            | —           |
| PB2DDR                 | —             | 0                      | 1          | —           |
| Pin function           | TIOCC3 output | PB2 input              | PB2 output | PUOA output |
|                        |               | TIOCC3 input           |            |             |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-20 PB1 Pin Function**

|                        |               |                        |            |            |   |
|------------------------|---------------|------------------------|------------|------------|---|
| PCOE                   | 0             |                        |            |            | 1 |
| TPU channel 3 setting* | Output        | Input or Initial Value |            |            | — |
| PB1DDR                 | —             | 0                      | 1          | —          |   |
| Pin function           | TIOCB3 output | PB1 input              | PB1 output | PCO output |   |
|                        |               | TIOCB3 input           |            |            |   |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

**Table 9-21 PB0 Pin Function**

|                        |               |                        |            |                               |   |
|------------------------|---------------|------------------------|------------|-------------------------------|---|
| PCIE                   | 0             |                        |            |                               | 1 |
| TPU channel 3 setting* | Output        | Input or Initial Value |            |                               |   |
| PB0DDR                 | —             | 0                      | 1          | 0                             |   |
| Pin function           | TIOCA3 output | PB0 input              | PB0 output | $\overline{\text{PCI}}$ input |   |
|                        |               | TIOCA3 input           |            |                               |   |

Note:\* For details on the TPU channel specification, refer to section 10, 16-Bit Timer Pulse Unit (TPU).

## 9.6 Port C

Port C is an 8-bit I/O port that also has other functions. Port C has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

- Port C data direction register (PCDDR)
- Port C data register (PCDR)
- Port C register (PORTC)
- Port C MOS pull-up control register (PCPCR)
- Port C open-drain control register (PCODR)

### 9.6.1 Port C Data Direction Register (PCDDR)

PCDDR is an 8-bit write-only register, the individual bits of which specify whether the pins of port C are used for input or output.

PCDDR cannot be read; if it is, an undefined value will be read.

| 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 1 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   |                                                                                                                                                                                   |

### 9.6.2 Port C Data Register (PCDR)

PCDR is an 8-bit readable/writable register that stores output data for the 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 I/O 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 |                                                                                          |

### 9.6.3 Port C Register (PORTC)

PORTC is an 8-bit read-only register that shows port C pin states.

PORTC cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PC7      | Undefined*    | 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      | Undefined*    | R   |                                                                                                                                                                                  |
| 5   | PC5      | Undefined*    | R   |                                                                                                                                                                                  |
| 4   | PC4      | Undefined*    | R   |                                                                                                                                                                                  |
| 3   | PC3      | Undefined*    | R   |                                                                                                                                                                                  |
| 2   | PC2      | Undefined*    | R   |                                                                                                                                                                                  |
| 1   | PC1      | Undefined*    | R   |                                                                                                                                                                                  |
| 0   | PC0      | Undefined*    | R   |                                                                                                                                                                                  |

Note:\* Determined by the states of pins PC7 to PC0.

### 9.6.4 Port C Pull-Up MOS Control Register (PCPCR)

PCPCR is an 8-bit read/write register that controls the on/off state of MOS input pull-up of 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 MOS input pull-up 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 |                                                                                                                           |

### 9.6.5 Port C Open-Drain Control Register (PCODR)

PCODR is an 8-bit read/write register that specifies an output type of port C.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PC7ODR   | 0             | R/W | When a pin is specified as an output port, setting the corresponding bit to 1 specifies pin output as open-drain and the MOS input pull-up to the off state. Clearing this bit to 0 specifies push-pull output. |
| 6   | PC6ODR   | 0             | R/W |                                                                                                                                                                                                                 |
| 5   | PC5ODR   | 0             | R/W |                                                                                                                                                                                                                 |
| 4   | PC4ODR   | 0             | R/W |                                                                                                                                                                                                                 |
| 3   | PC3ODR   | 0             | R/W |                                                                                                                                                                                                                 |
| 2   | PC2ODR   | 0             | R/W |                                                                                                                                                                                                                 |
| 1   | PC1ODR   | 0             | R/W |                                                                                                                                                                                                                 |
| 0   | PC0ODR   | 0             | R/W |                                                                                                                                                                                                                 |

### 9.6.6 Pin Functions

Port C pins also function as SCI\_1 and SCI\_0 I/O and interrupt input. The correspondence between the register specification and the pin functions is shown below.

PC7: This pin function is switched as shown below.

| PC7DDR       | 0         | 1          |
|--------------|-----------|------------|
| Pin function | PC7 input | PC7 output |

PC6: This pin function is switched as shown below.

| PC6DDR       | 0         | 1          |
|--------------|-----------|------------|
| Pin function | PC6 input | PC6 output |

PC5/SCK1/ $\overline{\text{IRQ5}}$ : The pin function is switched as shown below according to the operating mode, bit  $C/\overline{A}$  in the SCI1's SMR, bits CKE0 and CKE1 in SCR, and bit PC5DDR.

|                  |                                |            |             |             |            |
|------------------|--------------------------------|------------|-------------|-------------|------------|
| CKE1             | 0                              |            |             | 1           |            |
| $C/\overline{A}$ | 0                              |            |             | 1           | —          |
| CKE0             | 0                              |            | 1           | —           | —          |
| PC5DDR           | 0                              | 1          | —           | —           | —          |
| Pin function     | PC5 input                      | PC5 output | SCK1 output | SCK1 output | SCK1 input |
|                  | $\overline{\text{IRQ5}}$ input |            |             |             |            |

PC4/RxD1: The pin function is switched as shown below according to the operating mode, bit RE in the SCI1's SCR, and bit PC4DDR.

|              |           |            |            |
|--------------|-----------|------------|------------|
| RE           | 0         | 1          |            |
| PC4DDR       | 0         | 1          | —          |
| Pin function | PC4 input | PC4 output | RxD1 input |

PC3/TxD1: The pin function is switched as shown below according to the operating mode, bit TE in the SCI1's SCR, and bit PC3DDR.

|              |           |            |             |
|--------------|-----------|------------|-------------|
| TE           | 0         | 1          |             |
| PC3DDR       | 0         | 1          | —           |
| Pin function | PC3 input | PC3 output | TxD1 output |

PC2/SCK0/ $\overline{\text{IRQ4}}$ : The pin function is switched as shown below according to the operating mode, bit C/ $\overline{\text{A}}$  in the SCI0's SMR, bits CKE0 and CKE1 in SCR, and bit PC2DDR.

|                          |                                |            |             |             |            |
|--------------------------|--------------------------------|------------|-------------|-------------|------------|
| CKE1                     | 0                              |            |             |             | 1          |
| C/ $\overline{\text{A}}$ | 0                              |            |             | 1           | —          |
| CKE0                     | 0                              |            | 1           | —           | —          |
| PC2DDR                   | 0                              | 1          | —           | —           | —          |
| Pin function             | <u>PC2 input</u>               | PC2 output | SCK0 output | SCK0 output | SCK0 input |
|                          | $\overline{\text{IRQ4}}$ input |            |             |             |            |

PC1/RxD0: The pin function is switched as shown below according to the operating mode, bit RE in the SCI0's SCR, and bit PC1DDR.

|              |           |            |            |
|--------------|-----------|------------|------------|
| RE           | 0         | 1          |            |
| PC1DDR       | 0         | 1          | —          |
| Pin function | PC1 input | PC1 output | RxD0 input |

PC0/TxD0: The pin function is switched as shown below according to the operating mode, bit TE in the SCI0's SCR, and bit PC0DDR.

|              |           |            |             |
|--------------|-----------|------------|-------------|
| TE           | 0         | 1          |             |
| PC0DDR       | 0         | 1          | —           |
| Pin function | PC0 input | PC0 output | TxD0 output |

## 9.7 Port D

Port D is an 8-bit I/O port that also has other functions. Port D has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

- Port D data direction register (PDDDR)
- Port D data register (PDDR)
- Port D register (PORTD)
- Port D pull-up MOS control register (PDPCR)

### 9.7.1 Port D Data Direction Register (PDDDR)

PDDDR is an 8-bit write-only register, the individual bits of which specify whether the pins of port D are used for input or output.

PDDDR cannot be read; if it is, an undefined value will be read.

| 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 1 pin an output pin. Clearing this bit to 0 makes the pin an input pin. |
| 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   |                                                                                                                                                                                   |

### 9.7.2 Port D Data Register (PDDR)

PDDR is an 8-bit readable/writable register that stores output data for the 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 I/O 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 |                                                                                          |

### 9.7.3 Port D Register (PORTD)

PORTD is an 8-bit read-only register that shows port D pin states.

PORTD cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PD7      | Undefined*    | 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      | Undefined*    | R   |                                                                                                                                                                                  |
| 5   | PD5      | Undefined*    | R   |                                                                                                                                                                                  |
| 4   | PD4      | Undefined*    | R   |                                                                                                                                                                                  |
| 3   | PD3      | Undefined*    | R   |                                                                                                                                                                                  |
| 2   | PD2      | Undefined*    | R   |                                                                                                                                                                                  |
| 1   | PD1      | Undefined*    | R   |                                                                                                                                                                                  |
| 0   | PD0      | Undefined*    | R   |                                                                                                                                                                                  |

Note:\* Determined by the states of pins PD7 to PD0.

### 9.7.4 Port D Pull-up MOS Control Register (PDPCR)

PDPCR is an 8-bit readable/writable register that controls on/off states of the input pull-up MOS of port D.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                              |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------|
| 7   | PD7PCR   | 0             | R/W | When the pin is in its input state, the input pull-up MOS of the input pin is on when the corresponding bit is set to 1. |
| 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 |                                                                                                                          |

## 9.8 Port F

Port F is an 8-bit I/O port that also has other functions. Port F has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

- Port F data direction register (PFDDR)
- Port F data register (PFDR)
- Port F register (PORTF)

### 9.8.1 Port F Data Direction Register (PFDDR)

PFDDR is an 8-bit write-only register, the individual bits of which specify whether the pins of port F are used for input or output.

PFDDR cannot be read; if it is, an undefined value will be read.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                       |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PF7DDR   | 0             | W   | When a pin is specified as a general purpose I/O port, setting this bit to 1 makes the PF7 pin a $\phi$ output pin. Clearing this bit to 0 makes the pin an input pin.            |
| 6   | PF6DDR   | 0             | 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 pin. Clearing this bit to 0 makes the pin an input pin. |
| 5   | PF5DDR   | 0             | W   |                                                                                                                                                                                   |
| 4   | PF4DDR   | 0             | W   |                                                                                                                                                                                   |
| 3   | PF3DDR   | 0             | W   |                                                                                                                                                                                   |
| 2   | PF2DDR   | 0             | W   |                                                                                                                                                                                   |
| 1   | PF1DDR   | 0             | W   |                                                                                                                                                                                   |
| 0   | PF0DDR   | 0             | W   |                                                                                                                                                                                   |

### 9.8.2 Port F Data Register (PFDR)

PFDR is an 8-bit readable/writable register that stores output data for the port F pins.

| Bit | Bit Name | Initial Value | R/W | Description                                                                              |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------|
| 7   | —        | 0             | R/W | Reserved<br>Only 0 should be written to this bit.                                        |
| 6   | PF6DR    | 0             | R/W | Output data for a pin is stored when the pin is specified as a general purpose I/O port. |
| 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 |                                                                                          |

### 9.8.3 Port F Register (PORTF)

PORTF is an 8-bit read-only register that shows port F pin states.

PORTF cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PF7      | Undefined*    | 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      | Undefined*    | R   |                                                                                                                                                                                  |
| 5   | PF5      | Undefined*    | R   |                                                                                                                                                                                  |
| 4   | PF4      | Undefined*    | R   |                                                                                                                                                                                  |
| 3   | PF3      | Undefined*    | R   |                                                                                                                                                                                  |
| 2   | PF2      | Undefined*    | R   |                                                                                                                                                                                  |
| 1   | PF1      | Undefined*    | R   |                                                                                                                                                                                  |
| 0   | PF0      | Undefined*    | R   |                                                                                                                                                                                  |

Note:\* Determined by the states of pins PF7 to PF0.

#### 9.8.4 Pin Functions

Port F is an 8-bit I/O port. Port F pins also function as external interrupt input,  $\overline{\text{IRQ2}}$  and  $\overline{\text{IRQ3}}$ , A/D trigger input ( $\overline{\text{ADTRG}}$ ), and system clock output ( $\phi$ ).

PF7/  $\phi$ : The pin function is switched as shown below according to bit PF7DDR.

|              |           |               |
|--------------|-----------|---------------|
| PF7DDR       | 0         | 1             |
| Pin function | PF7 input | $\phi$ output |

PF6: The pin function is switched as shown below according to bit PF6DDR.

|              |           |            |
|--------------|-----------|------------|
| PF6DDR       | 0         | 1          |
| Pin function | PF6 input | PF6 output |

PF5: The pin function is switched as shown below according to bit PF5DDR.

|              |           |            |
|--------------|-----------|------------|
| PF5DDR       | 0         | 1          |
| Pin function | PF5 input | PF5 output |

PF4: The pin function is switched as shown below according to bit PF4DDR.

|              |           |            |
|--------------|-----------|------------|
| PF4DDR       | 0         | 1          |
| Pin function | PF4 input | PF4 output |

PF3/ $\overline{\text{ADTRG}}$ / $\overline{\text{IRQ3}}$ : The pin function is switched as shown below according to the operating mode, the bus mode, A/D converter bits TRGS1 and TRGS0, and bit PF3DDR.

|              |                                               |            |
|--------------|-----------------------------------------------|------------|
| PF3DDR       | 0                                             | 1          |
| Pin function | PF3 input                                     | PF3 output |
|              | $\overline{\text{ADTRG}}$ input* <sup>1</sup> |            |
|              | $\overline{\text{IRQ3}}$ input * <sup>2</sup> |            |

Notes: \*1  $\overline{\text{ADTRG}}$  input when TRGS0=TRGS1=1.

\*2 When used as an external interrupt input pin, do not use as an I/O pin for another function.

PF2: The pin function is switched as shown below according to bit PF2DDR.

|              |           |            |
|--------------|-----------|------------|
| PF2DDR       | 0         | 1          |
| Pin function | PF2 input | PF2 output |

PF1: The pin function is switched as shown below according to bit PF1DDR.

|              |           |            |
|--------------|-----------|------------|
| PF1DDR       | 0         | 1          |
| Pin function | PF1 input | PF1 output |

PF0/ $\overline{\text{IRQ2}}$ : The pin function is switched as shown below according to bit PF0DDR.

|              |                                |            |
|--------------|--------------------------------|------------|
| PFDDR        | 0                              | 1          |
| Pin function | PF0 input                      | PF0 output |
|              | $\overline{\text{IRQ2}}$ input |            |



# Section 10 16-Bit Timer Pulse Unit (TPU)

This LSI has an on-chip 16-bit timer pulse unit (TPU) comprised of six 16-bit timer channels.

The function list of the 16-bit timer unit and its block diagram are shown in table 10-1 and figure 10-1, respectively.

## 10.1 Features

- Maximum 16-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 operation:
    - Multiple timer counters (TCNT) can be written to simultaneously
    - Simultaneous clearing by compare match and input capture is possible
    - Register simultaneous input/output is possible by synchronous counter operation
    - A maximum 15-phase PWM output is possible in 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
- Programmable pulse generator (PPG) output trigger can be generated
- A/D converter conversion start trigger can be generated
- Module stop mode can be set

**Table 10-1 TPU Functions**

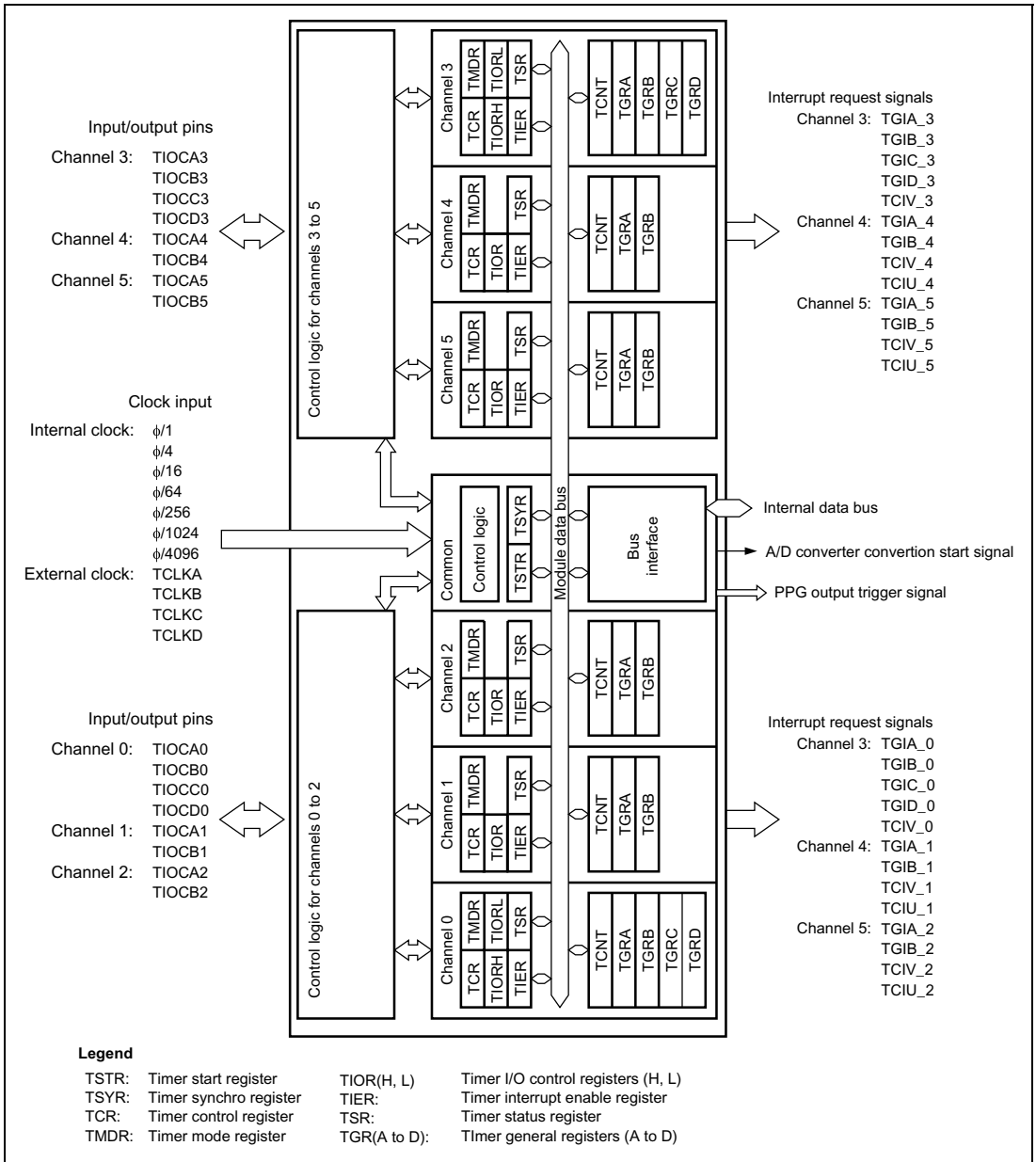
| Item                                   |                                         | Channel 0                               | Channel 1                               | Channel 2                               | Channel 3                               | Channel 4                               | Channel 5 |
|----------------------------------------|-----------------------------------------|-----------------------------------------|-----------------------------------------|-----------------------------------------|-----------------------------------------|-----------------------------------------|-----------|
| Count clock                            |                                         | ø/1                                     | ø/1                                     | ø/1                                     | ø/1                                     | ø/1                                     | ø/1       |
|                                        |                                         | ø/4                                     | ø/4                                     | ø/4                                     | ø/4                                     | ø/4                                     | ø/4       |
|                                        |                                         | ø/16                                    | ø/16                                    | ø/16                                    | ø/16                                    | ø/16                                    | ø/16      |
|                                        |                                         | ø/64                                    | ø/64                                    | ø/64                                    | ø/64                                    | ø/64                                    | ø/64      |
|                                        | TCLKA                                   | ø/256                                   | ø/1024                                  | ø/256                                   | ø/1024                                  | ø/256                                   | ø/256     |
|                                        | TCLKB                                   | TCLKA                                   | TCLKA                                   | ø/1024                                  | TCLKA                                   | TCLKA                                   | TCLKA     |
|                                        | TCLKC                                   | TCLKB                                   | TCLKB                                   | ø/4096                                  | TCLKC                                   | TCLKC                                   | TCLKC     |
| General registers                      | TCLKD                                   |                                         | TCLKC                                   | TCLKA                                   |                                         | TCLKD                                   |           |
|                                        |                                         |                                         |                                         |                                         |                                         |                                         |           |
| General registers                      | 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/<br>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<br>function              | TGR                                     | TGR                                     | TGR                                     | TGR                                     | TGR                                     | TGR                                     |           |
|                                        | compare<br>match or<br>input<br>capture | compare<br>match or<br>input<br>capture | compare<br>match or<br>input<br>capture | compare<br>match or<br>input<br>capture | compare<br>match or<br>input<br>capture | compare<br>match or<br>input<br>capture |           |
| Compare<br>match<br>output             | 0 output                                | o                                       | o                                       | o                                       | o                                       | o                                       | o         |
|                                        | 1 output                                | o                                       | o                                       | o                                       | o                                       | o                                       | o         |
|                                        | Toggle<br>output                        | o                                       | o                                       | o                                       | o                                       | o                                       | o         |
| Input capture<br>function              |                                         | o                                       | o                                       | o                                       | o                                       | o                                       | o         |
| Synchronous<br>operation               |                                         | o                                       | o                                       | o                                       | o                                       | o                                       | o         |
| PWM mode                               |                                         | o                                       | o                                       | o                                       | o                                       | o                                       | o         |
| Phase counting<br>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<br>compare<br>match or<br>input capture                                                                                                                                                                                                                   | TGR<br>compare<br>match or<br>input capture                                                                                                                                              | TGR<br>compare<br>match or<br>input capture                                                                                                                                              | TGR<br>compare<br>match or<br>input capture                                                                                                                                                                      | TGR<br>compare<br>match or<br>input capture                                                                                                                                              | TGR<br>compare<br>match or<br>input capture                                                                                                                                              |
| A/D converter trigger | TGRA_0<br>compare<br>match or<br>input capture                                                                                                                                                                                                                | TGRA_1<br>compare<br>match or<br>input capture                                                                                                                                           | TGRA_2<br>compare<br>match or<br>input capture                                                                                                                                           | TGRA_3<br>compare<br>match or<br>input capture                                                                                                                                                                   | TGRA_4<br>compare<br>match or<br>input capture                                                                                                                                           | TGRA_5<br>compare<br>match or<br>input capture                                                                                                                                           |
| PPG trigger           | TGRA_0/<br>TGRB_0<br>compare<br>match or<br>input capture                                                                                                                                                                                                     | TGRA_1/<br>TGRB_1<br>compare<br>match or<br>input capture                                                                                                                                | TGRA_2/<br>TGRB_2<br>compare<br>match or<br>input capture                                                                                                                                | TGRA_3/<br>TGRB_3<br>compare<br>match or<br>input capture                                                                                                                                                        | —                                                                                                                                                                                        | —                                                                                                                                                                                        |
| Interrupt sources     | 5 sources<br><ul style="list-style-type: none"> <li>• Compare match or input capture 0A</li> <li>• Compare match or input capture 0B</li> <li>• Compare match or input capture 0C</li> <li>• Compare match or input capture 0D</li> <li>• Overflow</li> </ul> | 4 sources<br><ul style="list-style-type: none"> <li>• Compare match or input capture 1A</li> <li>• Compare match or input capture 1B</li> <li>• Overflow</li> <li>• Underflow</li> </ul> | 4 sources<br><ul style="list-style-type: none"> <li>• Compare match or input capture 2A</li> <li>• Compare match or input capture 2B</li> <li>• Overflow</li> <li>• Underflow</li> </ul> | 5 sources<br><ul style="list-style-type: none"> <li>• Compare match or input capture 3A</li> <li>• Compare match or input capture 3C</li> <li>• Compare match or input capture 3D</li> <li>• Overflow</li> </ul> | 4 sources<br><ul style="list-style-type: none"> <li>• Compare match or input capture 4A</li> <li>• Compare match or input capture 4B</li> <li>• Overflow</li> <li>• Underflow</li> </ul> | 4 sources<br><ul style="list-style-type: none"> <li>• Compare match or input capture 5A</li> <li>• Compare match or input capture 5B</li> <li>• Overflow</li> <li>• Underflow</li> </ul> |

### Legend

○ : Possible

— : Not possible



**Figure 10-1 Block Diagram of TPU**

## 10.2 Input/Output Pins

**Table 10-2 TPU Pins**

| Channel | Symbol | I/O   | Function                                                                           |
|---------|--------|-------|------------------------------------------------------------------------------------|
| All     | TCLKA  | Input | External clock A input pin<br>(Channels 1 and 5 phase counting mode A phase input) |
|         | TCLKB  | Input | External clock B input pin<br>(Channels 1 and 5 phase counting mode B phase input) |
|         | TCLKC  | Input | External clock C input pin<br>(Channels 2 and 4 phase counting mode A phase input) |
|         | TCLKD  | Input | External clock D input pin<br>(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                    |

## 10.3 Register Descriptions

The TPU has the following registers. For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register. To distinguish registers in each channel, an underscore and the channel number are added as a suffix to the register name; TCR for channel 0 is expressed as TCR\_0.

- 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 synchro register (TSYR)

### 10.3.1 Timer Control Register (TCR)

The TCR registers are 8-bit readable/writable registers that control the TCNT operation for each channel. The TPU has a total of six TCR registers, one for each channel (channels 0 to 5). TCR register settings should be conducted only when TCNT operation is stopped.

| Bit | Bit Name | Initial value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | CCLR2    | 0             | R/W | Counter Clear 0 to 2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 6   | CCLR1    | 0             | R/W | These bits select the TCNT counter clearing source. See tables 10-3 and 10-4 for details.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 5   | CCLR0    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 4   | CKEG1    | 0             | R/W | Clock Edge 0 and 1<br><br>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. This setting is ignored if the input clock is $\phi/1$ , or when overflow/underflow of another channel is selected.<br><br>00: Count at rising edge<br>01: Count at falling edge<br>1X: Count at both edges<br><br><b>Legend</b> X: Don't care |
| 3   | CKEG0    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 2   | TPSC2    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 1   | TPSC1    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 0   | TPSC0    | 0             | R/W | These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See tables 10-5 to 10-10 for details.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

**Table 10-3 CCLR0 to CCLR2 (channels 0 and 3)**

| Channel | Bit 7<br>CCLR2 | Bit 6<br>CCLR1 | Bit 5<br>CCLR0 | Description                                                                                                              |
|---------|----------------|----------------|----------------|--------------------------------------------------------------------------------------------------------------------------|
| 0, 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 <sup>*1</sup> |
|         | 1              | 0              | 0              | TCNT clearing disabled                                                                                                   |
|         |                |                | 1              | TCNT cleared by TGRC compare match/input capture <sup>*2</sup>                                                           |
|         |                | 1              | 0              | TCNT cleared by TGRD compare match/input capture <sup>*2</sup>                                                           |
|         |                |                | 1              | TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation <sup>*1</sup> |

Notes: <sup>\*1</sup> Synchronous operation is set by setting the SYNC bit in TSYR to 1.

<sup>\*2</sup> 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.

**Table 10-4 CCLR0 to CCLR2 (channels 1, 2, 4, and 5)**

| Channel    | Bit 7<br>Reserved <sup>*2</sup> | Bit 6<br>CCLR1 | Bit 5<br>CCLR0 | Description                                                                                                              |
|------------|---------------------------------|----------------|----------------|--------------------------------------------------------------------------------------------------------------------------|
| 1, 2, 4, 5 | 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 <sup>*1</sup> |

Notes: <sup>\*1</sup> Synchronous operation is selected by setting the SYNC bit in TSYR to 1.

<sup>\*2</sup> Bit 7 is reserved in channels 1, 2, 4, and 5. It is always read as 0 and cannot be modified.

**Table 10-5 TPSC0 to TPSC2 (channel 0)**

| Channel | Bit 2<br>TPSC2 | Bit 1<br>TPSC1 | Bit 0<br>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 10-6 TPSC0 to TPSC2 (channel 1)**

| Channel | Bit 2<br>TPSC2 | Bit 1<br>TPSC1 | Bit 0<br>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        |

Note: This setting is ignored when channel 1 is in phase counting mode.

**Table 10-7 TPSC0 to TPSC2 (channel 2)**

| Channel | Bit 2<br>TPSC2 | Bit 1<br>TPSC1 | Bit 0<br>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 10-8 TPSC0 to TPSC2 (channel 3)**

| Channel | Bit 2<br>TPSC2 | Bit 1<br>TPSC1 | Bit 0<br>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 $\phi/4096$     |

**Table 10-9 TPSC0 to TPSC2 (channel 4)**

| Channel | Bit 2<br>TPSC2 | Bit 1<br>TPSC1 | Bit 0<br>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        |

Note: This setting is ignored when channel 4 is in phase counting mode.

**Table 10-10 TPSC0 to TPSC2 (channel 5)**

| Channel | Bit 2<br>TPSC2 | Bit 1<br>TPSC1 | Bit 0<br>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 |

Note: This setting is ignored when channel 5 is in phase counting mode.

### 10.3.2 Timer Mode Register (TMDR)

The TMDR registers are 8-bit readable/writable registers that are used to set the operating mode of each channel. The TPU has six TMDR registers, one for each channel. TMDR register settings should be changed only when TCNT operation is stopped.

| Bit                                                                                                 | Bit Name | Initial value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----------------------------------------------------------------------------------------------------|----------|---------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7                                                                                                   | —        | 1             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 6                                                                                                   | —        | 1             | —   | These bits are always read as 1 and cannot be modified.                                                                                                                                                                                                                                                                                                                                                                                               |
| 5                                                                                                   | BFB      | 0             | R/W | Buffer Operation B<br><br>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.<br>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.<br>0: TGRB operates normally<br>1: TGRB and TGRD used together for buffer operation |
| 4                                                                                                   | BFA      | 0             | R/W | Buffer Operation A<br><br>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.<br>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.<br>0: TGRA operates normally<br>1: TGRA and TGRC used together for buffer operation |
| 3                                                                                                   | MD3      | 0             | R/W | Modes 0 to 3                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 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. In a write, it should always be written with 0. See table 10-11 for details. |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

**Table 10-11 MD0 to MD3**

| Bit 3<br>MD3*1 | Bit 2<br>MD2*2 | Bit 1<br>MD1 | Bit 0<br>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              | X              | X            | X            | —                     |

**Legend**

X: 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.

### 10.3.3 Timer I/O Control Register (TIOR)

The TIOR registers are 8-bit readable/writable registers that control the TGR registers. The TPU has eight TIOR registers, two each for channels 0 and 3, and one each for channels 1, 2, 4, and 5.

Care is required as 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 B0 to B3<br>Specify the function of TGRB. |
| 6   | IOB2     | 0             | R/W |                                                       |
| 5   | IOB1     | 0             | R/W |                                                       |
| 4   | IOB0     | 0             | R/W |                                                       |
| 3   | IOA3     | 0             | R/W | I/O Control A0 to A3<br>Specify the function of TGRA. |
| 2   | IOA2     | 0             | R/W |                                                       |
| 1   | IOA1     | 0             | R/W |                                                       |
| 0   | IOA0     | 0             | R/W |                                                       |

#### TIORL\_0, TIORL\_3

| Bit | Bit Name | Initial value | R/W | Description                                           |
|-----|----------|---------------|-----|-------------------------------------------------------|
| 7   | IOD3     | 0             | R/W | I/O Control D0 to D3<br>Specify the function of TGRD. |
| 6   | IOD2     | 0             | R/W |                                                       |
| 5   | IOD1     | 0             | R/W |                                                       |
| 4   | IOD0     | 0             | R/W |                                                       |
| 3   | IOC3     | 0             | R/W | I/O Control C0 to C3<br>Specify the function of TGRC. |
| 2   | IOC2     | 0             | R/W |                                                       |
| 1   | IOC1     | 0             | R/W |                                                       |
| 0   | IOC0     | 0             | R/W |                                                       |

**Table 10-12 TIORH\_0 (channel 0)**

|               |               |               |               | Description                   |                                                                                                |
|---------------|---------------|---------------|---------------|-------------------------------|------------------------------------------------------------------------------------------------|
| Bit 7<br>IOB3 | Bit 6<br>IOB2 | Bit 5<br>IOB1 | Bit 4<br>IOB0 | TGRB_0<br>Function            | TIOCB0 Pin Function                                                                            |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                                                                |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match                                               |
|               |               | 1             | 0             |                               | Initial output is 0<br>1 output at compare match                                               |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match                                          |
|               |               | 1             | 0             |                               | Output disabled                                                                                |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match                                               |
|               | 1             | 0             | 0             |                               | Initial output is 1<br>1 output at compare match                                               |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match                                          |
|               |               | 1             | 0             |                               | Input capture at rising edge                                                                   |
|               |               |               | 1             |                               | Input capture at falling edge                                                                  |
|               |               | X             | X             |                               | Input capture at both edges                                                                    |
|               |               |               | X             |                               | Capture input source is channel 1/count clock<br>Input capture at TCNT_1 count- up/count-down* |

#### Legend

X: Don't care

Note: \* When bits TPSC0 to TPSC2 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.

**Table 10-13 TIORL\_0 (channel 0)**

|               |               |               |               | Description                     |                                                       |
|---------------|---------------|---------------|---------------|---------------------------------|-------------------------------------------------------|
| Bit 7<br>IOD3 | Bit 6<br>IOD2 | Bit 5<br>IOD1 | Bit 4<br>IOD0 | TGRD_0<br>Function              | TIOC0 Pin Function                                    |
| 0             | 0             | 0             | 0             | Output<br>Compare<br>register*2 | Output disabled                                       |
|               |               |               | 1             |                                 | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                                 | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                                 | Initial output is 0<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                                 | Output disabled                                       |
|               |               |               | 1             |                                 | Initial output is 1<br>0 output at compare match      |
|               |               | 1             | 0             |                                 | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                                 | Initial output is 1<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                                 | Input capture at rising edge                          |
|               |               |               | 1             |                                 | Input capture at falling edge                         |
|               |               | 1             | X             |                                 | Input capture at both edges                           |
|               |               |               | 1             |                                 | X                                                     |

#### Legend

X: Don't care

Notes: \*1 When bits TPSC0 to TPSC2 in TCR\_1 are set to B'000 and  $\emptyset/1$  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.

**Table 10-14 TIOR\_1 (channel 1)**

|               |               |               |               | Description                   |                                                                                                          |
|---------------|---------------|---------------|---------------|-------------------------------|----------------------------------------------------------------------------------------------------------|
| Bit 7<br>IOB3 | Bit 6<br>IOB2 | Bit 5<br>IOB1 | Bit 4<br>IOB0 | TGRB_1<br>Function            | TIOCB1 Pin Function                                                                                      |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                                                                          |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match                                                         |
|               |               |               | 0             |                               | Initial output is 0<br>1 output at compare match                                                         |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match                                                    |
|               |               | 1             | 0             |                               | Output disabled                                                                                          |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match                                                         |
|               |               |               | 0             |                               | Initial output is 1<br>1 output at compare match                                                         |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match                                                    |
|               | 1             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                                                                             |
|               |               |               | 1             |                               | Input capture at falling edge                                                                            |
|               |               |               | X             |                               | Input capture at both edges                                                                              |
|               |               |               | X             |                               | TGRC_0 compare match/ input capture<br>Input capture at generation of TGRC_0 compare match/input capture |

**Legend**

X: Don't care

**Table 10-15 TIOR\_2 (channel 2)**

|               |               |               |               | Description                   |                                                       |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------|
| Bit 7<br>IOB3 | Bit 6<br>IOB2 | Bit 5<br>IOB1 | Bit 4<br>IOB0 | TGRB_2<br>Function            | TIOCB2 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                               | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match |
| 1             | X             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                          |
|               |               |               | 1             |                               | Input capture at falling edge                         |
|               |               | 1             | X             |                               | Input capture at both edges                           |

**Legend**

X: Don't care

**Table 10-16 TIORH\_3 (channel 3)**

|               |               |               |               | Description                   |                                                       |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------|
| Bit 7<br>IOB3 | Bit 6<br>IOB2 | Bit 5<br>IOB1 | Bit 4<br>IOB0 | TGRB_3<br>Function            | TIOCB3 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                               | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match |
| 1             | 0             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                          |
|               |               |               | 1             |                               | Input capture at falling edge                         |
|               |               | 1             | X             |                               | Input capture at both edges                           |
|               |               | 1             | X             |                               | Capture input source is channel 4/count clock         |
|               |               |               | X             |                               | Input capture at TCNT_4 count-up/count-down*          |

#### Legend

X: Don't care

Note: \* When bits TPSC0 to TPSC2 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.

**Table 10-17 TIORL\_3 (channel 3)**

|               |               |               |               | Description                     |                                                       |
|---------------|---------------|---------------|---------------|---------------------------------|-------------------------------------------------------|
| Bit 7<br>IOD3 | Bit 6<br>IOD2 | Bit 5<br>IOD1 | Bit 4<br>IOD0 | TGRD_3<br>Function              | TIOCD3 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register*2 | Output disabled                                       |
|               |               |               | 1             |                                 | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                                 | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                                 | Initial output is 0<br>Toggle output at compare match |
|               |               | 1             | 0             |                                 | Output disabled                                       |
|               |               |               | 1             |                                 | Initial output is 1<br>0 output at compare match      |
|               | 1             | 0             | 0             | Input<br>capture<br>register*2  | Input capture at rising edge                          |
|               |               |               | 1             |                                 | Input capture at falling edge                         |
|               |               | 1             | X             |                                 | Input capture at both edges                           |
|               |               |               | X             |                                 | Capture input source is channel 4/count clock         |
|               |               | 1             | X             |                                 | Input capture at TCNT_4 count-up/count-down*1         |
|               |               |               | X             |                                 |                                                       |

#### Legend

X: Don't care

Notes: \*1 When bits TPSC0 to TPSC2 in TCR\_4 are set to B'000 and  $\emptyset/1$  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.

**Table 10-18 TIOR\_4 (channel 4)**

|               |               |               |               | Description                   |                                                                                                                                 |
|---------------|---------------|---------------|---------------|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Bit 7<br>IOB3 | Bit 6<br>IOB2 | Bit 5<br>IOB1 | Bit 4<br>IOB0 | TGRB_4<br>Function            | TIOCB4 Pin Function                                                                                                             |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                                                                                                 |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match                                                                                |
|               |               |               | 0             |                               | Initial output is 0<br>1 output at compare match                                                                                |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match                                                                           |
|               |               | 1             | 0             |                               | Output disabled                                                                                                                 |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match                                                                                |
|               |               |               | 0             |                               | Initial output is 1<br>1 output at compare match                                                                                |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match                                                                           |
|               | 1             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                                                                                                    |
|               |               |               | 1             |                               | Input capture at falling edge                                                                                                   |
|               |               |               | X             |                               | Input capture at both edges                                                                                                     |
|               |               |               | X             |                               | Capture input source is TGRC_3 compare match/input capture<br>Input capture at generation of TGRC_3 compare match/input capture |

**Legend**

X: Don't care

**Table 10-19 TIOR\_5 (channel 5)**

|               |               |               |               | Description                   |                                                       |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------|
| Bit 7<br>IOB3 | Bit 6<br>IOB2 | Bit 5<br>IOB1 | Bit 4<br>IOB0 | TGRB_5<br>Function            | TIOCB5 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                               | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match |
| 1             | X             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                          |
|               |               |               | 1             |                               | Input capture at falling edge                         |
|               |               | 1             | X             |                               | Input capture at both edges                           |

**Legend**

X: Don't care

Table 10-20 TIORH\_0 (channel 0)

| Bit 3<br>IOA3 | Bit 2<br>IOA2 | Bit 1<br>IOA1 | Bit 0<br>IOA0 | Description                   |                                                       |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------|
|               |               |               |               | TGRA_0<br>Function            | TIOCA0 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match      |
|               |               |               | 0             |                               | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match |
|               |               | 1             | 0             |                               | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match      |
|               |               |               | 0             |                               | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match |
|               | 1             | 0             | 0             | Input<br>capture<br>register  | Capture input source is TIOCA0 pin                    |
|               |               |               | 1             |                               | Input capture at rising edge                          |
|               |               |               | X             |                               | Input capture at falling edge                         |
|               |               |               | X             |                               | Input capture at both edges                           |
| 1             | 0             | 0             | 0             | Input<br>capture<br>register  | Capture input source is channel 1/count clock         |
|               |               |               | 1             |                               | Input capture at TCNT_1 count-up/count-down           |
|               |               |               | 0             |                               |                                                       |
|               |               |               | 1             |                               |                                                       |

**Legend**

X: Don't care

**Table 10-21 TIORL\_0 (channel 0)**

|               |               |               |               | Description                    |                                                       |
|---------------|---------------|---------------|---------------|--------------------------------|-------------------------------------------------------|
| Bit 3<br>IOC3 | Bit 2<br>IOC2 | Bit 1<br>IOC1 | Bit 0<br>IOC0 | TGRC_0<br>Function             | TIOCC0 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register* | Output disabled                                       |
|               |               |               | 1             |                                | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                                | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                                | Initial output is 0<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                                | Output disabled                                       |
|               |               |               | 1             |                                | Initial output is 1<br>0 output at compare match      |
|               |               | 1             | 0             |                                | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                                | Initial output is 1<br>Toggle output at compare match |
| 1             | 0             | 0             | 0             | Input<br>capture<br>register*  | Input capture at rising edge                          |
|               |               |               | 1             |                                | Input capture at falling edge                         |
|               |               | 1             | X             |                                | Input capture at both edges                           |
|               | 1             | X             | X             |                                | Capture input source is channel 1/count clock         |
|               |               |               |               |                                | Input capture at TCNT_1 count-up/count-down           |
|               |               |               |               |                                |                                                       |

#### Legend

X: Don't care

Note:\* 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.

Table 10-22 TIOR\_1 (channel 1)

|               |               |               |               | Description                   |                                                                                                                                           |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| Bit 3<br>IOA3 | Bit 2<br>IOA2 | Bit 1<br>IOA1 | Bit 0<br>IOA0 | TGRA_1<br>Function            | TIOCA1 Pin Function                                                                                                                       |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                                                                                                           |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match                                                                                          |
|               |               |               | 0             |                               | Initial output is 0<br>1 output at compare match                                                                                          |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match                                                                                     |
|               |               | 1             | 0             |                               | Output disabled                                                                                                                           |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match                                                                                          |
|               |               |               | 0             |                               | Initial output is 1<br>1 output at compare match                                                                                          |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match                                                                                     |
|               | 1             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                                                                                                              |
|               |               |               | 1             |                               | Input capture at falling edge                                                                                                             |
|               |               |               | X             |                               | Input capture at both edges                                                                                                               |
|               |               |               | X             |                               | Capture input source is TGRA_0 compare match/input capture<br>Input capture at generation of channel 0/TGRA_0 compare match/input capture |

**Legend**

X: Don't care

**Table 10-23 TIOR\_2 (channel 2)**

| Bit 3<br>IOA3 | Bit 2<br>IOA2 | Bit 1<br>IOA1 | Bit 0<br>IOA0 | Description                   |                                                       |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------|
|               |               |               |               | TGRA_2<br>Function            | TIOCA2 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match      |
|               |               |               | 0             |                               | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match |
|               |               | 1             | 0             |                               | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match      |
|               |               |               | 0             |                               | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match |
|               | 1             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                          |
|               |               |               | 1             |                               | Input capture at falling edge                         |
|               |               |               | X             |                               | Input capture at both edges                           |
|               |               |               | X             |                               |                                                       |

**Legend**

X: Don't care

**Table 10-24 TIORH\_3 (channel 3)**

|               |               |               |               | Description                   |                                                                                              |
|---------------|---------------|---------------|---------------|-------------------------------|----------------------------------------------------------------------------------------------|
| Bit 3<br>IOA3 | Bit 2<br>IOA2 | Bit 1<br>IOA1 | Bit 0<br>IOA0 | TGRA_3<br>Function            | TIOCA3 Pin Function                                                                          |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                                                              |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match                                             |
|               |               | 1             | 0             |                               | Initial output is 0<br>1 output at compare match                                             |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match                                        |
|               | 1             | 0             | 0             |                               | Output disabled                                                                              |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match                                             |
|               |               | 1             | 0             |                               | Initial output is 1<br>1 output at compare match                                             |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match                                        |
|               |               | X             | 0             |                               | Input capture at rising edge                                                                 |
|               |               |               | 1             |                               | Input capture at falling edge                                                                |
|               |               |               | X             |                               | Input capture at both edges                                                                  |
|               |               |               | X             |                               | Capture input source is channel 4/count clock<br>Input capture at TCNT_4 count-up/count-down |

**Legend**

X: Don't care

**Table 10-25 TIORL\_3 (channel 3)**

|               |               |               |               | Description                    |                                                                                              |
|---------------|---------------|---------------|---------------|--------------------------------|----------------------------------------------------------------------------------------------|
| Bit 3<br>IOC3 | Bit 2<br>IOC2 | Bit 1<br>IOC1 | Bit 0<br>IOC0 | TGRC_3<br>Function             | TIOCC3 Pin Function                                                                          |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register* | Output disabled                                                                              |
|               |               |               | 1             |                                | Initial output is 0<br>0 output at compare match                                             |
|               |               | 1             | 0             |                                | Initial output is 0<br>1 output at compare match                                             |
|               |               |               | 1             |                                | Initial output is 0<br>Toggle output at compare match                                        |
|               | 1             | 0             | 0             |                                | Output disabled                                                                              |
|               |               |               | 1             |                                | Initial output is 1<br>0 output at compare match                                             |
|               |               | 1             | 0             |                                | Initial output is 1<br>1 output at compare match                                             |
|               |               |               | 1             |                                | Initial output is 1<br>Toggle output at compare match                                        |
|               | 1             | 0             | 0             | Input<br>capture<br>register*  | Input capture at rising edge                                                                 |
|               |               |               | 1             |                                | Input capture at falling edge                                                                |
|               |               | 1             | X             |                                | Input capture at both edges                                                                  |
|               |               |               | X             |                                | Capture input source is channel 4/count clock<br>Input capture at TCNT_4 count-up/count-down |

#### Legend

X: Don't care

Note:\* 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.

**Table 10-26 TIOR\_4 (channel 4)**

|               |               |               |               | Description                   |                                                                                                                                 |
|---------------|---------------|---------------|---------------|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Bit 3<br>IOA3 | Bit 2<br>IOA2 | Bit 1<br>IOA1 | Bit 0<br>IOA0 | TGRA_4<br>Function            | TIOCA4 Pin Function                                                                                                             |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                                                                                                 |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match                                                                                |
|               |               |               | 0             |                               | Initial output is 0<br>1 output at compare match                                                                                |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match                                                                           |
|               |               | 1             | 0             |                               | Output disabled                                                                                                                 |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match                                                                                |
|               |               |               | 0             |                               | Initial output is 1<br>1 output at compare match                                                                                |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match                                                                           |
|               | 1             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                                                                                                    |
|               |               |               | 1             |                               | Input capture at falling edge                                                                                                   |
|               |               |               | X             |                               | Input capture at both edges                                                                                                     |
|               |               |               | X             |                               | Capture input source is TGRA_3 compare match/input capture<br>Input capture at generation of TGRA_3 compare match/input capture |

**Legend**

X: Don't care

**Table 10-27 TIOR\_5 (channel 5)**

|               |               |               |               | Description                   |                                                       |
|---------------|---------------|---------------|---------------|-------------------------------|-------------------------------------------------------|
| Bit 3<br>IOA3 | Bit 2<br>IOA2 | Bit 1<br>IOA1 | Bit 0<br>IOA0 | TGRA_5<br>Function            | TIOCA5 Pin Function                                   |
| 0             | 0             | 0             | 0             | Output<br>compare<br>register | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 0<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 0<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 0<br>Toggle output at compare match |
|               | 1             | 0             | 0             |                               | Output disabled                                       |
|               |               |               | 1             |                               | Initial output is 1<br>0 output at compare match      |
|               |               | 1             | 0             |                               | Initial output is 1<br>1 output at compare match      |
|               |               |               | 1             |                               | Initial output is 1<br>Toggle output at compare match |
| 1             | X             | 0             | 0             | Input<br>capture<br>register  | Input capture at rising edge                          |
|               |               |               | 1             |                               | Input capture at falling edge                         |
|               |               | 1             | X             |                               | Input capture at both edges                           |

#### Legend

X: Don't care

### 10.3.4 Timer Interrupt Enable Register (TIER)

The TIER registers are 8-bit readable/writable registers that control enabling or disabling of interrupt requests for each channel. The TPU has six TIER registers, one for each channel.

| 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<br/>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<br/>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<br/>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<br/>1: Interrupt requests (TGID) by TGFD bit enabled</p> |
| 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<br/>1: Interrupt requests (TGIC) by TGFC bit enabled</p> |

| Bit | Bit Name | Initial value | R/W | Description                                                                                                                                                                                                                                               |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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> |

### 10.3.5 Timer Status Register (TSR)

The TSR registers are 8-bit readable/writable registers that indicate the status of each channel. The TPU has six TSR registers, one for each channel.

| Bit | Bit Name | Initial value | R/W   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|-----|----------|---------------|-------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | TCFD     | 1             | R     | <p>Count Direction Flag</p> <p>Status flag that shows the direction in which TCNT counts in channels 1, 2, 4, and 5.</p> <p>In channels 0 and 3, bit 7 is reserved. It is always read as 1 and cannot be modified.</p> <p>0: TCNT counts down</p> <p>1: TCNT counts up</p>                                                                                                                                                                                                         |
| 6   | —        | 1             | —     | <p>Reserved</p> <p>This bit is always read as 1 and cannot be modified.</p>                                                                                                                                                                                                                                                                                                                                                                                                        |
| 5   | TCFU     | 0             | R/(W) | <p>Underflow Flag</p> <p>Status flag that indicates that TCNT underflow has occurred when channels 1, 2, 4, and 5 are set to phase counting mode. Only 0 can be written, for flag clearing.</p> <p>In channels 0 and 3, bit 5 is reserved. It is always read as 0 and cannot be modified.</p> <p>[Setting condition]</p> <p>When the TCNT value underflows (changes from H'0000 to H'FFFF)</p> <p>[Clearing condition]</p> <p>When 0 is written to TCFU after reading TCFU = 1</p> |
| 4   | TCFV     | 0             | R/(W) | <p>Overflow Flag</p> <p>Status flag that indicates that TCNT overflow has occurred. Only 0 can be written, for flag clearing.</p> <p>[Setting condition]</p> <p>When the TCNT value overflows (changes from H'FFFF to H'0000 )</p> <p>[Clearing condition]</p> <p>When 0 is written to TCFV after reading TCFV = 1</p>                                                                                                                                                             |

| Bit | Bit Name | Initial value | R/W   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----|----------|---------------|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3   | TGFD     | 0             | R/(W) | <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. Only 0 can be written, for flag clearing. In channels 1, 2, 4, and 5, 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"> <li>When TCNT = TGRD and TGRD is functioning as output compare register</li> <li>When TCNT value is transferred to TGRD by input capture signal and TGRD is functioning as input capture register</li> </ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When DTC is activated by TGID interrupt and the DISEL bit of MRB in DTC is 0</li> <li>When 0 is written to TGFD after reading TGFD = 1</li> </ul> |
| 2   | TGFC     | 0             | R/(W) | <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. Only 0 can be written, for flag clearing. In channels 1, 2, 4, and 5, 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"> <li>When TCNT = TGRC and TGRC is functioning as output compare register</li> <li>When TCNT value is transferred to TGRC by input capture signal and TGRC is functioning as input capture register</li> </ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When DTC is activated by TGIC interrupt and the DISEL bit of MRB in DTC is 0</li> <li>When 0 is written to TGFC after reading TGFC = 1</li> </ul> |

| Bit | Bit Name | Initial value | R/W   | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----|----------|---------------|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | TGFB     | 0             | R/(W) | <p>Input Capture/Output Compare Flag B</p> <p>Status flag that indicates the occurrence of TGRB input capture or compare match. Only 0 can be written, for flag clearing.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When TCNT = TGRB and TGRB is functioning as output compare register</li> <li>When TCNT value is transferred to TGRB by input capture signal and TGRB is functioning as input capture register</li> </ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When DTC is activated by TGIB interrupt and the DISEL bit of MRB in DTC is 0</li> <li>When 0 is written to TGFB after reading TGFB = 1</li> </ul> |
| 0   | TGFA     | 0             | R/(W) | <p>Input Capture/Output Compare Flag A</p> <p>Status flag that indicates the occurrence of TGRA input capture or compare match. Only 0 can be written, for flag clearing.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When TCNT = TGRA and TGRA is functioning as output compare register</li> <li>When TCNT value is transferred to TGRA by input capture signal and TGRA is functioning as input capture register</li> </ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When DTC is activated by TGIA interrupt and the DISEL bit of MRB in DTC is 0</li> <li>When 0 is written to TGFA after reading TGFA = 1</li> </ul> |

### 10.3.6 Timer Counter (TCNT)

The TCNT registers are 16-bit readable/writable counters. The TPU has six TCNT counters, one for each channel.

The TCNT counters are initialized to H'0000 by a reset, and in hardware standby mode.

The TCNT counters cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit.

### 10.3.7 Timer General Register (TGR)

The TGR registers are dual function 16-bit readable/writable registers, functioning as either output compare or input capture registers. The TPU has 16 TGR registers, four each for channels 0 and 3 and two 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.

### 10.3.8 Timer Start Register (TSTR)

TSTR is an 8-bit readable/writable register that selects operation/stoppage 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   | —        | 0             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 6   |          |               |     | Only 0 should be written to these bits.                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 5   | CST5     | 0             | R/W | Counter Start 0 to 5 (CST0 to CST5)                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 4   | CST4     |               |     | These bits select operation or stoppage for TCNT.<br>If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained.<br>If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value.<br><br>0: TCNT_0 to TCNT_5 count operation is stopped<br>1: TCNT_0 to TCNT_5 performs count operation |
| 3   | CST3     |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 2   | CST2     |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 1   | CST1     |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 0   | CST0     |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |

### 10.3.9 Timer Synchro Register (TSYR)

TSYR is an 8-bit readable/writable register that selects independent operation or synchronous operation for the channel 0 to 5 TCNT counters. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

| Bit | Bit Name | Initial value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | 0             | R/W | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| 6   | —        | 0             | R/W | Only 0 should be written to these bits.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 5   | SYNC5    | 0             | R/W | Timer Synchro 0 to 5                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| 4   | SYNC4    | 0             | R/W | These bits are used to select whether operation is independent of or synchronized with other channels.<br>When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible.<br><br>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 CCLR0 to CCLR2 in TCR.<br><br>0: TCNT_0 to TCNT_5 operates independently (TCNT presetting /clearing is unrelated to other channels)<br><br>1: TCNT_0 to TCNT_5 performs synchronous operation<br><br>TCNT synchronous presetting/synchronous clearing is possible |
| 3   | SYNC3    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 2   | SYNC2    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 1   | SYNC1    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 0   | SYNC0    | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |

## 10.4 Operation

### 10.4.1 Basic Functions

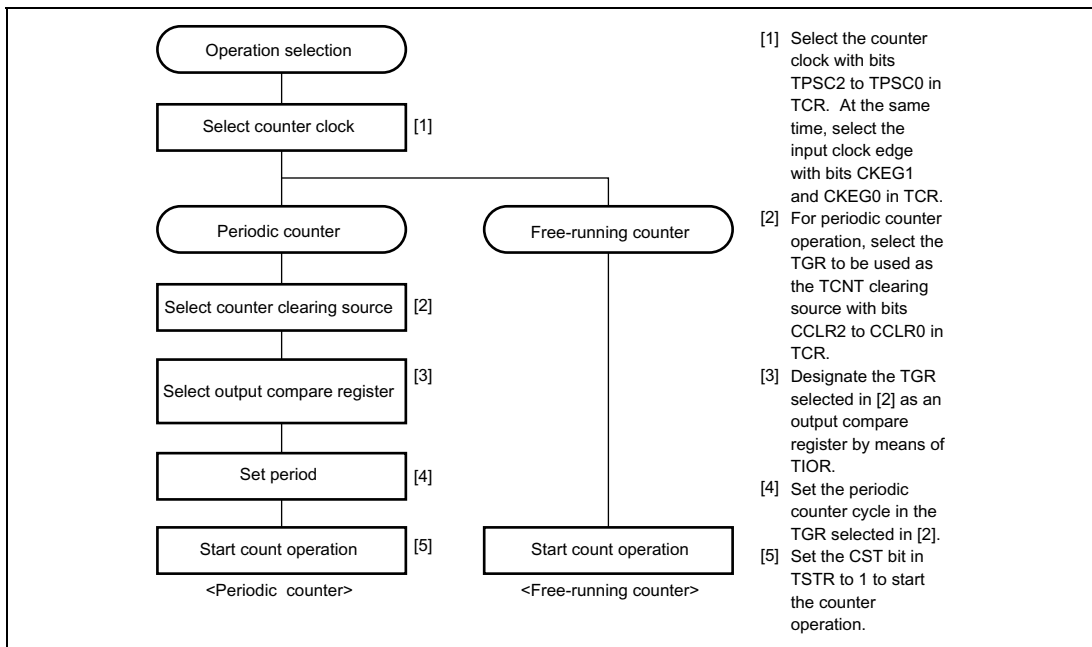
Each channel has a TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, synchronous counting, and external event counting.

Each TGR can be used as an input capture register or output compare register.

**Counter Operation:** When one of bits CST0 to CST5 is set to 1 in TSTR, the TCNT counter for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example.

## 1. Example of count operation setting procedure

Figure 10-2 shows an example of the count operation setting procedure.

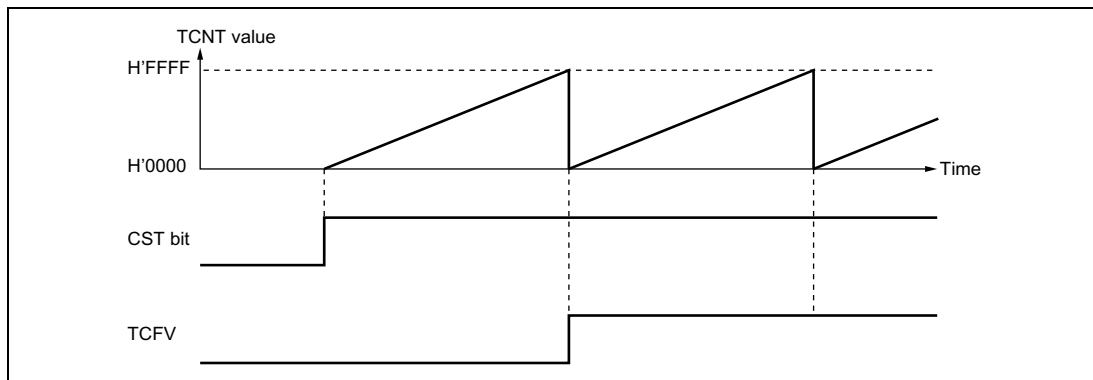


**Figure 10-2 Example of Counter Operation Setting Procedure**

## 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 (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 10-3 illustrates free-running counter operation.

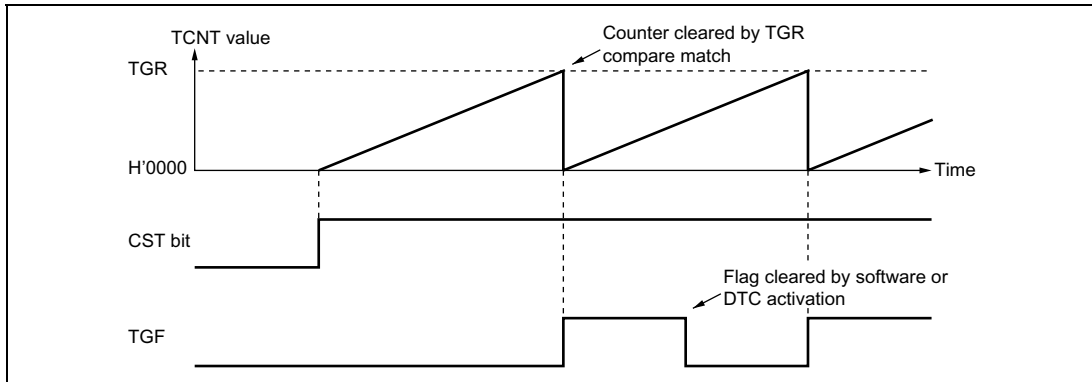


**Figure 10-3 Free-Running Counter Operation**

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 CCLR0 to CCLR2 in TCR. After the settings have been made, TCNT starts up-count 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 10-4 illustrates periodic counter operation.

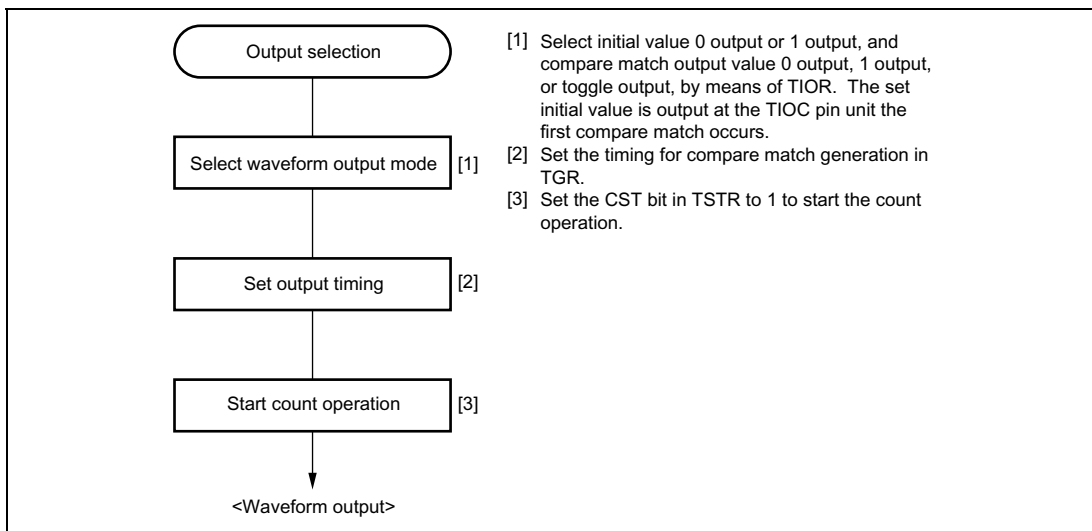


**Figure 10-4 Periodic Counter Operation**

**Waveform Output by Compare Match:** The TPU can perform 0, 1, or toggle output from the corresponding output pin using compare match.

1. Example of setting procedure for waveform output by compare match

Figure 10-5 shows an example of the setting procedure for waveform output by compare match.

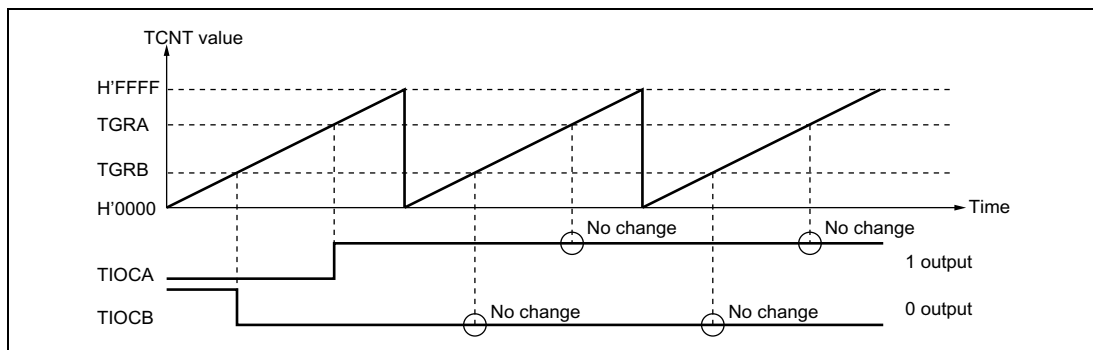


**Figure 10-5 Example of Setting Procedure for Waveform Output by Compare Match**

## 2. Examples of waveform output operation

Figure 10-6 shows an example of 0 output/1 output.

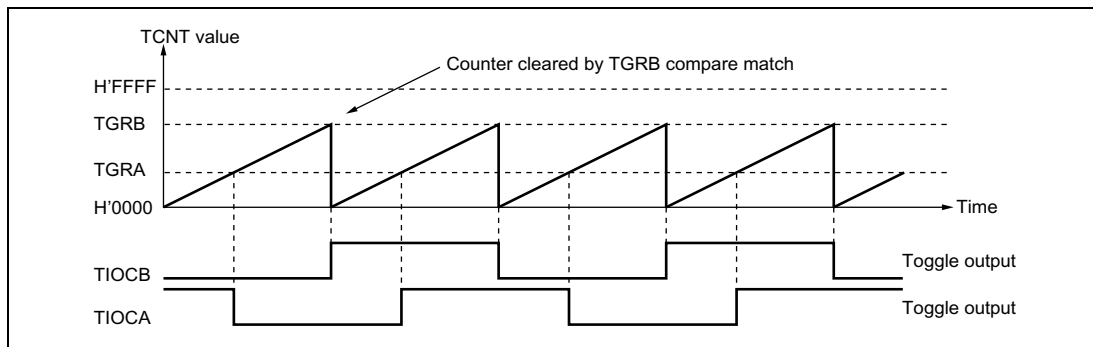
In this example TCNT has been designated as a free-running counter, and settings have been made such that 1 is output by compare match A, and 0 is output by compare match B. When the set level and the pin level coincide, the pin level does not change.



**Figure 10-6 Example of 0 Output/1 Output Operation**

Figure 10-7 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made such that the output is toggled by both compare match A and compare match B.



**Figure 10-7 Example of Toggle Output Operation**

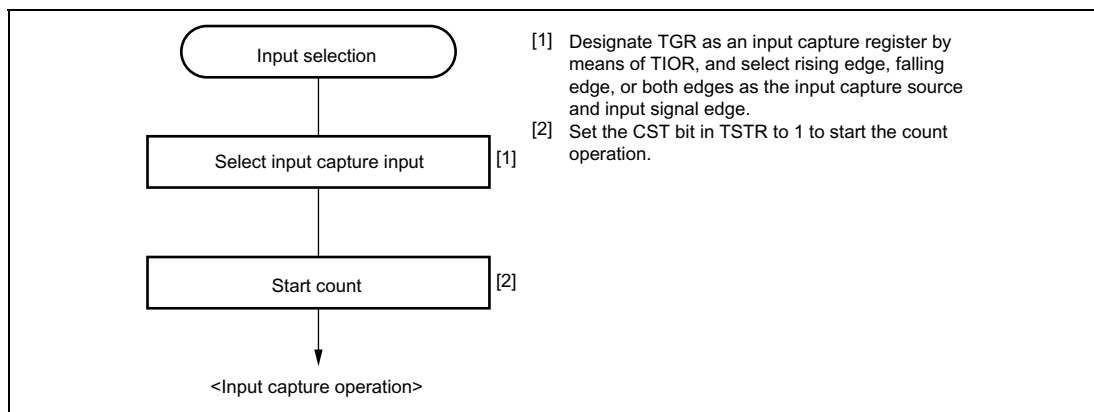
**Input Capture Function:** The TCNT value can be transferred to TGR on detection of the TIOC pin input edge.

Rising edge, falling edge, or both edges can be selected as the detected 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.

**Note:** When another channel's counter input clock is used as the input capture input for channels 0 and 3,  $\emptyset/1$  should not be selected as the counter input clock used for input capture input. Input capture will not be generated if  $\emptyset/1$  is selected.

#### 1. Example of input capture operation setting procedure

Figure 10-8 shows an example of the input capture operation setting procedure.

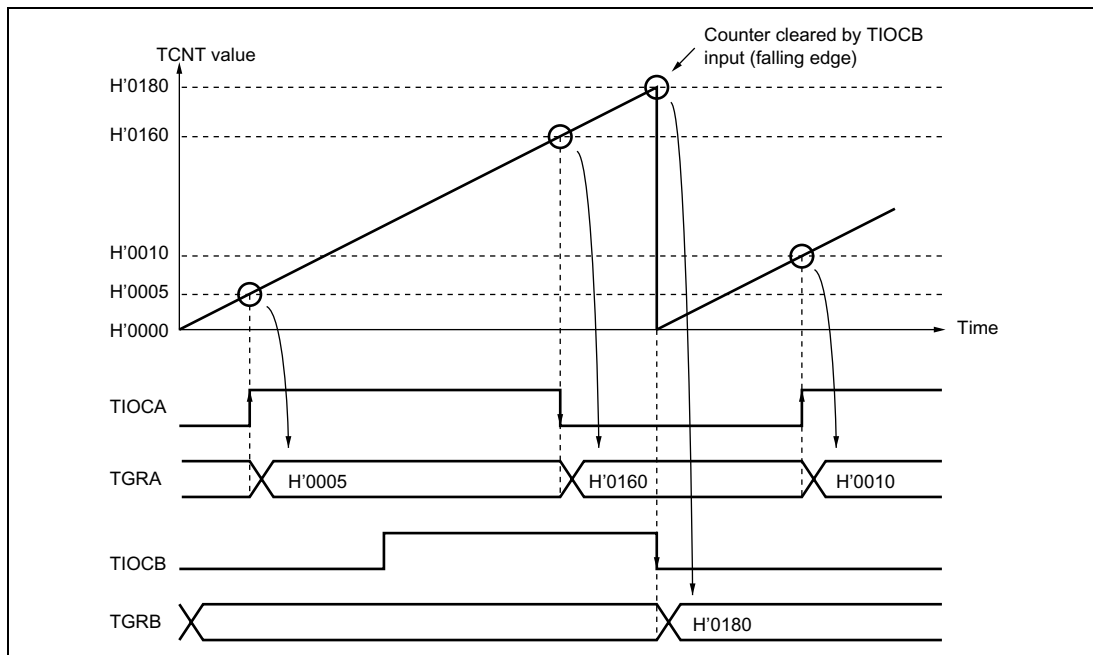


**Figure 10-8 Example of Input Capture Operation Setting Procedure**

## 2. Example of input capture operation

Figure 10-9 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, the 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 10-9 Example of Input Capture Operation**

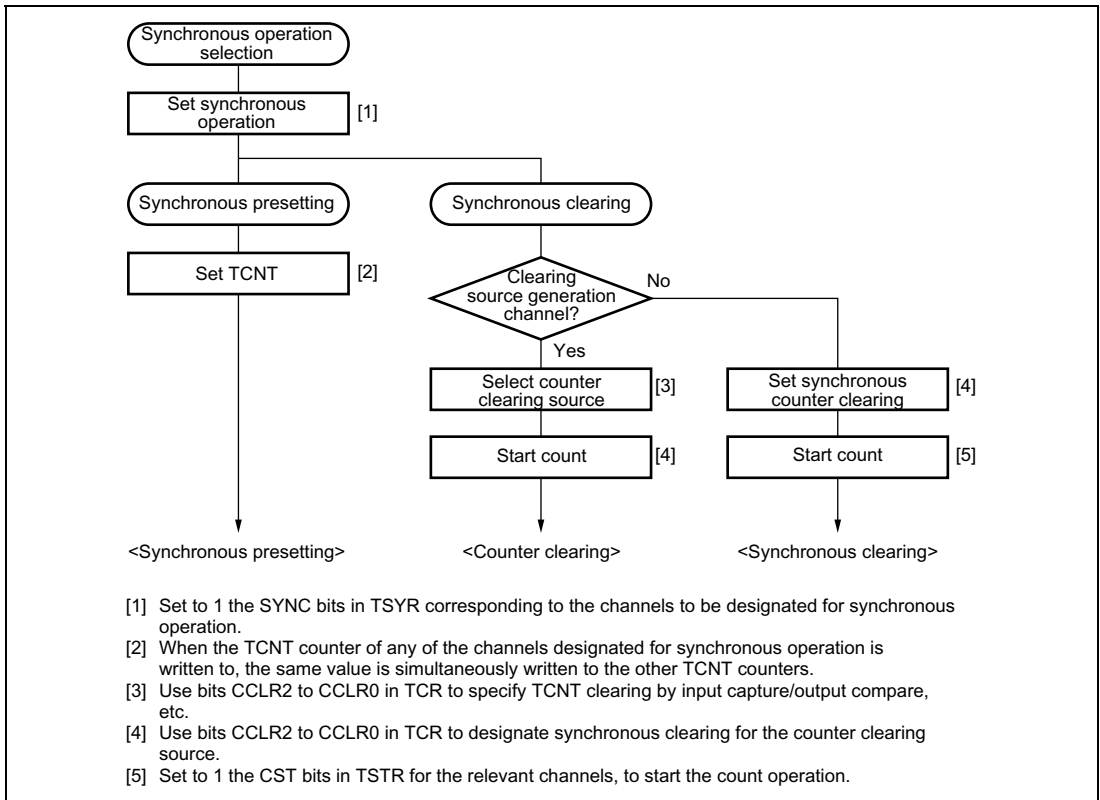
## 10.4.2 Synchronous Operation

In synchronous operation, the values in a number of TCNT counters can be rewritten simultaneously (synchronous presetting). Also, a number of TCNT counters can be cleared simultaneously by making the appropriate setting in TCR (synchronous clearing).

Synchronous operation enables TGR to be incremented with respect to a single time base.

Channels 0 to 5 can all be designated for synchronous operation.

**Example of Synchronous Operation Setting Procedure:** Figure 10-10 shows an example of the synchronous operation setting procedure.



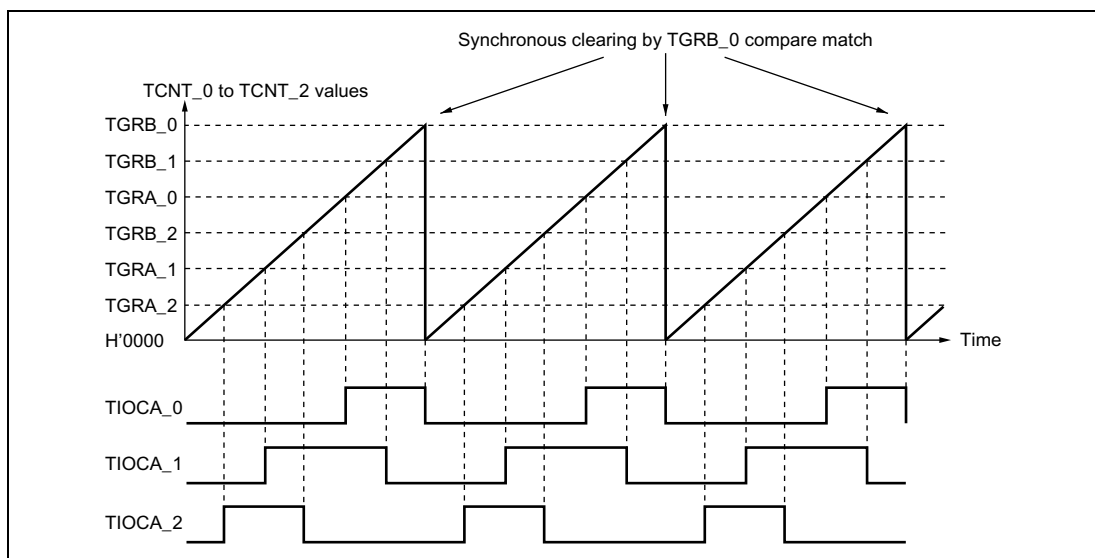
**Figure 10-10 Example of Synchronous Operation Setting Procedure**

**Example of Synchronous Operation:** Figure 10-11 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 TIOC0A, TIOC1A, and TIOC2A. At this time, synchronous presetting, and synchronous clearing by TGRB\_0 compare match, are performed for channel 0 to 2 TCNT counters, and the data set in TGRB\_0 is used as the PWM cycle.

For details of PWM modes, see section 10.4.5, PWM Modes.



**Figure 10-11 Example of Synchronous Operation**

### 10.4.3 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 as a compare match register.

Table 10-28 shows the register combinations used in buffer operation.

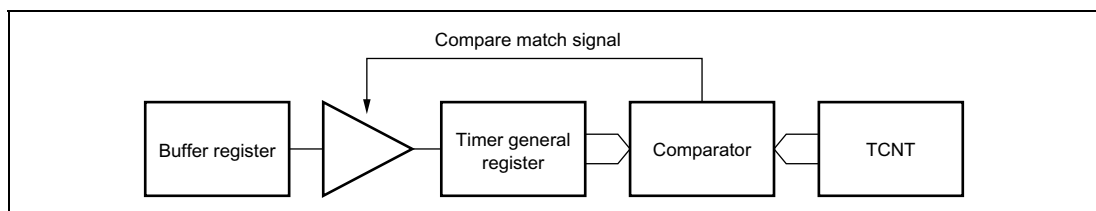
**Table 10-28 Register Combinations in Buffer Operation**

| Channel | Timer General Register | Buffer Register |
|---------|------------------------|-----------------|
| 0       | TGRA_0                 | TGRC_0          |
|         | TGRB_0                 | TGRD_0          |
| 3       | TGRA_3                 | TGRC_3          |
|         | TGRB_3                 | TGRD_3          |

- 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 10-12.

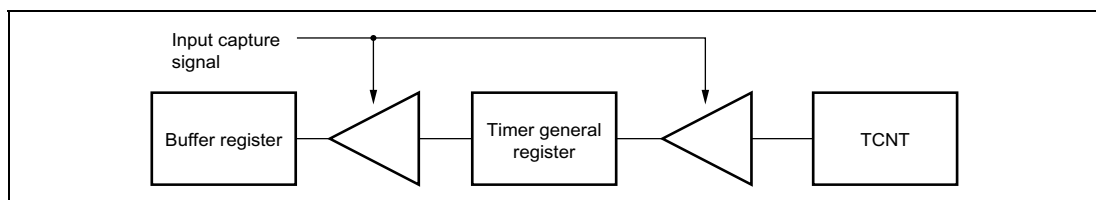


**Figure 10-12 Compare Match Buffer Operation**

- When TGR is an input capture register

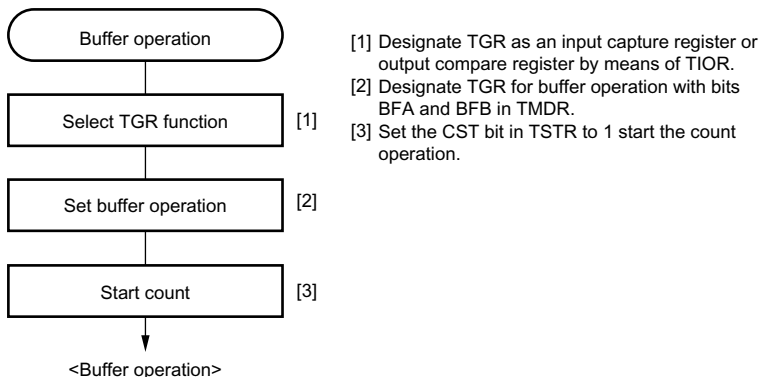
When input capture occurs, the value in TCNT is transferred to TGR and the value previously held in the timer general register is transferred to the buffer register.

This operation is illustrated in figure 10-13.



**Figure 10-13 Input Capture Buffer Operation**

**Example of Buffer Operation Setting Procedure:** Figure 10-14 shows an example of the buffer operation setting procedure.



**Figure 10-14 Example of Buffer Operation Setting Procedure**

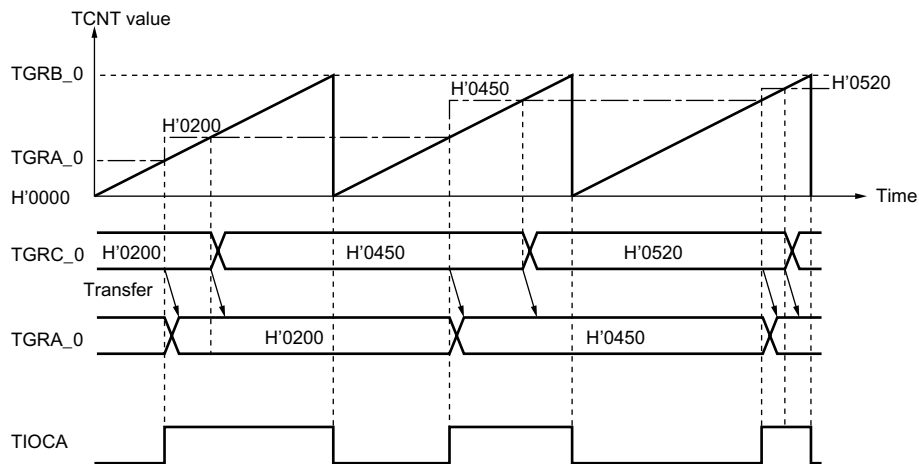
## Examples of Buffer Operation

### 1. When TGR is an output compare register

Figure 10-15 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 that compare match A occurs.

For details of PWM modes, see section 10.4.5, PWM Modes.



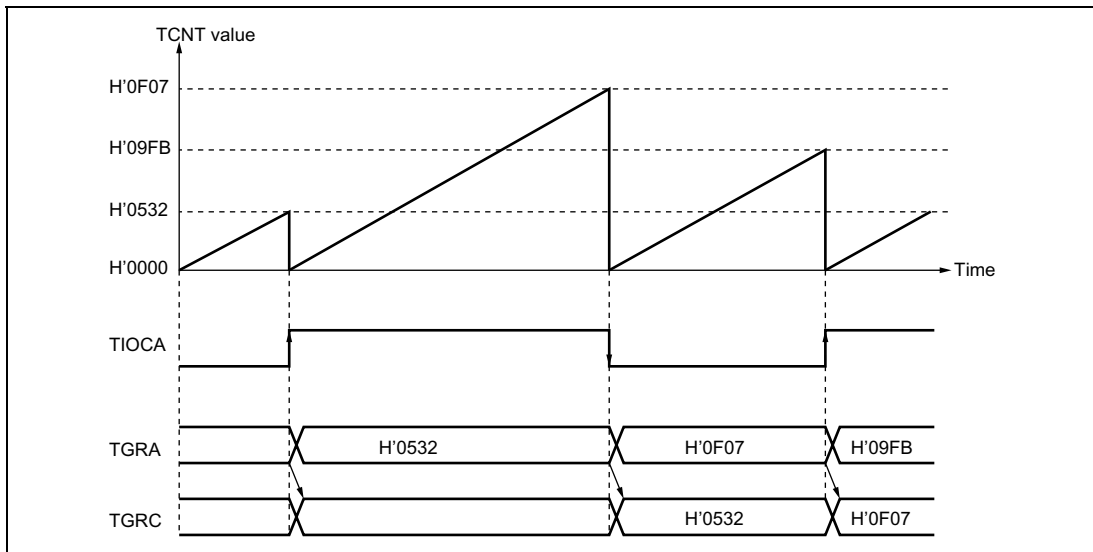
**Figure 10-15 Example of Buffer Operation (1)**

2. When TGR is an input capture register

Figure 10-16 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 the occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.



**Figure 10-16 Example of Buffer Operation (2)**

#### 10.4.4 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 upon overflow/underflow of TCNT\_2 (TCNT\_5) as set in bits TPSC0 to TPSC2 in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

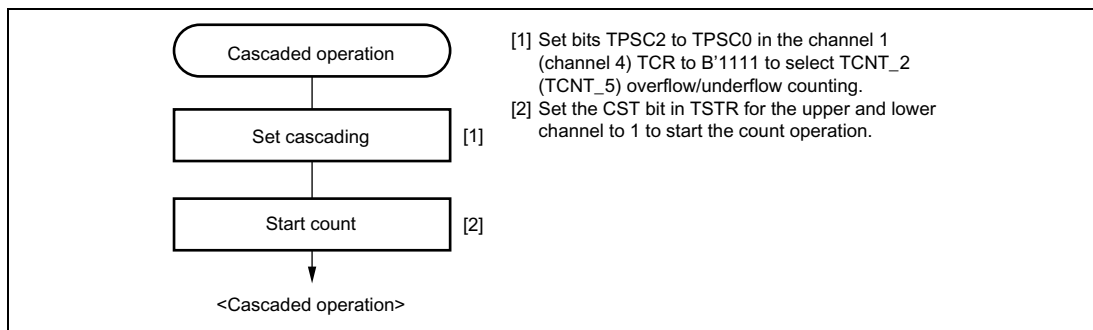
Table 10-29 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for channel 1 or 4, the counter clock setting is invalid and the counters operates independently in phase counting mode.

**Table 10-29 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        |

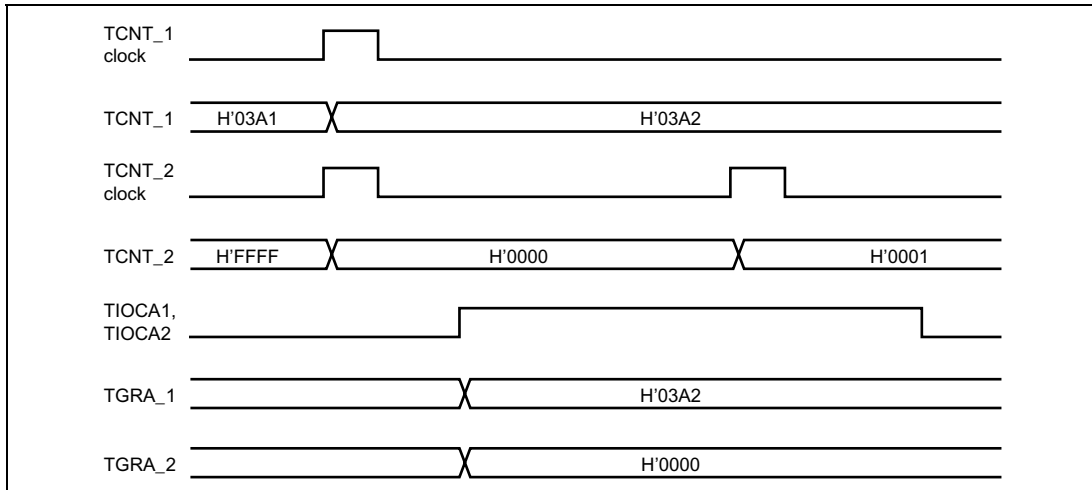
**Example of Cascaded Operation Setting Procedure:** Figure 10-17 shows an example of the setting procedure for cascaded operation.



**Figure 10-17 Cascaded Operation Setting Procedure**

**Examples of Cascaded Operation:** Figure 10-18 illustrates the operation when TCNT\_2 overflow/underflow counting has been set for TCNT\_1, when TGRA\_1 and TGRA\_2 have been designated as input capture registers, and when TIOC pin rising edge has been selected.

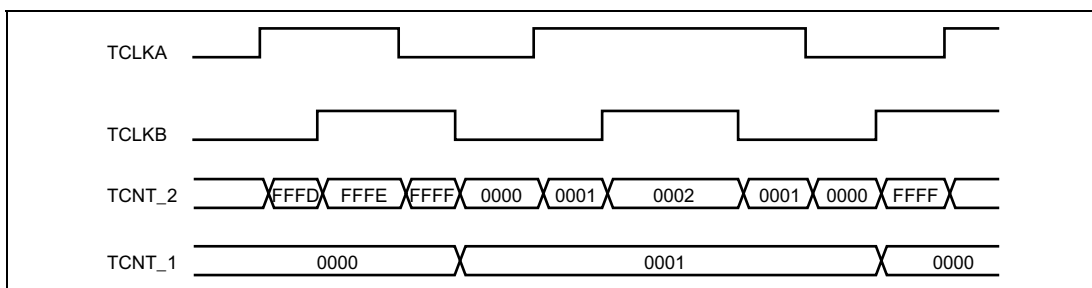
When a rising edge is input to the TIOCA1 and TIOCA2 pins simultaneously, the upper 16 bits of the 32-bit data are transferred to TGRA\_1, and the lower 16 bits to TGRA\_2.



**Figure 10-18 Example of Cascaded Operation (1)**

Figure 10-19 illustrates the operation when TCNT\_2 overflow/underflow counting 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 10-19 Example of Cascaded Operation (2)**

## 10.4.5 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. The output level can be selected as 0, 1, or toggle output in response to a compare match of each TGR.

TGR registers settings can be used to output a PWM waveform in the range of 0% to 100% duty.

Designating TGR compare match as the counter clearing source enables the period 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 output specified by bits IOA0 to IOA3 and IOC0 to IOC3 in TIOR is output from the TIOCA and TIOCC pins at compare matches A and C, and the output specified by bits IOB0 to IOB3 and IOD0 to IOD3 in TIOR is output at compare matches B and D. 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 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 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 in combination use with synchronous operation.

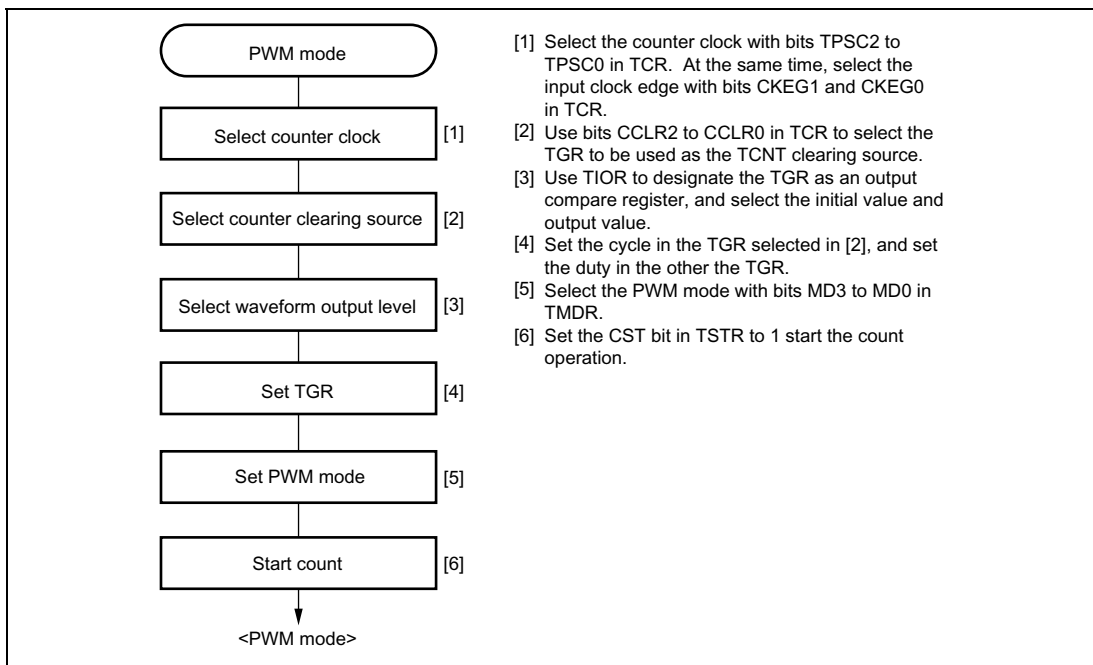
The correspondence between PWM output pins and registers is shown in table 10-30.

**Table 10-30 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       | TGR4A_4   | TIOCA4      | TIOCA4     |
|         | TGR4B_4   |             | TIOCB4     |
| 5       | TGRA_5    | TIOCA5      | TIOCA5     |
|         | TGRB_5    |             | TIOCB5     |

Note: In PWM mode 2, PWM output is not possible for the TGR register in which the period is set.

**Example of PWM Mode Setting Procedure:** Figure 10-20 shows an example of the PWM mode setting procedure.

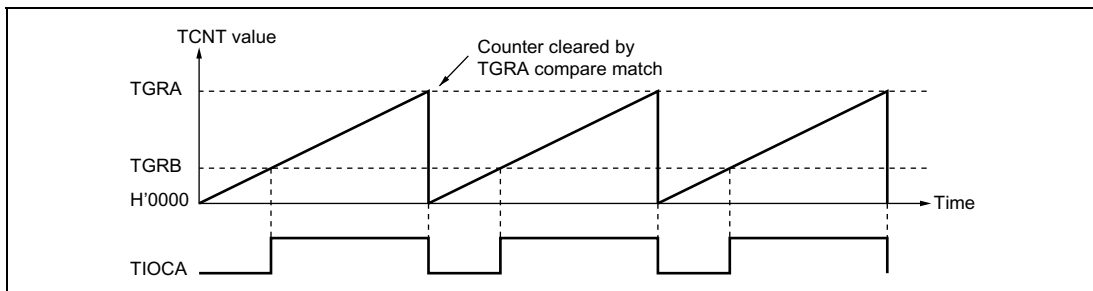


**Figure 10-20 Example of PWM Mode Setting Procedure**

**Examples of PWM Mode Operation:** Figure 10-21 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 period, and the values set in the TGRB registers are used as the duty levels.

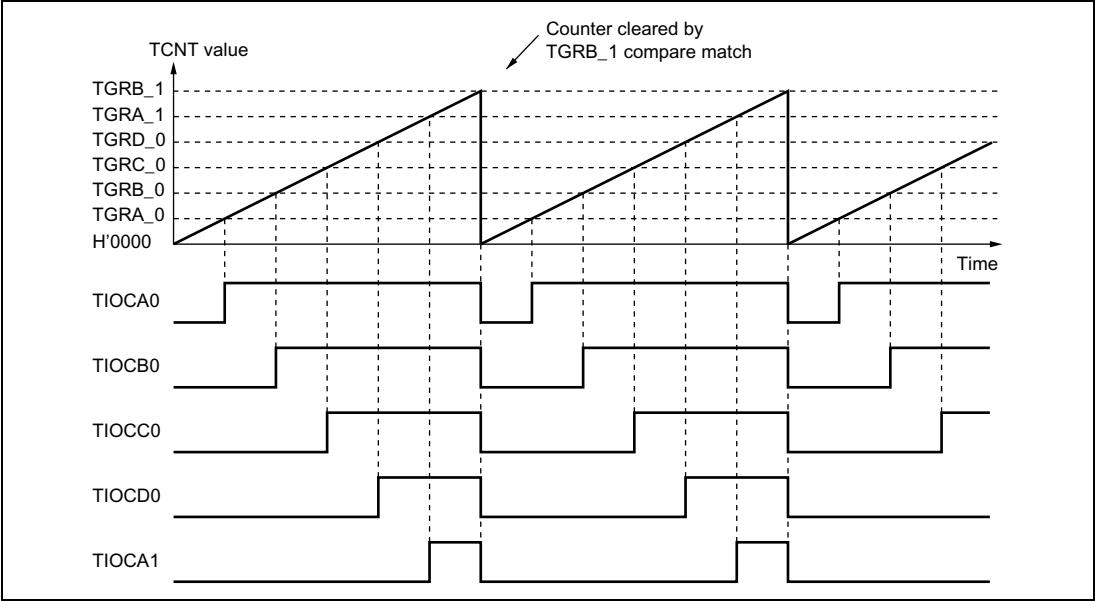


**Figure 10-21 Example of PWM Mode Operation (1)**

Figure 10-22 shows an example of PWM mode 2 operation.

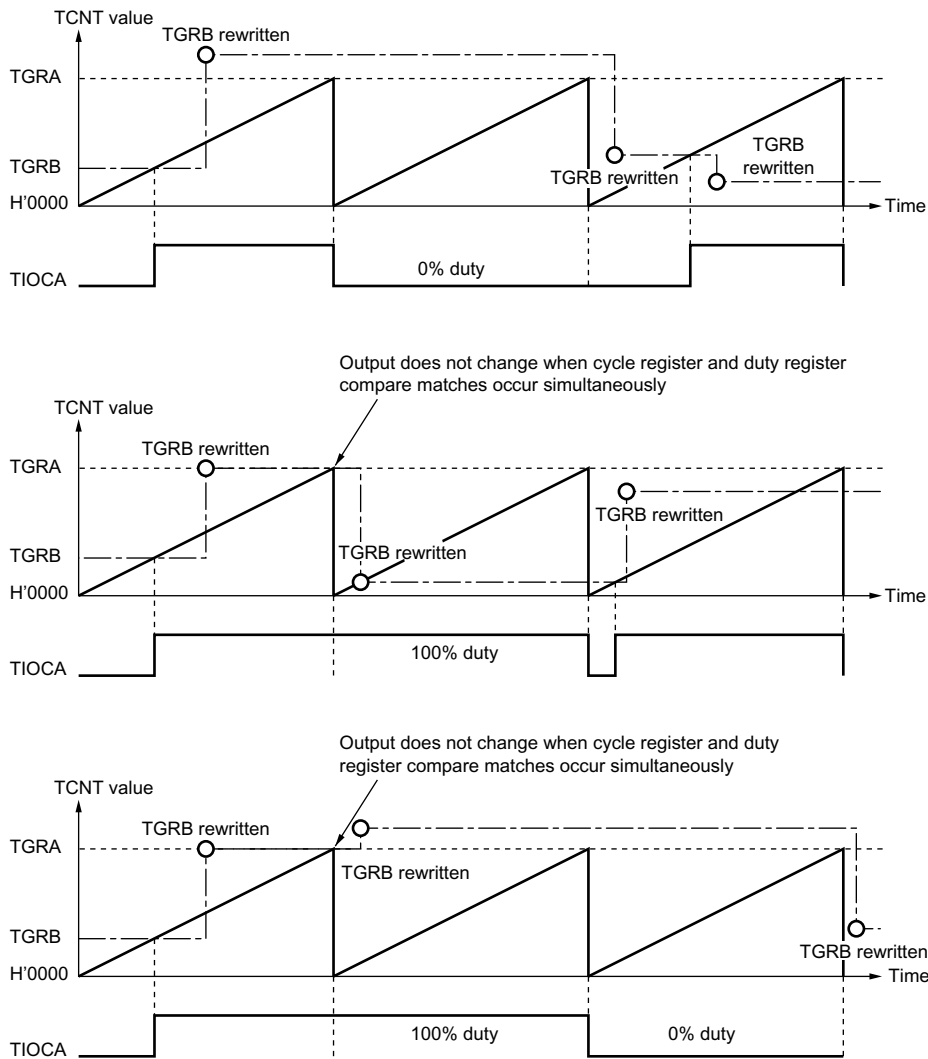
In this example, synchronous operation is designated for channels 0 and 1, TGRB\_1 compare match is set as the TCNT clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGR registers (TGRA\_0 to TGRD\_0, TGRA\_1), outputting a 5-phase PWM waveform.

In this case, the value set in TGRB\_1 is used as the cycle, and the values set in the other TGRs are used as the duty levels.



**Figure 10-22 Example of PWM Mode Operation (2)**

Figure 10-23 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode.



**Figure 10-23 Example of PWM Mode Operation (3)**

### 10.4.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and TCNT is incremented/decremented accordingly. This mode 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 TPSC0 to TPSC2 and bits CKEG0 and CKEG1 in TCR. However, the functions of bits CCLR0 and CCLR1 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.

If overflow occurs when TCNT is counting up, the TCFV flag in TSR is set; if underflow occurs when TCNT is counting down, the TCFU flag is set.

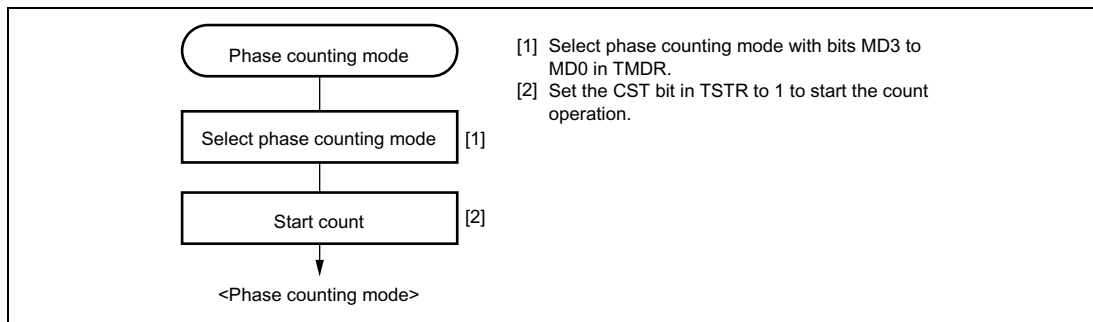
The TCFD bit in TSR is the count direction flag. Reading the TCFD flag reveals whether TCNT is counting up or down.

Table 10-31 shows the correspondence between external clock pins and channels.

**Table 10-31 Phase Counting Mode Clock Input Pins**

| 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   |

**Example of Phase Counting Mode Setting Procedure:** Figure 10-24 shows an example of the phase counting mode setting procedure.

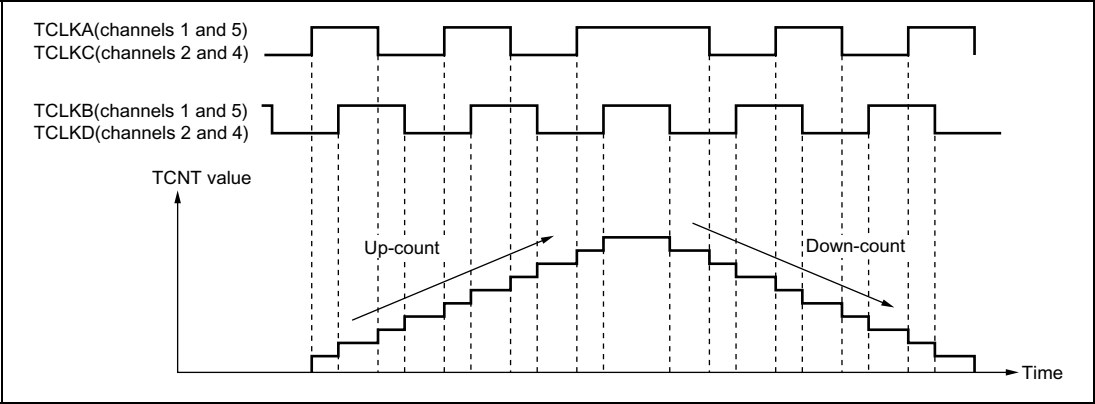


**Figure 10-24 Example of Phase Counting Mode Setting Procedure**

**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 10-25 shows an example of phase counting mode 1 operation, and table 10-32 summarizes the TCNT up/down-count conditions.



**Figure 10-25 Example of Phase Counting Mode 1 Operation**

**Table 10-32 Up/Down-Count Conditions in Phase Counting Mode 1**

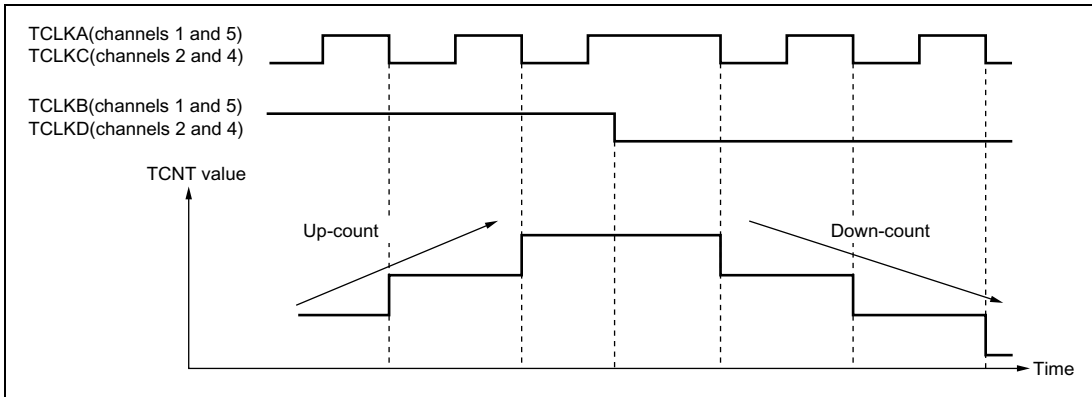
| TCLKA (Channels 1 and 5)<br>TCLKC (Channels 2 and 4)                               | TCLKB (Channels 1 and 5)<br>TCLKD (Channels 2 and 4)                                | Operation  |
|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|------------|
| 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

2. Phase counting mode 2

Figure 10-26 shows an example of phase counting mode 2 operation, and table 10-33 summarizes the TCNT up/down-count conditions.



**Figure 10-26 Example of Phase Counting Mode 2 Operation**

**Table 10-33 Up/Down-Count Conditions in Phase Counting Mode 2**

| <b>TCLKA (Channels 1 and 5)<br/>TCLKC (Channels 2 and 4)</b>                     | <b>TCLKB (Channels 1 and 5)<br/>TCLKD (Channels 2 and 4)</b>                      | <b>Operation</b> |
|----------------------------------------------------------------------------------|-----------------------------------------------------------------------------------|------------------|
| High level                                                                       |  | Don't care       |
| Low level                                                                        |  | Don't care       |
|  | Low level                                                                         | Don't care       |
|  | High level                                                                        | Up-count         |
| High level                                                                       |  | Don't care       |
| Low level                                                                        |  | Don't care       |
|  | High level                                                                        | Don't care       |
|  | Low level                                                                         | Down-count       |

**Legend**

 : Rising edge  
 : Falling edge

3. Phase counting mode 3

Figure 10-27 shows an example of phase counting mode 3 operation, and table 10-34 summarizes the TCNT up/down-count conditions.

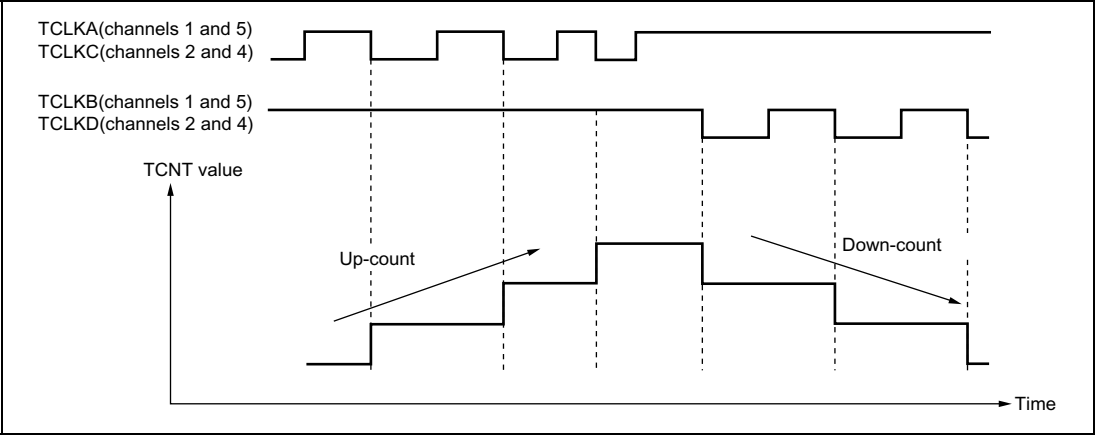


Figure 10-27 Example of Phase Counting Mode 3 Operation

Table 10-34 Up/Down-Count Conditions in Phase Counting Mode 3

| TCLKA (Channels 1 and 5)<br>TCLKC (Channels 2 and 4)                               | TCLKB (Channels 1 and 5)<br>TCLKD (Channels 2 and 4)                                | Operation  |
|------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|------------|
| High level                                                                         |    | Don't care |
| Low level                                                                          |    | Don't care |
|    | Low level                                                                           | Don't care |
|    | High level                                                                          | Up-count   |
| High level                                                                         |    | Down-count |
| Low level                                                                          |  | Don't care |
|  | High level                                                                          | Don't care |
|  | Low level                                                                           | Don't care |

Legend

-  : Rising edge
-  : Falling edge

4. Phase counting mode 4

Figure 10-28 shows an example of phase counting mode 4 operation, and table 10-35 summarizes the TCNT up/down-count conditions.

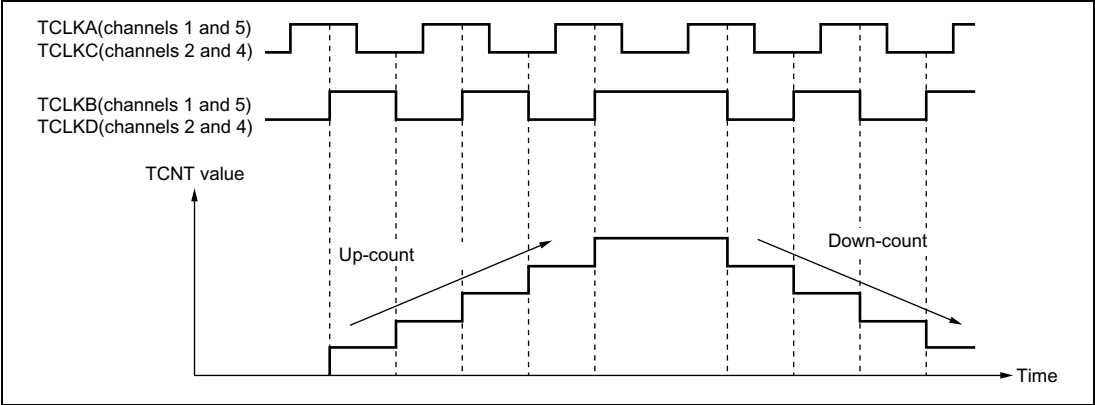


Figure 10-28 Example of Phase Counting Mode 4 Operation

Table 10-35 Up/Down-Count Conditions in Phase Counting Mode 4

| TCLKA (Channels 1 and 5)<br>TCLKC (Channels 2 and 4)                               | TCLKB (Channels 1 and 5)<br>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

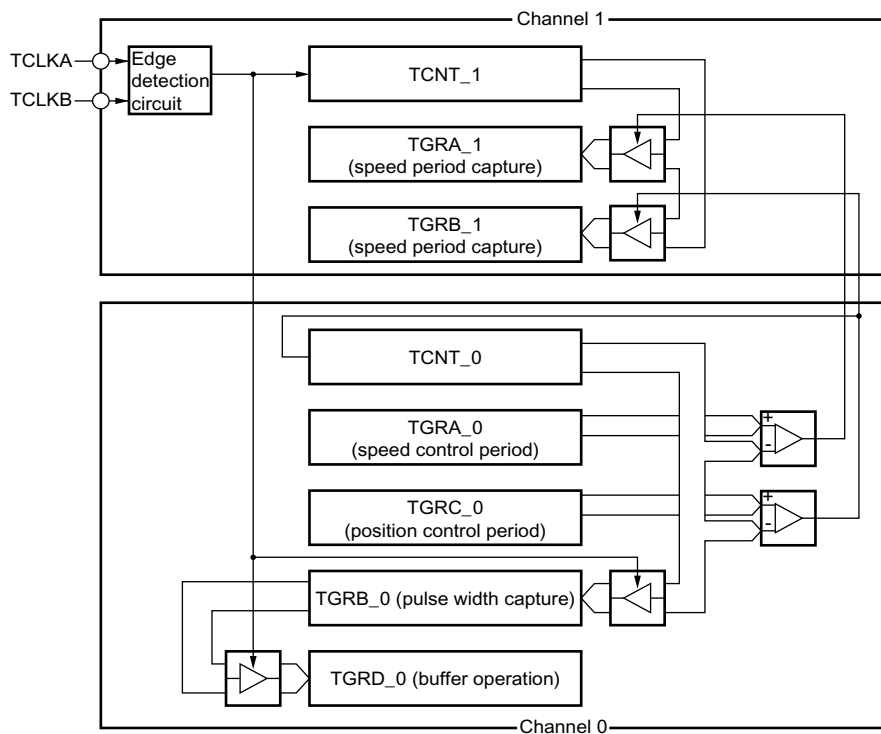
**Phase Counting Mode Application Example:** Figure 10-29 shows an example in which channel 1 is in phase counting mode, and channel 1 is coupled with channel 0 to input servo motor 2-phase encoder pulses in order to detect position or speed.

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 period and position control period. 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 the pulse widths of 2-phase encoder 4-multiplication pulses are detected.

TGRA\_1 and TGRB\_1 for channel 1 are designated for input capture, and channel 0 TGRA\_0 and TGRC\_0 compare matches are selected as the input capture source and store the up/down-counter values for the control periods.

This procedure enables the accurate detection of position and speed.



**Figure 10-29 Phase Counting Mode Application Example**

## 10.5 Interrupts

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/disabled bit, allowing the 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, however the priority order within a channel is fixed. For details, see section 5, Interrupt Controller.

Table 10-36 lists the TPU interrupt sources.

**Table 10-36 TPU Interrupts**

| Channel | Name   | Interrupt Source                   | Interrupt Flag | DTC          |
|---------|--------|------------------------------------|----------------|--------------|
|         |        |                                    |                | Activation   |
| 0       | TGIA_0 | TGRA_0 input capture/compare match | TGFA_0         | Possible     |
|         | TGIB_0 | TGRB_0 input capture/compare match | TGFB_0         | Possible     |
|         | TGIC_0 | TGRC_0 input capture/compare match | TGFC_0         | Possible     |
|         | TGID_0 | TGRD_0 input capture/compare match | TGFD_0         | Possible     |
|         | TCIV_0 | TCNT_0 overflow                    | TCFV_0         | Not possible |
| 1       | TGIA_1 | TGRA_1 input capture/compare match | TGFA_1         | Possible     |
|         | TGIB_1 | TGRB_1 input capture/compare match | TGFB_1         | Possible     |
|         | TCIV_1 | TCNT_1 overflow                    | TCFV_1         | Not possible |
|         | TCIU_1 | TCNT_1 underflow                   | TCFU_1         | Not possible |
| 2       | TGIA_2 | TGRA_2 input capture/compare match | TGFA_2         | Possible     |
|         | TGIB_2 | TGRB_2 input capture/compare match | TGFB_2         | Possible     |
|         | TCIV_2 | TCNT_2 overflow                    | TCFV_2         | Not possible |
|         | TCIU_2 | TCNT_2 underflow                   | TCFU_2         | Not possible |
| 3       | TGIA_3 | TGRA_3 input capture/compare match | TGFA_3         | Possible     |
|         | TGIB_3 | TGRB_3 input capture/compare match | TGFB_3         | Possible     |
|         | TGIC_3 | TGRC_3 input capture/compare match | TGFC_3         | Possible     |
|         | TGID_3 | TGRD_3 input capture/compare match | TGFD_3         | Possible     |
|         | TCIV_3 | TCNT_3 overflow                    | TCFV_3         | Not possible |
| 4       | TGIA_4 | TGRA_4 input capture/compare match | TGFA_4         | Possible     |
|         | TGIB_4 | TGRB_4 input capture/compare match | TGFB_4         | Possible     |
|         | TCIV_4 | TCNT_4 overflow                    | TCFV_4         | Not possible |
|         | TCIU_4 | TCNT_4 underflow                   | TCFU_4         | Not possible |
| 5       | TGIA_5 | TGRA_5 input capture/compare match | TGFA_5         | Possible     |
|         | TGIB_5 | TGRB_5 input capture/compare match | TGFB_5         | Possible     |
|         | TCIV_5 | TCNT_5 overflow                    | TCFV_5         | Not possible |
|         | TCIU_5 | TCNT_5 underflow                   | TCFU_5         | Not possible |

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

**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. 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. The TPU has six overflow interrupts, one for each channel.

**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 has four underflow interrupts, one each for channels 1, 2, 4, and 5.

## 10.6 DTC Activation

The DTC can be activated by the TGR input capture/compare match interrupt for a channel. For details, see section 8, Data Transfer Controller (DTC).

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.

## 10.7 A/D Converter Activation

The A/D converter can be activated by the TGRA input capture/compare match for a channel.

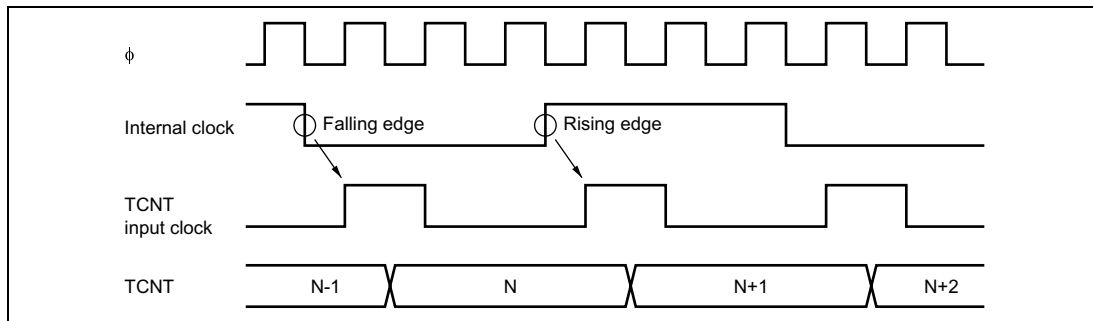
If the TTGE bit in TIER is set to 1 when the TGFA flag in TSR is set to 1 by the occurrence of a TGRA input capture/compare match on a particular channel, a request to begin A/D conversion is sent to the A/D converter. If the TPU conversion start trigger has been selected on the A/D converter side at this time, A/D conversion is begun.

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.

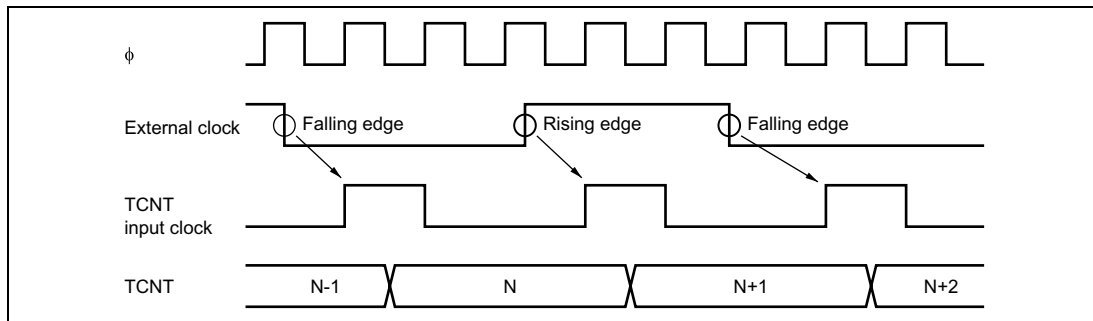
## 10.8 Operation Timing

### 10.8.1 Input/Output Timing

**TCNT Count Timing:** Figure 10-30 shows TCNT count timing in internal clock operation, and figure 10-31 shows TCNT count timing in external clock operation.



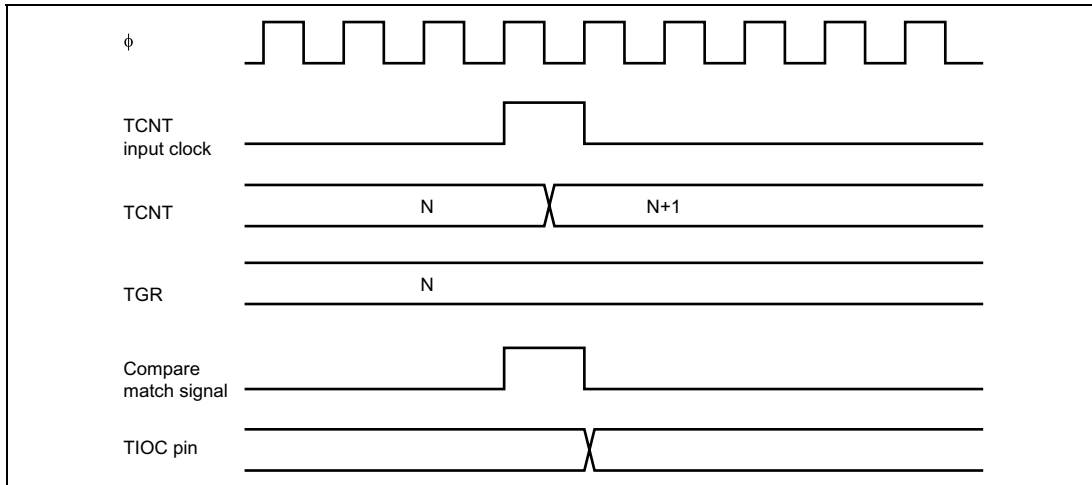
**Figure 10-30 Count Timing in Internal Clock Operation**



**Figure 10-31 Count Timing in External Clock Operation**

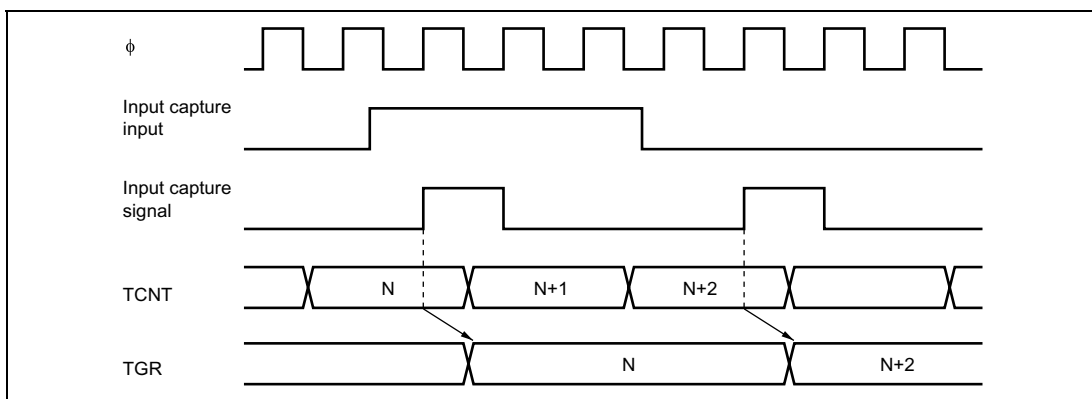
**Output Compare Output Timing:** A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the output value set in TIOR is output at the output compare output pin. After a match between TCNT and TGR, the compare match signal is not generated until the TCNT input clock is generated.

Figure 10-32 shows output compare output timing.



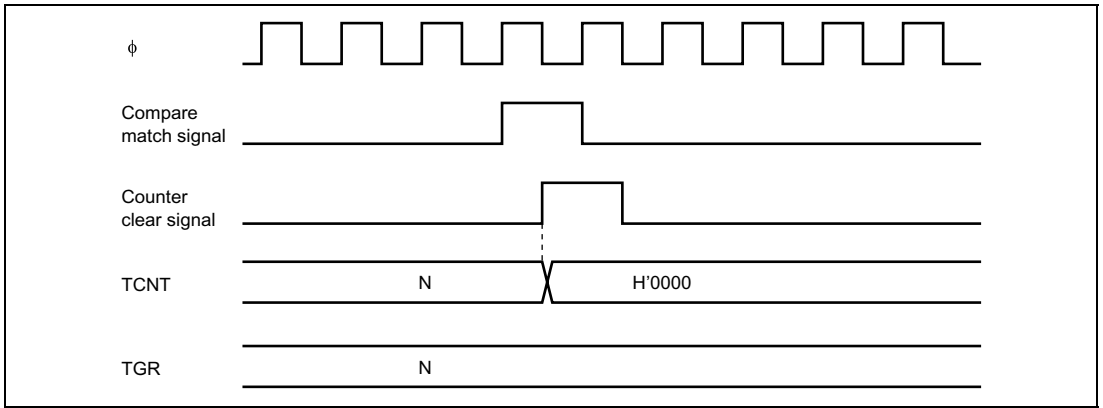
**Figure 10-32 Output Compare Output Timing**

**Input Capture Signal Timing:** Figure 10-33 shows input capture signal timing.

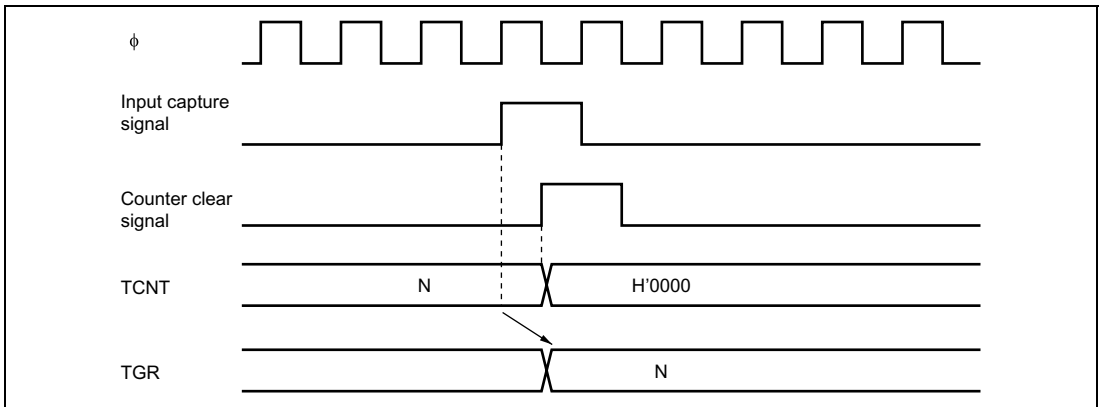


**Figure 10-33 Input Capture Input Signal Timing**

**Timing for Counter Clearing by Compare Match/Input Capture:** Figure 10-34 shows the timing when counter clearing on compare match is specified, and figure 10-35 shows the timing when counter clearing on input capture is specified.

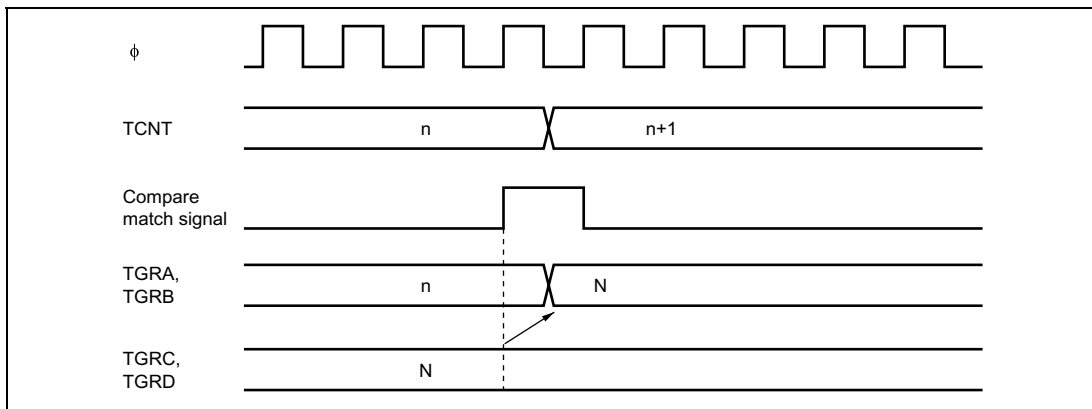


**Figure 10-34 Counter Clear Timing (Compare Match)**

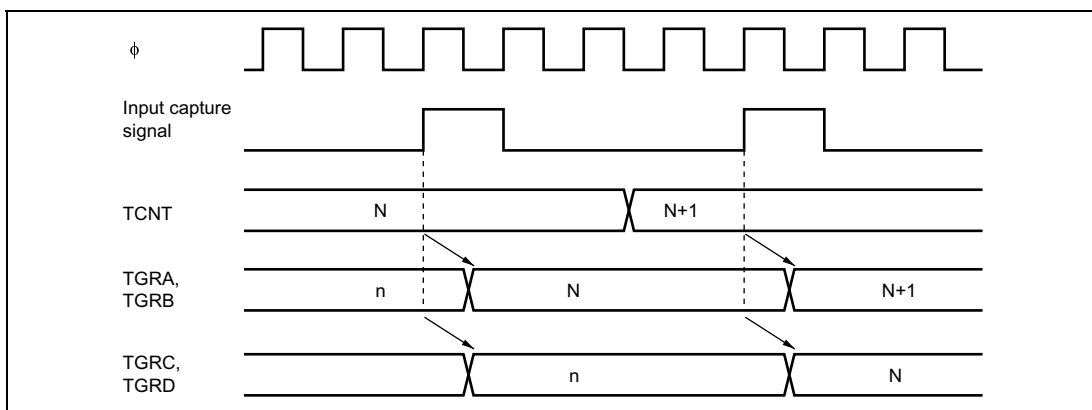


**Figure 10-35 Counter Clear Timing (Input Capture)**

**Buffer Operation Timing:** Figures 10-36 and 10-37 show the timing in buffer operation.



**Figure 10-36 Buffer Operation Timing (Compare Match)**



**Figure 10-37 Buffer Operation Timing (Input Capture)**

10.8.2 Interrupt Signal Timing

**TGF Flag Setting Timing in Case of Compare Match:** Figure 10-38 shows the timing for setting of the TGF flag in TSR on compare match, and TGI interrupt request signal timing.

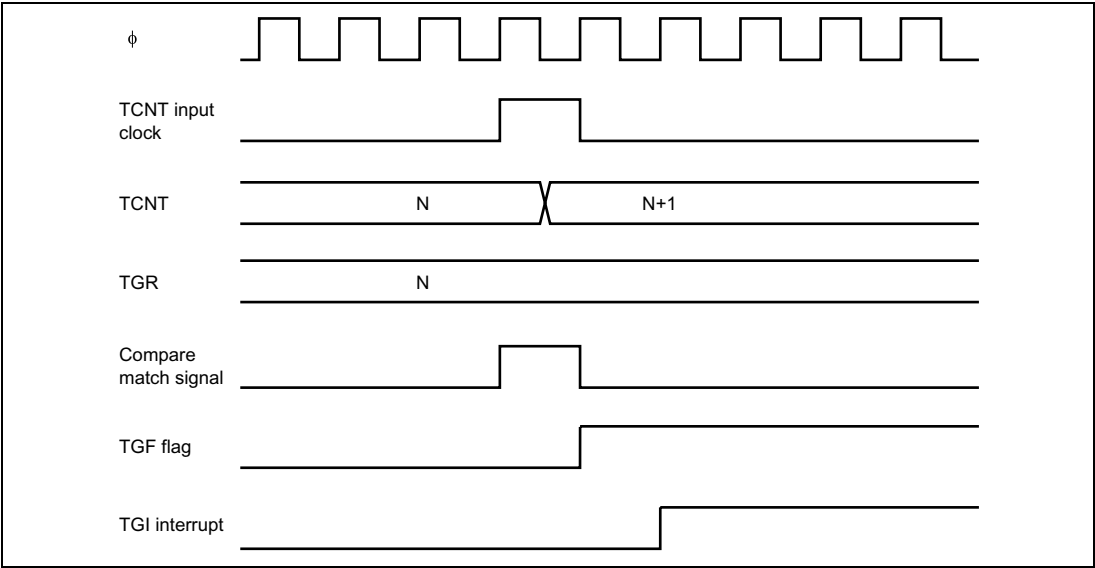


Figure 10-38 TGI Interrupt Timing (Compare Match)

**TGF Flag Setting Timing in Case of Input Capture:** Figure 10-39 shows the timing for setting of the TGF flag in TSR on input capture, and TGI interrupt request signal timing.

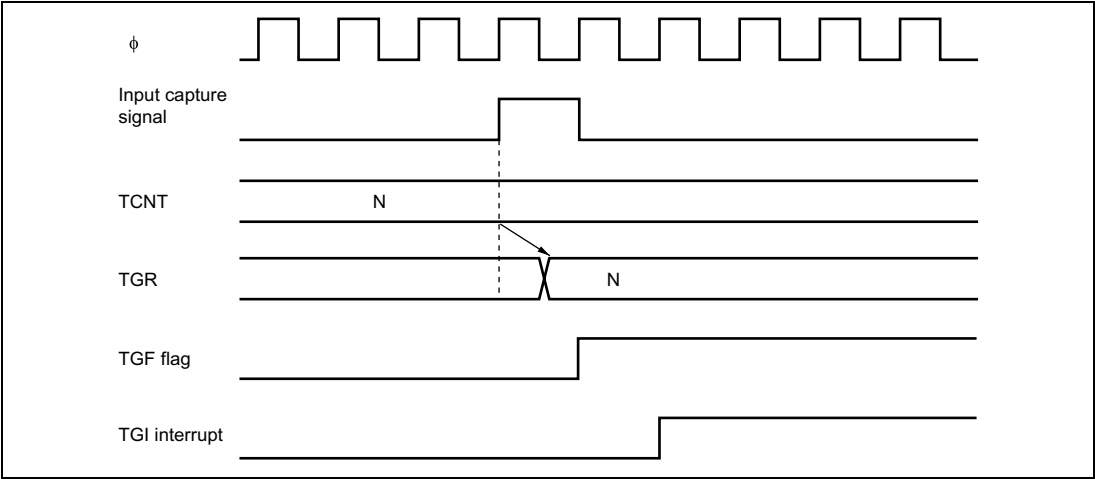
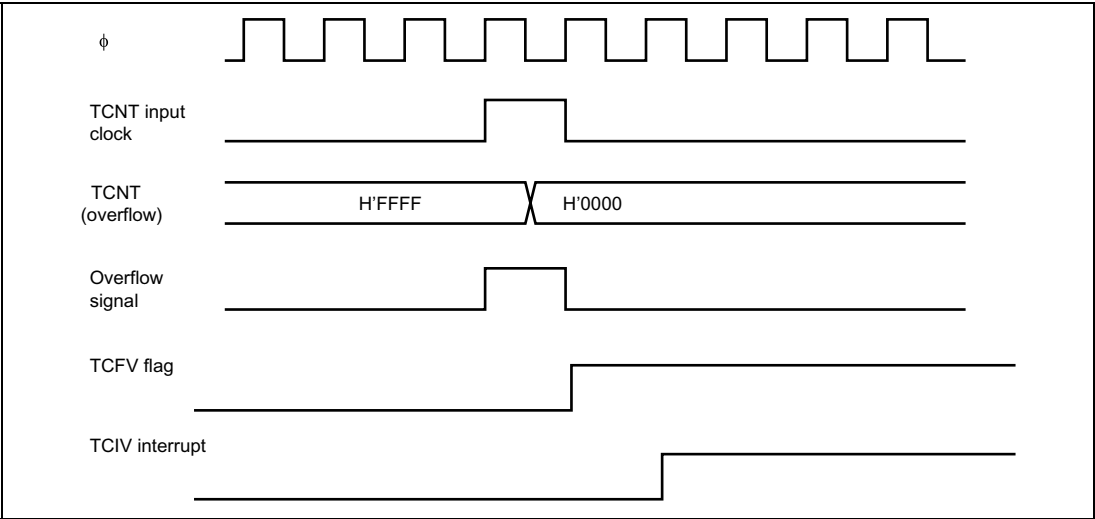


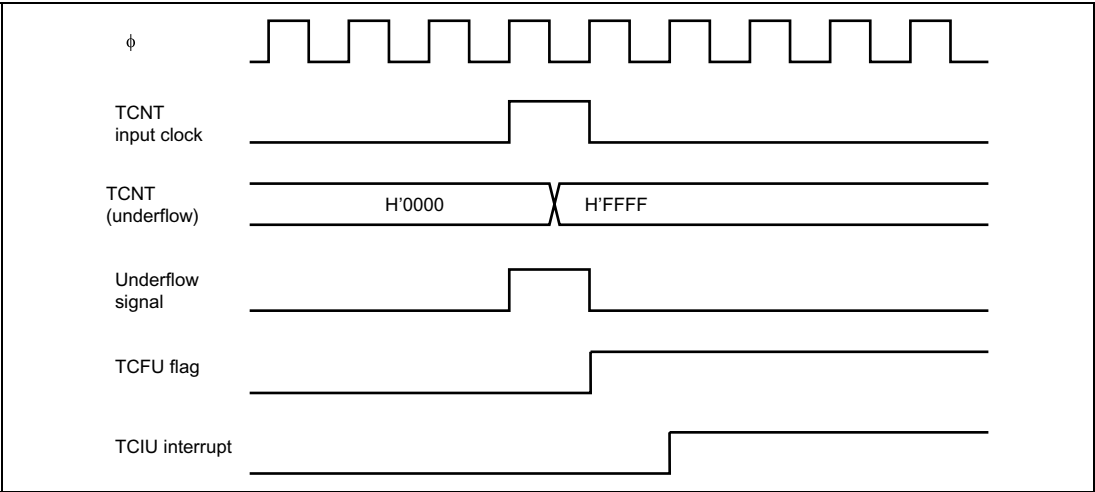
Figure 10-39 TGI Interrupt Timing (Input Capture)

**TCFV Flag/TCFU Flag Setting Timing:** Figure 10-40 shows the timing for setting of the TCFV flag in TSR on overflow, and TCIV interrupt request signal timing.

Figure 10-41 shows the timing for setting of the TCFU flag in TSR on underflow, and TCIU interrupt request signal timing.

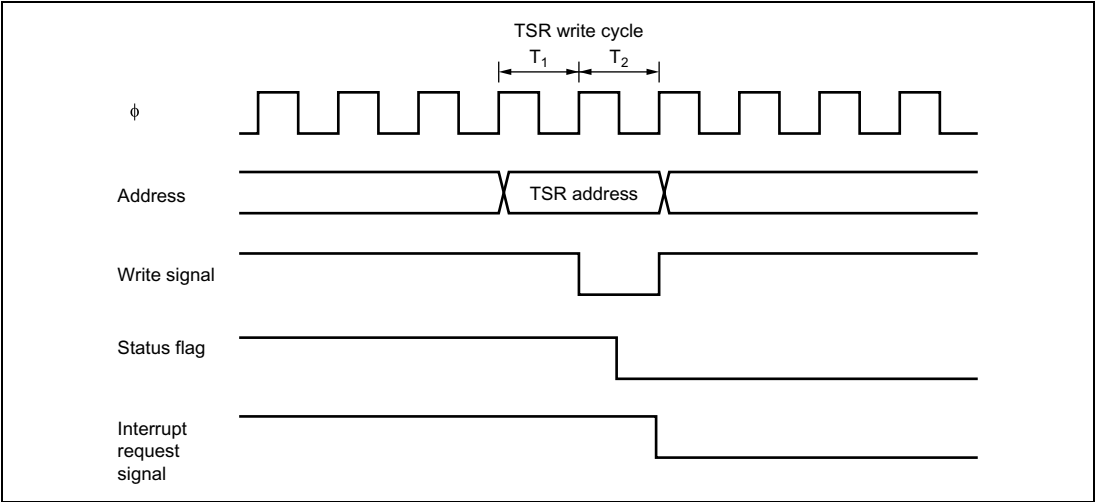


**Figure 10-40 TCIV Interrupt Setting Timing**

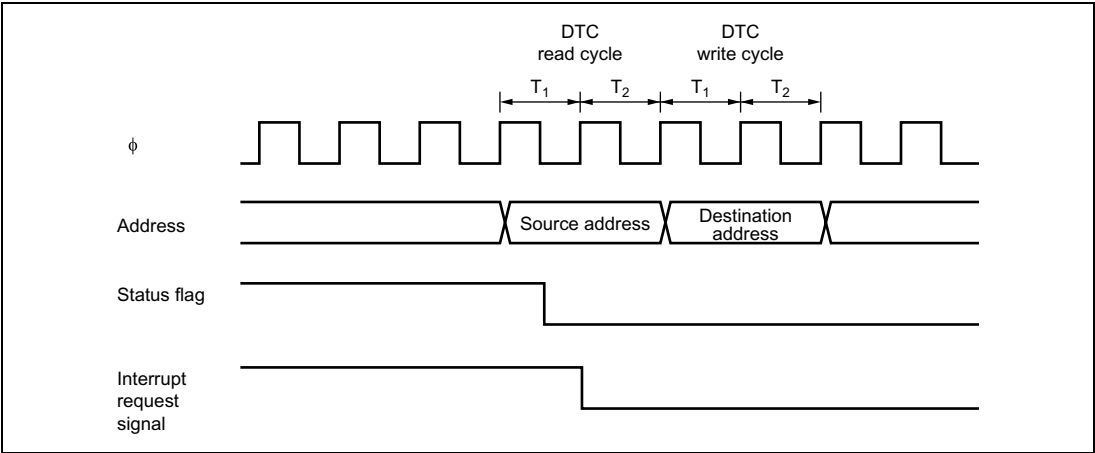


**Figure 10-41 TCIU Interrupt Setting Timing**

**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 is activated, the flag is cleared automatically. Figure 10-42 shows the timing for status flag clearing by the CPU, and figure 10-43 shows the timing for status flag clearing by the DTC.



**Figure 10-42 Timing for Status Flag Clearing by CPU**



**Figure 10-43 Timing for Status Flag Clearing by DTC Activation**

## 10.9 Usage Notes

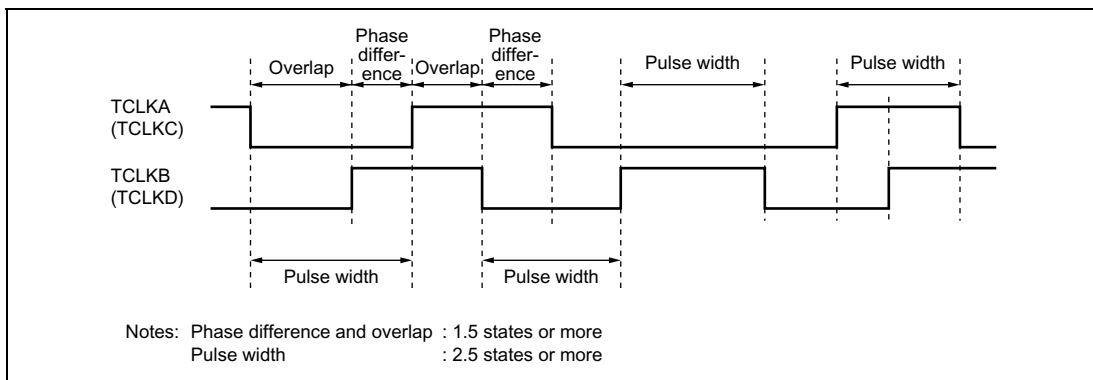
### 10.9.1 Module Stop Mode Setting

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 20, Power-Down Modes.

### 10.9.2 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 at narrower pulse widths.

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 10-44 shows the input clock conditions in phase counting mode.



**Figure 10-44 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode**

### 10.9.3 Caution on Period Setting

When counter clearing on 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:

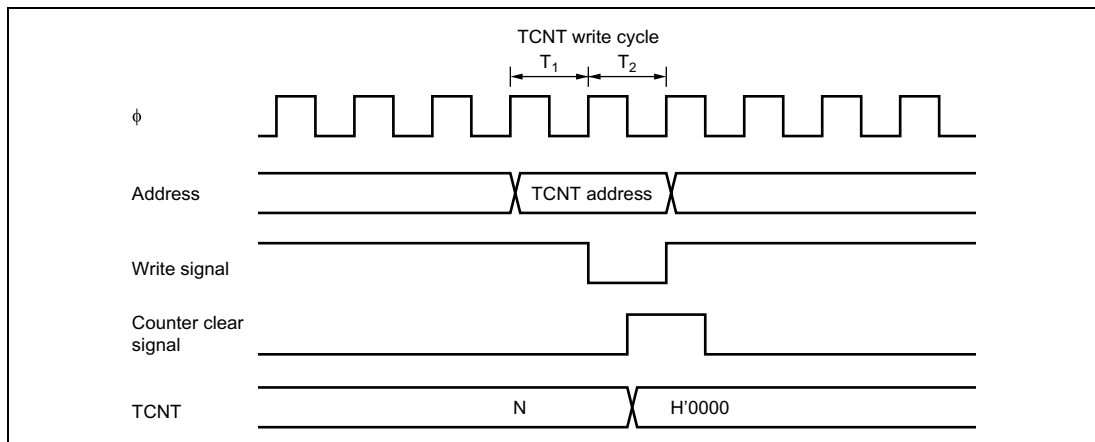
$$f = \frac{\varnothing}{(N + 1)}$$

Where  $f$  : Counter frequency  
 $\phi$  : Operating frequency  
 $N$  : TGR set value

#### 10.9.4 Contention between TCNT Write and Clear Operations

If the counter clear 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 10-45 shows the timing in this case.

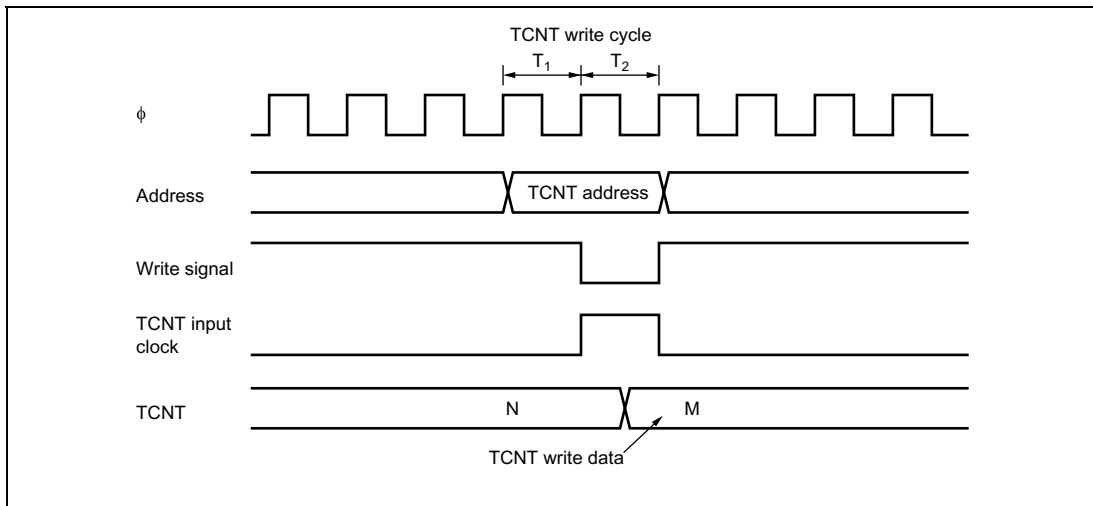


**Figure 10-45 Contention between TCNT Write and Clear Operations**

### 10.9.5 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 10-46 shows the timing in this case.

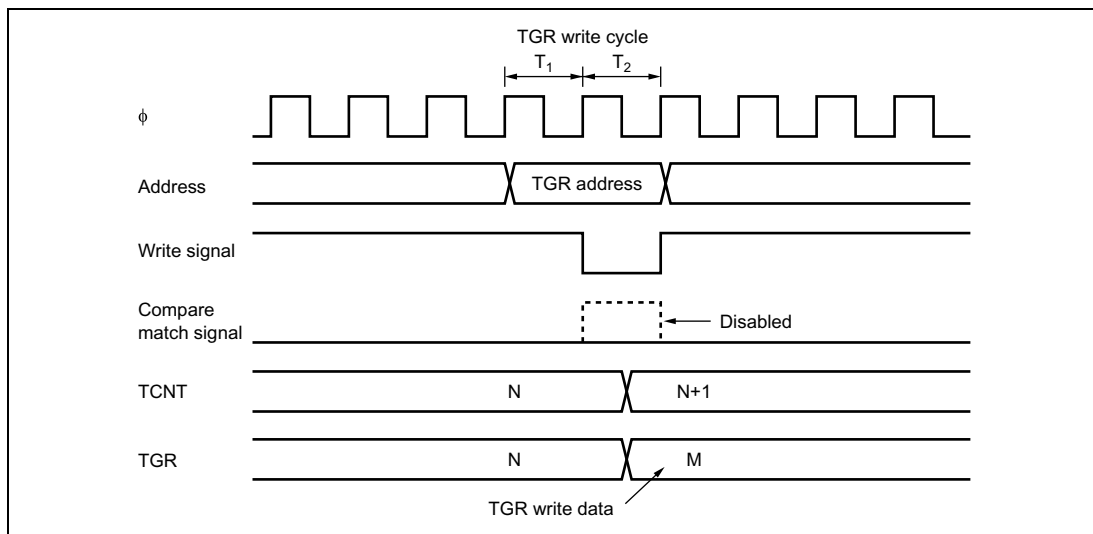


**Figure 10-46 Contention between TCNT Write and Increment Operations**

### 10.9.6 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 inhibited. A compare match does not occur even if the previous value is written.

Figure 10-47 shows the timing in this case.

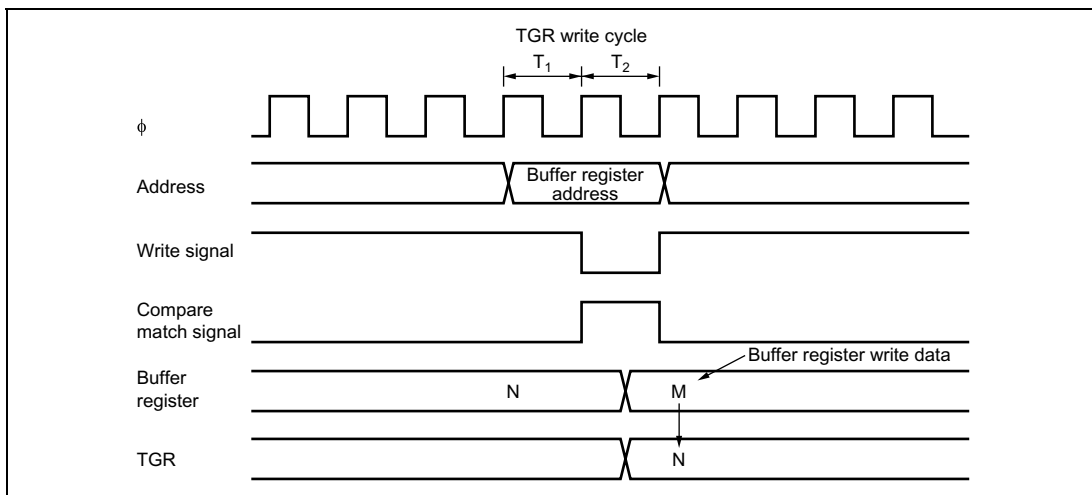


**Figure 10-47 Contention between TGR Write and Compare Match**

### 10.9.7 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 that is transferred to TGR by the buffer operation will be that in the buffer prior to the write.

Figure 10-48 shows the timing in this case.

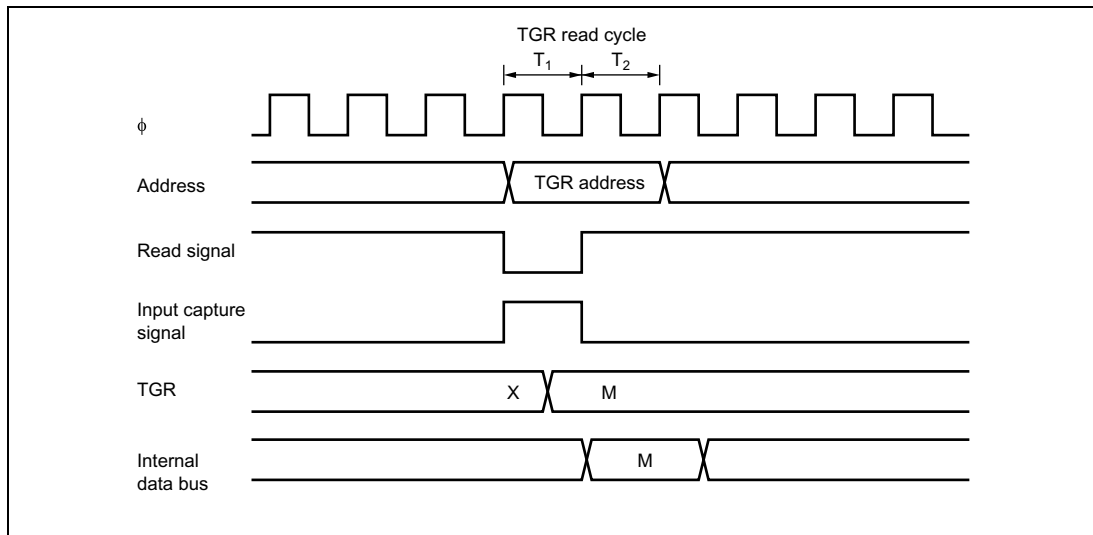


**Figure 10-48 Contention between Buffer Register Write and Compare Match**

### 10.9.8 Contention between TGR Read and Input Capture

If an input capture signal is generated in the  $T_1$  state of a TGR read cycle, the data that is read will be that in the buffer after input capture transfer.

Figure 10-49 shows the timing in this case.

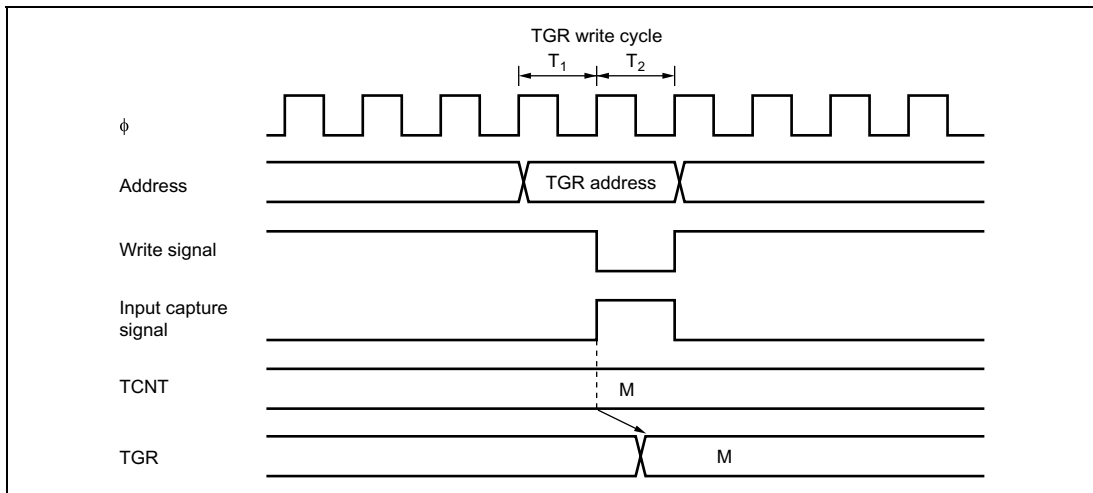


**Figure 10-49 Contention between TGR Read and Input Capture**

### 10.9.9 Contention between TGR Write and Input Capture

If an 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 10-50 shows the timing in this case.

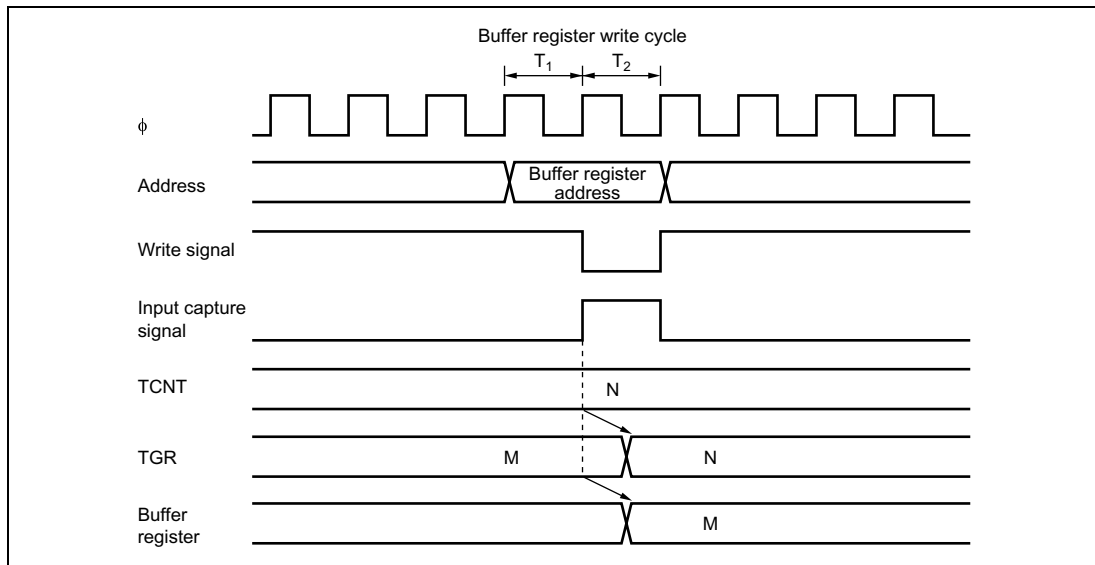


**Figure 10-50 Contention between TGR Write and Input Capture**

### 10.9.10 Contention between Buffer Register Write and Input Capture

If an 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 10-51 shows the timing in this case.

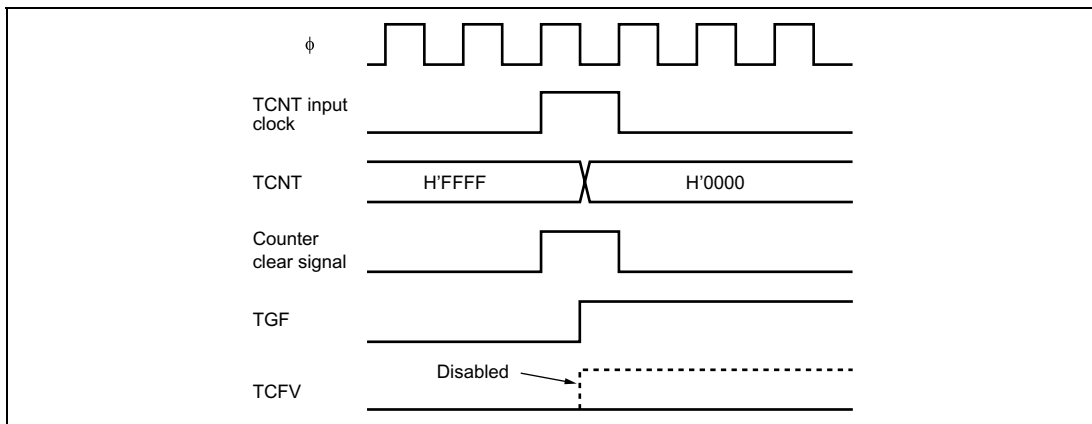


**Figure 10-51 Contention between Buffer Register Write and Input Capture**

### 10.9.11 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 10-52 shows the operation timing when a TGR compare match is specified as the clearing source, and when H'FFFF is set in TGR.

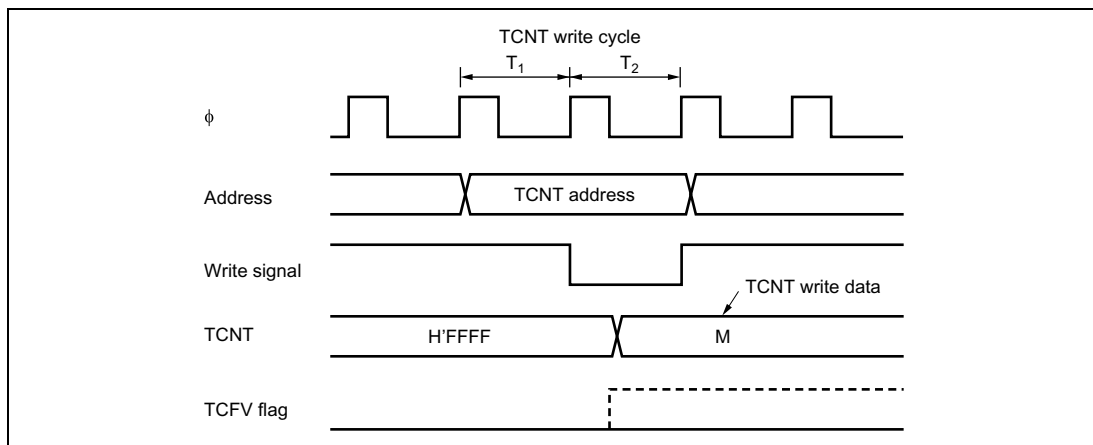


**Figure 10-52 Contention between Overflow and Counter Clearing**

### 10.9.12 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, and overflow/underflow occurs, the TCNT write takes precedence and the TCFV/TCFU flag in TSR is not set.

Figure 10-53 shows the operation timing when there is contention between TCNT write and overflow.



**Figure 10-53 Contention between TCNT Write and Overflow**

### 10.9.13 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.

### 10.9.14 Interrupts in 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.

# Section 11 Motor Management Timer (MMT)

Motor Management Timer (MMT) can output 6-phase PWM waveforms with non-overlap times.

Figure 11-1 shows a block diagram of the MMT.

## 11.1 Features

- Triangular wave comparison type 6-phase PWM waveform output with non-overlap times
- Non-overlap times generated by timer dead time counters
- Toggle output synchronized with PWM period
- Counter clearing on an external signal
- Data transfer by DTC activation
- Output-off functions
- PWM output halted by external signal
- PWM output halted when oscillation stops
- Module stop mode can be set



## 11.2 Input/Output Pins

Table 11-1 shows the pin configuration of the MMT.

**Table 11-1 Pin Configuration**

| Name                    | I/O    | Function                                   |
|-------------------------|--------|--------------------------------------------|
| $\overline{\text{PCI}}$ | Input  | Counter clear signal input                 |
| PCO                     | Output | Toggle output synchronized with PWM period |
| PUOA                    | Output | PWMU phase output (positive phase)         |
| PUOB                    | Output | PWMU phase output (negative phase)         |
| PVOA                    | Output | PWMV phase output (positive phase)         |
| PVOB                    | Output | PWMV phase output (negative phase)         |
| PWOA                    | Output | PWMW phase output (positive phase)         |
| PWOB                    | Output | PWMW phase output (negative phase)         |

## 11.3 Register Descriptions

The MMT has the following registers. These registers are initialized on power-on reset or in hardware standby mode. They are not initialized in module stop mode or software standby mode. For details on register addresses, refer to Appendix A, On-Chip I/O Register.

- Timer mode register (TMDR)
- Timer control register (TCNR)
- Timer status register (TSR)
- Timer counter (TCNT)
- Timer buffer register U (TBRU)
- Timer buffer register V (TBRV)
- Timer buffer register W (TBRW)
- Timer general register UU (TGRUU)
- Timer general register VU (TGRVU)
- Timer general register WU (TGRWU)
- Timer general register U (TGRU)
- Timer general register V (TGRV)
- Timer general register W (TGRW)
- Timer general register UD (TGRUD)
- Timer general register VD (TGRVD)
- Timer general register WD (TGRWD)
- Timer dead time counter 0 (TDCNT0)

- Timer dead time counter 1 (TDCNT1)
- Timer dead time counter 2 (TDCNT2)
- Timer dead time counter 3 (TDCNT3)
- Timer dead time counter 4 (TDCNT4)
- Timer dead time counter 5 (TDCNT5)
- Timer dead time data register (TDDR)
- Timer period buffer register (TPBR)
- Timer period data register (TPDR)
- MMT pin control register (MMTPC)

### 11.3.1 Timer Mode Register (TMDR)

The timer mode register (TMDR) sets the operating mode and selects the PWM output level.

| Bit    | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                      |
|--------|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7 to 4 | —        | 0             | —   | Reserved<br>These bits are always read as 0 and should only be written with 0.                                                                                                                                   |
| 3      | OLSN     | 0             | R/W | Output Level Select N<br>Selects the negative phase output level in the operating modes.<br>0: Active level is low<br>1: Active level is high                                                                    |
| 2      | OLSP     | 0             | R/W | Output Level Select P<br>Selects the positive phase output level in the operating modes.<br>0: Active level is low<br>1: Active level is high                                                                    |
| 1      | MD1      | 0             | R/W | Modes 0 to 3                                                                                                                                                                                                     |
| 0      | MD0      | 0             | R/W | These bits set the timer operating mode.<br>00: Operation halted<br>01: Operating mode 1 (transfer at crest)<br>10: Operating mode 2 (transfer at trough)<br>11: Operating mode 3 (transfer at crest and trough) |

### 11.3.2 Timer Control Register (TCNR)

The timer control register (TCNR) controls the enabling or disabling of interrupt requests, selects the enabling or disabling of register access, selects counter operation or halting, and controls the enabling or disabling of toggle output synchronized with the PWM period.

| Bit    | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                               |
|--------|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7      | —        | 0             | R/W | Reserved<br>This bit is always read as 0. Only 0 should be written to this bit.                                                                                                                                                                                                                                                                           |
| 6      | CST      | 0             | R/W | Timer Counter Start<br>Selects operation or halting of the timer counter (TCNT) and timer dead time counter (TDCNT).<br>0: TCNT and TDCNT operation is halted<br>1: TCNT and TDCNT perform count operations                                                                                                                                               |
| 5      | RPRO     | 0             | R/W | Register Protect<br>Enables or disables the reading of registers other than TSR, and enables or disables writes to registers other than TBRU to TBRW, TPBR, and TSR. Writes to TCNR itself are also disabled. Note that reset input is necessary in order to write to these registers again.<br>0: Register access enabled<br>1: Register access disabled |
| 4 to 2 | —        | 0             | —   | Reserved<br>These bits are always read as 0. Only 0 should be written to these bits.                                                                                                                                                                                                                                                                      |
| 1      | TGIEN    | 0             | R/W | TGR Interrupt Enable N<br>Enables or disables interrupt requests by the TGFN bit when TGFN is set to 1 in the TSR register.<br>0: Interrupt requests by TGFN bit disabled<br>1: Interrupt requests by TGFN bit enabled                                                                                                                                    |
| 0      | TGIEM    | 0             | R/W | TGR Interrupt Enable M<br>Enables or disables interrupt requests by the TGFM bit when TGFM is set to 1 in the TSR register.<br>0: Interrupt requests by TGFM bit disabled<br>1: Interrupt requests by TGFM bit enabled                                                                                                                                    |

### 11.3.3 Timer Status Register (TSR)

The timer status register holds status flags.

| Bit    | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                      |
|--------|----------|---------------|--------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7      | TCFG     | 1             | R      | Count Direction Flag<br>Status flag that indicates the count direction of the TCNT counter.<br>0: TCNT counts down<br>1: TCNT counts up                                                                                          |
| 6 to 2 | —        | 0             | —      | Reserved<br>These bits are always read as 0 and should only be written with 0.                                                                                                                                                   |
| 1      | TGFN     | 0             | R/(W)* | Output Compare Flag N<br>Status flag that indicates a compare match between TCNT and 2Td (Td: TDDR value).<br>[Setting condition]<br>When TCNT = 2Td<br>[Clearing condition]<br>When 0 is written to TGFN after reading TGFN = 1 |
| 0      | TGFM     | 0             | R/(W)* | Output Compare Flag M<br>Status flag that indicates a compare match between TCNT and the TPDR register.<br>[Setting condition]<br>When TCNT = TPDR<br>[Clearing condition]<br>When 0 is written to TGFM after reading TGFM = 1   |

Note: \* Can only be written with 0 for flag clearing.

### 11.3.4 Timer Counter (TCNT)

The timer counter (TCNT) is a 16-bit counter. Only 16-bit access can be used on TCNT; 8-bit access is not possible.

### **11.3.5 Timer Buffer Registers (TBR)**

The timer buffer registers (TBR) function as 16-bit buffer registers. The MMT has three TBR registers; TBRU, TBRV, and TBRW, each of which have two addresses; a buffer operation address (shown first) and a free operation address (shown second). A value written to the buffer operation address is transferred to the corresponding TGR at the timing set in bits MD1 and MD0 in the timer mode register (TMDR). A value set in the free operation address is transferred to the corresponding TGR immediately. Only 16-bit access can be used on the TBR registers; 8-bit access is not possible.

### **11.3.6 Timer General Registers (TGR)**

The timer general registers (TGR) function as 16-bit compare registers. The MMT has nine TGR registers, that are compared with the TCNT counter in the operating modes. Only 16-bit access can be used on the TGR registers; 8-bit access is not possible.

### **11.3.7 Timer Dead Time Counters (TDCNT)**

The timer dead time counters (TDCNT) are 16-bit read-only counters. Only 16-bit access can be used on the TDCNT counters; 8-bit access is not possible.

### **11.3.8 Timer Dead Time Data Register (TDDR)**

The timer dead time data register (TDDR) is a 16-bit register that sets the positive phase and negative phase non-overlap time (dead time). Only 16-bit access can be used on TDDR; 8-bit access is not possible.

### **11.3.9 Timer Period Buffer Register (TPBR)**

The timer period buffer register (TPBR) is a 16-bit register that functions as a buffer register for the TPDR register. A value of 1/2 the PWM carrier period should be set as the TPBR value. The TPBR value is transferred to the TPDR register at the transfer timing set in the TMDR register. Only 16-bit access can be used on TPBR; 8-bit access is not possible.

### **11.3.10 Timer Period Data Register (TPDR)**

The timer period data register (TPDR) functions as a 16-bit compare register. In the operating modes, the TPDR register value is constantly compared with the TCNT counter value, and when they match the TCNT counter changes its count direction from up to down. Only 16-bit access can be used on TPDR; 8-bit access is not possible.

### 11.3.11 MMT Pin Control Register (MMTPC)

The MMTPC is an 8-bit readable/writable register that enables or disables usage of the MMT pins.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                      |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------|
| 7   | PWOBE    | 0             | R/W | PWOB enabled<br>0: PB7 pin is port I/O pin/TPU I/O pin<br>1: PB7 pin is PWOB output pin of MMT                   |
| 6   | PWOAE    | 0             | R/W | PWOA enabled<br>0: PB6 pin is port I/O pin/TPU I/O pin<br>1: PB6 pin is PWOA output pin of MMT                   |
| 5   | PVOBE    | 0             | R/W | PVOB enabled<br>0: PB5 pin is port I/O pin/TPU I/O pin<br>1: PB5 pin is PVOB output pin of MMT                   |
| 4   | PVOAE    | 0             | R/W | PVOA enabled<br>0: PB4 pin is port I/O pin/TPU I/O pin<br>1: PB4 pin is PVOA output pin of MMT                   |
| 3   | PUOBE    | 0             | R/W | PUOB enabled<br>0: PB3 pin is port I/O pin/TPU I/O pin<br>1: PB3 pin is PUOB output pin of MMT                   |
| 2   | PUOAE    | 0             | R/W | PUOA enabled<br>0: PB2 pin is port I/O pin/TPU I/O pin<br>1: PB2 pin is PUOA output pin of MMT                   |
| 1   | PCOE     | 0             | R/W | PCO enabled<br>0: PB1 pin is port I/O pin/TPU I/O pin<br>1: PB1 pin is PCO output pin of MMT                     |
| 0   | PCIE     | 0             | R/W | PCI enabled<br>0: PB0 pin is port I/O pin/TPU I/O pin<br>1: PB0 pin is $\overline{\text{PCI}}$ output pin of MMT |

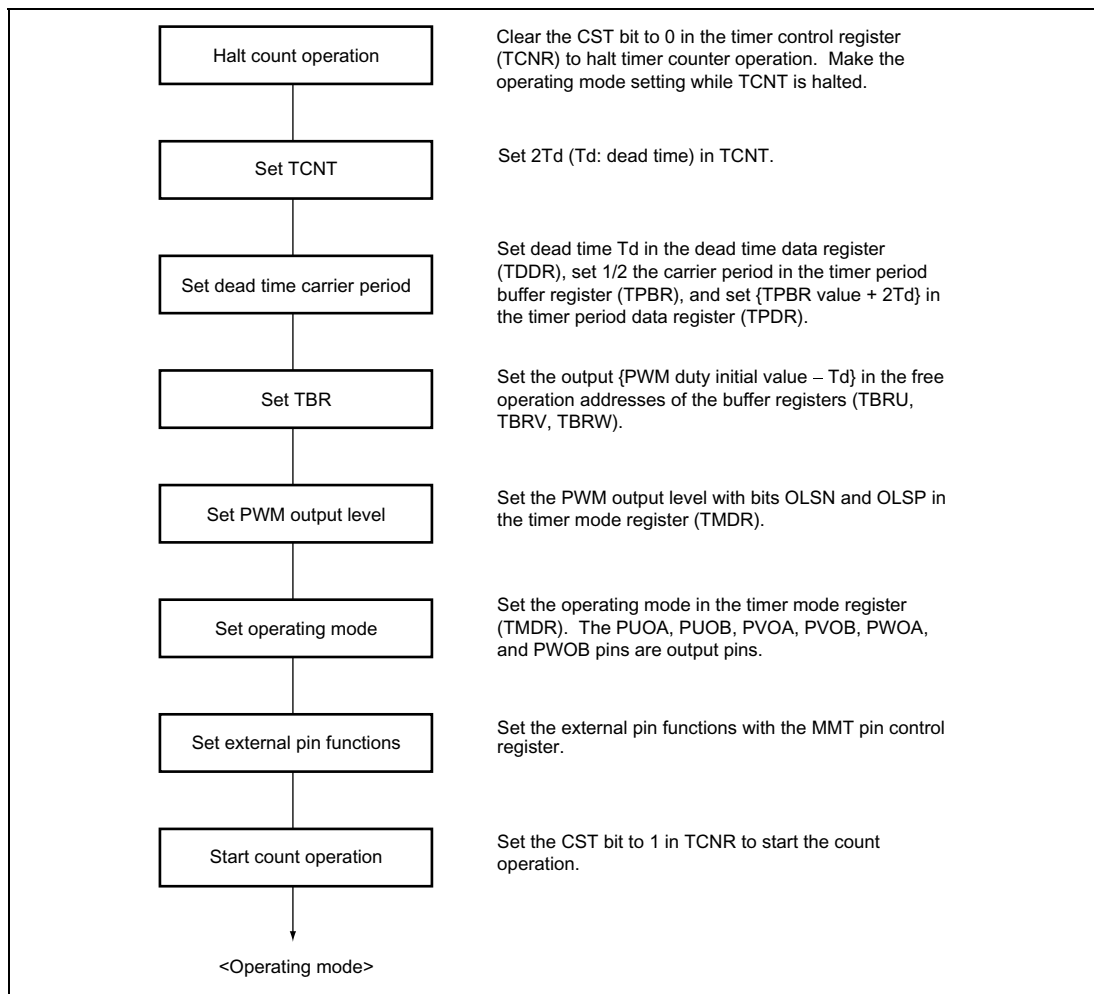
## 11.4 Operation

When the operating mode is selected, a 3-phase PWM waveform is output with a non-overlap relationship between the positive and negative phases.

The PUOA, PUOB, PVOA, PVOB, PWOA, and PWOB pins are PWM output pins, the PCO pin functions as a toggle output synchronized with the PWM waveform, and the  $\overline{\text{PCI}}$  pin functions as the counter clear signal input. The TCNT counter performs up- and down-count operations, whereas the TDCNT counters perform up-count operations.

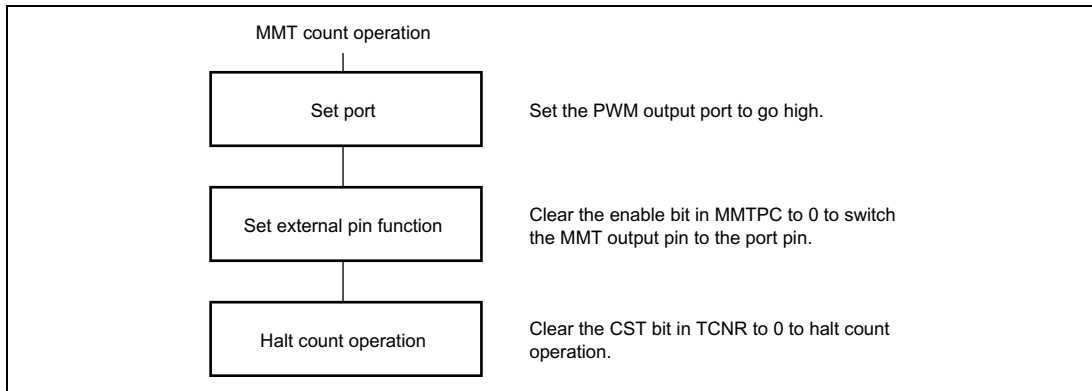
## 11.4.1 Sample Setting Procedure

An example of the operating mode setting procedure is shown in figure 11-2.



**Figure 11-2 Sample Operating Mode Setting Procedure**

Figure 11-3 shows the sample MMT canceling procedure.



**Figure 11-3 MMT Canceling Procedure**

**Count Operation:** Set 2Td (Td: value set in TDDR) as the initial value of the TCNT counter when CST bit in TCNR is set to 0.

When the CST bit is set to 1, TCNT counts up to {value set in TPBR + 2Td}, and then starts counting down. When TCNT reaches 2Td, it starts counting up again, and continues in this way.

TCNT is constantly compared with TGRU, TGRV, and TGRW. In addition, it is compared with TGRUU, TGRVU, TGRWU, and TPDR when counting up, and with TGRUD, TGRVD, TGRWD, and 2Td when counting down.

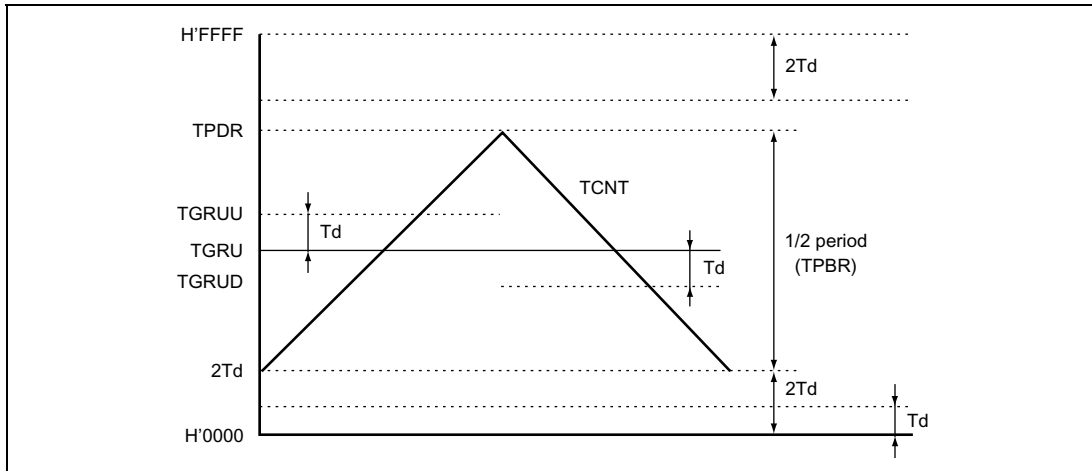
TDCNT0 to TDCNT5 are read-only counters. It is not necessary to set their initial values.

TDCNT0, TDCNT2, and TDCNT4 start counting up at the falling edge of a positive phase compare match output when TCNT is counting down. When they become equal to TDDR they are cleared to 0 and halt.

TDCNT1, TDCNT3, and TDCNT5 start counting up at the falling edge of a negative phase compare match output when TCNT is counting up. When they match TDDR they are cleared to 0 and halt.

TDCNT0 to TDCNT5 are compared with TDDR only while a count operation is in progress. No count operation is performed when the TDDR value is 0.

Figure 11-4 shows an example of the TCNT count operation.



**Figure 11-4 Example of TCNT Count Operation**

**Register Operation:** In the operating modes, four buffer registers and ten compare registers are used.

The registers that are constantly compared with the TCNT counter are TGRU, TGRV, and TGRW. In addition, TGRUU, TGRVU, TGRWU, and TPDR are compared with TCNT when TCNT is counting up, and TGRUD, TGRVD, TGRWD are compared with TCNT when TCNT is counting down. The buffer register for TPDR is TPBR; the buffer register for TGRUU, TGRU, and TGRUD is TBRU; the buffer register for TGRVU, TGRV, and TGRVD is TBRV; and the buffer register for TGRWU, TGRW, and TGRWD is TBRW.

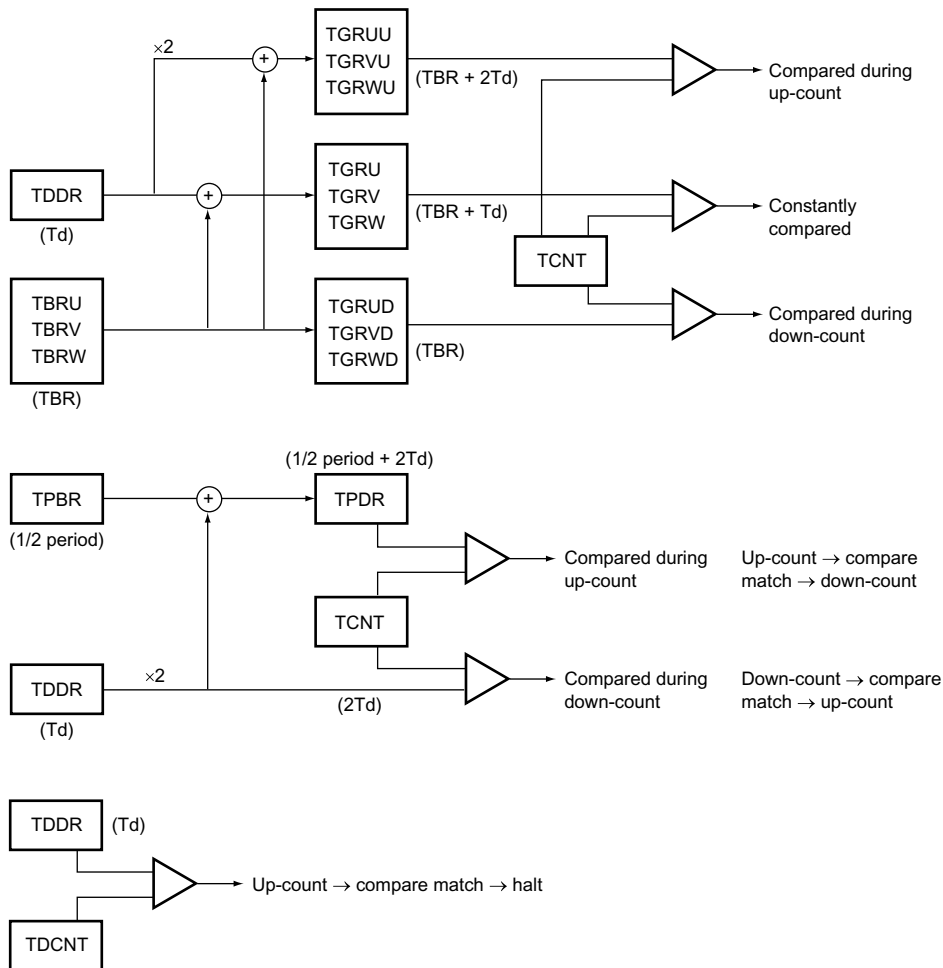
To change compare register data, the new data should be written to the corresponding buffer register. The buffer registers can be read and written to at all times. Data written to TPBR and the buffer operation addresses for TBRU to TBRW is transferred at the timing specified by bits MD1 and MD0 in the timer mode register (TMDR). Data written to the free operation addresses for TBRU to TBRW is transferred immediately.

After data transfer is completed, the relationship between the compare registers and buffer registers is as follows:

$\text{TGRU (TGRV, TGRW) value} = \text{TBRU (TBRV, TBRW) value} + T_d$  ( $T_d$ : value set in TDDR)  
 $\text{TGRUU (TGRVU, TGRWU) value} = \text{TBRU (TBRV, TBRW) value} + 2T_d$   
 $\text{TGRUD (TGRVD, TGRWD) value} = \text{TBRU (TBRV, TBRW) value}$   
 $\text{TPDR value} = \text{TPBR value} + 2T_d$

The values of TBRU to TBRW should always be set in the range H'0000 to H'FFFF – 2Td, and the value of TPBR should always be set in the range H'0000 to H'FFFF – 4Td.

Figure 11-5 shows examples of counter and register operations.



**Figure 11-5 Examples of Counter and Register Operations**

**Initial Settings:** In the operating modes, there are five registers that require initialization.

Make the following register settings before setting the operating mode with bits MD1 and MD0 in the timer mode register (TMDR).

Set the timer period buffer register (TPBR) to 1/2 the PWM carrier period, set dead time Td in the timer dead time data register (TDDR) (when outputting an ideal waveform, Td = H'0000), and set {TPBR value + 2Td} in the timer period data register (TPDR).

Set {PWM duty initial value – Td} in the free write operation addresses for TBRU to TBRW.

The values of TBRU to TBRW should always be set in the range H'0000 to H'FFFF – 2Td, and the value of TPBR should always be set in the range H'0000 to H'FFFF – 4Td.

**PWM Output Active Level Setting:** In the operating modes, the active level of PWM pulses is set with bits OLSN and OLSP in the timer mode register (TMDR).

The output level can be set for the three positive phases and the three negative phases of 6-phase output. The operating mode must be exited before setting or changing the output level.

**Dead Time Setting:** In the operating modes, PWM pulses are output with a non-overlap relationship between the positive and negative phases. This non-overlap time is known as the dead time. The non-overlap time is set in the timer dead time data register (TDDR). The dead time generation waveform is generated by comparing the value set in TDDR with the timer dead time counters (TDCNT) for each phase. The operating mode must be exited before changing the contents of TDDR.

**PWM Period Setting:** In the operating modes, 1/2 the PWM pulse period is set in the TPBR register. The TPBR value should always be set in the range H'0000 to H'FFFF – 4Td. The value set in TPBR is transferred to TPDR at the timing selected with bits MD1 and MD0 in the timer mode register (TMDR). After the transfer, the value in TPDR is {TPBR value + 2Td}.

The new PWM period is effective from the next period when data is updated at the TCNT counter crest, and from the same period when data is updated at the trough.

**Register Updating:** In the operating modes, buffer registers are used to update compare register data. Update data can be written to a buffer register at all times. The buffer register value is transferred to the compare register at the timing set by bits MD1 and MD0 in the timer mode register (TMDR) (except in the case of a write to the free operation address for TBRU to TBRW, in which case the value is transferred to the corresponding compare register immediately).

**Initial Output in Operating Modes:** The initial output in the operating modes is determined by the initial values of TBRU to TBRW.

Table 11-2 shows the relationship between the initial value of TBRU to TBRW and the initial output.

**Table 11-2 Initial Values of TBRU to TBRW and Initial Output**

| Initial Value of TBRU to TBRW | Initial Output     |                    |
|-------------------------------|--------------------|--------------------|
|                               | OLSP = 1, OLSN = 1 | OLSP = 0, OLSN = 0 |
| TBR = H'0000                  | Positive phase: 1  | Positive phase: 0  |
|                               | Negative phase: 0  | Negative phase: 1  |
| H'0000 < TBR ≤ Td             | Positive phase: 0  | Positive phase: 1  |
|                               | Negative phase: 0  | Negative phase: 1  |
| Td < TBR ≤ H'FFFF – 2Td       | Positive phase: 0  | Positive phase: 1  |
|                               | Negative phase: 1  | Negative phase: 0  |

**PWM Output Generation in Operating Modes:** In the operating modes, a 3-phase PWM waveform is output with a non-overlap relationship between the positive and negative phases. This non-overlap time is called the dead time.

The PWM waveform is generated from an output generation waveform generated by ANDing the compare output waveform with the dead time generation waveform. Waveform generation for one phase (the U-phase) is shown here. The V-phase and W-phase waveforms are generated in the same way.

#### 1. Compare Output Waveform

The compare output waveform is generated by comparing the values in the TCNT counter and the TGR registers.

For compare output waveform U phase A (CMOUA), 0 is output if  $TGRUU > TCNT$  in the T1 interval (when TCNT is counting up), and 1 is output if  $TGRUU \leq TCNT$ . In the T2 interval (when TCNT is counting down), 0 is output if  $TGRU > TCNT$ , and 1 is output if  $TGRU \leq TCNT$ .

For compare output waveform U phase B (CMOUB), 1 is output if  $TGRU > TCNT$  in the T1 interval, and 0 is output if  $TGRU \leq TCNT$ . In the T2 interval, 1 is output if  $TGRUD > TCNT$ , and 0 is output if  $TGRUD \leq TCNT$ .

#### 2. Dead Time Generation Waveform

For dead time generation waveform U phases A (DTGUA) and B (DTGUB), 1 is output as the initial value.

TDCNT0 starts counting at the falling edge of CMOUA. DTGUA outputs 0 if TDCNT0 is counting, and 1 otherwise.

TDCNT1 starts counting at the falling edge of CMOUB. DTGUB outputs 0 if TDCNT1 is counting, and 1 otherwise.

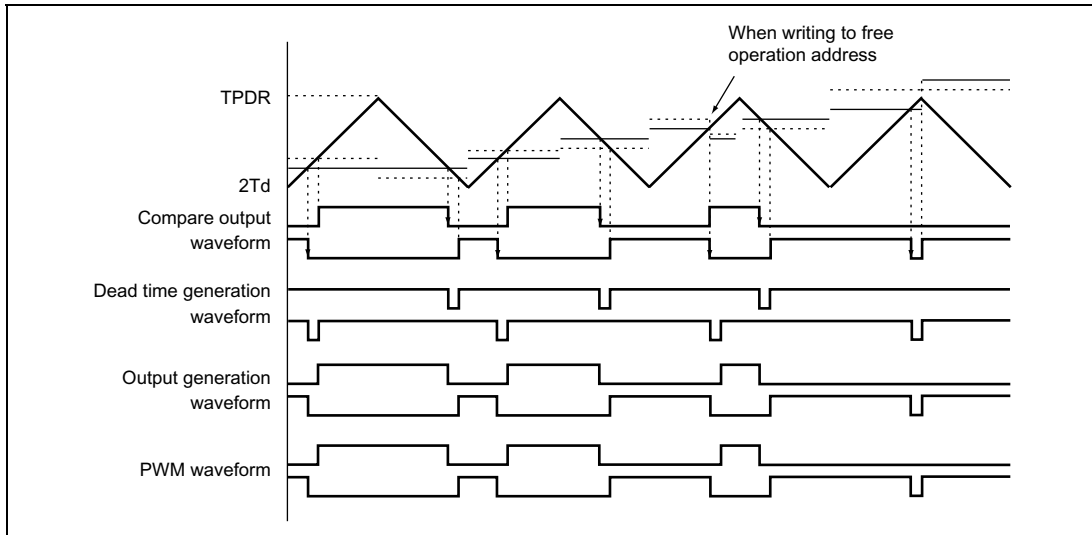
#### 3. Output Generation Waveform

Output generation waveform U phase A (OGUA) is generated by ANDing CMOUA and DTGUB, and output generation waveform U phase B (OGUB) is generated by ANDing CMOUB and DTGUA.

#### 4. PWM Waveform

The PWM waveform is generated by converting the output generation waveform to the output level set in bits OLSN and OLSP in the timer mode register (TMDR).

Figure 11-6 shows an example of PWM waveform generation (operating mode 3, OLSN = 1, OLSP = 1).



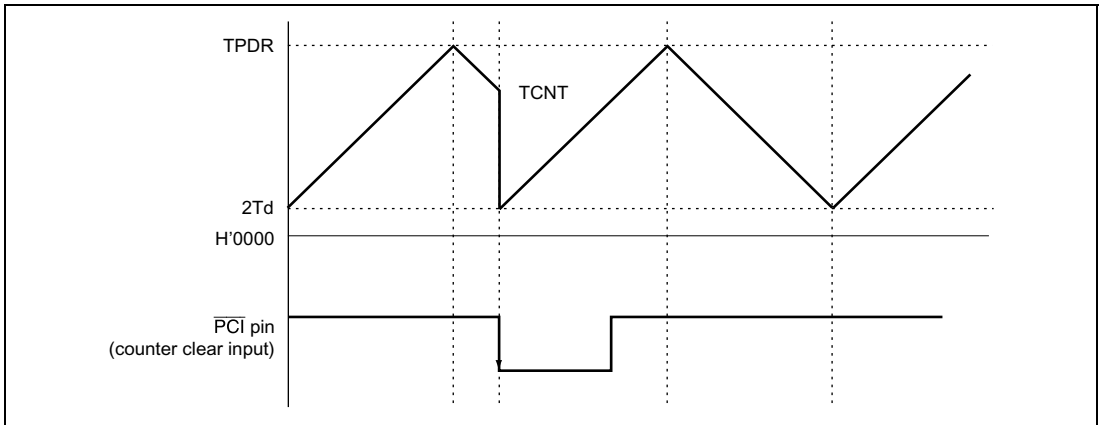
**Figure 11-6 Example of PWM Waveform Generation**

**0% to 100% Duty Output:** In the operating modes, PWM waveforms with any duty from 0% to 100% can be output. The output PWM duty is set using the buffer registers (TBRU to TBRW).

100% duty output is performed when the buffer register (TBRU to TBRW) value is set to H'0000. The waveform in this case has positive phase in the 100% on state. 0% duty output is performed when a value greater than the TPDR value is set as the buffer register (TBRU to TBRW) value. The waveform in this case has positive phase in the 100% off state.

**External Counter Clear Function:** In the operating modes, the TCNT counter can be cleared from an external source. When using the counter clear function, the  $\overline{\text{PCI}}$  pin function should be set to input using the MMT pin control register.

At the falling edge of  $\overline{\text{PCI}}$ , the TCNT counter is cleared to 2Td (initial set value), counts up until it reaches the TPDR value, and then starts counting down. When the count reaches 2Td, TCNT starts counting up again, and this sequence is repeated. An example of counter clearing is shown in figure 11-7.

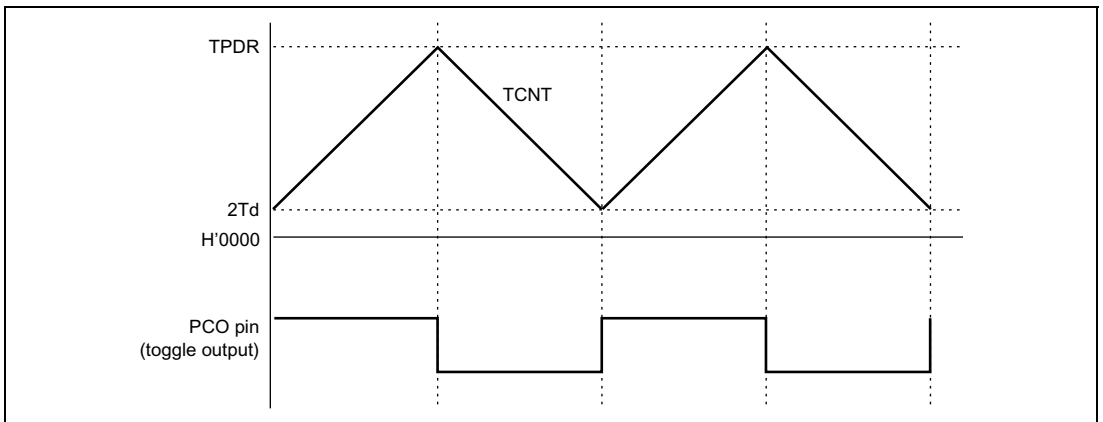


**Figure 11-7 Example of TCNT Counter Clearing**

**Toggle Output Synchronized with PWM Period:** In the operating modes, output can be toggled synchronously with the PWM carrier period. When outputting the PWM period, the PCO pin function should be set to output using the MMT pin control register. An example of the toggle output waveform is shown in figure 11-8.

PWM output is toggled according to the TCNT count direction. The toggle output pin is PCO. PCO outputs 1 when TCNT is counting up, and 0 when counting down.

**Output State when Counter is Halted:** When TCNT is halted, the state of the PWM outputpin is retained. The TCNT value is also retained.



**Figure 11-8 Example of Toggle Output Waveform Synchronized with PWM Period**

## 11.4.2 Output Protection Functions

Operating mode output has the following protection functions:

- Halting PWM output by external signal  
The 6-phase PWM output pins can be placed in the high-impedance state automatically by inputting a specified external signal. There are four external signal input pins. For details, see section 11.8, Port Output Enable (POE).
- Halting MMT output when oscillation stops  
The 6-phase PWM output pins are placed in the high-impedance state automatically when stoppage of the clock input is detected. However, pin states are not guaranteed when the clock is restarted.

## 11.5 Interrupts

When the TGF<sub>M</sub> (TGF<sub>N</sub>) flag is set to 1 in the timer status register (TSR) by a compare match between TCNT and the TPDR register (2T<sub>d</sub>), and if the TGI<sub>EM</sub> (TGI<sub>EN</sub>) bit setting in the timer control register (TCNR) is 1, an interrupt is requested. The interrupt request is cleared by clearing the TGF flag to 0.

**Table 11-3 MMT Interrupt Sources**

| Name  | Interrupt Source                               | Interrupt Flag | DTC Activation |
|-------|------------------------------------------------|----------------|----------------|
| TGIMN | Compare match between TCNT and TPDR            | TGFM           | Yes            |
| TGINN | Compare match between TCNT and 2T <sub>d</sub> | TGFN           | Yes            |

The on-chip DTC can be activated by a compare match between TCNT and TPDR or between TCNT and 2T<sub>d</sub>.

# 11.6    Operation Timing

## 11.6.1    Input/Output Timing

**TCNT and TDCNT Count Timing:** Figure 11-9 shows the TCNT and TDCNT count timing.

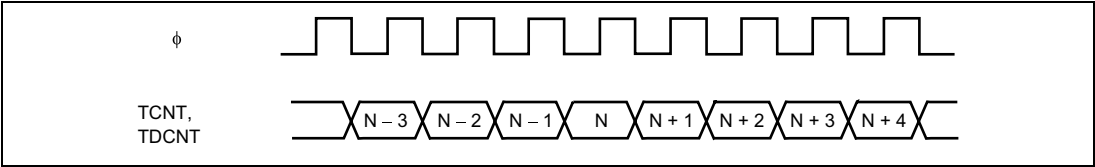


Figure 11-9    Count Timing

**TCNT Counter Clearing Timing:** Figure 11-10 shows the timing of TCNT counter clearing by an external signal.

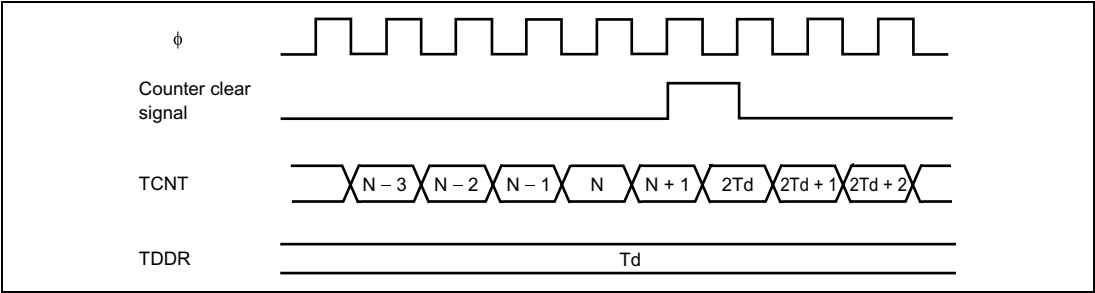
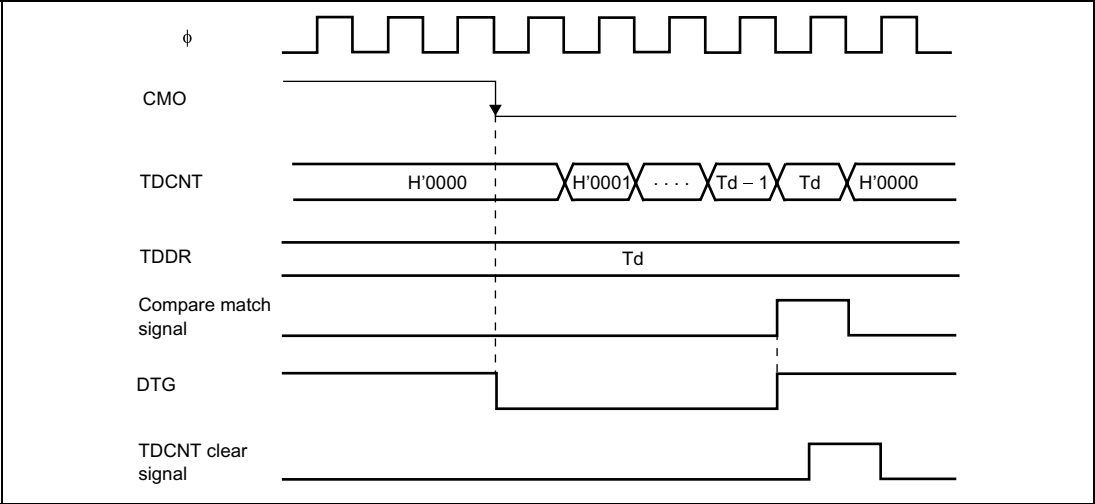


Figure 11-10    TCNT Counter Clearing Timing

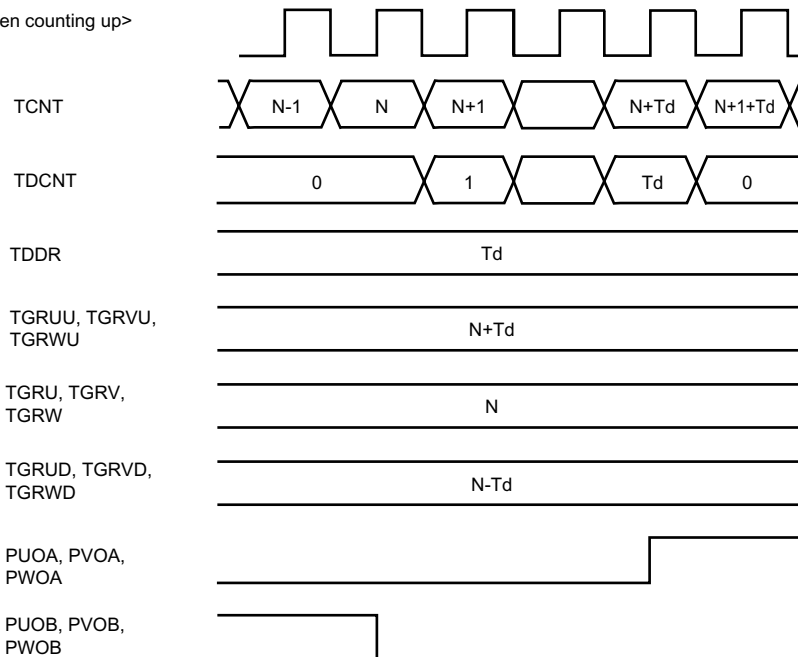
**TDCNT Operation Timing:** Figure 11-11 shows the TDCNT operation timing.



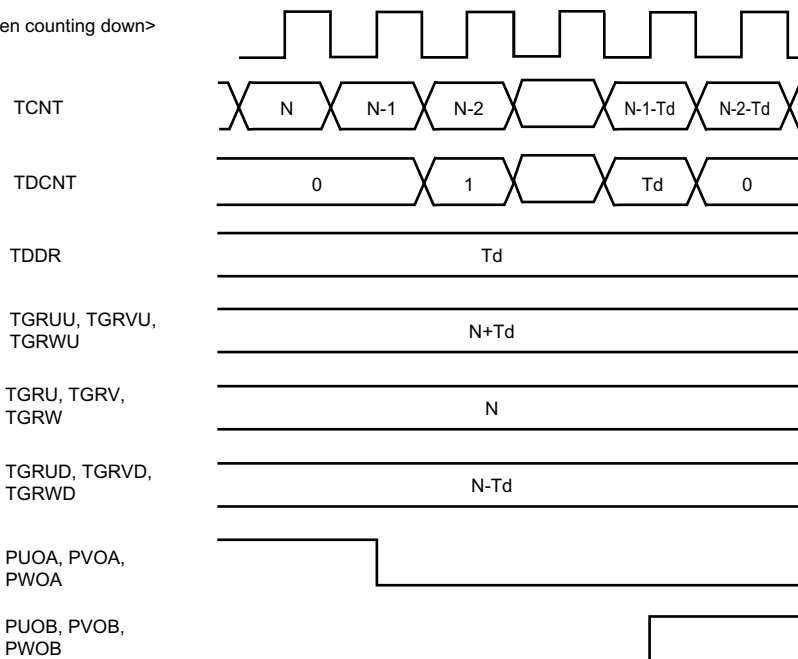
**Figure 11-11 TDCNT Operation Timing**

**Dead Time Genaration Timing:** Figure 11-12 shows the timing for dead time generation.

<When counting up>

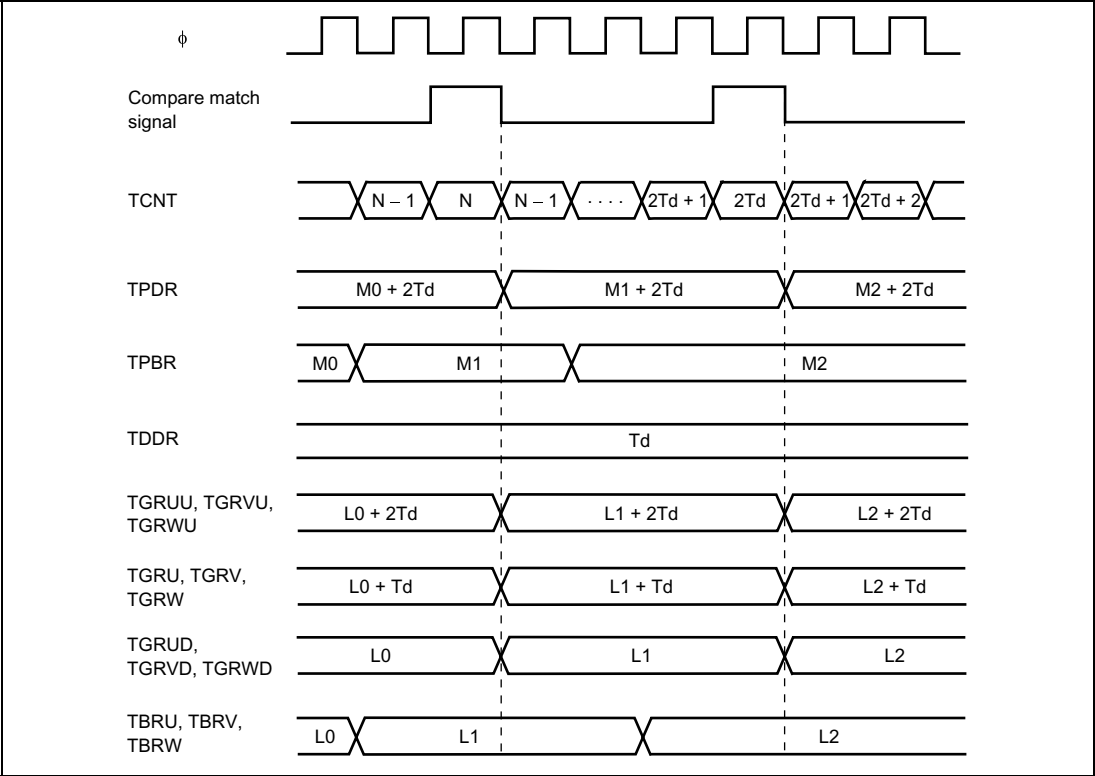


<When counting down>



**Figure 11-12 Dead Time Generation Timing**

**Buffer Operation Timing:** Figure 11-13 shows the compare match buffer operation timing.



**Figure 11-13 Buffer Operation Timing**

### 11.6.2 Interrupt Signal Timing

**Timing of TGF Flag Setting by Compare Match:** Figure 11-14 shows the timing of setting of the TGF flag in the timer status register (TSR) on a compare match between TCNT and TPDR, and the timing of the TGI interrupt request signal. The timing is the same for a compare match between TCNT and 2Td.

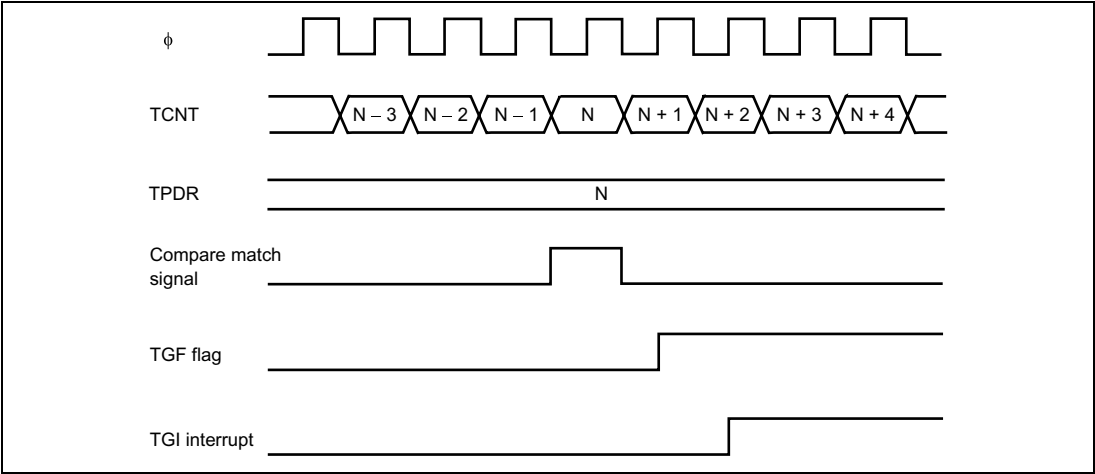


Figure 11-14 TGI Interrupt Timing

**Status Flag Clearing Timing:** A status flag is cleared when the CPU reads 1 from the flag, then 0 is written to it. When the DTC controller is activated, the flag is cleared automatically. Figure 11-15 shows the timing of status flag clearing by the CPU, and Figure 11-16 shows the timing of status flag clearing by the DTC.

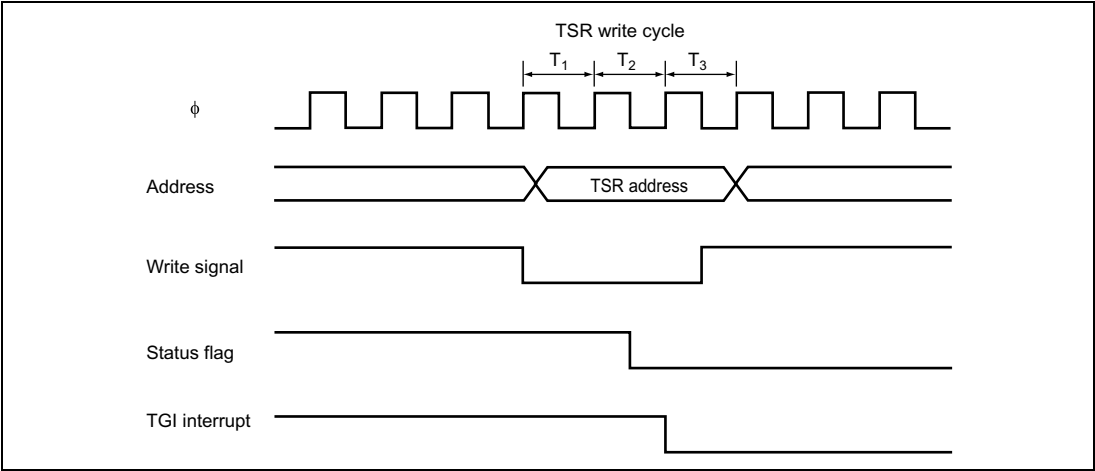
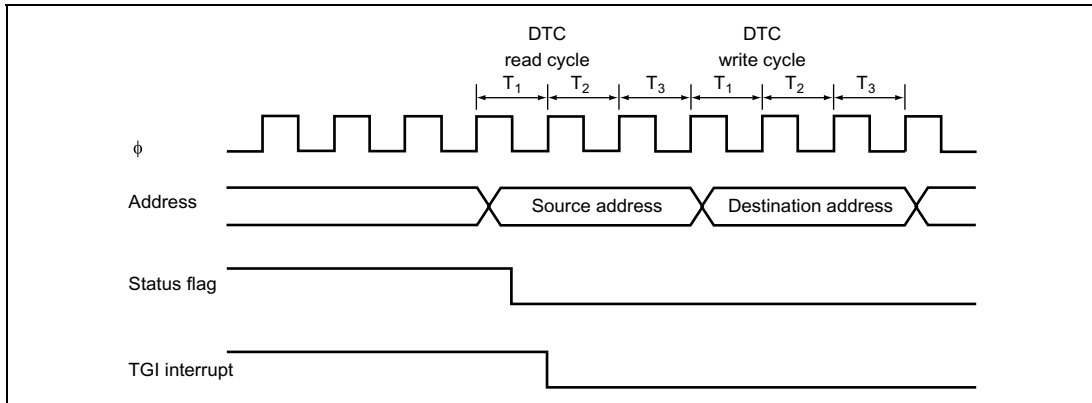


Figure 11-15 Timing of Status Flag Clearing by CPU



**Figure 11-16 Timing of Status Flag Clearing by DTC Controller**

## 11.7 Usage Notes

### 11.7.1 Module Stop Mode Setting

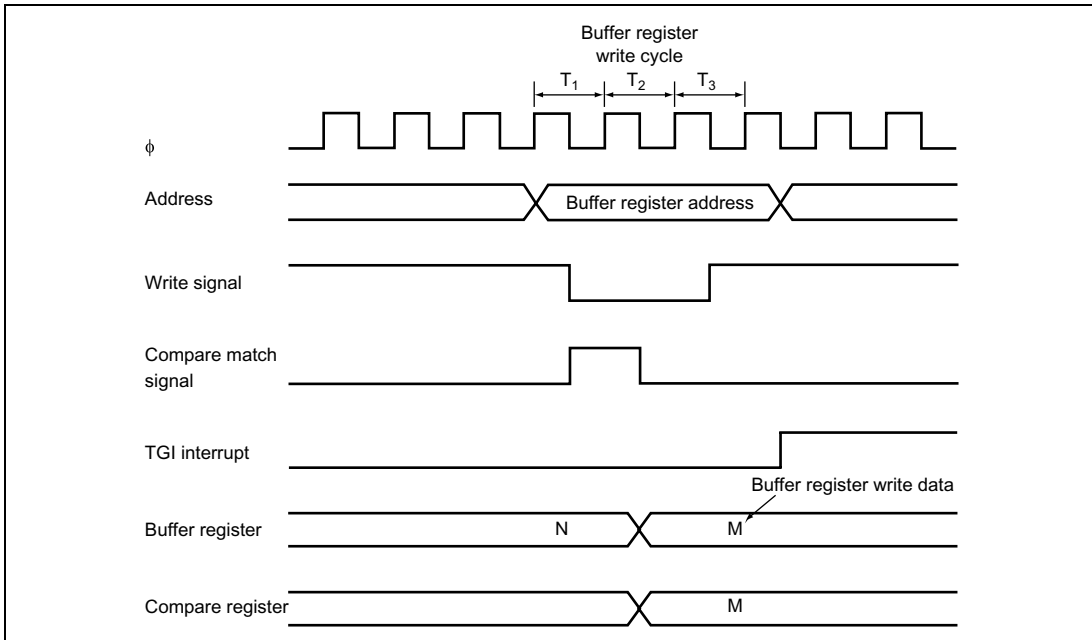
MMT operation can be disabled or enabled using the module stop control register. The initial setting is for MMT operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 20, Power-Down Modes.

### 11.7.2 Notes for MMT Operation

Note that the kinds of operation and contention described below occur during MMT operation.

**Contention between Buffer Register Write and Compare Match:** If a compare match occurs in the  $T_2$  state of a buffer register (TBRU to TBRW, or TPBR) write cycle, data is transferred from the buffer register to the compare register (TGR or TPDR) by a buffer operation. The data transferred is the buffer register write data.

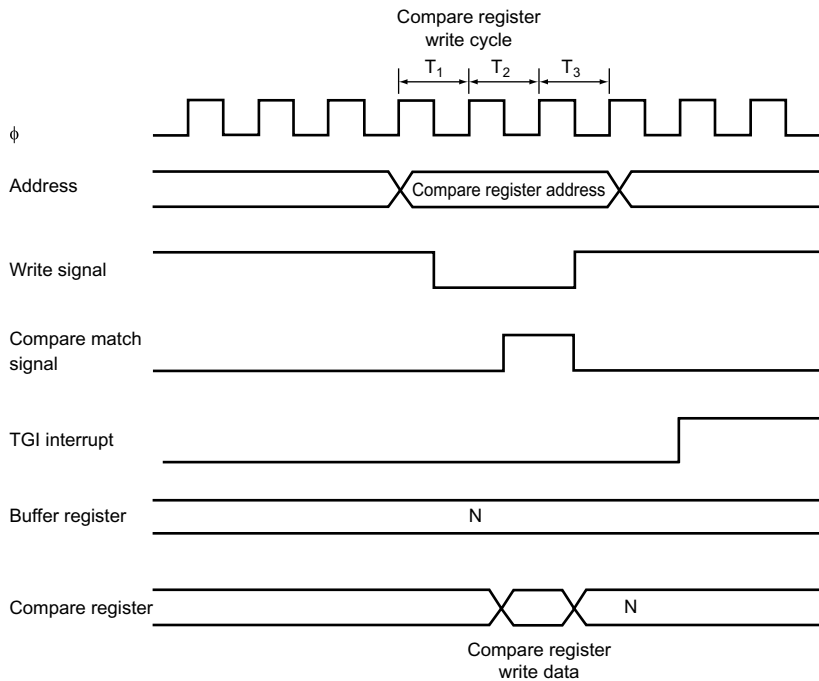
Figure 11-17 shows the timing in this case.



**Figure 11-17 Contention between Buffer Register Write and Compare Match**

**Contention between Compare Register Write and Compare Match:** If a compare match occurs in the  $T_3$  state of a compare register (TGR or TPDR) write cycle, the compare register write is not performed, and data is transferred from the buffer register (TBRU, TBRV, TBRW, or TPBR) to the compare register by a buffer operation.

Figure 11-18 shows the timing in this case.



**Figure 11-18 Contention between Compare Register Write and Compare Match**

## 11.8 Port Output Enable (POE)

The port output enable (POE) circuit enables the MMT's output pins (POUA, POUB, POVA, POVB, POWA, POWB, and PCO) to be placed in the high-impedance state by varying the input at pins POE0 to POE3. An interrupt can also be requested at the same time.

### 11.8.1 Features

The POE circuit has the following features:

- Falling edge,  $P\phi/8 \times 16$  times,  $P\phi/16 \times 16$  times, or  $P\phi/128 \times 16$  times low-level sampling can be set for each of input pins POE0 to POE3.
- The MMT's output pins can be placed in the high-impedance state on sampling of a falling edge or low level at pins  $\overline{POE0}$  to  $\overline{POE3}$ .
- An interrupt request can be initiated by input level sampling.

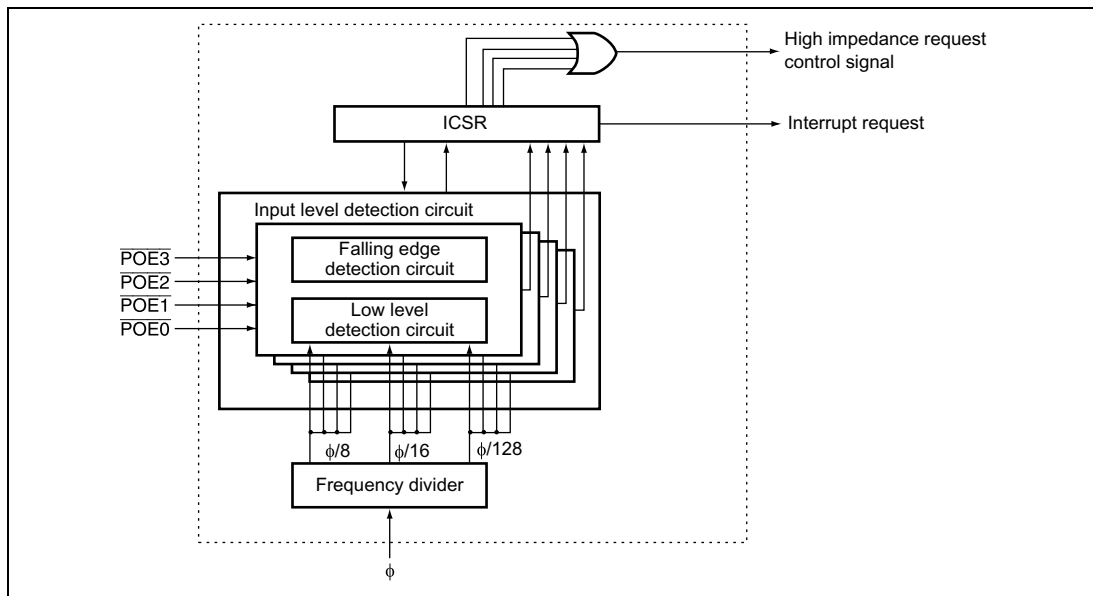


Figure 11-19 Block Diagram of POE

## 11.8.2 Input/Output Pins

Table 11-4 shows the pin configuration of the POE circuit.

**Table 11-4 Pin Configuration**

| Name                          | Abbreviation | I/O   | Function                                                                    |
|-------------------------------|--------------|-------|-----------------------------------------------------------------------------|
| Port output enable input pins | POE0–POE3    | Input | Input request signals for placing MMT's output pins in high-impedance state |

## 11.8.3 Register Descriptions

The POE circuit has the following registers. The ICSR registers are initialized by a reset or in hardware standby mode. However, they are not initialized in software standby mode or sleep mode and retain their previous values. For details on register addresses, refer to Appendix A, On-Chip I/O Register.

- Input level control/status register (ICSR)
- POE pin control register (POEPC)

**Input Level Control/Status Register (ICSR) :** The input level control/status register (ICSR) is a 16-bit readable/writable register that selects the input mode for pins  $\overline{\text{POE0}}$  to  $\overline{\text{POE3}}$ , controls enabling or disabling of interrupts, and holds status information.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                                                               |
|-----|----------|---------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | POE3F    | 0             | R/(W)* | <p>POE3 Flag</p> <p>Indicates that a high impedance request has been input to the POE3 pin.</p> <p>[Clearing condition]</p> <p>When 0 is written to POE3F after reading POE3F = 1</p> <p>[Setting condition]</p> <p>When the input set by bits 6 and 7 of ICSR occurs at the POE3 pin</p> |

| Bit     | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                                                                                                                             |
|---------|----------|---------------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 14      | POE2F    | 0             | R/(W)* | <p>POE2 Flag</p> <p>Indicates that a high impedance request has been input to the <math>\overline{\text{POE2}}</math> pin.</p> <p>[Setting condition]</p> <p>When the input set by bits 4 and 5 of ICSR occurs at the <math>\overline{\text{POE2}}</math> pin</p> <p>[Clearing condition]</p> <p>When 0 is written to POE2F after reading POE2F = 1</p> |
| 13      | POE1F    | 0             | R/(W)* | <p>POE1 Flag</p> <p>Indicates that a high impedance request has been input to the <math>\overline{\text{POE1}}</math> pin.</p> <p>[Setting condition]</p> <p>When the input set by bits 2 and 3 of ICSR occurs at the <math>\overline{\text{POE1}}</math> pin</p> <p>[Clearing condition]</p> <p>When 0 is written to POE1F after reading POE1F = 1</p> |
| 12      | POE0F    | 0             | R/(W)* | <p>POE0 Flag</p> <p>Indicates that a high impedance request has been input to the <math>\overline{\text{POE0}}</math> pin.</p> <p>[Setting condition]</p> <p>When the input set by bits 0 and 1 of ICSR occurs at the <math>\overline{\text{POE0}}</math> pin</p> <p>[Clearing condition]</p> <p>When 0 is written to POE0F after reading POE0F = 1</p> |
| 11 to 9 | —        | 0             | —      | <p>Reserved</p> <p>These bits are always read as 0. Only 0 should be written to these bits.</p>                                                                                                                                                                                                                                                         |
| 8       | PIE      | 0             | R/W    | <p>Port Interrupt Enable</p> <p>Enables or disables an interrupt request when 1 is set in any of bits POE0F to POE3F in ICSR.</p> <p>0: Interrupt request disabled</p> <p>1: Interrupt request enabled</p>                                                                                                                                              |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | POE3M1   | 0             | R/W | POE3 Modes 1 and 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 6   | POE3M0   | 0             | R/W | <p>These bits select the input mode of the <math>\overline{\text{POE3}}</math> pin.</p> <p>00: Request accepted at falling edge of <math>\overline{\text{POE3}}</math> input</p> <p>01: <math>\overline{\text{POE3}}</math> input is sampled for low level 16 times every <math>P_0/8</math> clock, and request is accepted when all samples are low level</p> <p>10: <math>\overline{\text{POE3}}</math> input is sampled for low level 16 times every <math>P_0/16</math> clock, and request is accepted when all samples are low level</p> <p>11: <math>\overline{\text{POE3}}</math> input is sampled for low level 16 times every <math>P_0/128</math> clock, and request is accepted when all samples are low level</p> |
| 5   | POE2M1   | 0             | R/W | POE2 Modes 1 and 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 4   | POE2M0   | 0             | R/W | <p>These bits select the input mode of the <math>\overline{\text{POE2}}</math> pin.</p> <p>00: Request accepted at falling edge of <math>\overline{\text{POE2}}</math> input</p> <p>01: <math>\overline{\text{POE2}}</math> input is sampled for low level 16 times every <math>P_0/8</math> clock, and request is accepted when all samples are low level</p> <p>10: <math>\overline{\text{POE2}}</math> input is sampled for low level 16 times every <math>P_0/16</math> clock, and request is accepted when all samples are low level</p> <p>11: <math>\overline{\text{POE2}}</math> input is sampled for low level 16 times every <math>P_0/128</math> clock, and request is accepted when all samples are low level</p> |
| 3   | POE1M1   | 0             | R/W | POE1 Modes 1 and 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 2   | POE1M0   | 0             | R/W | <p>These bits select the input mode of the <math>\overline{\text{POE1}}</math> pin.</p> <p>00: Request accepted at falling edge of <math>\overline{\text{POE1}}</math> input</p> <p>01: <math>\overline{\text{POE1}}</math> input is sampled for low level 16 times every <math>P_0/8</math> clock, and request is accepted when all samples are low level</p> <p>10: <math>\overline{\text{POE1}}</math> input is sampled for low level 16 times every <math>P_0/16</math> clock, and request is accepted when all samples are low level</p> <p>11: <math>\overline{\text{POE1}}</math> input is sampled for low level 16 times every <math>P_0/128</math> clock, and request is accepted when all samples are low level</p> |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | POE0M1   | 0             | R/W | POE0 Modes 1 and 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| 0   | POE0M0   | 0             | R/W | These bits select the input mode of the $\overline{\text{POE0}}$ pin.<br>00: Request accepted at falling edge of $\overline{\text{POE0}}$ input<br>01: $\overline{\text{POE0}}$ input is sampled for low level 16 times every $P_0/8$ clock, and request is accepted when all samples are low level<br>10: $\overline{\text{POE0}}$ input is sampled for low level 16 times every $P_0/16$ clock, and request is accepted when all samples are low level<br>11: $\overline{\text{POE0}}$ input is sampled for low level 16 times every $P_0/128$ clock, and request is accepted when all samples are low level |

Note: \*Only 0 can be written, for flag clearing.

**POE Pin Control Register (POEPC) :** POEPC is a 8-bit readable/writable register that controls enabling or disabling of the POE pin.

| Bit    | Bit Name | Initial Value | R/W | Description                                                                                                          |
|--------|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------|
| 7      | POE3E    | 0             | R/W | POE3 enabled<br>0: PA3 pin is port I/O pin/SCK2 I/O pin<br>1: PA3 pin is $\overline{\text{POE3}}$ input pin of POE   |
| 6      | POE2E    | 0             | R/W | POE2 enabled<br>0: PA2 pin is port I/O pin/RxD2 input pin<br>1: PA2 pin is $\overline{\text{POE2}}$ input pin of POE |
| 5      | POE1E    | 0             | R/W | POE1 enabled<br>0: PA1 pin is port I/O pin/TxD output pin<br>1: PA1 pin is $\overline{\text{POE1}}$ input pin of POE |
| 4      | POE0E    | 0             | R/W | POE0 enabled<br>0: PA0 pin is port I/O pin<br>1: PA0 pin is $\overline{\text{POE0}}$ input pin of POE                |
| 3 to 0 | —        | Undefined     | —   | Reserved                                                                                                             |

## 11.8.4 Operation

**Input Level Detection:** When the input condition set in ICSR occurs on any one of the  $\overline{\text{POE}}$  pins, the MMT output pins go to the high-impedance state.

- Pins placed in the high-impedance state (the MMT's output pins)  
The 7 pins PWOB, PWOA, PVOB, PVOA, PUOB, PUOA, PCO are placed in the high-impedance state.

**Note:** When used as an output port or TPU output pin, a pin will not enter the high-impedance state.

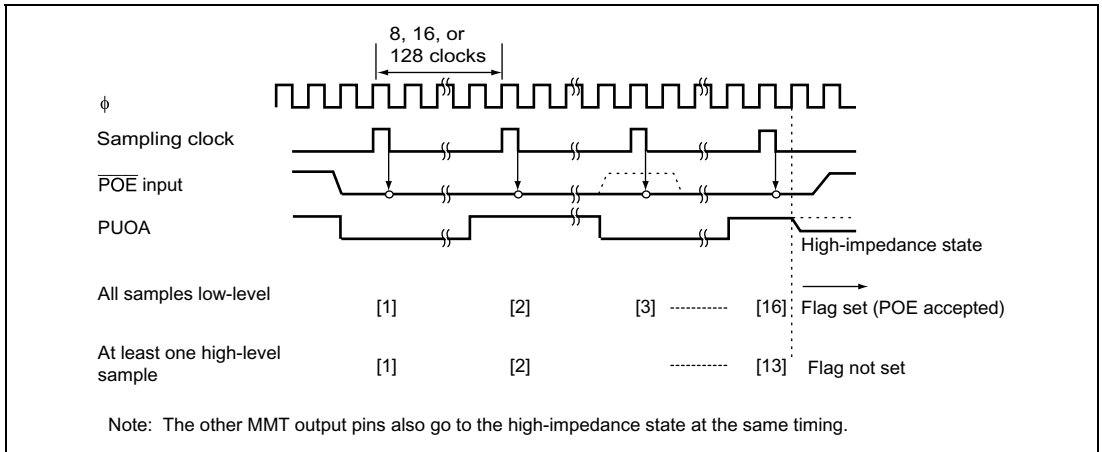
### 1. Falling edge detection

When a transition from high- to low-level input occurs on a  $\overline{\text{POE}}$  pin.

### 2. Low level detection

Figure 11-20 shows the low level detection operation. Low level sampling is performed 16 times in succession using the sampling clock set in ICSR. The input is not accepted if a high level is detected even once among these samples.

The timing of entry of the MMT's output pins into the high-impedance state from the sampling clock is the same for falling edge detection and low level detection.



**Figure 11-20 Low Level Detection Operation**

**Exiting High-Impedance State:** The MMT output pins that have entered the high-impedance state are released from this state by restoring them to their initial states by means of a power-on reset, or by clearing all the POE flags in ICSR (POE0F to POE3F: bits 12 to 15).

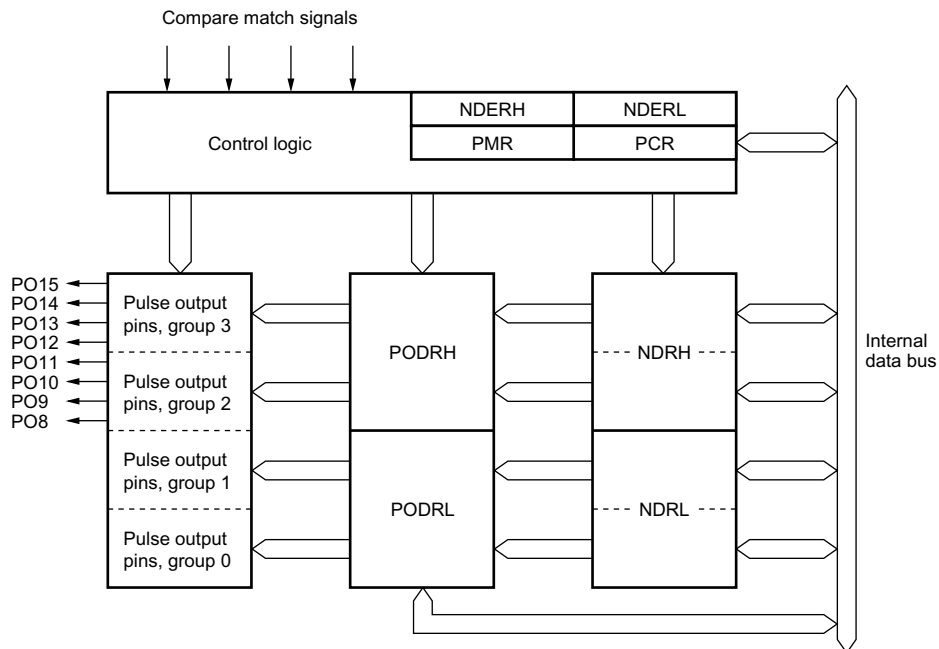


## Section 12 Programmable Pulse Generator (PPG)

The programmable pulse generator provides pulse outputs using the 16-bit timer pulse unit (TPU) as a time base. The PPG pulse outputs are divided into 4-bit groups (group 2 and group 3) that can operate both simultaneously and independently. The block diagram of the PPG is shown in figure 12-1.

### 12.1 Features

- 8-bit output data
- Two output groups
- Selectable output trigger signals
- Non-overlap mode
- Can operate in tandem with the data transfer controller (DTC)
- Settable inverted output
- Module stop mode can be set



#### Legend

PMR : PPG output mode register  
 PCR : PPG output control register  
 NDERH : Next data enable register H  
 NDERL : Next data enable register L  
 NDRH : Next data register H  
 NDRL : Next data register L  
 PODRH : Output data register H  
 PODRL : Output data register L

**Figure 12-1 Block Diagram of PPG**

## 12.2 Input/Output Pins

Table 12-1 summarizes the I/O pins of the PPG.

**Table 12-1 PPG I/O Pins**

| Pin Name | I/O    | Function             |
|----------|--------|----------------------|
| PO15     | Output | Group 3 pulse output |
| PO14     | Output |                      |
| PO13     | Output |                      |
| PO12     | Output |                      |
| PO11     | Output | Group 2 pulse output |
| PO10     | Output |                      |
| PO9      | Output |                      |
| PO8      | Output |                      |

## 12.3 Register Descriptions

The PPG has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

- PPG output control register (PCR)
- PPG output mode register (PMR)
- Next data enable register H (NDERH)
- Next data enable register L (NDERL)
- Output data register H (PODRH)
- Output data register L (PODRL)
- Next data register H (NDRH)
- Next data register L (NDRL)

### 12.3.1 Next Data Enable Registers H,L (NDERH, NDERL)

NDERH and NDERL are an 8-bit readable/writable register that enables or disables pulse output on a bit-by-bit basis. The corresponding DDR also needs to be set to 1 in order to enable pulse output by the PPG.

#### NDERH

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                              |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | NDER15   | 0             | R/W | Next Data Enable 8 to 15                                                                                                                                                                                                 |
| 6   | NDER14   | 0             | R/W | When a bit is set to 1 for pulse output by NDRH, the value in the corresponding NDRH bit is transferred to the PODRH bit by the selected output trigger. Values are not transferred from NDRH to PODRH for cleared bits. |
| 5   | NDER13   | 0             | R/W |                                                                                                                                                                                                                          |
| 4   | NDER12   | 0             | R/W |                                                                                                                                                                                                                          |
| 3   | NDER11   | 0             | R/W |                                                                                                                                                                                                                          |
| 2   | NDER10   | 0             | R/W |                                                                                                                                                                                                                          |
| 1   | NDER9    | 0             | R/W |                                                                                                                                                                                                                          |
| 0   | NDER8    | 0             | R/W |                                                                                                                                                                                                                          |

#### NDERL

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                              |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | NDER7    | 0             | R/W | Next Data Enable 0 to 7                                                                                                                                                                                                  |
| 6   | NDER6    | 0             | R/W | When a bit is set to 1 for pulse output by NDRL, the value in the corresponding NDRL bit is transferred to the PODRL bit by the selected output trigger. Values are not transferred from NDRL to PODRL for cleared bits. |
| 5   | NDER5    | 0             | R/W |                                                                                                                                                                                                                          |
| 4   | NDER4    | 0             | R/W |                                                                                                                                                                                                                          |
| 3   | NDER3    | 0             | R/W |                                                                                                                                                                                                                          |
| 2   | NDER2    | 0             | R/W |                                                                                                                                                                                                                          |
| 1   | NDER1    | 0             | R/W |                                                                                                                                                                                                                          |
| 0   | NDER0    | 0             | R/W |                                                                                                                                                                                                                          |

### 12.3.2 Output Data Registers H,L (PODRH, PODRL)

PODRH and PODRL are 8-bit readable/writable registers that store output data for use in pulse output. A bit that has been set for pulse output by NDER is read-only and cannot be modified.

#### PODRH

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                     |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | POD15    | 0             | R/W | Output Data Register 8 to 15                                                                                                                                                                                                                                                    |
| 6   | POD14    | 0             | R/W | For bits that have been set to pulse output by NDERH, the output trigger transfers NDRH values to this register during PPG operation. While NDERH is set to 1, the CPU cannot write to this register. While NDERH is cleared, the initial output value of the pulse can be set. |
| 5   | POD13    | 0             | R/W |                                                                                                                                                                                                                                                                                 |
| 4   | POD12    | 0             | R/W |                                                                                                                                                                                                                                                                                 |
| 3   | POD11    | 0             | R/W |                                                                                                                                                                                                                                                                                 |
| 2   | POD10    | 0             | R/W |                                                                                                                                                                                                                                                                                 |
| 1   | POD9     | 0             | R/W |                                                                                                                                                                                                                                                                                 |
| 0   | POD8     | 0             | R/W |                                                                                                                                                                                                                                                                                 |

#### PODRL

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                      |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | POD15    | 0             | R/W | Output Data Register 0 to 7                                                                                                                                                                                                                                                      |
| 6   | POD14    | 0             | R/W | For bits which have been set to pulse output by NDERL, the output trigger transfers NDRL values to this register during PPG operation. While NDERL is set to 1, the CPU cannot write to this register. While NDERL is cleared, the initial output value of the pulse can be set. |
| 5   | POD13    | 0             | R/W |                                                                                                                                                                                                                                                                                  |
| 4   | POD12    | 0             | R/W |                                                                                                                                                                                                                                                                                  |
| 3   | POD11    | 0             | R/W |                                                                                                                                                                                                                                                                                  |
| 2   | POD10    | 0             | R/W |                                                                                                                                                                                                                                                                                  |
| 1   | POD9     | 0             | R/W |                                                                                                                                                                                                                                                                                  |
| 0   | POD8     | 0             | R/W |                                                                                                                                                                                                                                                                                  |

### 12.3.3 Next Data Registers H,L (NDRH, NDRL)

NDRH and NDRL are an 8-bit readable/writable register that stores the data for the next pulse output. The NDR addresses differ depending on whether pulse output groups have the same output trigger or different output triggers.

#### NDRH

If pulse output groups 2 and 3 have the same output trigger, all eight bits are mapped to the same address and can be accessed at one time, as shown below.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------|
| 7   | NDR15    | 0             | R/W | Next Data Register 8 to 15                                                                                      |
| 6   | NDR14    | 0             | R/W | The register contents are transferred to the corresponding PODRH bits by the output trigger specified with PCR. |
| 5   | NDR13    | 0             | R/W |                                                                                                                 |
| 4   | NDR12    | 0             | R/W |                                                                                                                 |
| 3   | NDR11    | 0             | R/W |                                                                                                                 |
| 2   | NDR10    | 0             | R/W |                                                                                                                 |
| 1   | NDR9     | 0             | R/W |                                                                                                                 |
| 0   | NDR8     | 0             | R/W |                                                                                                                 |

If pulse output groups 2 and output pulse groups 3 have different output triggers, the upper 4 bits and the lower 4 bits are mapped to different addresses, as shown below.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------|
| 7   | NDR15    | 0             | R/W | Next Data Register 12 to 15                                                                                     |
| 6   | NDR14    | 0             | R/W | The register contents are transferred to the corresponding PODRH bits by the output trigger specified with PCR. |
| 5   | NDR13    | 0             | R/W |                                                                                                                 |
| 4   | NDR12    | 0             | R/W |                                                                                                                 |
| 3   | —        | 1             | —   | Reserved                                                                                                        |
| 2   | —        | 1             | —   | These bits are always read as 1 and cannot be modified.                                                         |
| 1   | —        | 1             | —   |                                                                                                                 |
| 0   | —        | 1             | —   |                                                                                                                 |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------|
| 7   | —        | 1             | —   | Reserved                                                                                                        |
| 6   | —        | 1             | —   | These bits are always read as 1 and cannot be modified.                                                         |
| 5   | —        | 1             | —   |                                                                                                                 |
| 4   | —        | 1             | —   |                                                                                                                 |
| 3   | NDR11    | 0             | R/W | Next Data Register 8 to 11                                                                                      |
| 2   | NDR10    | 0             | R/W | The register contents are transferred to the corresponding PODRH bits by the output trigger specified with PCR. |
| 1   | NDR9     | 0             | R/W |                                                                                                                 |
| 0   | NDR8     | 0             | R/W |                                                                                                                 |

## NDRL

If pulse output groups 0 and 1 have the same output trigger, all eight bits are mapped to the same address and can be accessed at one time, as shown below.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------|
| 7   | NDR7     | 0             | R/W | Next Data Register 0 to 7                                                                                       |
| 6   | NDR6     | 0             | R/W | The register contents are transferred to the corresponding PODRL bits by the output trigger specified with PCR. |
| 5   | NDR5     | 0             | R/W |                                                                                                                 |
| 4   | NDR4     | 0             | R/W |                                                                                                                 |
| 3   | NDR3     | 0             | R/W |                                                                                                                 |
| 2   | NDR2     | 0             | R/W |                                                                                                                 |
| 1   | NDR1     | 0             | R/W |                                                                                                                 |
| 0   | NDR0     | 0             | R/W |                                                                                                                 |

If pulse output groups 0 and output pulse groups 1 have different output triggers, upper 4 bits and lower 4 bits are mapped to the different addresses as shown below.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------|
| 7   | NDR7     | 0             | R/W | Next Data Register 4 to 7                                                                                       |
| 6   | NDR6     | 0             | R/W | The register contents are transferred to the corresponding PODRL bits by the output trigger specified with PCR. |
| 5   | NDR5     | 0             | R/W |                                                                                                                 |
| 4   | NDR4     | 0             | R/W |                                                                                                                 |
| 3   | —        | 1             | —   | Reserved                                                                                                        |
| 2   | —        | 1             | —   | These bits are always read as 1 and cannot be modified.                                                         |
| 1   | —        | 1             | —   |                                                                                                                 |
| 0   | —        | 1             | —   |                                                                                                                 |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                     |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------|
| 7   | —        | 1             | —   | Reserved                                                                                                        |
| 6   | —        | 1             | —   | 1 is always read and write is disabled.                                                                         |
| 5   | —        | 1             | —   |                                                                                                                 |
| 4   | —        | 1             | —   |                                                                                                                 |
| 3   | NDR3     | 0             | R/W | Next Data Register 3 to 0                                                                                       |
| 2   | NDR2     | 0             | R/W | The register contents are transferred to the corresponding PODRL bits by the output trigger specified with PCR. |
| 1   | NDR1     | 0             | R/W |                                                                                                                 |
| 0   | NDR0     | 0             | R/W |                                                                                                                 |

### 12.3.4 PPG Output Control Register (PCR)

PCR is an 8-bit readable/writable register that selects output trigger signals on a group-by-group basis. For details on output trigger selection, refer to section 12.3.5, PPG Output Mode Register (PMR).

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                            |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | G3CMS1   | 1             | R/W | Group 3 Compare Match Select 0 and 1                                                                                                                                                                   |
| 6   | G3CMS0   | 1             | R/W | Select output trigger of pulse output group 3.<br>00: Compare match in TPU channel 0<br>01: Compare match in TPU channel 1<br>10: Compare match in TPU channel 2<br>11: Compare match in TPU channel 3 |
| 5   | G2CMS1   | 1             | R/W | Group 2 Compare Match Select 0 and 1                                                                                                                                                                   |
| 4   | G2CMS0   | 1             | R/W | Select output trigger of pulse output group 2.<br>00: Compare match in TPC channel 0<br>01: Compare match in TPC channel 1<br>10: Compare match in TPC channel 2<br>11: Compare match in TPC channel 3 |
| 3   | G1CMS1   | 1             | R/W | Reserved                                                                                                                                                                                               |
| 2   | G1CMS0   | 1             | R/W |                                                                                                                                                                                                        |
| 1   | G0CMS1   | 1             | R/W | Reserved                                                                                                                                                                                               |
| 0   | G0CMS0   | 1             | R/W |                                                                                                                                                                                                        |

### 12.3.5 PPG Output Mode Register (PMR)

The PMR is an 8-bit readable/writable register that selects the pulse output mode of the PPG for each group. If inverted output is selected, a low-level pulse is output when PODRH is 1 and a high-level pulse is output when PODRH is 0. If non-overlapping operation is selected, PPG updates its output values on compare match A or B of the TPU that becomes the output trigger. For details, refer to section 12.4.5, Non-Overlapping Pulse Output.

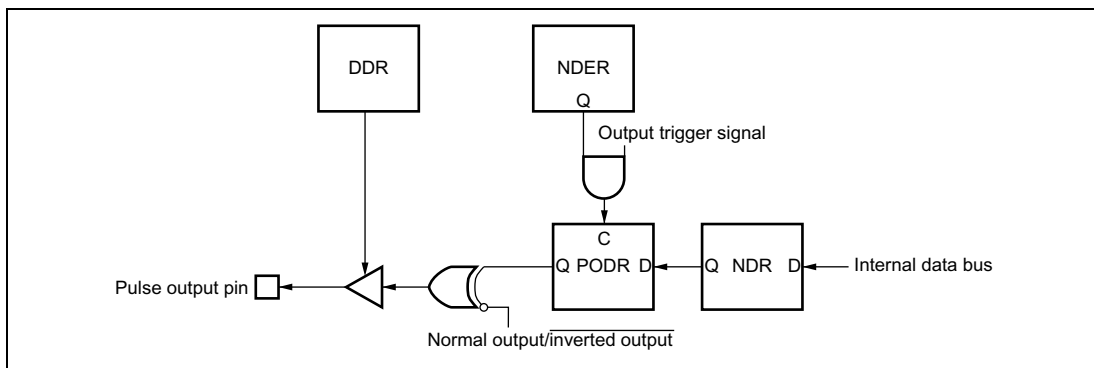
| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                    |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | G3INV    | 1             | R/W | Group 3 Inversion<br>Selects direct output or inverted output for pulse output group 3.<br>0: Inverted output<br>1: Direct output                                                                                                                                                              |
| 6   | G2INV    | 1             | R/W | Group 2 Inversion<br>Selects direct output or inverted output for pulse output group 2.<br>0: Inverted output<br>1: Direct output                                                                                                                                                              |
| 5   | —        | 1             | R/W | Reserved                                                                                                                                                                                                                                                                                       |
| 4   | —        | 1             | R/W |                                                                                                                                                                                                                                                                                                |
| 3   | G3NOV    | 0             | R/W | Group 3 Non-Overlap<br>Selects normal or non-overlapping operation for pulse output group 3.<br>0: Normal operation (output values updated at compare match A in the selected TPU channel)<br>1: Non-overlapping operation (output values at compare match A or B in the selected TPU channel) |
| 2   | G2NOV    | 0             | R/W | Group 2 Non-Overlap<br>Selects normal or non-overlapping operation for pulse output group 2.<br>0: Normal operation (output values updated at compare match A in the selected TPU channel)<br>1: Non-overlapping operation (output values at compare match A or B in the selected TPU channel) |
| 1   | —        | 0             | R/W | Reserved                                                                                                                                                                                                                                                                                       |
| 0   | —        | 0             | R/W |                                                                                                                                                                                                                                                                                                |

## 12.4 Operation

### 12.4.1 Overview

Figure 12-2 shows a block diagram of the PPG. PPG pulse output is enabled when the corresponding bits in P1DDR and NDER are set to 1. An initial output value is determined by its corresponding PODR initial setting. When the compare match event specified by PCR occurs, the corresponding NDR bit contents are transferred to PODR to update the output values.

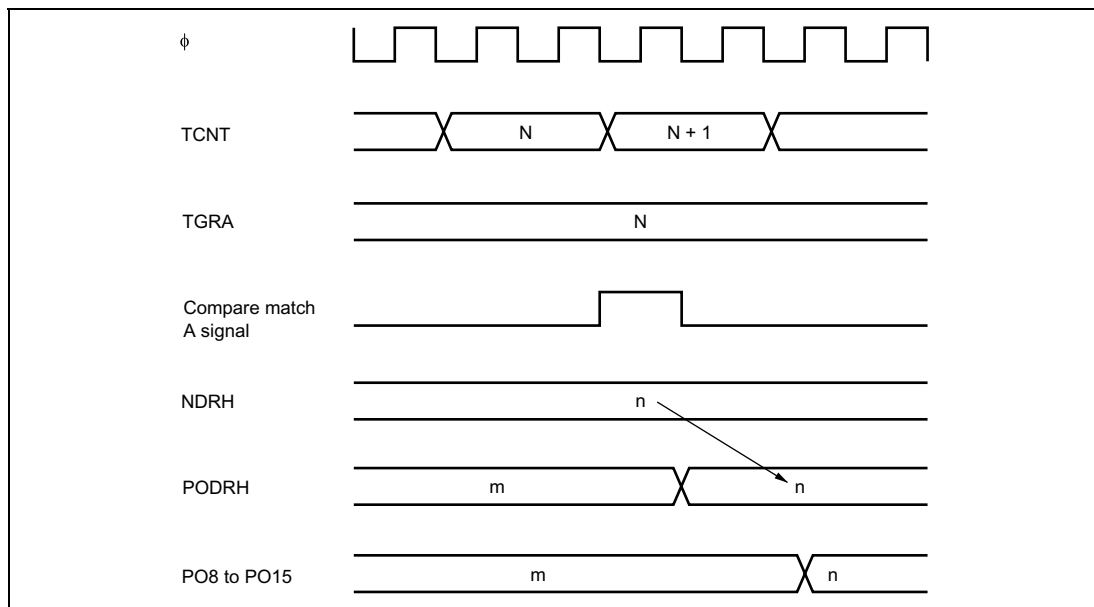
The sequential output of up to 8 bits of data is possible by writing new output data to NDR before the next compare match.



**Figure 12-2 PPG Output Operation**

## 12.4.2 Output Timing

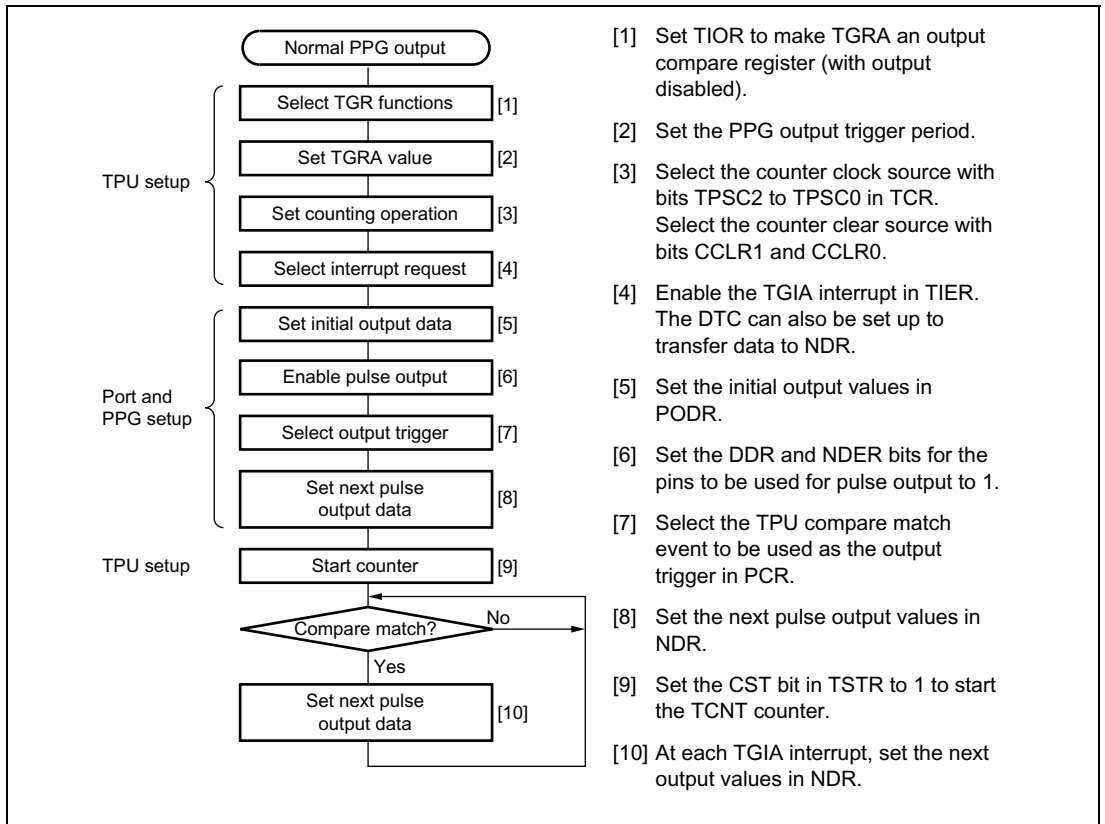
If pulse output is enabled, the contents of NDR contents are transferred to PODR and output when the specified compare match event occurs. Figure 12-3 shows the timing of these operations for the case of normal output in groups 2 and 3, triggered by compare match A.



**Figure 12-3 Timing of Transfer and Output of NDR Contents (Example)**

### 12.4.3 Sample Setup Procedure for Normal Pulse Output

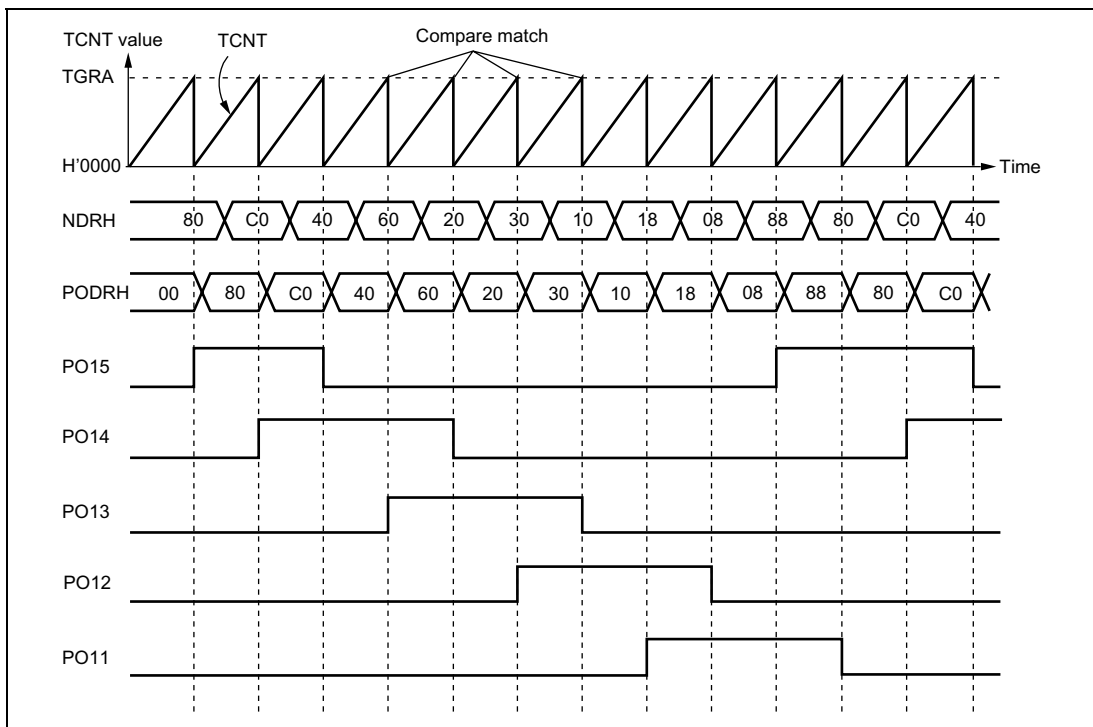
Figure 12-4 shows a sample procedure for setting up normal pulse output.



**Figure 12-4 Setup Procedure for Normal Pulse Output (Example)**

## 12.4.4 Example of Normal Pulse Output (Example of Five-Phase Pulse Output)

Figure 12-5 shows an example in which pulse output is used for cyclic five-phase pulse output.



**Figure 12-5 Normal Pulse Output Example (Five-Phase Pulse Output)**

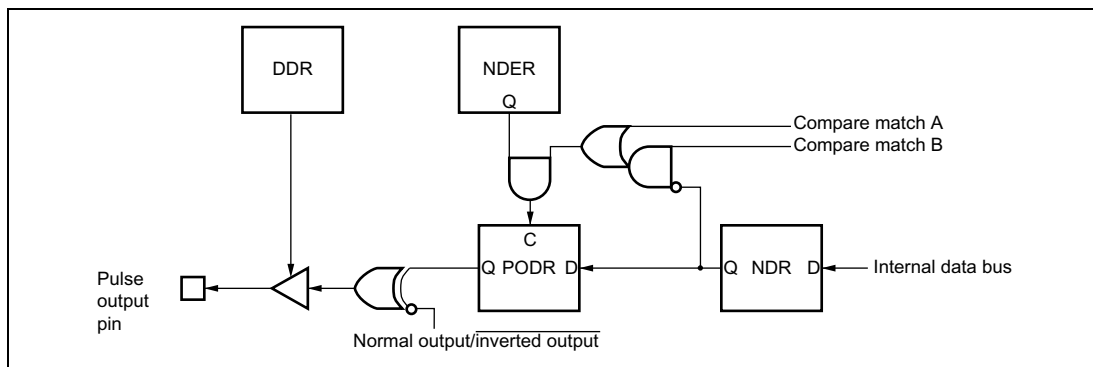
1. Set up TGRA of the TPU that is used as the output trigger to be an output compare register. Set a frequency in TGRA so the counter will be cleared on compare match A. Set the TGIEA bit of TIER to 1 to enable the compare match/input capture A (TGIA) interrupt.
2. Write H'F8 in P1DDR and NDERH, and set the G3CMS0, G3CMS1, G2CMS0, and G2CMS1 bits in PCR to select compare match in the TPU channel set up in the previous step to be the output trigger. Write output data H'80 in NDRH.
3. When compare match A occurs, the NDRH contents are transferred to PODRH and output. The TGIA interrupt handling routine writes the next output data (H'C0) in NDRH.
4. Five-phase overlapping pulse output (one or two phases active at a time) can be obtained subsequently by writing H'40, H'60, H'20, H'30, H'10, H'18, H'08, H'88... at successive TGIA interrupts. If the DTC is set for activation by this interrupt, pulse output can be obtained without imposing a load on the CPU.

### 12.4.5 Non-Overlapping Pulse Output

During non-overlapping operation, transfer from NDR to PODR is performed as follows:

- NDR bits are always transferred on PODR bits on compare match A.
- On compare match B, NDR bits are transferred only if their value is 0. Bits are not transferred if their value is 1.

Figure 12-6 illustrates the non-overlapping pulse output operation.

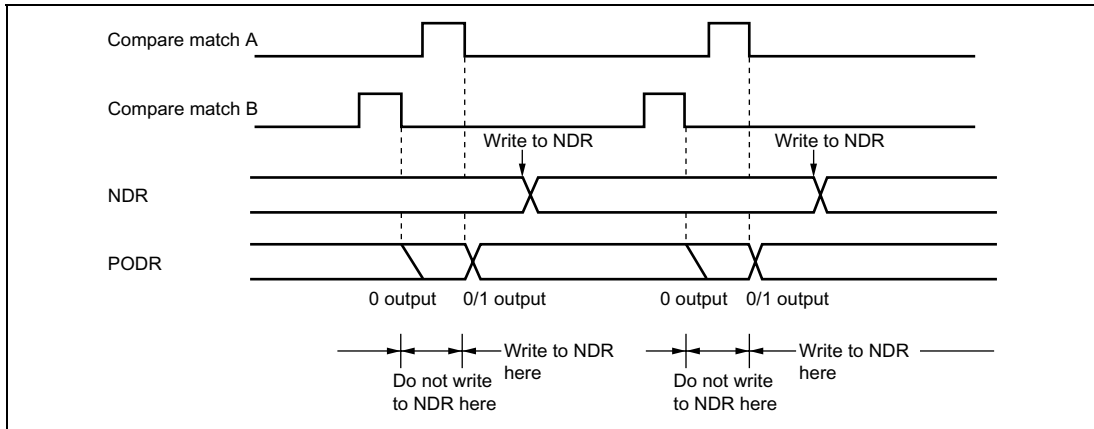


**Figure 12-6 Non-Overlapping Pulse Output**

Therefore, 0 data can be transferred ahead of 1 data by making compare match B occur before compare match A. The NDR contents should not be altered during the interval between compare match B and compare match A (the non-overlap margin).

This can be accomplished by having the TGIA interrupt handling routine write the next data in NDR, or by having the TGIA interrupt activate the DTC. Note, however, that the next data must be written before the next compare match B occurs.

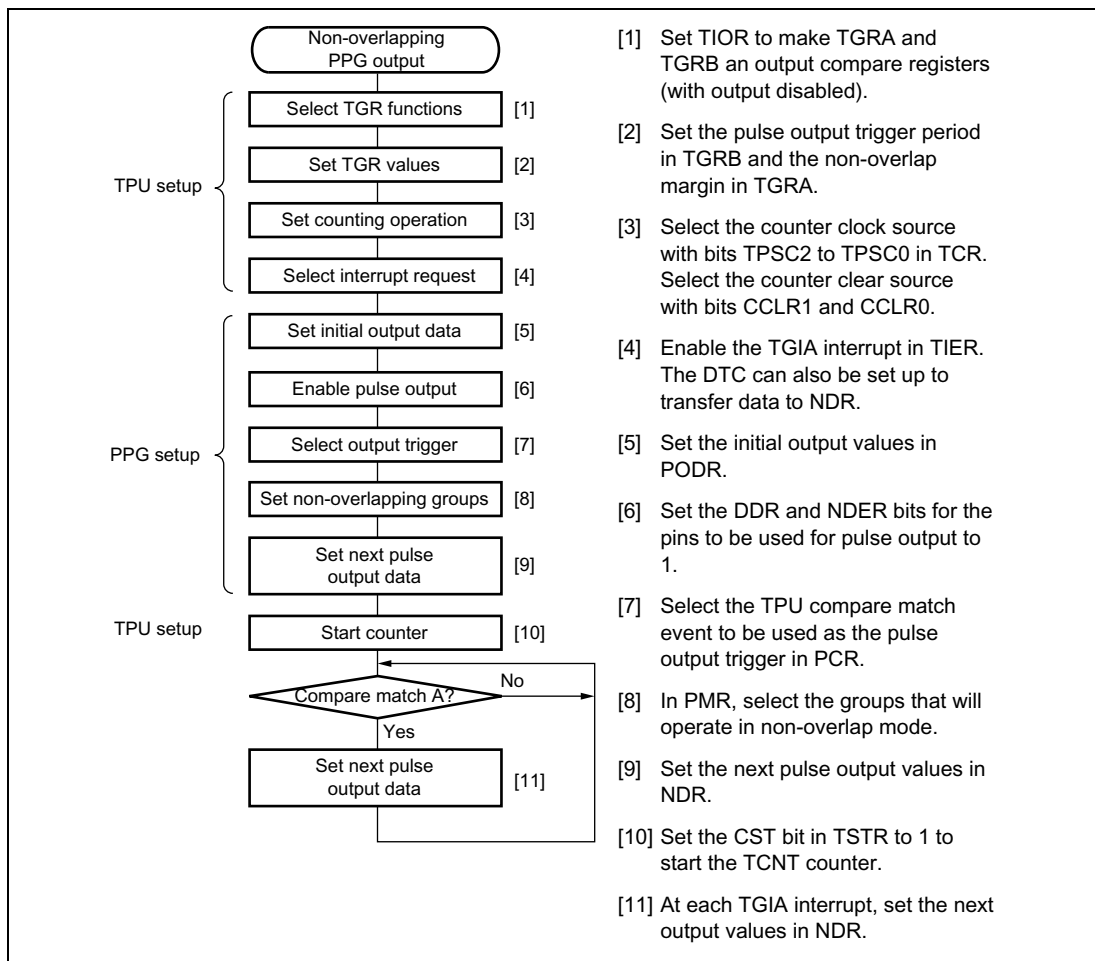
Figure 12-7 shows the timing of this operation.



**Figure 12-7 Non-Overlapping Operation and NDR Write Timing**

## 12.4.6 Sample Setup Procedure for Non-Overlapping Pulse Output

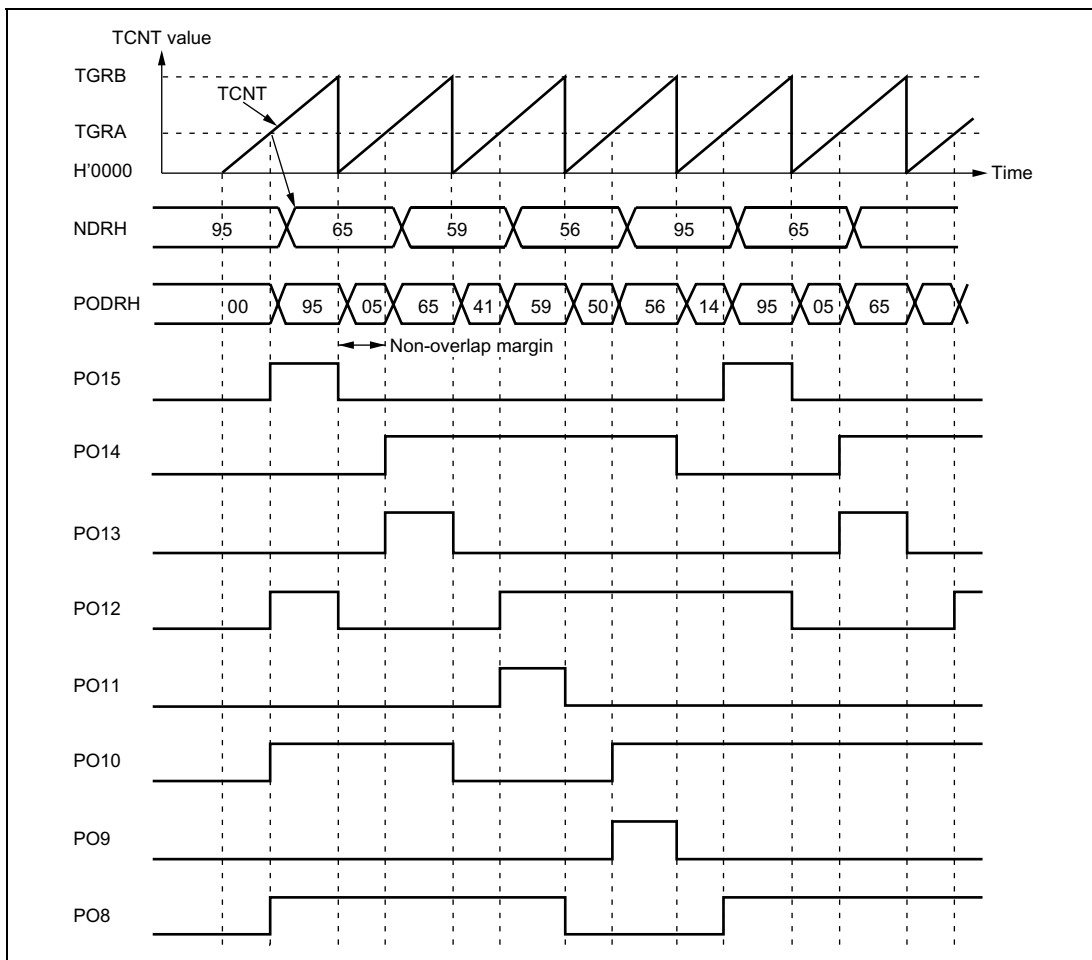
Figure 12-8 shows a sample procedure for setting up non-overlapping pulse output.



**Figure 12-8 Setup Procedure for Non-Overlapping Pulse Output (Example)**

### 12.4.7 Example of Non-Overlapping Pulse Output (Example of Four-Phase Complementary Non-Overlapping Output)

Figure 12-9 shows an example in which pulse output is used for four-phase complementary non-overlapping pulse output.



**Figure 12-9 Non-Overlapping Pulse Output Example (Four-Phase Complementary)**

1. Set up the TPU channel to be used as the output trigger channel such that TGRA and TGRB are output compare registers. Set the trigger period in TGRB and the non-overlap margin in TGRA, and set the counter to be cleared on compare match B. Set the TGIEA bit in TIER to 1 to enable the TGIA interrupt.
2. Write H'FF in P1DDR and NDERH, and set the G3CMS1, G3CMS0, G2CMS1, and G2CMS0 bits in PCR to select compare match in the TPU channel set up in the previous step to be the output trigger. Set the G3NOV and G2NOV bits in PMR to 1 to select non-overlapping output. Write output data H'95 in NDRH.
3. The timer counter in the TPU channel starts. When a compare match with TGRB occurs, outputs change from 1 to 0. When a compare match with TGRA occurs, outputs change from 0 to 1 (the change from 0 to 1 is delayed by the value set in TGRA). The TGIA interrupt handling routine writes the next output data (H'65) in NDRH.
4. Four-phase complementary non-overlapping pulse output can be obtained subsequently by writing H'59, H'56, H'95... at successive TGIA interrupts. If the DTC is set for activation by this interrupt, pulse output can be obtained without imposing a load on the CPU.

## 12.4.8 Inverted Pulse Output

If the G3INV, G2INV, G1INV, and G0INV bits in PMR are cleared to 0, values that are the inverse of the PODR contents can be output.

Figure 12-10 shows the outputs when G3INV and G2INV are cleared to 0, in addition to the settings of figure 12-9.

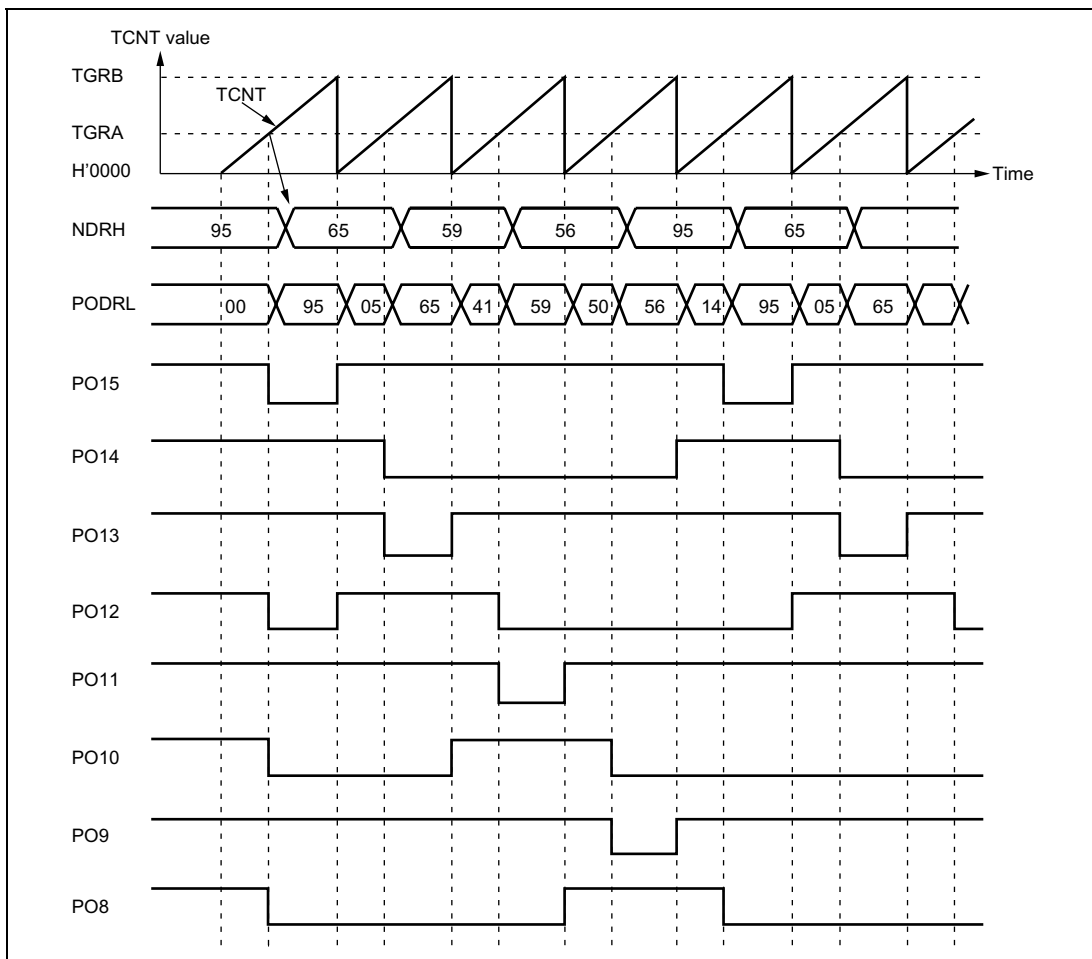
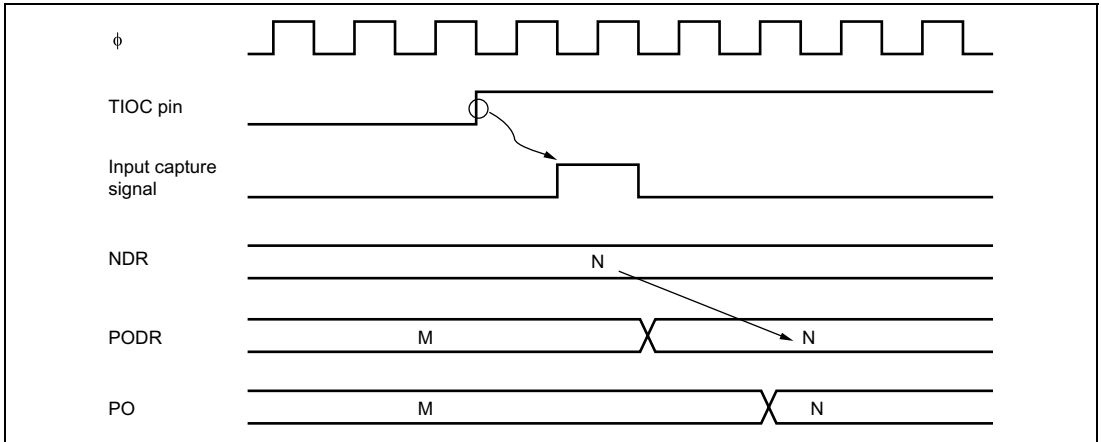


Figure 12-10 Inverted Pulse Output (Example)

### 12.4.9 Pulse Output Triggered by Input Capture

Pulse output can be triggered by TPU input capture as well as by compare match. If TGRA functions as an input capture register in the TPU channel selected by PCR, pulse output will be triggered by the input capture signal.

Figure 12-11 shows the timing of this output.



**Figure 12-11 Pulse Output Triggered by Input Capture (Example)**

## 12.5 Usage Notes

### 12.5.1 Module Stop Mode Setting

PPG operation can be disabled or enabled using the module stop control register. The initial setting is for PPG operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 20, Power-Down Modes.

### 12.5.2 Operation of Pulse Output Pins

Pins PO8 to PO15 are also used for other peripheral functions such as the TPU. When output by another peripheral function is enabled, the corresponding pins cannot be used for pulse output. Note, however, that data transfer from NDR bits to PODR bits takes place, regardless of the usage of the pins.

Pin functions should be changed only under conditions in which the output trigger event will not occur.

## Section 13 Watchdog Timer

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 eight counter input clocks.
- Switchable between watchdog timer mode and interval timer mode.

In watchdog timer mode

- If the counter overflows, it is possible to select whether this LSI is internally reset or not.

In interval timer mode

- If the counter overflows, the WDT generates an interval timer interrupt (WOVI).

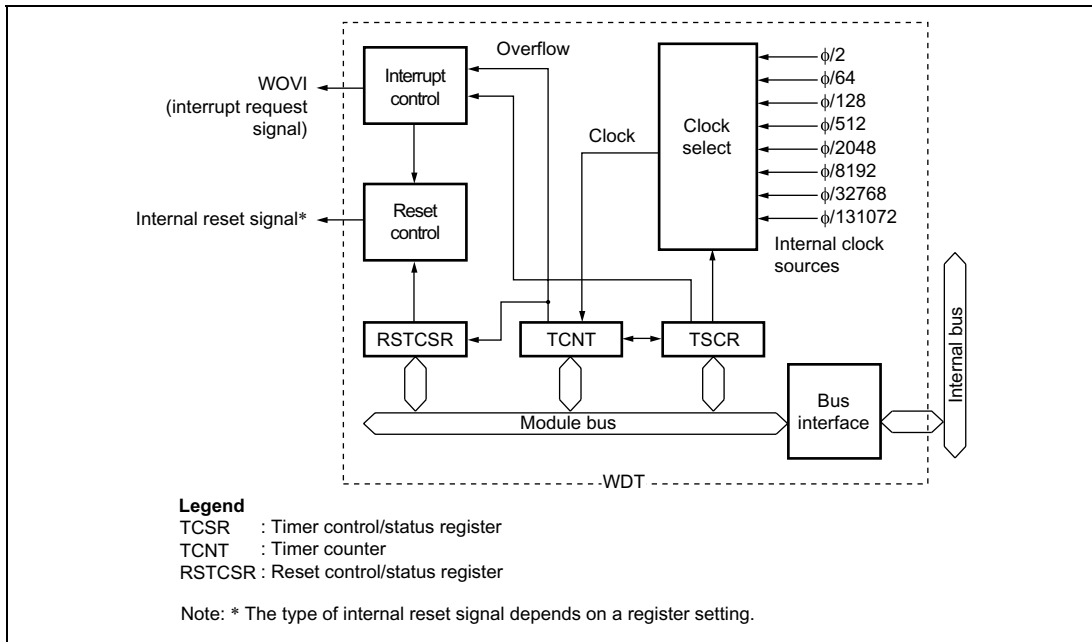


Figure 13-1 Block Diagram of WDT

## 13.2 Register Descriptions

The WDT has the following three registers. For details, refer to Appendix A, On-Chip I/O Register. 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 13.5.1, Notes on Register Access.

- Timer control/status register (TCSR)
- Timer counter (TCNT)
- Reset control/status register (RSTCSR)

### 13.2.1 Timer Counter (TCNT)

TCNT is an 8-bit readable/writable up-counter. TCNT is initialized to H'00 by a reset, when the TME bit in TCSR is cleared to 0.

### 13.2.2 Timer Control/Status Register (TCSR)

TCSR is an 8-bit readable/writable register. Its functions include selecting the clock source to be input to TCNT, and selecting the timer mode.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----|----------|---------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | OVF      | 0             | R/(W)* | <p>Overflow Flag</p> <p>Indicates that TCNT has overflowed. Only a write of 0 is permitted, 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 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</p> <p>1: Watchdog timer mode</p>                                                                                                                                                                                                                                                                        |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5   | TME      | 0             | R/W | Timer Enable<br><br>When this bit is set to 1, TCNT starts counting.<br>When this bit is cleared, TCNT stops counting and is initialized to H'00.                                                                                                                                                                                                                                                                                                                                                                      |
| 4   | —        | 1             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| 3   | —        | 1             | —   | These bits are always read as 1 and cannot be modified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 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 for $\phi = 20$ MHz is enclosed in parentheses.<br><br>000: Clock $\phi/2$ (frequency: 25.6 $\mu$ s)<br>001: Clock $\phi/64$ (frequency: 819.2 $\mu$ s)<br>010: Clock $\phi/128$ (frequency: 1.6 ms)<br>011: Clock $\phi/512$ (frequency: 6.6 ms)<br>100: Clock $\phi/2048$ (frequency: 26.2 ms)<br>101: Clock $\phi/8192$ (frequency: 104.9 ms)<br>110: Clock $\phi/32768$ (frequency: 419.4 ms)<br>111: Clock $\phi/131072$ (frequency: 1.68 s) |
| 0   | CKS0     | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

Note: \* Only 0 can be written, for flag clearing.

### 13.2.3 Reset Control/Status Register (RSTCSR)

RSTCSR is an 8-bit readable/writable register that 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.</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<br/>(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>Selects the type of internal reset generated if TCNT overflows during watchdog timer operation.</p> <p>0: Power-on reset</p> <p>1: Setting prohibited</p>                                                                                                                                                                                                               |
| 4   | —        | 1             | —      | Reserved                                                                                                                                                                                                                                                                                                                                                                                       |
| 3   | —        | 1             | —      | These bits are always read as 1 and cannot be modified.                                                                                                                                                                                                                                                                                                                                        |
| 2   | —        | 1             | —      |                                                                                                                                                                                                                                                                                                                                                                                                |
| 1   | —        | 1             | —      |                                                                                                                                                                                                                                                                                                                                                                                                |
| 0   | —        | 1             | —      |                                                                                                                                                                                                                                                                                                                                                                                                |

Note: \* Only 0 can be written, for flag clearing.

## 13.3 Operation

### 13.3.1 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.

When the WDT is used as a watchdog timer, and if TCNT overflows without being rewritten because of a system malfunction or other error, an internal reset occurs and the internal chip states can be reset.

TCNT does not overflow while the system is operating normally. Software must prevent TCNT overflows by rewriting the TCNT value (normally by writing H'00) before overflows occurs.

In this case, select power-on reset by setting the RSTS bit of the RSTCSR to 0.

The internal reset signal is output for 518 states.

If a reset caused by a signal input to the  $\overline{RES}$  pin occurs at the same time as a reset caused by a WDT overflow, the  $\overline{RES}$  pin reset has priority and the WOVF bit in RSTCSR is cleared to 0.

The WDTOVF signal is output for 132 states when the RSTE bit = 1 of RSTCSR, and for 130 states when the RSTE bit = 0.

When the TCNT overflows in watchdog timer mode, the WOVF bit of the RSTCSR is set to 1.

If the RSTE bit of the RSTCSR has been set to 1, an internal reset signal for the entire LSI is generated at TCNT overflow.

### 13.3.2 Interval Timer Mode

When the WDT is used as an interval timer, an interval timer interrupt (WOVI) is generated each time the TCNT overflows. Therefore, an interrupt can be generated at intervals.

When the TCNT overflows in interval timer mode, an interval timer interrupt (WOVI) is requested at the time the OVF bit of the TCSR is set to 1.

## 13.4 Interrupts

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.

**Table 13-1 WDT Interrupt Source**

| Name | Interrupt Source | Interrupt Flag | DTC Activation |
|------|------------------|----------------|----------------|
| WOVI | TCNT overflow    | WOVF           | Impossible     |

## 13.5 Usage Notes

### 13.5.1 Notes on Register Access

The watchdog timer's TCNT, TCSR, and RSTCSR registers differ from other registers in being more difficult to write to. The procedures for writing to and reading these registers are given below.

#### Writing to TCNT, TCSR, and RSTCSR

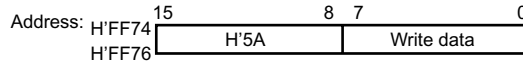
These registers must be written to by a word transfer instruction. They cannot be written to by a byte transfer instruction.

TCNT and TCSR both have the same write address. Therefore, the relative condition shown in figure 13-2 needs to be satisfied in order to write to TCNT or TCSR. The transfer instruction writes the lower byte data to TCNT or TCSR according to the satisfied condition.

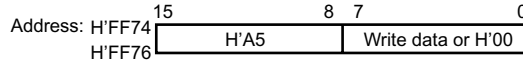
To write to RSTCSR, execute a word transfer instruction for address H'FF76. A byte transfer instruction cannot write to RSTCSR.

The method of writing 0 to the WOVF bit differs from that of writing to the RSTE and RSTS bits. To write 0 to the WOVF bit, satisfy the condition shown in figure 13-2. If satisfied, the transfer instruction clears the WOVF bit to 0, but has no effect on the RSTE and RSTS bits. To write to the RSTE and RSTS bits, satisfy the condition shown in figure 13-2. If satisfied, the transfer instruction writes the values in bits 5 and 6 of the lower byte into the RSTE and RSTS bits, respectively, but has no effect on the WOVF bit.

TCNT write  
Writing to RSTE and RSTS bits



TCSR write  
Writing 0 to WOVF bit



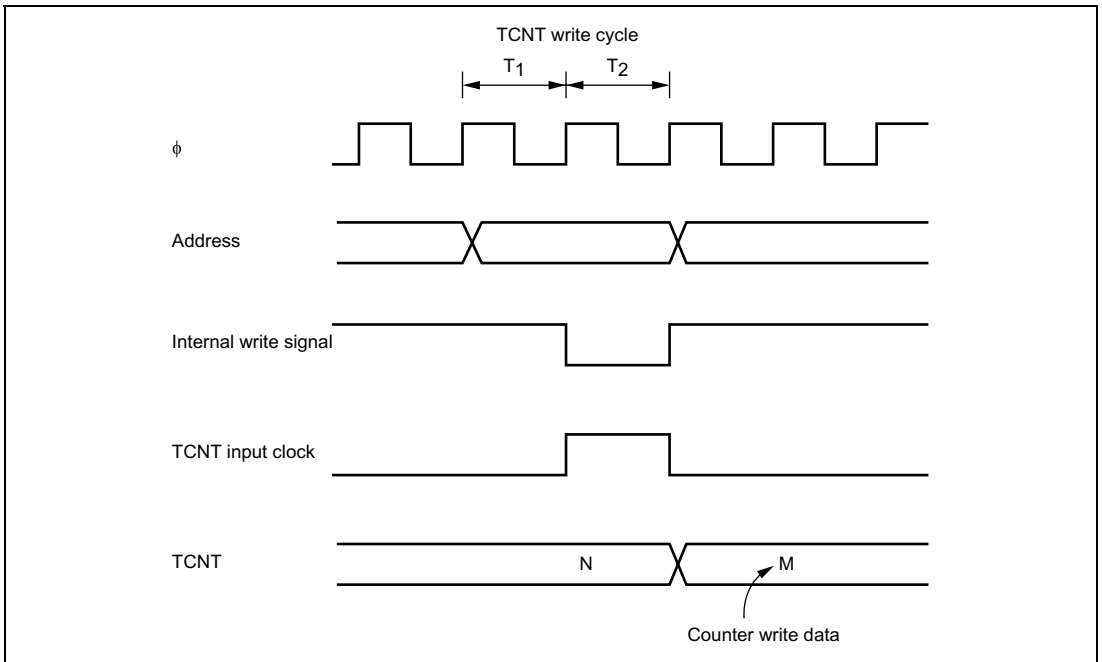
**Figure 13-2 Writing to TCNT, TCSR, and RSTCSR (Example for WDT0)**

## Reading TCNT, TCSR, and RSTCSR (WDT0)

These registers are read in the same way as other registers. The read addresses are H'FF74 for TCSR, H'FF75 for TCNT, and H'FF77 for RSTCSR.

### 13.5.2 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 13-3 shows this operation.



**Figure 13-3 Contention between TCNT Write and Increment**

### **13.5.3 Changing Value of CKS2 to CKS0**

If bits CKS0 to CKS2 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 bits CKS0 to CKS2.

### **13.5.4 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.

### **13.5.5 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 and TCSR of the WDT 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.

### **13.5.6 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.

# Section 14 Serial Communication Interface (SCI)

This LSI has three independent serial communication interface (SCI) channels. 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 extension function.

Figure 14-1 shows a block diagram of the SCI.

## 14.1 Features

- 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).
- 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

## Clocked Synchronous mode

- Data length: 8 bits
- Receive error detection: Overrun errors detected

## 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

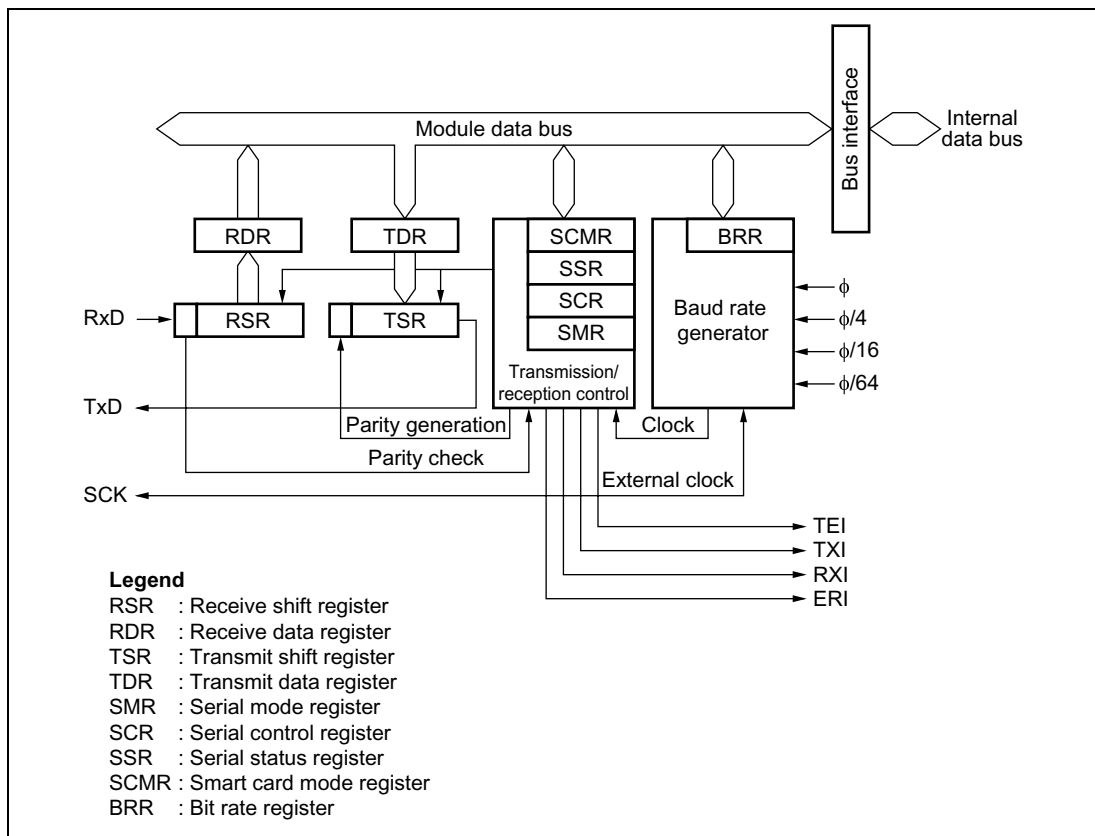


Figure 14-1 Block Diagram of SCI

## 14.2 Input/Output Pins

Table 14-1 shows the serial pins for each SCI channel.

**Table 14-1 Pin Configuration**

| Channel | Pin Name* | 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       | SCK2      | I/O    | SCI2 clock input/output   |
|         | RxD2      | Input  | SCI2 receive data input   |
|         | TxD2      | Output | SCI2 transmit data output |

Note: \*Pin names SCK, RxD, and TxD are used in the text for all channels, omitting the channel designation.

## 14.3 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, On-Chip 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)

### **14.3.1 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.

### **14.3.2 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.

### **14.3.3 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.

### **14.3.4 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.

### 14.3.5 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<br>0: Asynchronous mode<br>1: Clocked synchronous mode                                                                                                                                                                                                                                   |
| 6   | CHR          | 0             | R/W | Character Length (enabled only in asynchronous mode)<br>0: Selects 8 bits as the data length.<br>1: Selects 7 bits as the data length. LSB-first is fixed and the MSB of TDR is not transmitted in transmission.<br>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)<br>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. |
| 4   | O/ $\bar{E}$ | 0             | R/W | Parity Mode (enabled only when the PE bit is 1 in asynchronous mode)<br>0: Selects even parity.<br>1: Selects odd parity.                                                                                                                                                                                   |
| 3   | STOP         | 0             | R/W | Stop Bit Length (enabled only in asynchronous mode)<br>Selects the stop bit length in transmission.<br>0: 1 stop bit<br>1: 2 stop bits<br>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.                     |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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/<math>\bar{E}</math> bit settings are invalid in multiprocessor mode.</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: <math>\emptyset</math> clock (n = 0)</p> <p>01: <math>\emptyset/4</math> clock (n = 1)</p> <p>10: <math>\emptyset/16</math> clock (n = 2)</p> <p>11: <math>\emptyset/64</math> clock (n = 3)</p> <p>For the relationship between the bit rate register setting and the baud rate, see section 14.3.9, Bit Rate Register (BRR). n is the decimal representation of the value of n in BRR (see section 14.3.9, 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 14.7.8, Clock Output Control.</p>                                                                                                                        |
| 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 14.7.3, Block Transfer Mode.</p>                                                                                                                                                                                                                                                                                        |
| 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.<br/>1: Selects odd parity.</p> <p>For details on setting this bit in Smart Card interface mode, refer to section 14.7.2, Data Format (Except for Block Transfer Mode).</p>                                                                                                                                                                   |
| 3   | BCP1              | 0             | R/W | Basic Clock Pulse 1 and 2                                                                                                                                                                                                                                                                                                                                                                                                                           |
| 2   | BCP0              | 0             | R/W | <p>These bits specify the number of basic clock periods in a 1-bit transfer interval on the Smart Card interface.</p> <p>00: 32 clock (S = 32)<br/>01: 64 clock (S = 64)<br/>10: 372 clock (S = 372)<br/>11: 256 clock (S = 256)</p> <p>For details, refer to section 14.7.4, Receive Data Sampling Timing and Reception Margin in Smart Card Interface Mode. S stands for the value of S in BRR (see section 14.3.9, Bit Rate Register (BRR)).</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: <math>\emptyset</math> clock (<math>n = 0</math>)</p> <p>01: <math>\emptyset/4</math> clock (<math>n = 1</math>)</p> <p>10: <math>\emptyset/16</math> clock (<math>n = 2</math>)</p> <p>11: <math>\emptyset/64</math> clock (<math>n = 3</math>)</p> <p>For the relationship between the bit rate register setting and the baud rate, see section 14.3.9, Bit Rate Register (BRR). <math>n</math> is the decimal representation of the value of <math>n</math> in BRR (see section 14.3.9, Bit Rate Register (BRR)).</p> |

### 14.3.6 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 14.8, Interrupts. 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 | Transmit Interrupt Enable<br>When this bit is set to 1, the TXI interrupt request is enabled.                                                                                                                                                                                                                                                                                                                                                                               |
| 6   | RIE      | 0             | R/W | Receive Interrupt Enable<br>When this bit is set to 1, RXI and ERI interrupt requests are enabled.                                                                                                                                                                                                                                                                                                                                                                          |
| 5   | TE       | 0             | R/W | Transmit Enable<br>When this bit s set to 1, transmission is enabled.                                                                                                                                                                                                                                                                                                                                                                                                       |
| 4   | RE       | 0             | R/W | Receive Enable<br>When this bit is set to 1, reception is enabled.                                                                                                                                                                                                                                                                                                                                                                                                          |
| 3   | MPIE     | 0             | R/W | Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 in asynchronous mode)<br>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 14.5, Multiprocessor Communication Function. |
| 2   | TEIE     | 0             | R/W | Transmit End Interrupt Enable<br>This bit is set to 1, TEI interrupt request is enabled.                                                                                                                                                                                                                                                                                                                                                                                    |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | CKE1     | 0             | R/W | Clock Enable 0 and 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 0   | CKE0     | 0             | R/W | <p>Selects the clock source and SCK pin function.</p> <p>Asynchronous mode</p> <p>00: Internal clock<br/>SCK pin functions as I/O port</p> <p>01: Internal clock<br/>Outputs a clock of the same frequency as the bit rate from the SCK pin.</p> <p>1X: External clock<br/>Inputs a clock with a frequency 16 times the bit rate from the SCK pin.</p> <p>Clocked synchronous mode</p> <p>0X: Internal clock (SCK pin functions as clock output)</p> <p>1X: External clock (SCK pin functions as clock input)</p> |

---

#### Legend

X: 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 | Transmit Interrupt Enable<br>When this bit is set to 1, TXI interrupt request is enabled.                                                                                                                                                                                                                                                                                                                                                     |
| 6   | RIE      | 0             | R/W | Receive Interrupt Enable<br>When this bit is set to 1, RXI and ERI interrupt requests are enabled.                                                                                                                                                                                                                                                                                                                                            |
| 5   | TE       | 0             | R/W | Transmit Enable<br>When this bit is set to 1, transmission is enabled.                                                                                                                                                                                                                                                                                                                                                                        |
| 4   | RE       | 0             | R/W | Receive Enable<br>When this bit is set to 1, reception is enabled.                                                                                                                                                                                                                                                                                                                                                                            |
| 3   | MPIE     | 0             | R/W | Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 in asynchronous mode)<br>Write 0 to this bit in Smart Card interface mode.                                                                                                                                                                                                                                                                                          |
| 2   | TEIE     | 0             | R/W | Transmit End Interrupt Enable<br>Write 0 to this bit in Smart Card interface mode.                                                                                                                                                                                                                                                                                                                                                            |
| 1   | CKE1     | 0             | R/W | Clock Enable 0 and 1                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 0   | CKE0     | 0             |     | Enables or disables clock output from the SCK pin. The clock output can be dynamically switched in GSM mode. For details, refer to section 14.7.8, Clock Output Control.<br><br>When the GM bit in SMR is 0:<br>00: Output disabled (SCK pin can be used as an I/O port pin)<br>01: Clock output<br>1X: Reserved<br><br>When the GM bit in SMR is 1:<br>00: Output fixed low<br>01: Clock output<br>10: Output fixed high<br>11: Clock output |

### Legend

X: Don't care

### 14.3.7 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 | <p>Transmit Data Register Empty</p> <p>Displays whether TDR contains transmit data.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"><li>• When the TE bit in SCR is 0</li><li>• When data is transferred from TDR to TSR and data can be written to TDR</li></ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"><li>• When 0 is written to TDRE after reading TDRE = 1</li><li>• When the DTC is activated by a TXI interrupt request and writes data to TDR</li></ul>                                              |
| 6   | RDRF     | 0             | R/W | <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"><li>• When 0 is written to RDRF after reading RDRF = 1</li><li>• When the DTC is activated by an RXI interrupt and transferred data from RDR</li></ul> <p>The RDRF flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0.</p> |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5   | ORER     | 0             | R/W | <p>Overflow Error</p> <p>[Setting condition]</p> <p>When the next serial reception is completed while RDRF = 1</p> <p>[Clearing condition]</p> <p>When 0 is written to ORER after reading ORER = 1</p>                                                                                                                                                                                                                                  |
| 4   | FER      | 0             | R/W | <p>Framing Error</p> <p>[Setting condition]</p> <p>When the stop bit is 0</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>                                                                                                                                                                                                          |
| 3   | PER      | 0             | R/W | <p>Parity Error</p> <p>[Setting condition]</p> <p>When a parity error is detected during reception</p> <p>[Clearing condition]</p> <p>When 0 is written to PER after reading PER = 1</p>                                                                                                                                                                                                                                                |
| 2   | TEND     | 1             | R   | <p>Transmit End</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When the TE bit in SCR is 0</li> <li>When TDRE = 1 at transmission of the last bit of a 1-byte serial transmit character</li> </ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When 0 is written to TDRE after reading TDRE = 1</li> <li>When the DTC is activated by a TXI interrupt and writes data to TDR</li> </ul> |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                             |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | MPB      | 0             | R   | Multiprocessor Bit<br><br>MPB stores the multiprocessor bit in the receive data. When the RE bit in SCR is cleared to 0 its previous state is retained. |
| 0   | MPBT     | 0             | R/W | Multiprocessor Bit Transfer<br><br>MPBT stores the multiprocessor bit to be added to the transmit data.                                                 |

Smart Card Interface Mode (When SMIF in SCMR is 1)

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | TDRE     | 1             | R/W | Transmit Data Register Empty<br><br>Displays whether TDR contains transmit data.<br>[Setting conditions]<br><ul style="list-style-type: none"> <li>When the TE bit in SCR is 0</li> <li>When data is transferred from TDR to TSR and data can be written to TDR</li> </ul><br>[Clearing conditions]<br><ul style="list-style-type: none"> <li>When 0 is written to TDRE after reading TDRE = 1</li> <li>When the DTC is activated by a TXI interrupt request and writes data to TDR</li> </ul>                                        |
| 6   | RDRF     | 0             | R/W | Receive Data Register Full<br><br>Indicates that the received data is stored in RDR.<br>[Setting condition]<br>When serial reception ends normally and receive data is transferred from RSR to RDR<br><br>[Clearing conditions]<br><ul style="list-style-type: none"> <li>When 0 is written to RDRF after reading RDRF = 1</li> <li>When the DTC is activated by an RXI interrupt and transferred data from RDR</li> </ul><br>The RDRF flag is not affected and retains their previous values when the RE bit in SCR is cleared to 0. |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                            |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 5   | ORER     | 0             | R/W | <p>Overflow Error</p> <p>[Setting condition]</p> <p>When the next serial reception is completed while RDRF = 1</p> <p>[Clearing condition]</p> <p>When 0 is written to ORER after reading ORER = 1</p> |
| 4   | ERS      | 0             | R/W | <p>Error Signal Status</p> <p>[Setting condition]</p> <p>When the low level of the error signal is sampled</p> <p>[Clearing conditions]</p> <p>When 0 is written to ERS after reading ERS = 1</p>      |
| 3   | PER      | 0             | R/W | <p>Parity Error</p> <p>[Setting condition]</p> <p>When a parity error is detected during reception</p> <p>[Clearing condition]</p> <p>When 0 is written to PER after reading PER = 1</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"> <li>When the TE bit in SCR is 0 and the ERS bit is also 0</li> <li>When the ERS bit is 0 and the TDRE bit is 1 after the specified interval following transmission of 1-byte data.</li> </ul> <p>The timing of bit setting differs according to the register setting as follows:</p> <p>When GM = 0 and BLK = 0, 2.5 etu after transmission starts</p> <p>When GM = 0 and BLK = 1, 1.5 etu after transmission starts</p> <p>When GM = 1 and BLK = 0, 1.0 etu after transmission starts</p> <p>When GM = 1 and BLK = 1, 1.0 etu after transmission starts</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When 0 is written to TDRE after reading TDRE = 1</li> <li>When the DTC is activated by a TXI interrupt and writes data to TDR</li> </ul> |
| 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>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

### 14.3.8 Smart Card Mode Register (SCMR)

SCMR is a register that selects Smart Card interface mode and its format.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                    |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | 1             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                                                       |
| 6   | —        | 1             | —   | These bits are always read as 1.                                                                                                                                                                                                                                                                                                                                                               |
| 5   | —        | 1             | —   |                                                                                                                                                                                                                                                                                                                                                                                                |
| 4   | —        | 1             | —   |                                                                                                                                                                                                                                                                                                                                                                                                |
| 3   | SDIR     | 0             | R/W | Smart Card Data Transfer Direction<br>Selects the serial/parallel conversion format.<br>0: LSB-first in transfer<br>1: MSB-first in transfer<br>The bit setting is valid only when the transfer data format is 8 bits. For 7-bit data, LSB-first is fixed.                                                                                                                                     |
| 2   | SINV     | 0             | R/W | Smart Card Data Invert<br>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.<br>0: TDR contents are transmitted as they are.<br>Receive data is stored as it is in RDR<br>1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR |
| 1   | —        | 1             | —   | Reserved<br>This bit is always read as 1.                                                                                                                                                                                                                                                                                                                                                      |
| 0   | SMIF     | 0             | R/W | Smart Card Interface Mode Select<br>This bit is set to 1 to make the SCI operate in Smart Card interface mode.<br>0: Normal asynchronous mode or clocked synchronous mode<br>1: Smart card interface mode                                                                                                                                                                                      |

### 14.3.9 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 14-2 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 14-2 The Relationships between The N Setting in BRR and Bit Rate B**

| Mode                      | Bit Rate                                                         | Error                                                                                                                     |
|---------------------------|------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| Asynchronous Mode         | $B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (N + 1)}$ | $\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N + 1)} - 1 \right\} \times 100$ |
| Clocked Synchronous Mode  | $B = \frac{\phi \times 10^6}{8 \times 2^{2n-1} \times (N + 1)}$  |                                                                                                                           |
| Smart Card Interface Mode | $B = \frac{\phi \times 10^6}{S \times 2^{2n-1} \times (N + 1)}$  | $\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{B \times S \times 2^{2n-1} \times (N + 1)} - 1 \right\} \times 100$  |

Notes: B: Bit rate (bit/s)

N: BRR setting for baud rate generator ( $0 \leq N \leq 255$ )

$\phi$ : Operating frequency (MHz)

n and S: Determined by the SMR settings shown in the following tables.

| SMR Setting |      |   |
|-------------|------|---|
| CKS1        | CKS0 | n |
| 0           | 0    | 0 |
| 0           | 1    | 1 |
| 1           | 0    | 2 |
| 1           | 1    | 3 |

| SMR Setting |      |     |
|-------------|------|-----|
| BCP1        | BCP0 | S   |
| 0           | 0    | 32  |
| 0           | 1    | 64  |
| 1           | 0    | 372 |
| 1           | 1    | 256 |

Table 14-3 shows sample N settings in BRR in normal asynchronous mode. Table 14-4 shows the maximum bit rate for each frequency in normal asynchronous mode. Table 14-6 shows sample N settings in BRR in clocked synchronous mode. Table 14-8 shows sample N settings in BRR in Smart Card interface mode. In Smart Card interface mode, S (the number of basic clock periods in a 1-bit transfer interval) can be selected. For details, refer to section 14.7.4, Receive Data Sampling Timing and Reception Margin in Smart Card Interface Mode. Tables 14.5 and 14.7 show the maximum bit rates with external clock input.

**Table 14-3 BRR Settings for Various Bit Rates (Asynchronous Mode)**

| Bit Rate<br>(bit/s) | Operating Frequency $\phi$ (MHz) |     |           |        |     |           |   |     |           |
|---------------------|----------------------------------|-----|-----------|--------|-----|-----------|---|-----|-----------|
|                     | 4                                |     |           | 4.9152 |     |           | 5 |     |           |
|                     | n                                | N   | Error (%) | n      | N   | Error (%) | n | N   | Error (%) |
| 110                 | 2                                | 70  | 0.03      | 2      | 86  | 0.31      | 2 | 88  | −0.25     |
| 150                 | 1                                | 207 | 0.16      | 1      | 255 | 0.00      | 2 | 64  | 0.16      |
| 300                 | 1                                | 103 | 0.16      | 1      | 127 | 0.00      | 1 | 129 | 0.16      |
| 600                 | 0                                | 207 | 0.16      | 0      | 255 | 0.00      | 1 | 64  | 0.16      |
| 1200                | 0                                | 103 | 0.16      | 0      | 127 | 0.00      | 0 | 129 | 0.16      |
| 2400                | 0                                | 51  | 0.16      | 0      | 63  | 0.00      | 0 | 64  | 0.16      |
| 4800                | 0                                | 25  | 0.16      | 0      | 31  | 0.00      | 0 | 32  | −1.36     |
| 9600                | 0                                | 12  | 0.16      | 0      | 15  | 0.00      | 0 | 15  | 1.73      |
| 19200               | —                                | —   | —         | 0      | 7   | 0.00      | 0 | 7   | 1.73      |
| 31250               | 0                                | 3   | 0.00      | 0      | 4   | −1.70     | 0 | 4   | 0.00      |
| 38400               | —                                | —   | —         | 0      | 3   | 0.00      | 0 | 3   | 1.73      |

| Bit Rate<br>(bit/s) | Operating Frequency $\phi$ (MHz) |     |           |       |     |           |        |     |           |   |     |           |
|---------------------|----------------------------------|-----|-----------|-------|-----|-----------|--------|-----|-----------|---|-----|-----------|
|                     | 6                                |     |           | 6.144 |     |           | 7.3728 |     |           | 8 |     |           |
|                     | 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      | — | —   | —         |

# Operating Frequency $\phi$ (MHz)

| Bit Rate<br>(bit/s) | 9.8304 |     |           | 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      |

# Operating Frequency $\phi$ (MHz)

| Bit Rate<br>(bit/s) | 14 |     |           | 14.7456 |     |           | 16 |     |           | 17.2032 |     |           |
|---------------------|----|-----|-----------|---------|-----|-----------|----|-----|-----------|---------|-----|-----------|
|                     | 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 | 0.13      | 2       | 191 | 0.00      | 2  | 207 | 0.13      | 2       | 223 | 0.00      |
| 300                 | 2  | 90  | 0.13      | 2       | 95  | 0.00      | 2  | 103 | 0.13      | 2       | 111 | 0.00      |
| 600                 | 1  | 181 | 0.13      | 1       | 191 | 0.00      | 1  | 207 | 0.13      | 1       | 223 | 0.00      |
| 1200                | 1  | 90  | 0.13      | 1       | 95  | 0.00      | 1  | 103 | 0.13      | 1       | 111 | 0.00      |
| 2400                | 0  | 181 | 0.13      | 0       | 191 | 0.00      | 0  | 207 | 0.13      | 0       | 223 | 0.00      |
| 4800                | 0  | 90  | 0.13      | 0       | 95  | 0.00      | 0  | 103 | 0.13      | 0       | 111 | 0.00      |
| 9600                | 0  | 45  | -0.93     | 0       | 47  | 0.00      | 0  | 51  | 0.13      | 0       | 55  | 0.00      |
| 19200               | 0  | 22  | -0.93     | 0       | 23  | 0.00      | 0  | 25  | 0.13      | 0       | 27  | 0.00      |
| 31250               | 0  | 13  | 0.00      | 0       | 14  | -1.70     | 0  | 15  | 0.00      | 0       | 13  | 1.20      |
| 38400               | —  | —   | —         | 0       | 11  | 0.00      | 0  | 12  | 0.13      | 0       | 13  | 0.00      |

| Bit Rate<br>(bit/s) | Operating Frequency $\phi$ (MHz) |     |           |         |     |           |    |     |           |
|---------------------|----------------------------------|-----|-----------|---------|-----|-----------|----|-----|-----------|
|                     | 18                               |     |           | 19.6608 |     |           | 20 |     |           |
|                     | 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      |

**Table 14-4 Maximum Bit Rate for Each Frequency (Asynchronous Mode)**

| $\phi$ (MHz) | Maximum Bit Rate (bit/s) | n | N | $\phi$ (MHz) | Maximum Bit Rate (bit/s) | n | N |
|--------------|--------------------------|---|---|--------------|--------------------------|---|---|
| 4            | 125000                   | 0 | 0 | 12           | 375000                   | 0 | 0 |
| 4.9152       | 153600                   | 0 | 0 | 12.288       | 384000                   | 0 | 0 |
| 5            | 156250                   | 0 | 0 | 14           | 437500                   | 0 | 0 |
| 6            | 187500                   | 0 | 0 | 14.7456      | 460800                   | 0 | 0 |
| 6.144        | 192000                   | 0 | 0 | 16           | 500000                   | 0 | 0 |
| 7.3728       | 230400                   | 0 | 0 | 17.2032      | 537600                   | 0 | 0 |
| 8            | 250000                   | 0 | 0 | 18           | 562500                   | 0 | 0 |
| 9.8304       | 307200                   | 0 | 0 | 19.6608      | 614400                   | 0 | 0 |
| 10           | 312500                   | 0 | 0 | 20           | 625000                   | 0 | 0 |

**Table 14-5 Maximum Bit Rate with External Clock Input (Asynchronous Mode)**

| <b>ø (MHz)</b> | <b>External Input Clock (MHz)</b> | <b>Maximum Bit Rate (bit/s)</b> | <b>ø (MHz)</b> | <b>External Input Clock (MHz)</b> | <b>Maximum Bit Rate (bit/s)</b> |
|----------------|-----------------------------------|---------------------------------|----------------|-----------------------------------|---------------------------------|
| 4              | 1.0000                            | 62500                           | 12             | 3.0000                            | 187500                          |
| 4.9152         | 1.2288                            | 76800                           | 12.288         | 3.0720                            | 192000                          |
| 5              | 1.2500                            | 78125                           | 14             | 3.5000                            | 218750                          |
| 6              | 1.5000                            | 93750                           | 14.7456        | 3.6864                            | 230400                          |
| 6.144          | 1.5360                            | 96000                           | 16             | 4.0000                            | 250000                          |
| 7.3728         | 1.8432                            | 115200                          | 17.2032        | 4.3008                            | 268800                          |
| 8              | 2.0000                            | 125000                          | 18             | 4.5000                            | 281250                          |
| 9.8304         | 2.4576                            | 153600                          | 19.6608        | 4.9152                            | 307200                          |
| 10             | 2.5000                            | 156250                          | 20             | 5.0000                            | 312500                          |

**Table 14-6 BRR Settings for Various Bit Rates (Clocked Synchronous Mode)**

| Bit Rate<br>(bit/s) | Operating Frequency $\phi$ (MHz) |     |   |     |    |     |    |     |    |     |
|---------------------|----------------------------------|-----|---|-----|----|-----|----|-----|----|-----|
|                     | 4                                |     | 8 |     | 10 |     | 16 |     | 20 |     |
|                     | n                                | N   | n | N   | n  | N   | n  | N   | n  | N   |
| 110                 | —                                | —   |   |     |    |     |    |     |    |     |
| 250                 | 2                                | 249 | 3 | 124 | —  | —   | 3  | 249 |    |     |
| 500                 | 2                                | 124 | 2 | 249 | —  | —   | 3  | 124 | —  | —   |
| 1k                  | 1                                | 249 | 2 | 124 | —  | —   | 2  | 249 | —  | —   |
| 2.5k                | 1                                | 99  | 1 | 199 | 1  | 249 | 2  | 99  | 2  | 124 |
| 5k                  | 0                                | 199 | 1 | 99  | 1  | 124 | 1  | 199 | 1  | 249 |
| 10k                 | 0                                | 99  | 0 | 199 | 0  | 249 | 1  | 99  | 1  | 124 |
| 25k                 | 0                                | 39  | 0 | 79  | 0  | 99  | 0  | 159 | 0  | 199 |
| 50k                 | 0                                | 19  | 0 | 39  | 0  | 49  | 0  | 79  | 0  | 99  |
| 100k                | 0                                | 9   | 0 | 19  | 0  | 24  | 0  | 39  | 0  | 49  |
| 250k                | 0                                | 3   | 0 | 7   | 0  | 9   | 0  | 15  | 0  | 19  |
| 500k                | 0                                | 1   | 0 | 3   | 0  | 4   | 0  | 7   | 0  | 9   |
| 1M                  | 0                                | 0*  | 0 | 1   |    |     | 0  | 3   | 0  | 4   |
| 2.5M                |                                  |     |   |     | 0  | 0*  |    |     | 0  | 1   |
| 5M                  |                                  |     |   |     |    |     |    |     | 0  | 0*  |

**Legend**

Blank : Cannot be set.

— : Can be set, but there will be a degree of error.

\* : Continuous transfer is not possible.

**Table 14-7 Maximum Bit Rate with External Clock Input (Clocked Synchronous Mode)**

| $\phi$ (MHz) | External Input<br>Clock (MHz) | Maximum Bit<br>Rate (bit/s) | $\phi$ (MHz) | External Input<br>Clock (MHz) | Maximum Bit<br>Rate (bit/s) |
|--------------|-------------------------------|-----------------------------|--------------|-------------------------------|-----------------------------|
| 4            | 0.6667                        | 666666.7                    | 14           | 2.3333                        | 2333333.3                   |
| 6            | 1.0000                        | 1.000000.0                  | 16           | 2.6667                        | 2666666.7                   |
| 8            | 1.3333                        | 1333333.3                   | 18           | 3.0000                        | 3000000.0                   |
| 10           | 1.6667                        | 1666666.7                   | 20           | 3.3333                        | 3333333.3                   |
| 12           | 2.0000                        | 2000000.0                   |              |                               |                             |

**Table 14-8 Examples of Bit Rate for Various BRR Settings (Smart Card Interface Mode)**  
(When n = 0 and S = 372)

| Bit Rate<br>(bit/s) | Operating Frequency $\phi$ (MHz) |   |           |       |   |           |         |   |           |       |   |           |
|---------------------|----------------------------------|---|-----------|-------|---|-----------|---------|---|-----------|-------|---|-----------|
|                     | 7.1424                           |   |           | 10.00 |   |           | 10.7136 |   |           | 13.00 |   |           |
|                     | n                                | N | Error (%) | n     | N | Error (%) | n       | N | Error (%) | n     | N | Error (%) |
| 9600                | 0                                | 0 | 0.00      | 0     | 1 | 30        | 0       | 1 | 25        | 0     | 1 | 8.99      |

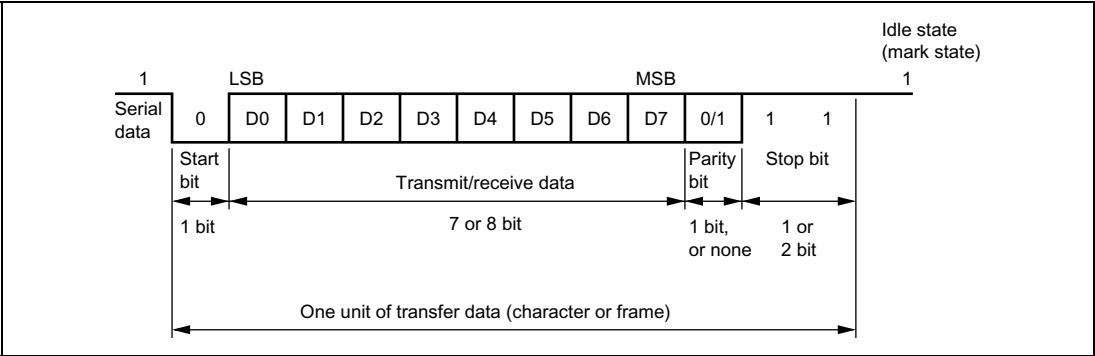
| Bit Rate<br>(bit/s) | Operating Frequency $\phi$ (MHz) |   |           |       |   |           |       |   |           |       |   |           |
|---------------------|----------------------------------|---|-----------|-------|---|-----------|-------|---|-----------|-------|---|-----------|
|                     | 14.2848                          |   |           | 16.00 |   |           | 18.00 |   |           | 20.00 |   |           |
|                     | n                                | N | Error (%) | n     | N | Error (%) | n     | N | Error (%) | n     | N | Error (%) |
| 9600                | 0                                | 1 | 0.00      | 0     | 1 | 12.01     | 0     | 2 | 15.99     | 0     | 2 | 6.60      |

**Table 14-9 Maximum Bit Rate at Various Frequencies (Smart Card Interface Mode)**  
(When S = 372)

| $\phi$ (MHz) | Maximum Bit Rate (bit/s) | n | N | $\phi$ (MHz) | Maximum Bit Rate (bit/s) | n | N |
|--------------|--------------------------|---|---|--------------|--------------------------|---|---|
| 7.1424       | 9600                     | 0 | 0 | 14.2848      | 19200                    | 0 | 0 |
| 10.00        | 13441                    | 0 | 0 | 16.00        | 21505                    | 0 | 0 |
| 10.7136      | 14400                    | 0 | 0 | 18.00        | 24194                    | 0 | 0 |
| 13.00        | 17473                    | 0 | 0 | 20.00        | 26882                    | 0 | 0 |

# 14.4     Operation in Asynchronous Mode

Figure 14-2 shows the general format for asynchronous serial communication. One frame consists of a start bit (low level), followed by data (in LSB-first order), a parity bit (high or low level), 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. When the transmission line goes to the space state (low level), the SCI recognizes a start bit and starts serial communication. In asynchronous serial communication, the communication line is usually held in the mark state (high level). The SCI monitors the communication 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. 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 14-2   Data Format in Asynchronous Communication**  
**(Example with 8-Bit Data, Parity, Two Stop Bits)**

## 14.4.1     Data Transfer Format

Table 14-10 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 14.5, Multiprocessor Communication Function.

**Table 14-10 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

#### 14.4.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a basic clock with a frequency of 16 times the transfer rate. In reception, the SCI samples the falling edge of the start bit using the basic clock, and performs internal synchronization. Receive data is latched internally at the rising edge of the 8th pulse of the basic clock as shown in figure 14-3. Thus, the reception margin in asynchronous mode is given by formula (1) below.

$$M = \left\{ \left( 0.5 - \frac{1}{2N} \right) - \frac{D - 0.5}{N} - (L - 0.5) F \right\} \times 100 [\%]$$

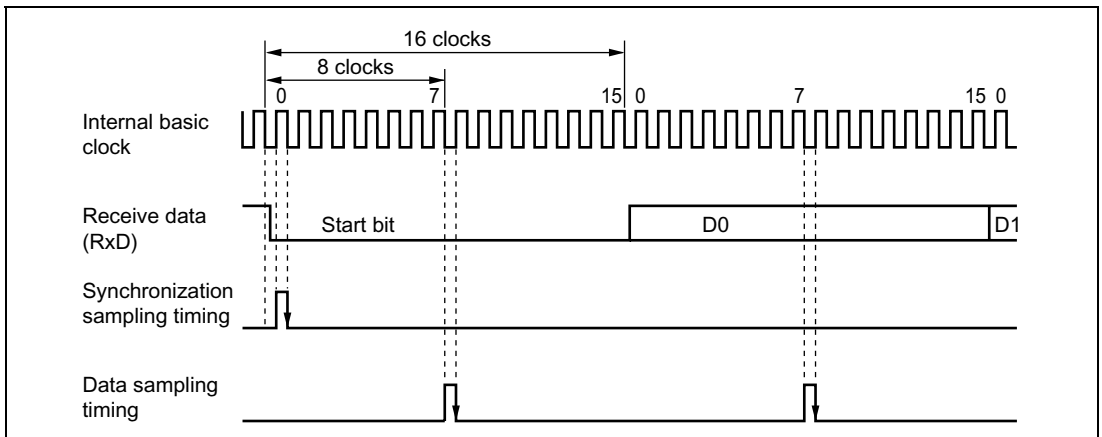
... Formula (1)

Where M : Reception margin  
N : Ratio of bit rate to clock (N = 16)  
D : Clock duty (D = 0.5 to 1.0)  
L : Frame length (L = 9 to 12)  
F : Absolute value of clock rate deviation

Assuming values of F (absolute value of clock rate deviation) = 0 and D (clock duty) = 0.5 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.

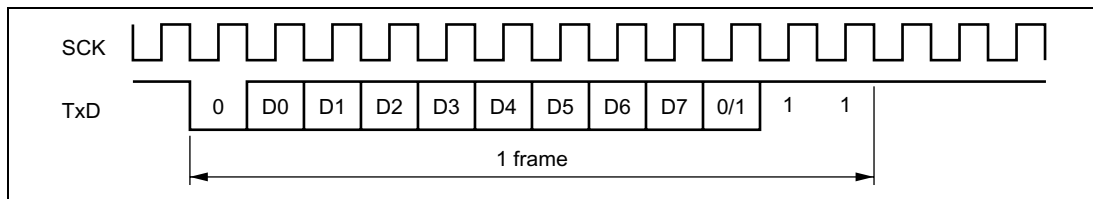


**Figure 14-3 Receive Data Sampling Timing in Asynchronous Mode**

### 14.4.3 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. 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 14-4.



**Figure 14-4 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode)**

#### 14.4.4 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 below. 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.

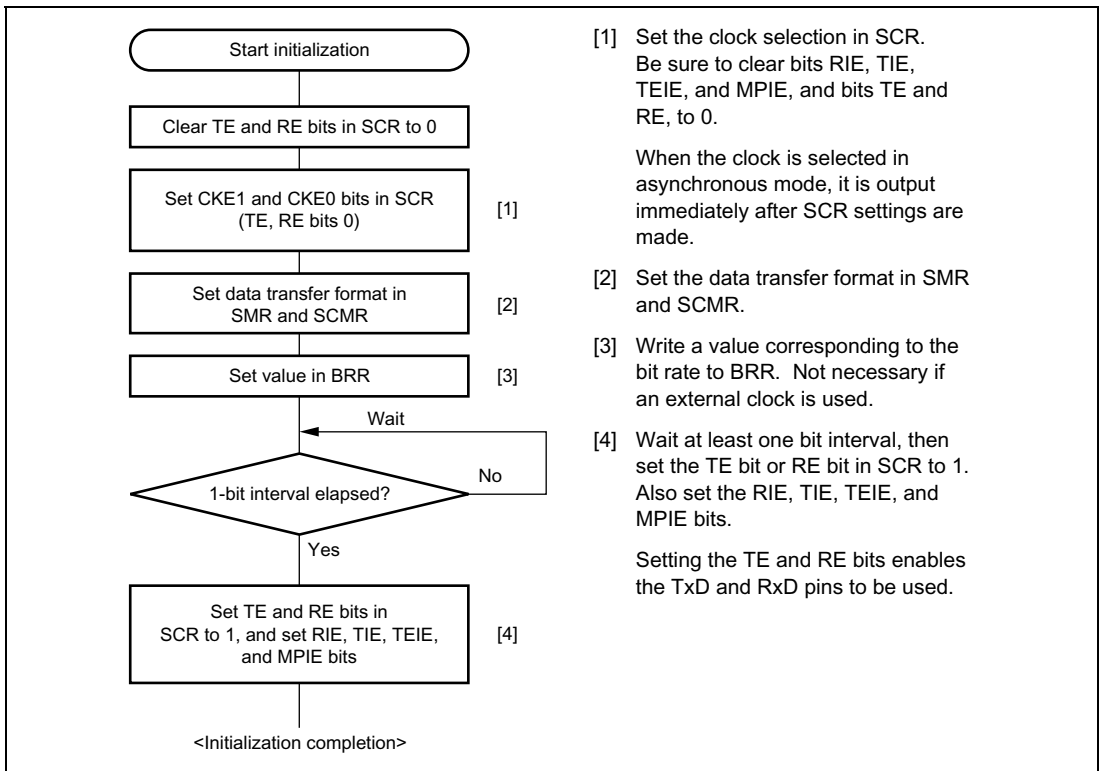


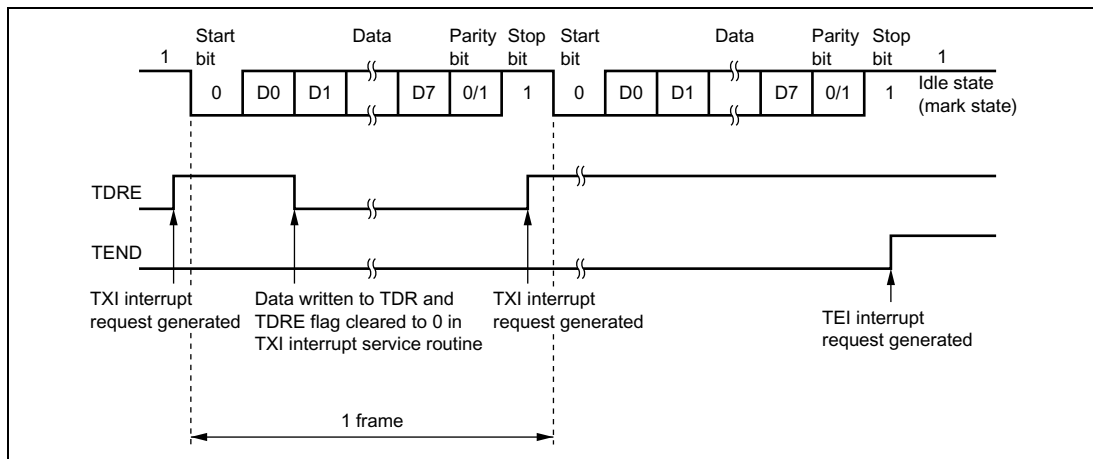
Figure 14-5 Sample SCI Initialization Flowchart

### 14.4.5 Data Transmission (Asynchronous Mode)

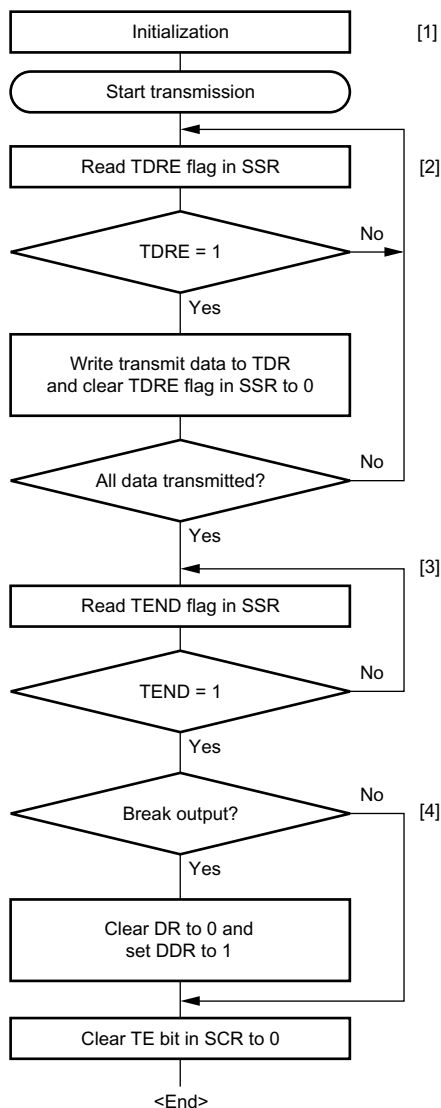
Figure 14-6 shows an example of operation for transmission in asynchronous mode. In transmission, the SCI operates as described below.

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 14-7 shows a sample flowchart for transmission in asynchronous mode.



**Figure 14-6 Example of Operation in Transmission in Asynchronous Mode  
(Example with 8-Bit Data, Parity, One Stop Bit)**



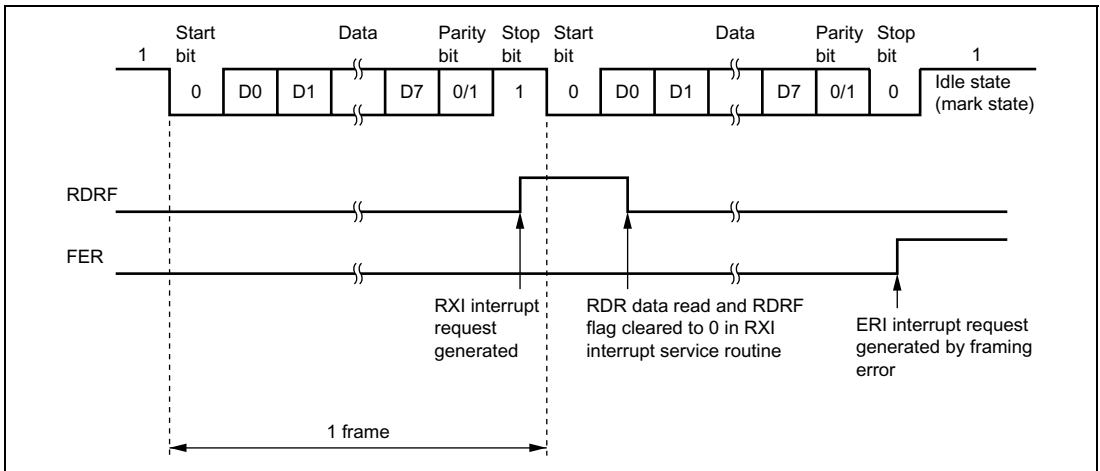
- [1] SCI initialization:  
The TxD pin is automatically designated as the transmit data output pin.  
After the TE bit is set to 1, a frame of 1s is output, and transmission is enabled.
- [2] SCI status check and transmit data write:  
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.
- [3] Serial transmission continuation procedure:  
To continue serial transmission, 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 DTC 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 DDR for the port corresponding to the TxD pin to 1, clear DR to 0, then clear the TE bit in SCR to 0.

**Figure 14-7 Sample Serial Transmission Flowchart**

#### 14.4.6 Serial Data Reception (Asynchronous Mode)

Figure 14-8 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 ORE 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.



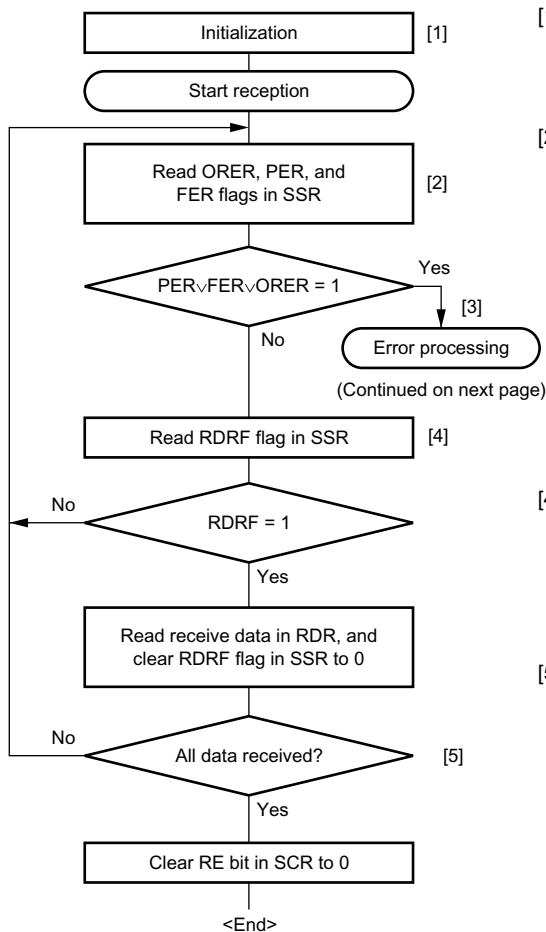
**Figure 14-8 Example of SCI Operation in Reception**  
(Example with 8-Bit Data, Parity, One Stop Bit)

Table 14-11 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 14-9 shows a sample flow chart for serial data reception.

**Table 14-11 SSR Status Flags and Receive Data Handling**

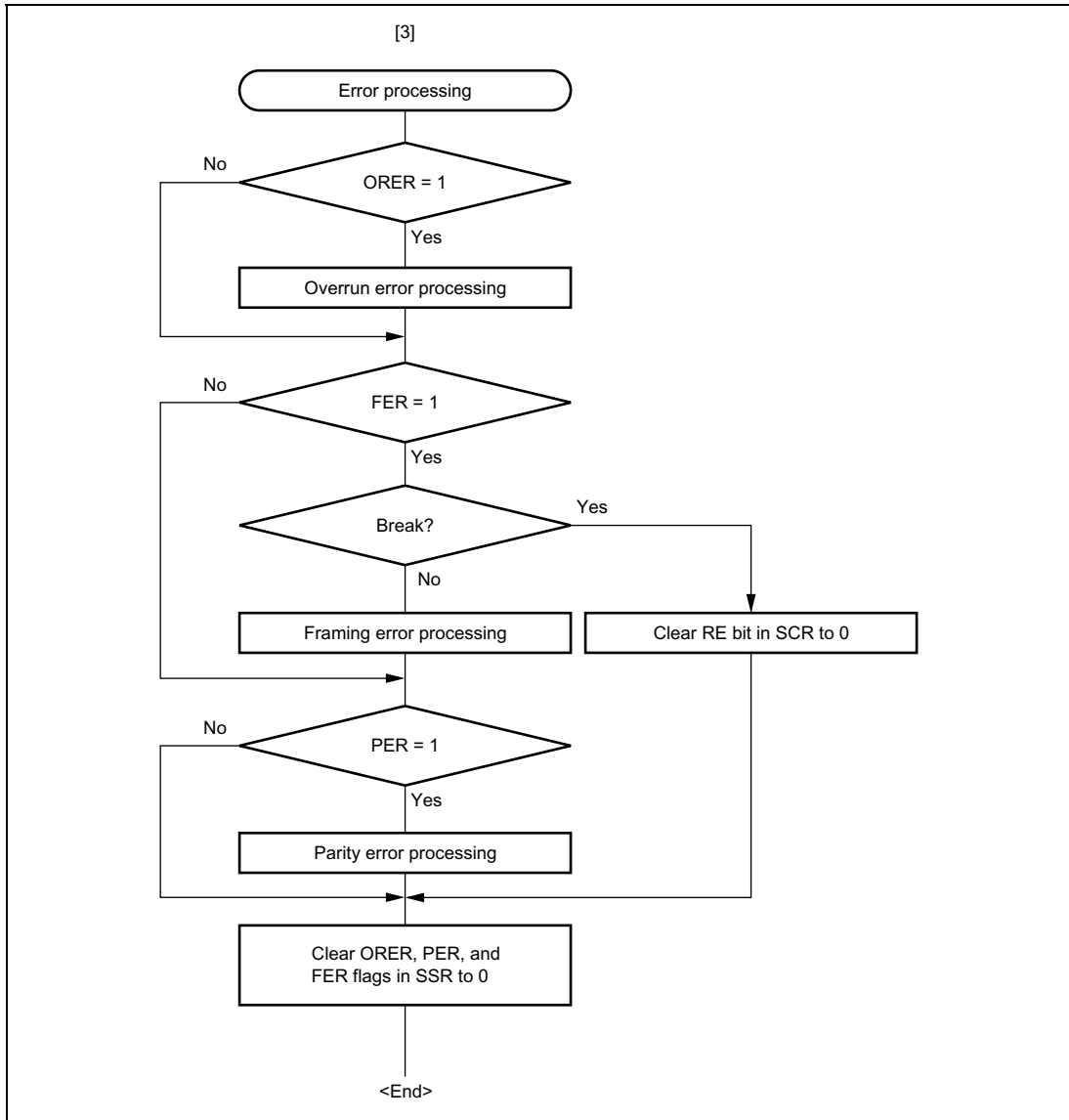
| SSR Status Flag |      |     |     | Receive Data       | Receive Error Type                           |
|-----------------|------|-----|-----|--------------------|----------------------------------------------|
| RDRF*           | ORER | FER | PER |                    |                                              |
| 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 DTC is activated by an RXI interrupt and the RDR value is read.

**Figure 14-9 Sample Serial Reception Data Flowchart (1)**



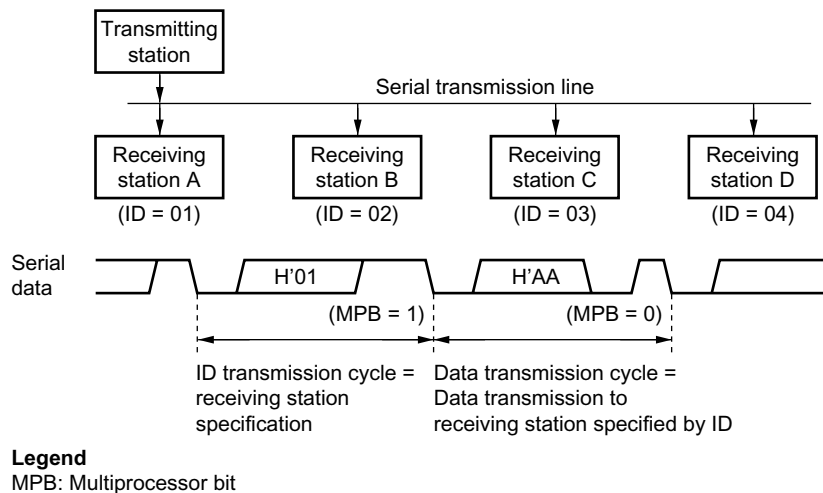
**Figure 14-9 Sample Serial Reception Data Flowchart (2)**

## 14.5 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 14-10 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 14-10 Example of Communication Using Multiprocessor Format  
(Transmission of Data H'AA to Receiving Station A)**

## 14.5.1 Multiprocessor Serial Data Transmission

Figure 14-11 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.

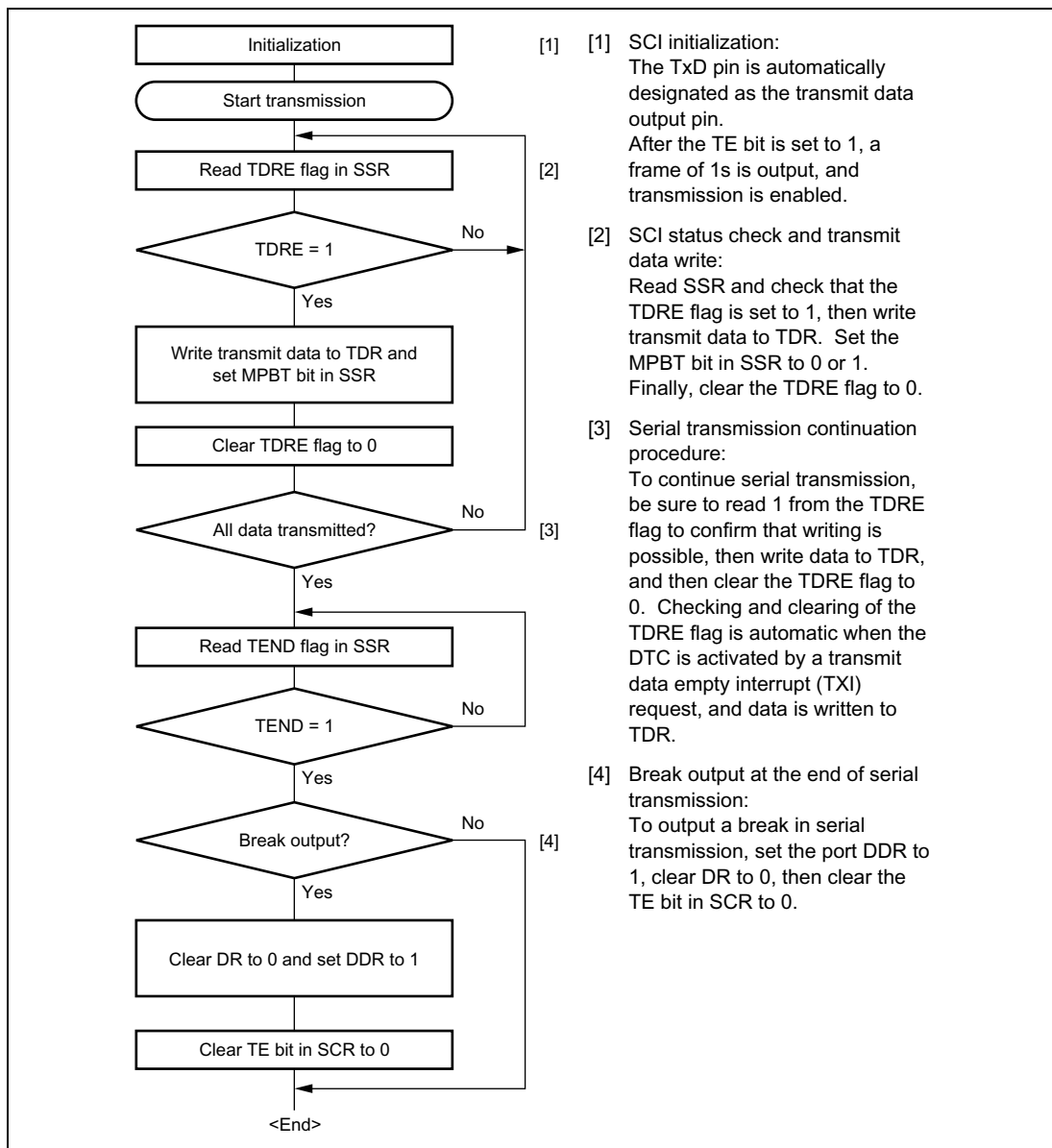
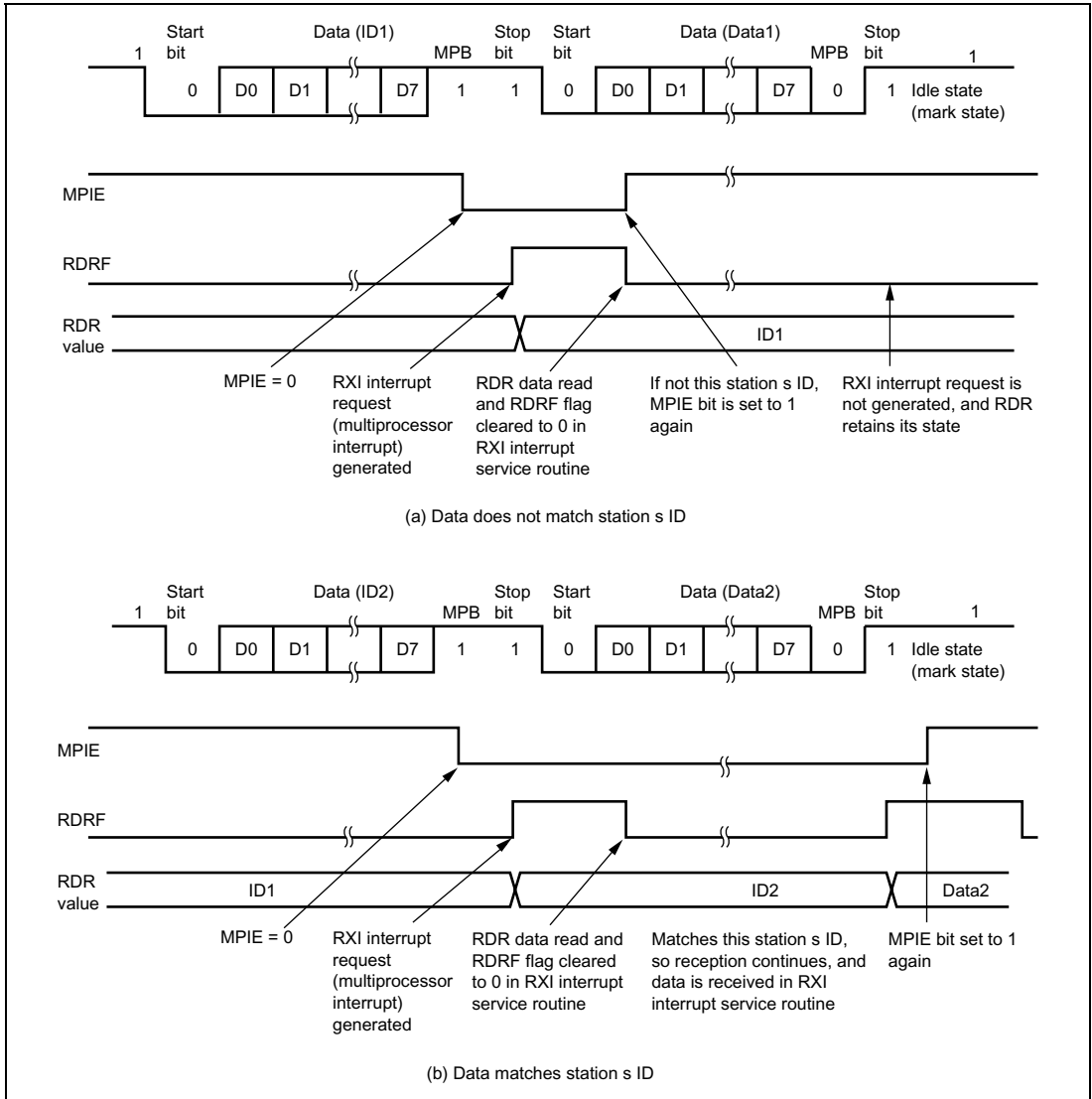


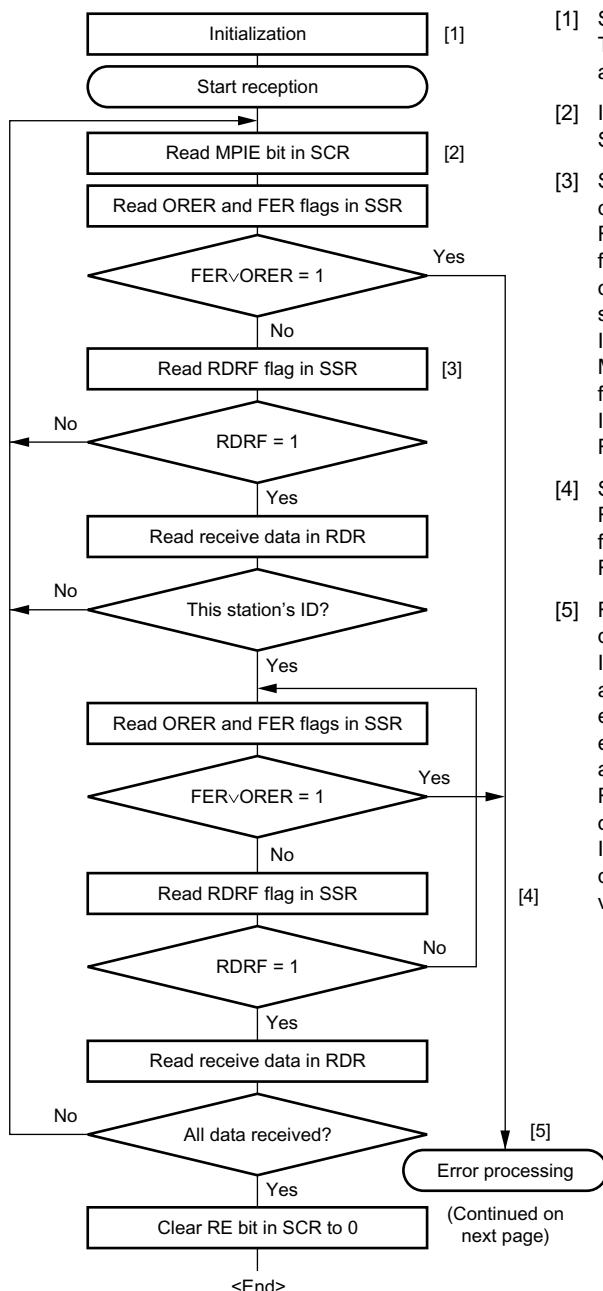
Figure 14-11 Sample Multiprocessor Serial Transmission Flowchart

## 14.5.2 Multiprocessor Serial Data Reception

Figure 14-13 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 14-12 shows an example of SCI operation for multiprocessor format reception.



**Figure 14-12 Example of SCI Operation in Reception**  
(Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)



[1] SCI initialization:  
The RxD pin is automatically designated as the receive data input pin.

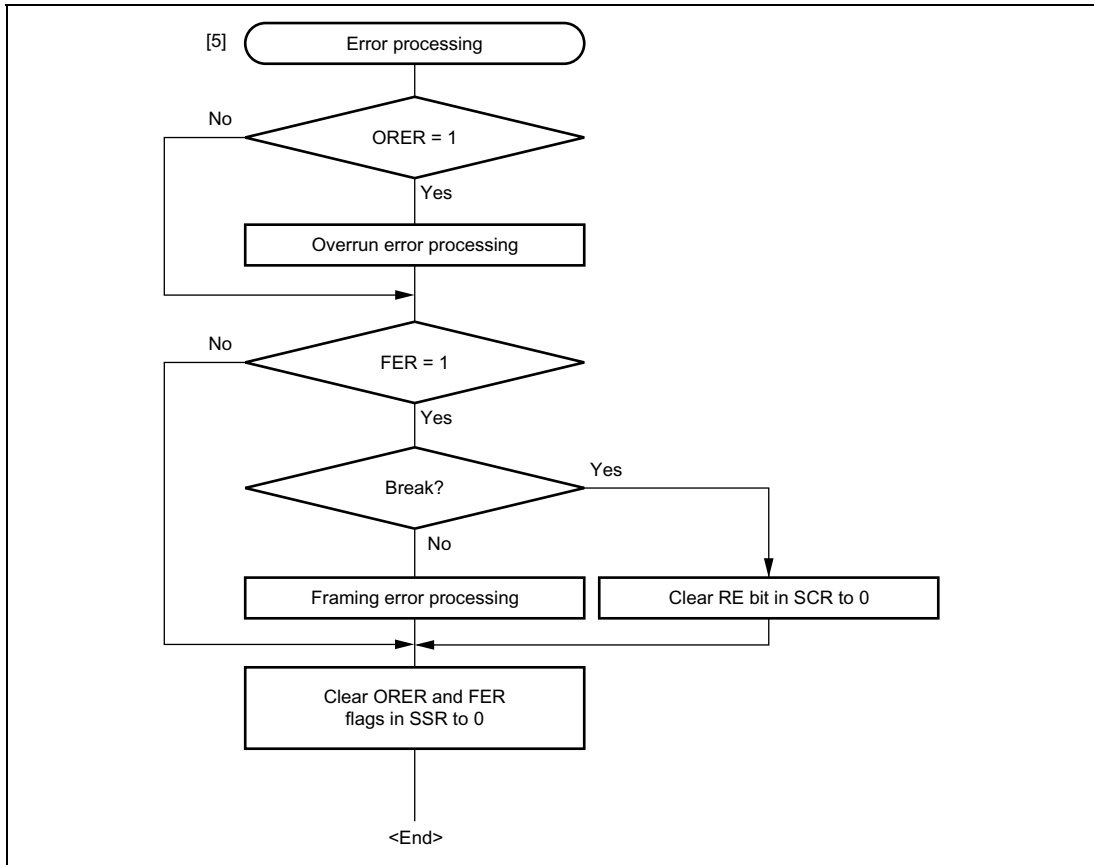
[2] ID reception cycle:  
Set the MPIE bit in SCR to 1.

[3] SCI status check, ID reception and comparison:  
Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR and compare it with this station's ID.  
If the data is not this station's ID, set the MPIE bit to 1 again, and clear the RDRF flag to 0.  
If the data is this station's ID, clear the RDRF flag to 0.

[4] SCI status check and data reception:  
Read SSR and check that the RDRF flag is set to 1, then read the data in RDR.

[5] Receive error processing and break detection:  
If a receive error occurs, read the ORER and FER flags in SSR to identify the error. After performing the appropriate error processing, ensure that the ORER and FER flags are all cleared to 0. Reception cannot be resumed if either of these flags is set to 1.  
In the case of a framing error, a break can be detected by reading the RxD pin value.

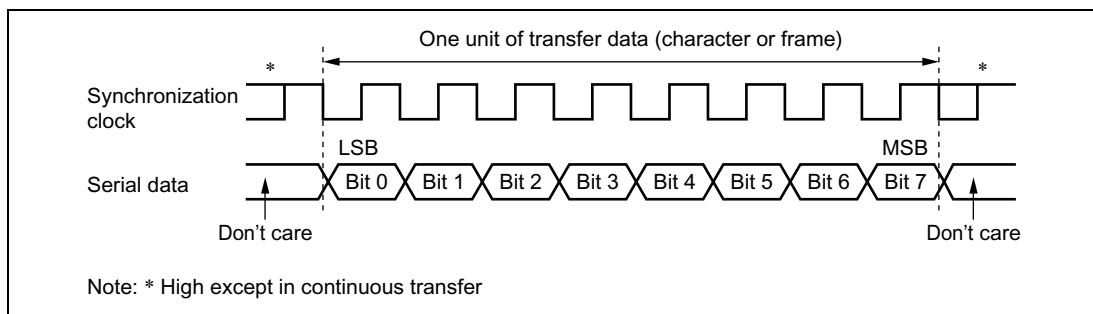
Figure 14-13 Sample Multiprocessor Serial Reception Flowchart (1)



**Figure 14-13 Sample Multiprocessor Serial Reception Flowchart (2)**

## 14.6 Operation in Clocked Synchronous Mode

Figure 14-14 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.



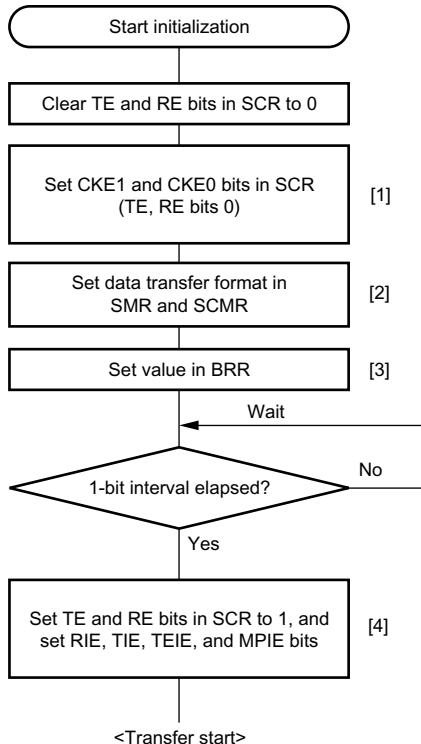
**Figure 14-14 Data Format in Synchronous Communication (For LSB-First)**

### 14.6.1 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.

## 14.6.2 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 14-15. 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.



- [1] Set the clock selection in SCR. Be sure to clear bits RIE, TIE, TEIE, and MPPIE, TE and RE, to 0.
- [2] Set the data transfer format in SMR and SCMR.
- [3] Write a value corresponding to the bit rate to BRR. Not necessary if an external clock is used.
- [4] Wait at least one bit interval, then set the TE bit or RE bit in SCR to 1. Also set the RIE, TIE, TEIE, and MPPIE bits. Setting the TE and RE bits enables the TxD and RxD pins to be used.

Note: In simultaneous transmit and receive operations, the TE and RE bits should both be cleared to 0 or set to 1 simultaneously.

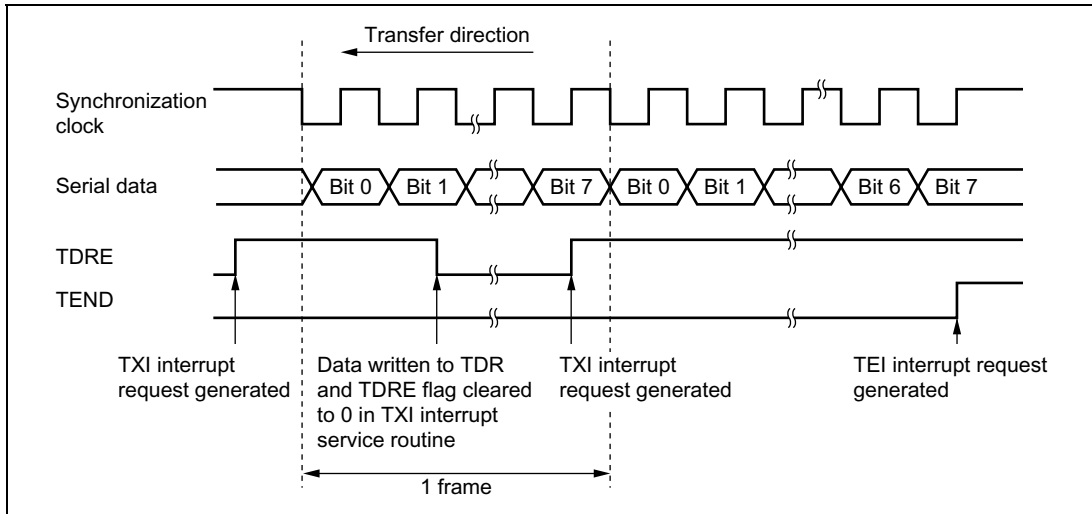
**Figure 14-15 Sample SCI Initialization Flowchart**

### 14.6.3 Serial Data Transmission (Clocked Synchronous Mode)

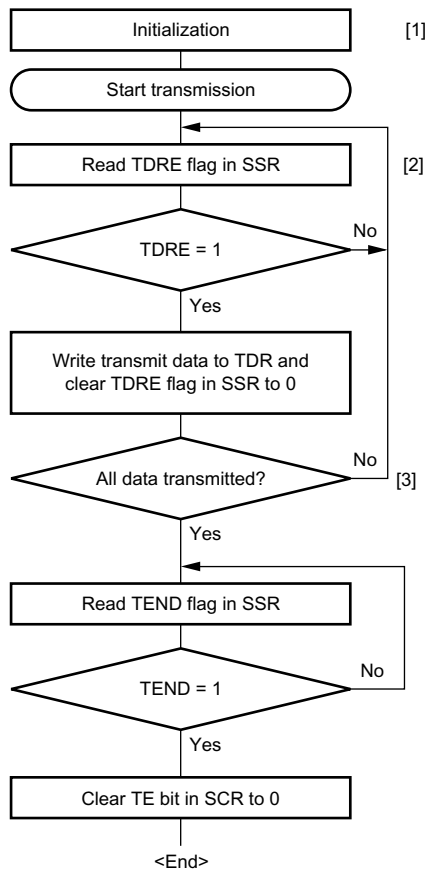
Figure 14-16 shows an example of SCI operation for transmission in clocked synchronous mode. In serial transmission, the SCI operates as described below.

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 14-17 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 14-16 Sample SCI Transmission Operation in Clocked Synchronous Mode**



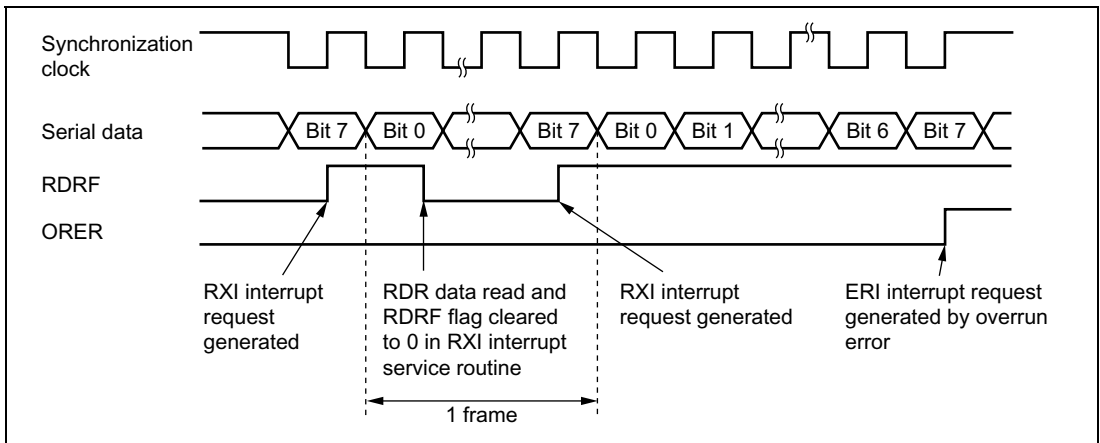
- [1] SCI initialization:  
The TxD pin is automatically designated as the transmit data output pin.
- [2] SCI status check and transmit data write:  
Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR and clear the TDRE flag to 0.
- [3] Serial transmission continuation 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 DTC is activated by a transmit data empty interrupt (TXI) request and data is written to TDR.

**Figure 14-17 Sample Serial Transmission Flowchart**

#### 14.6.4 Serial Data Reception (Clocked Synchronous Mode)

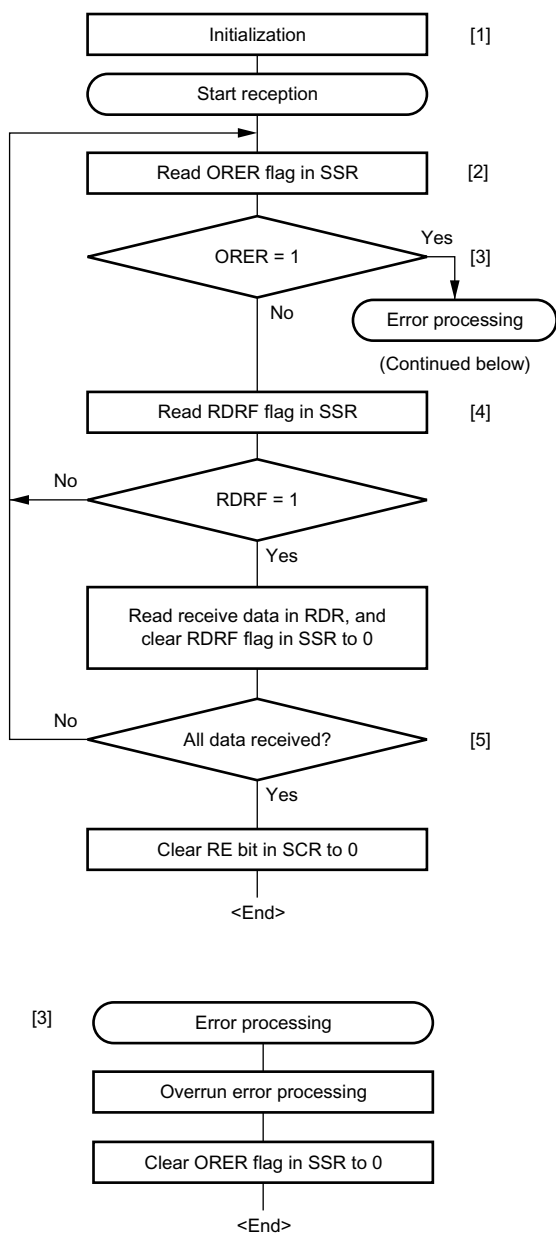
Figure 14-18 shows an example of SCI operation for reception in clocked synchronous mode. In serial reception, the SCI operates as described below.

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 14-18 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 14-19 shows a sample flow chart for serial data reception.

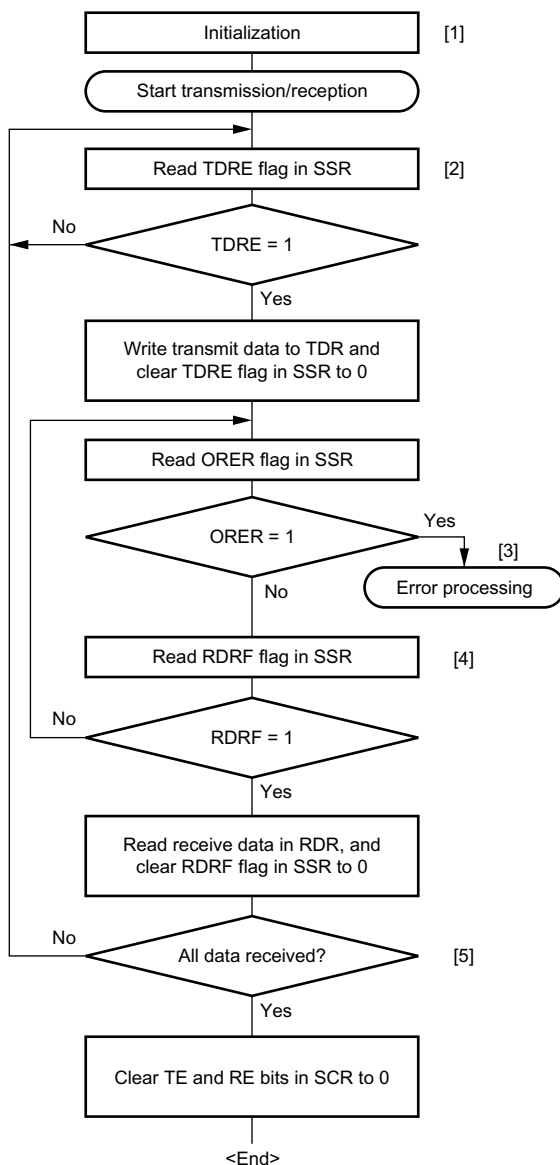


- [1] SCI initialization:  
The Rx/D pin is automatically designated as the receive data input pin.
- [2] [3] Receive error processing:  
If a receive error occurs, read the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag to 0. Transfer cannot be resumed if the ORER flag is set to 1.
- [4] SCI status check and receive data read:  
Read SSR and check that the RDRF flag is set to 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 MSB (bit 7) of the current frame is received, reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0 should be finished. The RDRF flag is cleared automatically when the DTC is activated by a receive data full interrupt (RXI) request and the RDR value is read.

**Figure 14-19 Sample Serial Reception Flowchart**

#### **14.6.5 Simultaneous Serial Data Transmission and Reception (Clocked Synchronous Mode)**

Figure 14-20 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.



- [1] SCI initialization:  
The TxD pin is designated as the transmit data output pin, and the RxD pin is designated as the receive data input pin, enabling simultaneous transmit and receive operations.
- [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.  
Transition of the TDRE flag from 0 to 1 can also be identified by a TXI interrupt.
- [3] Receive error processing:  
If a receive error occurs, read the ORER flag in SSR, and after performing the appropriate error processing, clear the ORER flag to 0. Transmission/reception cannot be resumed if the ORER flag is set to 1.
- [4] SCI status check and receive data read:  
Read SSR and check that the RDRF flag is set to 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 transmission/reception continuation procedure:  
To continue serial transmission/reception, before the MSB (bit 7) of the current frame is received, finish reading the RDRF flag, reading RDR, and clearing the RDRF flag to 0. Also, before the MSB (bit 7) of the current frame is transmitted, read 1 from the TDRE flag to confirm that writing is possible. Then write data to TDR and clear the TDRE flag to 0. Checking and clearing of the TDRE flag is automatic when the DTC is activated by a transmit data empty interrupt (TXI) request and data is written to TDR. Also, the RDRF flag is cleared automatically when the DTC is activated by a receive data full interrupt (RXI) request and the RDR value is read.

Note: 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 simultaneously.

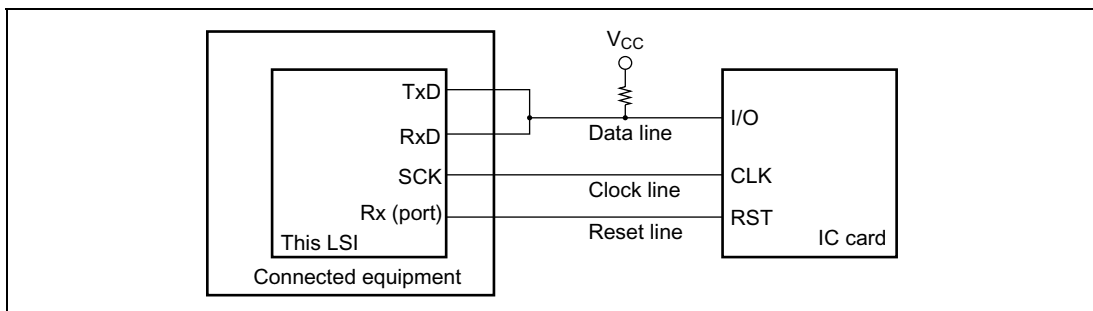
**Figure 14-20 Sample Flowchart of Simultaneous Serial Transmit and Receive Operations**

## 14.7 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.

### 14.7.1 Pin Connection Example

Figure 14-21 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.

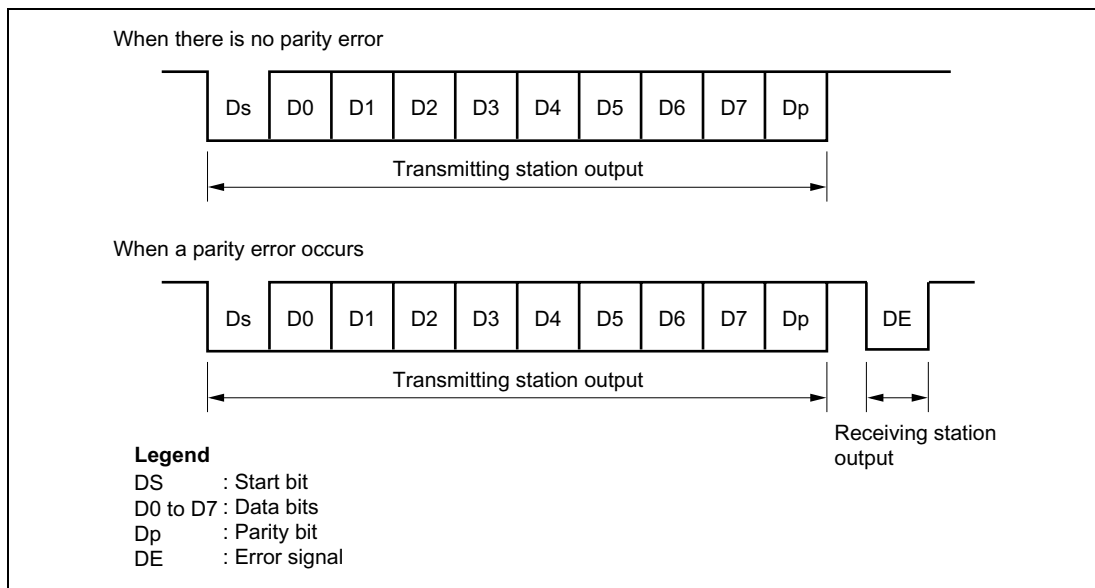


**Figure 14-21 Schematic Diagram of Smart Card Interface Pin Connections**

## 14.7.2 Data Format (Except for Block Transfer Mode)

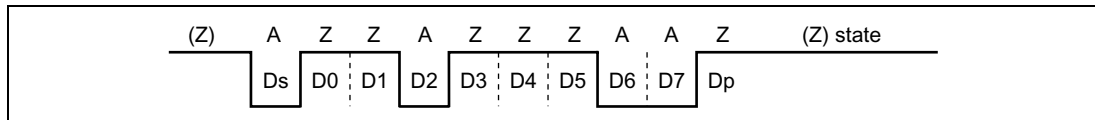
Figure 14-22 shows the transfer data format in Smart Card interface mode.

- 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.



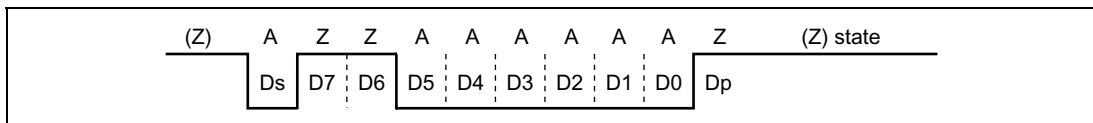
**Figure 14-22 Normal Smart Card Interface Data Format**

Data transfer with other types of IC cards (direct convention and inverse convention) are performed as described in the following.



**Figure 14-23 Direct Convention (SDIR = SINV = 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 O/ $\overline{E}$  bit in SMR to 0 to select even parity mode.



**Figure 14-24 Inverse Convention (SDIR = SINV = O/ $\overline{E}$  = 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.

### 14.7.3 Block Transfer Mode

Operation in block transfer mode is the same as that in SCI asynchronous 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.

## 14.7.4 Receive Data Sampling Timing and Reception Margin in Smart Card Interface Mode

In Smart Card interface mode, the SCI operates on a basic clock with a frequency of 32, 64, 372, or 256 times the transfer rate (fixed at 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 basic clock, and performs internal synchronization. As shown in figure 14-25, by sampling receive data at the rising-edge of the 16th, 32nd, 186th, or 128th pulse of the basic 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)

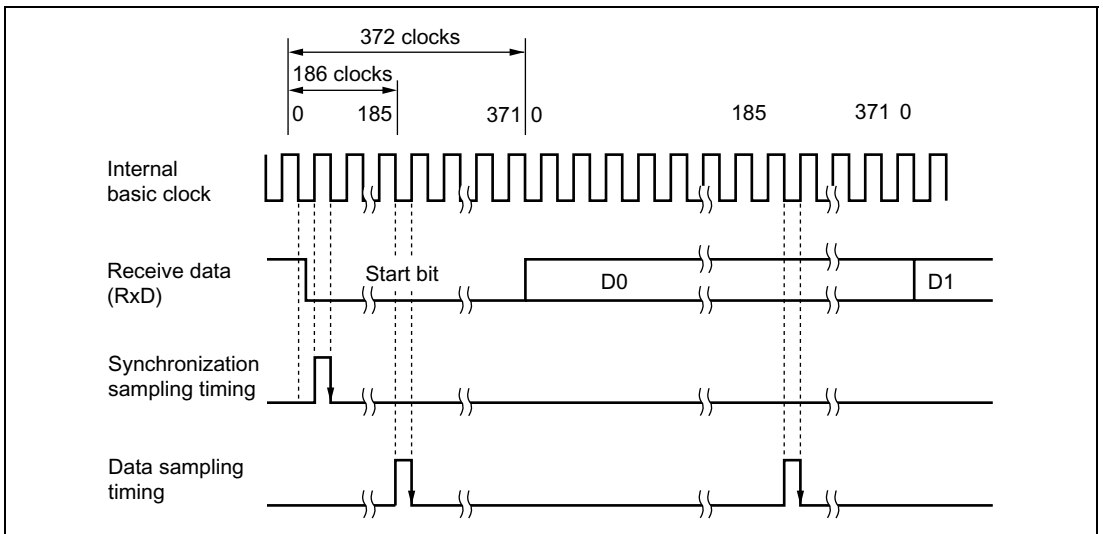
D: Clock duty (D = 0 to 1.0)

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.

$$\begin{aligned} M &= (0.5 - 1/2 \times 372) \times 100\% \\ &= 49.866\% \end{aligned}$$



**Figure 14-25 Receive Data Sampling Timing in Smart Card Mode  
(Using Clock of 372 Times the Transfer Rate)**

### 14.7.5 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/ $\overline{E}$ , BCP0, BCP1, CKS0, CKS1 bits in SMR. Set the PE bit to 1.
4. Set the SMIF, SDIR, and SINV 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.

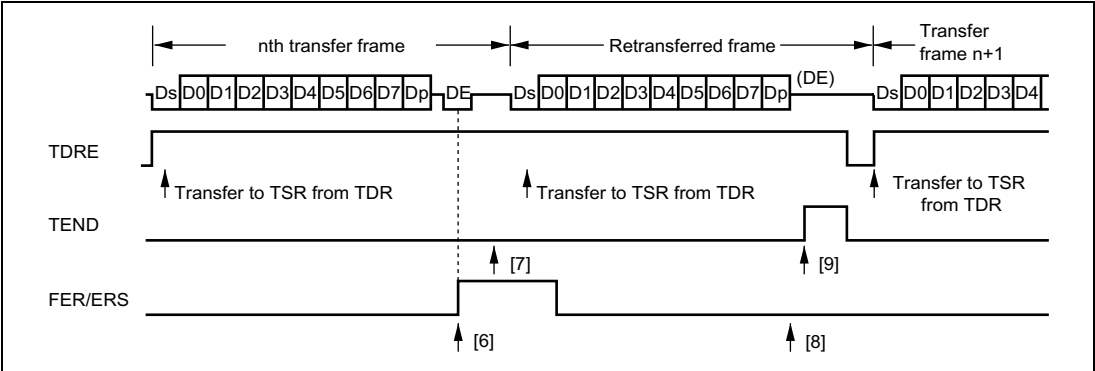
### 14.7.6 Data Transmission (Except for Block Transfer Mode)

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 14-26 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 kept cleared to 0 until 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.

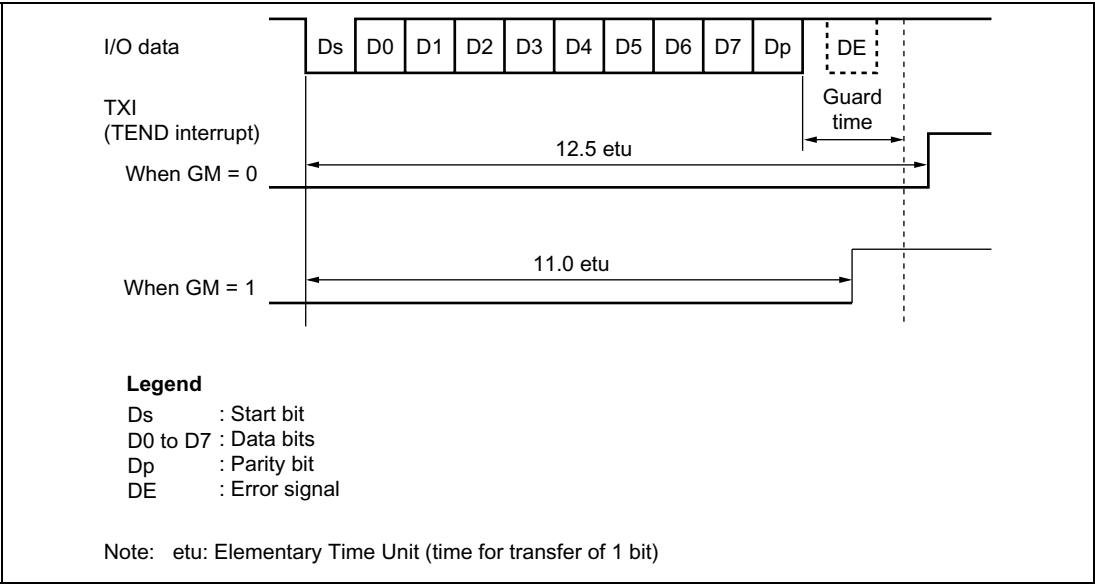
Figure 14-28 shows a flowchart for transmission. The sequence of transmit operations can be performed automatically by specifying the DTC to be activated with a TXI interrupt source. In a transmit operation, the TDRE flag is set to 1 at the same time as the TEND flag in SSR is set, and a TXI interrupt 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. The TDRE and TEND flags are automatically cleared to 0 when data is transferred by the DTC. In the event of an 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 8, Data Transfer Controller (DTC).

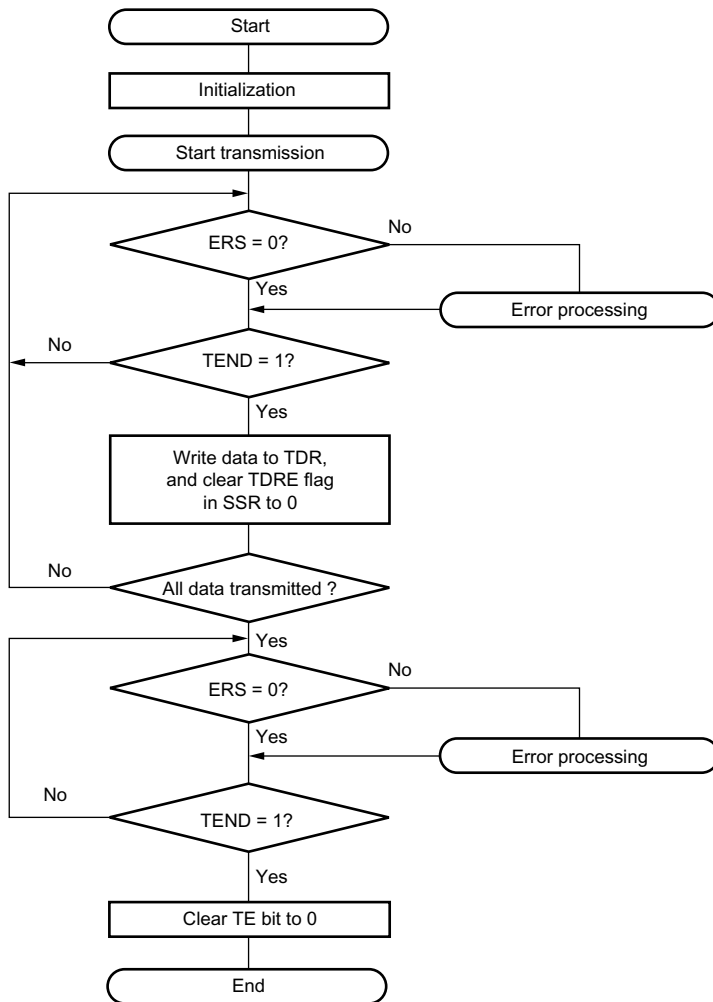


**Figure 14-26 Retransfer Operation in SCI Transmit Mode**

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 14-27.



**Figure 14-27 TEND Flag Generation Timing in Transmission Operation**



**Figure 14-28 Example of Transmission Processing Flow**

### 14.7.7 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 14-29 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 14-30 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. The RDRF flag is cleared to 0 automatically when data is transferred by the DTC. 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 14.4, Operation in Asynchronous Mode.

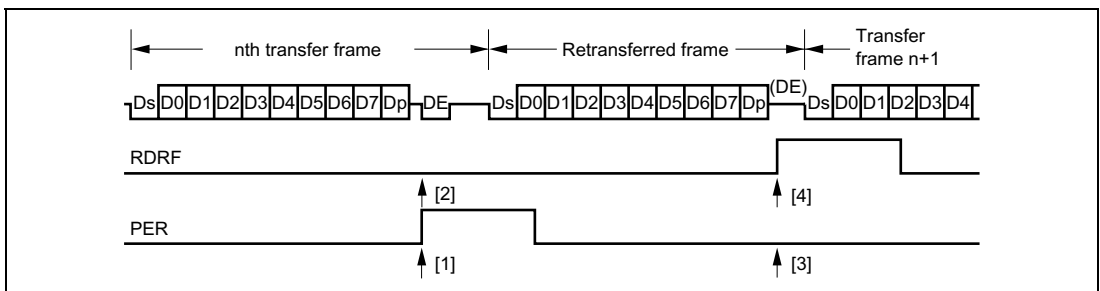
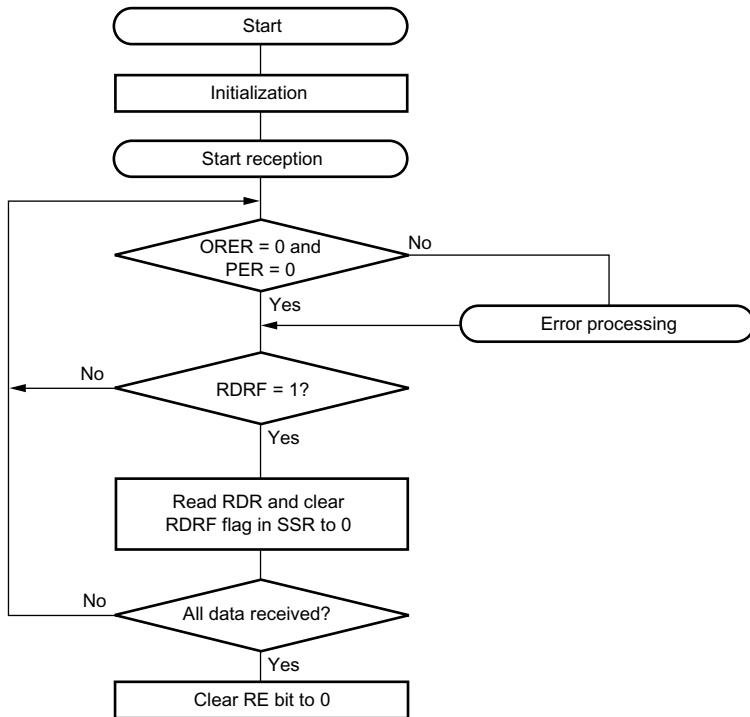


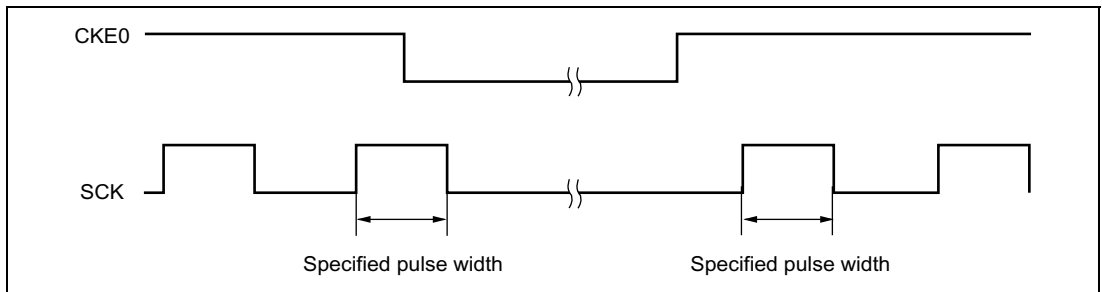
Figure 14-29 Retransfer Operation in SCI Receive Mode



**Figure 14-30 Example of Reception Processing Flow**

### 14.7.8 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 14-31 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 14-31 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:**

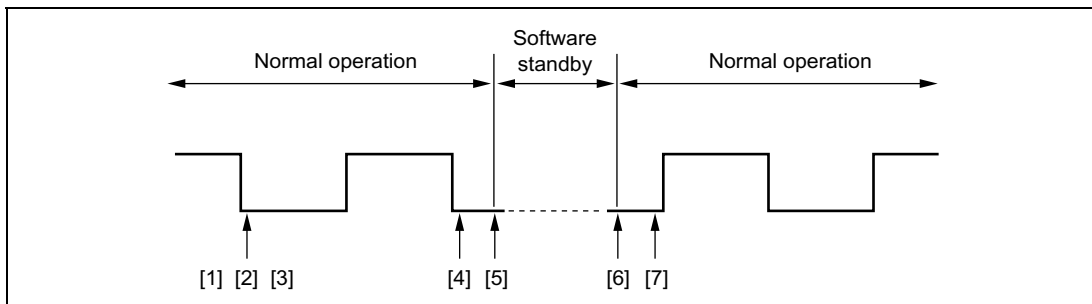
1. Set the data register (DR) and data direction register (DDR) corresponding to the SCK pin to the value for the fixed output state in software standby mode.
2. Write 0 to the TE bit and RE bit in the serial control register (SCR) to halt transmit/receive operation. At the same time, set the CKE1 bit 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 14-32 Clock Halt and Restart Procedure**

## 14.8 Interrupts

### 14.8.1 Interrupts in Normal Serial Communication Interface Mode

Table 14-12 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 DTC.

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 DTC to transfer data. The RDRF flag is cleared to 0 automatically when data is transferred by the DTC.

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.

**Table 14-12 SCI Interrupt Sources**

| Channel | Name  | Interrupt Source    | Interrupt Flag | DTC Activation |
|---------|-------|---------------------|----------------|----------------|
| 0       | ERI_0 | Receive Error       | ORER, FER, PER | Not possible   |
|         | RXI_0 | Receive Data Full   | RDRF           | Possible       |
|         | TXI_0 | Transmit Data Empty | TDRE           | Possible       |
|         | TEI_0 | Transmission End    | TEND           | Not possible   |
| 1       | ERI_1 | Receive Error       | ORER, FER, PER | Not possible   |
|         | RXI_1 | Receive Data Full   | RDRF           | Possible       |
|         | TXI_1 | Transmit Data Empty | TDRE           | Possible       |
|         | TEI_1 | Transmission End    | TEND           | Not possible   |
| 2       | ERI_2 | Receive Error       | ORER, FER, PER | Not possible   |
|         | RXI_2 | Receive Data Full   | RDRF           | Possible       |
|         | TXI_2 | Transmit Data Empty | TDRE           | Possible       |
|         | TEI_2 | Transmission End    | TEND           | Not possible   |

## 14.8.2 Interrupts in Smart Card Interface Mode

Table 14-13 shows the interrupt sources in Smart Card interface mode. The transmit end interrupt (TEI) request cannot be used in this mode.

**Table 14-13 SCI Interrupt Sources**

| Channel | Name  | Interrupt Source         | Interrupt Flag | DTC Activation |
|---------|-------|--------------------------|----------------|----------------|
| 0       | ERI_0 | Receive Error, detection | ORER, PER, ERS | Not possible   |
|         | RXI_0 | Receive Data Full        | RDRF           | Possible       |
|         | TXI_0 | Transmit Data Empty      | TEND           | Possible       |
| 1       | ERI_1 | Receive Error, detection | ORER, PER, ERS | Not possible   |
|         | RXI_1 | Receive Data Full        | RDRF           | Possible       |
|         | TXI_1 | Transmit Data Empty      | TEND           | Possible       |
| 2       | ERI_2 | Receive Error, detection | ORER, PER, ERS | Not possible   |
|         | RXI_2 | Receive Data Full        | RDRF           | Possible       |
|         | TXI_2 | Transmit Data Empty      | TEND           | Possible       |

In Smart Card interface mode, as in normal serial communication interface mode, transfer can be carried out using the DTC. In transmit operations, the TDRE flag is also set to 1 at the same time as the TEND flag in SSR is set, and a TXI interrupt is generated. If the TXI request is designated beforehand as a DTC activation source, the DTC will be activated by the TXI request, and transmit data will be transferred. The TDRE and TEND flags are automatically cleared to 0 when data is transferred by the DTC. In the event of an 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. Hence, 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 transferring 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 8, Data Transfer Controller (DTC).

In receive operations, 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. The RDRF flag is cleared to 0 automatically when data is transferred by the DTC. If an error occurs, an error flag is set but the RDRF flag is not. Consequently, the DTC is not activated, instead, an ERI interrupt request is sent to the CPU. Therefore, the error flag should be cleared.

## **14.9 Usage Notes**

### **14.9.1 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 20, Power-Down Modes.

### **14.9.2 Break Detection and Processing**

When framing error 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.

### **14.9.3 Mark State and Break Detection**

When TE is 0, the TxD pin is used as an I/O port whose direction (input or output) and level are determined by DR and 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 PCR and PDR 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 PCR to 1 and PDR 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.

### **14.9.4 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.

# Section 15 Hitachi Controller Area Network (HCAN)

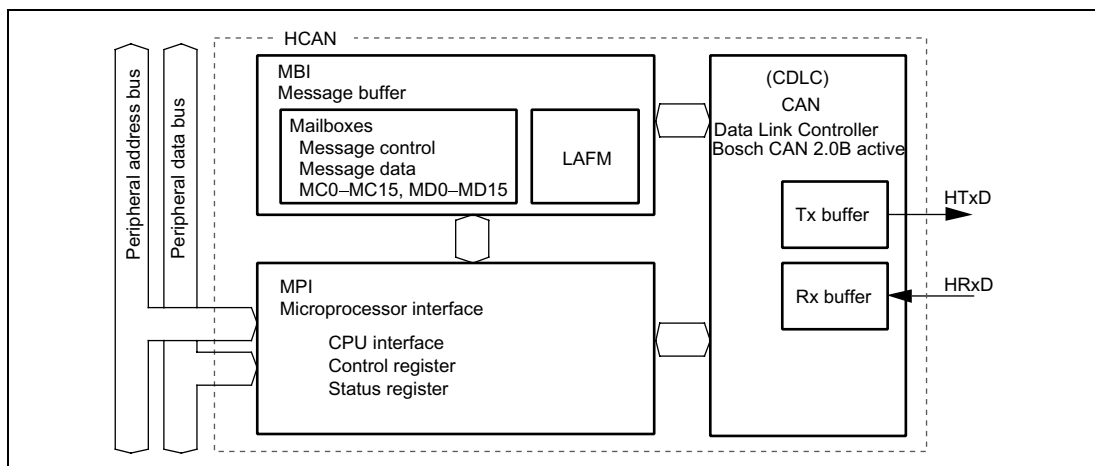
The HCAN is a module for controlling a controller area network (CAN) for realtime communication in vehicular and industrial equipment systems, etc. For details on CAN specification, refer to Bosch CAN Specification Version 2.0 1991, Robert Bosch GmbH.

The block diagram of the HCAN is shown in figure 15-1.

## 15.1 Features

- CAN version: Bosch 2.0B active compatible
  - Communication systems: NRZ (Non-Return to Zero) system (with bit-stuffing function)
  - Broadcast communication system
  - Transmission path: Bidirectional 2-wire serial communication
  - Communication speed: Max. 1 Mbps
  - Data length: 0 to 8 bytes
- Number of channels: 1
- Data buffers: 16 (one receive-only buffer and 15 buffers settable for transmission/reception)
- Data transmission: Two methods
  - Mailbox (buffer) number order (low-to-high)
  - Message priority (identifier) reverse-order (high-to-low)
- Data reception: Two methods
  - Message identifier match (transmit/receive-setting buffers)
  - Reception with message identifier masked (receive-only)
- CPU interrupts: 12
  - Error interrupt
  - Reset processing interrupt
  - Message reception interrupt
  - Message transmission interrupt
- HCAN operating modes
- Support for various modes
  - Hardware reset
  - Software reset
  - Normal status (error-active, error-passive)
  - Bus off status
  - HCAN configuration mode
  - HCAN sleep mode
  - HCAN halt mode

- Other features
  - DTC can be activated by message reception mailbox (HCAN mailbox 0 only)
- Module stop mode can be set



**Figure 15-1 HCAN Block Diagram**

- Message Buffer Interface (MBI)
 

The MBI, consisting of mailboxes and a local acceptance filter mask (LAFM), stores CAN transmit/receive messages (identifiers, data, etc.) Transmit messages are written by the CPU. For receive messages, the data received by the CDLC is stored automatically.
- Microprocessor Interface (MPI)
 

The MPI, consisting of a bus interface, control register, status register, etc., controls HCAN internal data, status, and so forth.
- CAN Data Link Controller (CDLC)
 

The CDLC transmits and receives of messages conforming to the Bosch CAN Ver. 2.0B active standard (data frames, remote frames, error frames, overload frames, inter-frame spacing), as well as CRC checking, bus arbitration, and other functions.

## 15.2 Input/Output Pins

Table 15-1 shows the HCAN's pins.

When using HCAN pins, settings must be made in the HCAN configuration mode (during initialization: MCR0 = 1 and GSR3 = 1).

**Table 15-1 HCAN Pins**

| Name                   | Abbreviation | Input/Output | Function                 |
|------------------------|--------------|--------------|--------------------------|
| HCAN transmit data pin | HTxD         | Output       | CAN bus transmission pin |
| HCAN receive data pin  | HRxD         | Input        | CAN bus reception pin    |

A bus driver is necessary for the interface between the pins and the CAN bus. A Philips PCA82C250 compatible model is recommended.

## 15.3 Register Descriptions

The HCAN has the following registers. For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register.

- Master control register (MCR)
- General status register (GSR)
- Bit configuration register (BCR)
- Mailbox configuration register (MBCR)
- Transmit wait register (TXPR)
- Transmit wait cancel register (TXCR)
- Transmit acknowledge register (TXACK)
- Abort acknowledge register (ABACK)
- Receive complete register (RXPR)
- Remote request register (RFPR)
- Interrupt register (IRR)
- Mailbox interrupt mask register (MBIMR)
- Interrupt mask register (IMR)
- Receive error counter (REC)
- Transmit error counter (TEC)
- Unread message status register (UMSR)
- Local acceptance filter mask H (LAFMH)
- Local acceptance filter mask L (LAFML)
- Message control (8-bit × 8 registers × 16 sets) (MC0 to MC15)

- Message data (8-bit × 8 registers × 16 sets) (MD0 to MD15)
- HCAN Monitor Register (HCANMON)

### 15.3.1 Master Control Register (MCR)

The master control register (MCR) is an 8-bit register that controls the HCAN.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                             |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | MCR7     | 0             | R/W | HCAN Sleep Mode Release<br>When this bit is set to 1, the HCAN automatically exits HCAN sleep mode on detection of CAN bus operation.                                                   |
| 6   | —        | 0             | R   | Reserved<br>This bit is always read as 0. Only 0 should be written to this bit.                                                                                                         |
| 5   | MCR5     | 0             | R/W | HCAN Sleep Mode<br>When this bit is set to 1, the HCAN transits to HCAN sleep mode. When this bit is cleared to 0, HCAN sleep mode is released.                                         |
| 4   | —        | 0             | R   | Reserved                                                                                                                                                                                |
| 3   | —        | 0             | R   | These bits are always read as 0. Only 0 should be written to these bits.                                                                                                                |
| 2   | MCR2     | 0             | R/W | Message Transmission Method<br>0: Transmission order determined by message identifier priority<br>1: Transmission order determined by mailbox (buffer) number priority (TXPR1 > TXPR15) |
| 1   | MCR1     | 0             | R/W | Halt Request<br>When this bit is set to 1, the HCAN transits to HCAN HALT mode. When this bit is cleared to 0, HCAN HALT mode is released.                                              |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0   | MCR0     | 1             | R/W | <p>Reset Request</p> <p>When this bit is set to 1, the HCAN transits to reset mode. For details, refer to section 15.4.1, Hardware and Software Reset.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• Hardware standby</li> <li>• Software standby</li> <li>• 1-write (software reset)</li> </ul> <p>[Clearing condition]</p> <p>When 0 is written to this bit while the GSR3 bit in GSR is 1</p> |

### 15.3.2 General Status Register (GSR)

The general status register (GSR) is an 8-bit that indicates the status of the CAN bus.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | 0             | R   | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 6   | —        | 0             | R   | These bits are always read as 0. Only 0 should be written to these bits.                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| 5   | —        | 0             | R   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 4   | —        | 0             | R   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| 3   | GSR3     | 1             | R   | <p>Reset Status Bit</p> <p>Indicates whether the HCAN module is in the normal operating state or the reset state. This bit cannot be modified.</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>• When entering configuration mode after the HCAN internal reset has finished</li> <li>• Sleep mode</li> </ul> <p>[Clearing condition]</p> <p>When entering normal operation mode after the MCR0 bit in MCR is cleared to 0 (Note that there is a delay between clearing of the MCR0 bit and the GSR3 bit.)</p> |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2   | GSR2     | 1             | R   | <p>Message Transmission Status Flag</p> <p>Flag that indicates whether the module is currently in the message transmission period. This bit cannot be modified.</p> <p>[Setting condition]</p> <p>Start of message transmission (SOF)</p> <p>[Clearing condition]</p> <p>Interval of three bits after EOF (End of Frame)</p>                               |
| 1   | GSR1     | 0             | R   | <p>Transmit/Receive Warning Flag</p> <p>This bit cannot be modified.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• When <math>TEC &lt; 96</math> and <math>REC &lt; 96</math></li> <li>• When <math>TEC \geq 256</math></li> </ul> <p>[Setting condition]</p> <p>When <math>TEC \geq 96</math> or <math>REC \geq 96</math></p> |
| 0   | GSR0     | 0             | R   | <p>Bus Off Flag</p> <p>This bit cannot be modified.</p> <p>[Setting condition]</p> <p>When <math>TEC \geq 256</math> (bus off state)</p> <p>[Clearing condition]</p> <p>Recovery from bus off state</p>                                                                                                                                                    |

### 15.3.3 Bit Configuration Register (BCR)

The bit configuration register (BCR) is a 16-bit register that is used to set HCAN bit timing parameters and the baud rate prescaler. For details on parameters, refer to section 15.4.2, Initialization after Hardware Reset.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                       |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | BCR7     | 0             | R/W | Re-Synchronization Jump Width (SJW)                                                                                                                                                                                               |
| 14  | BCR6     | 0             | R/W | Set the maximum bit synchronization width.<br>00: 1 time quantum<br>01: 2 time quanta<br>10: 3 time quanta<br>11: 4 time quanta                                                                                                   |
| 13  | BCR5     | 0             | R/W | Baud Rate Prescaler (BRP)                                                                                                                                                                                                         |
| 12  | BCR4     | 0             | R/W | Set the length of time quantum.                                                                                                                                                                                                   |
| 11  | BCR3     | 0             | R/W | 000000: $2 \times$ system clock                                                                                                                                                                                                   |
| 10  | BCR2     | 0             | R/W | 000001: $4 \times$ system clock                                                                                                                                                                                                   |
| 9   | BCR1     | 0             | R/W | 000010: $6 \times$ system clock                                                                                                                                                                                                   |
| 8   | BCR0     | 0             | R/W | :<br>111111: $128 \times$ system clock                                                                                                                                                                                            |
| 7   | BCR15    | 0             | R/W | Bit Sample Point (BSP)<br>Sets the point at which data is sampled.<br>0: Bit sampling at one point (end of time segment 1 (TSEG1))<br>1: Bit sampling at three points (end of TSEG1 and preceding and following one time quantum) |
| 6   | BCR14    | 0             | R/W | Time Segment 2 (TSEG2)                                                                                                                                                                                                            |
| 5   | BCR13    | 0             | R/W | Set the TSEG2 width within a range of 2 to 8 time quanta.                                                                                                                                                                         |
| 4   | BCR12    | 0             | R/W | 000: Setting prohibited<br>001: 2 time quanta<br>010: 3 time quanta<br>011: 4 time quanta<br>100: 5 time quanta<br>101: 6 time quanta<br>110: 7 time quanta<br>111: 8 time quanta                                                 |

| Bit | Bit Name | Initial Value | R/W | Description                             |
|-----|----------|---------------|-----|-----------------------------------------|
| 3   | BCR11    | 0             | R/W | Time Segment 1 (TSEG1)                  |
| 2   | BCR10    | 0             | R/W | Set the TSEG1 (PRSEG + PHSEG1) width to |
| 1   | BCR9     | 0             | R/W | between 4 and 16 time quanta.           |
| 0   | BCR8     | 0             | R/W | 0000: Setting prohibited                |
|     |          |               |     | 0001: Setting prohibited                |
|     |          |               |     | 0010: Setting prohibited                |
|     |          |               |     | 0011: 4 time quanta                     |
|     |          |               |     | 0100: 5 time quanta                     |
|     |          |               |     | 0101: 6 time quanta                     |
|     |          |               |     | 0110: 7 time quanta                     |
|     |          |               |     | 0111: 8 time quanta                     |
|     |          |               |     | 1000: 9 time quanta                     |
|     |          |               |     | 1001: 10 time quanta                    |
|     |          |               |     | 1010: 11 time quanta                    |
|     |          |               |     | 1011: 12 time quanta                    |
|     |          |               |     | 1100: 13 time quanta                    |
|     |          |               |     | 1101: 14 time quanta                    |
|     |          |               |     | 1110: 15 time quanta                    |
|     |          |               |     | 1111: 16 time quanta                    |

### 15.3.4 Mailbox Configuration Register (MBCR)

The mailbox configuration register (MBCR) is a 16-bit register that is used to set the transfer direction for each mailbox.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                             |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | MBCR7    | 0             | R/W | These bits set the transfer direction for the corresponding mailboxes from 1 to 15. MBCRn determines the transfer direction for mailbox n (n =1 to 15). |
| 14  | MBCR6    | 0             | R/W |                                                                                                                                                         |
| 13  | MBCR5    | 0             | R/W |                                                                                                                                                         |
| 12  | MBCR4    | 0             | R/W |                                                                                                                                                         |
| 11  | MBCR3    | 0             | R/W |                                                                                                                                                         |
| 10  | MBCR2    | 0             | R/W | 0: Corresponding mailbox is set for transmission<br>1: Corresponding mailbox is set for reception                                                       |
| 9   | MBCR1    | 0             | R/W |                                                                                                                                                         |
| 8   | —        | 1             | R   | Bit 8 is reserved. This bit is always read as 1 and the write value should always be 1.                                                                 |
| 7   | MBCR15   | 0             | R/W |                                                                                                                                                         |
| 6   | MBCR14   | 0             | R/W |                                                                                                                                                         |
| 5   | MBCR13   | 0             | R/W |                                                                                                                                                         |
| 4   | MBCR12   | 0             | R/W |                                                                                                                                                         |
| 3   | MBCR11   | 0             | R/W |                                                                                                                                                         |
| 2   | MBCR10   | 0             | R/W |                                                                                                                                                         |
| 1   | MBCR9    | 0             | R/W |                                                                                                                                                         |
| 0   | MBCR8    | 0             | R/W |                                                                                                                                                         |

### 15.3.5 Transmit Wait Register (TXPR)

The transmit wait register (TXPR) is a 16-bit register that is used to set a transmit wait after a transmit message is stored in a mailbox (buffer) (CAN bus arbitration wait).

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                        |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | TXPR7    | 0             | R/W | These bits set a transmit wait (CAN bus arbitration wait) for the corresponding mailboxes 1 to 15. When TXPRn (n = 1 to 15) is set to 1, the message in mailbox n becomes the transmit wait state. |
| 14  | TXPR6    | 0             | R/W |                                                                                                                                                                                                    |
| 13  | TXPR5    | 0             | R/W |                                                                                                                                                                                                    |
| 12  | TXPR4    | 0             | R/W |                                                                                                                                                                                                    |
| 11  | TXPR3    | 0             | R/W | [Clearing conditions]                                                                                                                                                                              |
| 10  | TXPR2    | 0             | R/W | • Completion of message transmission                                                                                                                                                               |
| 9   | TXPR1    | 0             | R/W | • Completion of transmission cancellation                                                                                                                                                          |
| 8   | —        | 0             | R   | Bit 8 is reserved. This bit is always read as 1 and the write value should always be 1.                                                                                                            |
| 7   | TXPR15   | 0             | R/W |                                                                                                                                                                                                    |
| 6   | TXPR14   | 0             | R/W |                                                                                                                                                                                                    |
| 5   | TXPR13   | 0             | R/W |                                                                                                                                                                                                    |
| 4   | TXPR12   | 0             | R/W |                                                                                                                                                                                                    |
| 3   | TXPR11   | 0             | R/W |                                                                                                                                                                                                    |
| 2   | TXPR10   | 0             | R/W |                                                                                                                                                                                                    |
| 1   | TXPR9    | 0             | R/W |                                                                                                                                                                                                    |
| 0   | TXPR8    | 0             | R/W |                                                                                                                                                                                                    |

### 15.3.6 Transmit Wait Cancel Register (TXCR)

The transmit wait cancel register (TXCR) is a 16-bit register that controls the cancellation of transmit wait messages in mailboxes (buffers).

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                   |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | TXCR7    | 0             | R/W | These bits cancel the transmit wait message in the corresponding mailboxes 1 to 15. When TXCRn (n = 1 to 15) is set to 1, the transmit wait message in mailbox n is canceled. |
| 14  | TXCR6    | 0             | R/W |                                                                                                                                                                               |
| 13  | TXCR5    | 0             | R/W |                                                                                                                                                                               |
| 12  | TXCR4    | 0             | R/W |                                                                                                                                                                               |
| 11  | TXCR3    | 0             | R/W | [Clearing condition]                                                                                                                                                          |
| 10  | TXCR2    | 0             | R/W | Completion of TXPR clearing when transmit message is canceled normally                                                                                                        |
| 9   | TXCR1    | 0             | R/W | Bit 8 is reserved. This bit is always read as 0 and the write value should always be 0.                                                                                       |
| 8   | —        | 0             | R   |                                                                                                                                                                               |
| 7   | TXCR15   | 0             | R/W |                                                                                                                                                                               |
| 6   | TXCR14   | 0             | R/W |                                                                                                                                                                               |
| 5   | TXCR13   | 0             | R/W |                                                                                                                                                                               |
| 4   | TXCR12   | 0             | R/W |                                                                                                                                                                               |
| 3   | TXCR11   | 0             | R/W |                                                                                                                                                                               |
| 2   | TXCR10   | 0             | R/W |                                                                                                                                                                               |
| 1   | TXCR9    | 0             | R/W |                                                                                                                                                                               |
| 0   | TXCR8    | 0             | R/W |                                                                                                                                                                               |

### 15.3.7 Transmit Acknowledge Register (TXACK)

The transmit acknowledge register (TXACK) is a 16-bit register containing status flags that indicate the normal transmission of mailbox (buffer) transmit messages.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                        |
|-----|----------|---------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | TXACK7   | 0             | R/(W)* | These bits are status flags that indicate error-free transmission of the transmit message in the corresponding mailboxes 1 to 15. When the message in mailbox n (n = 1 to 15) has been transmitted error-free, TXACKn is set to 1. |
| 14  | TXACK6   | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 13  | TXACK5   | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 12  | TXACK4   | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 11  | TXACK3   | 0             | R/(W)* | [Setting condition]                                                                                                                                                                                                                |
| 10  | TXACK2   | 0             | R/(W)* | Completion of message transmission for corresponding mailbox                                                                                                                                                                       |
| 9   | TXACK1   | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 8   | —        | 0             | R      | [Clearing condition]                                                                                                                                                                                                               |
| 7   | TXACK15  | 0             | R/(W)* | Writing 1                                                                                                                                                                                                                          |
| 6   | TXACK14  | 0             | R/(W)* | Bit 8 is reserved. This bit is always read as 0 and the write value should always be 0.                                                                                                                                            |
| 5   | TXACK13  | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 4   | TXACK12  | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 3   | TXACK11  | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 2   | TXACK10  | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 1   | TXACK9   | 0             | R/(W)* |                                                                                                                                                                                                                                    |
| 0   | TXACK8   | 0             | R/(W)* |                                                                                                                                                                                                                                    |

Note: \* Only 1 can be written to clear the flag.

### 15.3.8 Abort Acknowledge Register (ABACK)

The abort acknowledge register (ABACK) is a 16-bit register containing status flags that indicate the normal cancellation (aborting) of mailbox (buffer) transmit messages.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                |
|-----|----------|---------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | ABACK7   | 0             | R/(W)* | These bits are status flags that indicate error-free cancellation (abortion) of the transmit message in the corresponding mailboxes 1 to 15. When the message in mailbox n (n = 1 to 15) has been canceled error-free, ABACKn is set to 1. |
| 14  | ABACK6   | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 13  | ABACK5   | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 12  | ABACK4   | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 11  | ABACK3   | 0             | R/(W)* | [Setting condition]                                                                                                                                                                                                                        |
| 10  | ABACK2   | 0             | R/(W)* | Completion of transmit message cancellation for corresponding mailbox                                                                                                                                                                      |
| 9   | ABACK1   | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 8   | —        | 0             | R      | [Clearing condition]                                                                                                                                                                                                                       |
| 7   | ABACK15  | 0             | R/(W)* | Writing 1                                                                                                                                                                                                                                  |
| 6   | ABACK14  | 0             | R/(W)* | Bit 8 is reserved. This bit is always read as 0. The write value should always be 0.                                                                                                                                                       |
| 5   | ABACK13  | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 4   | ABACK12  | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 3   | ABACK11  | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 2   | ABACK10  | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 1   | ABACK9   | 0             | R/(W)* |                                                                                                                                                                                                                                            |
| 0   | ABACK8   | 0             | R/(W)* |                                                                                                                                                                                                                                            |

Note: \* Only 1 can be written to clear the flag.

### 15.3.9 Receive Complete Register (RXPR)

The receive complete register (RXPR) is a 16-bit register containing status flags that indicate the normal reception of messages in mailboxes (buffers). For reception of a remote frame, when a bit in this register is set to 1, the corresponding remote request register (RFPR) bit is also set to 1 simultaneously.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                  |
|-----|----------|---------------|--------|----------------------------------------------------------------------------------------------|
| 15  | RXPR7    | 0             | R/(W)* | When the message in mailbox n (n = 0 to 15) has been received error-free, RXPRn is set to 1. |
| 14  | RXPR6    | 0             | R/(W)* |                                                                                              |
| 13  | RXPR5    | 0             | R/(W)* | [Setting condition]                                                                          |
| 12  | RXPR4    | 0             | R/(W)* | Completion of message (data frame or remote frame) reception in corresponding mailbox        |
| 11  | RXPR3    | 0             | R/(W)* |                                                                                              |
| 10  | RXPR2    | 0             | R/(W)* | [Clearing condition]                                                                         |
| 9   | RXPR1    | 0             | R/(W)* | Writing 1                                                                                    |
| 8   | RXPR0    | 0             | R/(W)* |                                                                                              |
| 7   | RXPR15   | 0             | R/(W)* |                                                                                              |
| 6   | RXPR14   | 0             | R/(W)* |                                                                                              |
| 5   | RXPR13   | 0             | R/(W)* |                                                                                              |
| 4   | RXPR12   | 0             | R/(W)* |                                                                                              |
| 3   | RXPR11   | 0             | R/(W)* |                                                                                              |
| 2   | RXPR10   | 0             | R/(W)* |                                                                                              |
| 1   | RXPR9    | 0             | R/(W)* |                                                                                              |
| 0   | RXPR8    | 0             | R/(W)* |                                                                                              |

Note: \* Only 1 can be written to clear the flag.

### 15.3.10 Remote Request Register (RFPR)

The remote request register (RFPR) is a 16-bit register containing status flags that indicate normal reception of remote frames in mailboxes (buffers). When a bit in this register is set to 1, the corresponding receive complete register (RXPR) bit is also set to 1 simultaneously.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                             |
|-----|----------|---------------|--------|---------------------------------------------------------------------------------------------------------|
| 15  | RFPR7    | 0             | R/(W)* | When mailbox n (n = 0 to 15) has received the remote frame error-free, RFPRn (n = 0 to 15) is set to 1. |
| 14  | RFPR6    | 0             | R/(W)* |                                                                                                         |
| 13  | RFPR5    | 0             | R/(W)* | [Setting condition]                                                                                     |
| 12  | RFPR4    | 0             | R/(W)* |                                                                                                         |
| 11  | RFPR3    | 0             | R/(W)* | Completion of remote frame reception in corresponding mailbox                                           |
| 10  | RFPR2    | 0             | R/(W)* |                                                                                                         |
| 9   | RFPR1    | 0             | R/(W)* | [Clearing condition]                                                                                    |
| 8   | RFPR0    | 0             | R/(W)* |                                                                                                         |
| 7   | RFPR15   | 0             | R/(W)* | Writing 1                                                                                               |
| 6   | RFPR14   | 0             | R/(W)* |                                                                                                         |
| 5   | RFPR13   | 0             | R/(W)* |                                                                                                         |
| 4   | RFPR12   | 0             | R/(W)* |                                                                                                         |
| 3   | RFPR11   | 0             | R/(W)* |                                                                                                         |
| 2   | RFPR10   | 0             | R/(W)* |                                                                                                         |
| 1   | RFPR9    | 0             | R/(W)* |                                                                                                         |
| 0   | RFPR8    | 0             | R/(W)* |                                                                                                         |

Note: \* Only 1 can be written to clear the flag.

### 15.3.11 Interrupt Register (IRR)

The interrupt register (IRR) is a 16-bit interrupt status flag register.

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                       |
|-----|----------|---------------|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | IRR7     | 0             | R/(W)* | Overload Frame<br>[Setting condition]<br>When an overload frame is transmitted<br><br>[Clearing condition]<br>Writing 1                                                                                                           |
| 14  | IRR6     | 0             | R/(W)* | Bus Off Interrupt Flag<br>Status flag indicating the bus off state caused by the transmit error counter.<br>[Setting condition]<br>When $TEC \geq 256$<br><br>[Clearing condition]<br>Writing 1                                   |
| 13  | IRR5     | 0             | R/(W)* | Error Passive Interrupt Flag<br>Status flag indicating the error passive state caused by the transmit/receive error counter.<br>[Setting condition]<br>When $TEC \geq 128$ or $REC \geq 128$<br>[Clearing condition]<br>Writing 1 |

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                                                                                                                                |
|-----|----------|---------------|--------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 12  | IRR4     | 0             | R/(W)* | <p>Receive Overload Warning Interrupt Flag</p> <p>Status flag indicating the error warning state caused by the receive error counter.</p> <p>[Setting condition]</p> <p>When <math>REC \geq 96</math></p> <p>[Clearing condition]</p> <p>Writing 1</p>                                                                                                     |
| 11  | IRR3     | 0             | R/(W)* | <p>Transmit Overload Warning Interrupt Flag</p> <p>Status flag indicating the error warning state caused by the transmit error counter.</p> <p>[Setting condition]</p> <p>When <math>TEC \geq 96</math></p> <p>[Clearing condition]</p> <p>Writing 1</p>                                                                                                   |
| 10  | IRR2     | 0             | R      | <p>Remote Frame Request Interrupt Flag</p> <p>Status flag indicating that a remote frame has been received in a mailbox (buffer).</p> <p>[Setting condition]</p> <p>When remote frame reception is completed, when corresponding MBIMR = 0</p> <p>[Clearing condition]</p> <p>Clearing of all bits in RFPR (remote request register)</p>                   |
| 9   | IRR1     | 0             | R      | <p>Receive Message Interrupt Flag</p> <p>Status flag indicating that a mailbox (buffer) receive message has been received normally.</p> <p>[Setting condition]</p> <p>When data frame or remote frame reception is completed, when corresponding MBIMR = 0</p> <p>[Clearing condition]</p> <p>Clearing of all bits in RXPR (receive complete register)</p> |

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|-----|----------|---------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 8   | IRR0     | 1             | R/(W)* | <p>Reset Interrupt Flag</p> <p>Status flag indicating that the HCAN module has been reset. This bit cannot be masked by the interrupt mask register (IMR). If this bit is not cleared to 0 after entering power-on reset or returning from software standby mode, interrupt processing will start immediately when the interrupt controller enables interrupts.</p> <p>[Setting condition]</p> <p>When the reset operation has finished after entering power-on reset or software standby mode</p> <p>[Clearing condition]</p> <p>Writing 1</p> |
| 7   | —        | 0             | —      | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 6   | —        | 0             | —      | These bits are always read as 0. Only 0 should be written to these bits.                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 5   | —        | 0             | —      |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| 4   | IRR12    | 0             | R/(W)* | <p>Bus Operation Interrupt Flag</p> <p>Status flag indicating detection of a dominant bit due to bus operation when the HCAN module is in HCAN sleep mode.</p> <p>[Setting condition]</p> <p>Bus operation (dominant bit) detection in HCAN sleep mode</p> <p>[Clearing condition]</p> <p>Writing 1</p>                                                                                                                                                                                                                                         |
| 3   | —        | 0             | —      | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |
| 2   | —        | 0             | —      | These bits are always read as 0. Only 0 should be written to these bits.                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

| Bit | Bit Name | Initial Value | R/W    | Description                                                                                                                                                                                                                                                                                                                   |
|-----|----------|---------------|--------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1   | IRR9     | 0             | R      | <p>Unread Interrupt Flag</p> <p>Status flag indicating that a receive message has been overwritten before being read.</p> <p>[Setting condition]</p> <p>When UMSR (unread message status register) is set</p> <p>[Clearing condition]</p> <p>Clearing of all bits in UMSR (unread message status register)</p>                |
| 0   | IRR8     | 0             | R/(W)* | <p>Mailbox Empty Interrupt Flag</p> <p>Status flag indicating that the next transmit message can be stored in the mailbox.</p> <p>[Setting condition]</p> <p>When TXPR (transmit wait register) is cleared by completion of transmission or completion of transmission abort</p> <p>[Clearing condition]</p> <p>Writing 1</p> |

Note: \* Only 1 can be written to clear the flag.

### 15.3.12 Mailbox Interrupt Mask Register (MBIMR)

The mailbox interrupt mask register (MBIMR) is a 16-bit register that controls the enabling or disabling of individual mailbox (buffer) interrupt requests.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                      |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | MBIMR7   | 1             | R/W | Mailbox Interrupt Mask (MBIMRx)                                                                                                                                                                  |
| 14  | MBIMR6   | 1             | R/W | When MBIMRn (n = 1 to 15) is cleared to 0, the interrupt request in mailbox n is enabled. When set to 1, the interrupt request is masked.                                                        |
| 13  | MBIMR5   | 1             | R/W |                                                                                                                                                                                                  |
| 12  | MBIMR4   | 1             | R/W | The interrupt source in a transmit mailbox is TXPR clearing caused by transmission end or transmission cancellation. The interrupt source in a receive mailbox is RXPR setting on reception end. |
| 11  | MBIMR3   | 1             | R/W |                                                                                                                                                                                                  |
| 10  | MBIMR2   | 1             | R/W |                                                                                                                                                                                                  |
| 9   | MBIMR1   | 1             | R/W |                                                                                                                                                                                                  |
| 8   | MBIMR0   | 1             | R/W |                                                                                                                                                                                                  |
| 7   | MBIMR15  | 1             | R/W |                                                                                                                                                                                                  |
| 6   | MBIMR14  | 1             | R/W |                                                                                                                                                                                                  |
| 5   | MBIMR13  | 1             | R/W |                                                                                                                                                                                                  |
| 4   | MBIMR12  | 1             | R/W |                                                                                                                                                                                                  |
| 3   | MBIMR11  | 1             | R/W |                                                                                                                                                                                                  |
| 2   | MBIMR10  | 1             | R/W |                                                                                                                                                                                                  |
| 1   | MBIMR9   | 1             | R/W |                                                                                                                                                                                                  |
| 0   | MBIMR8   | 1             | R/W |                                                                                                                                                                                                  |

### 15.3.13 Interrupt Mask Register (IMR)

The interrupt mask register (IMR) is a 16-bit register containing flags that enable or disable requests by individual interrupt sources. The interrupt flag cannot be masked.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                  |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 15  | IMR7     | 1             | R/W | Overload Frame/Bus Off Recovery Interrupt Mask<br>When this bit is cleared to 0, OVR0 (interrupt request by IRR7) is enabled. When set to 1, OVR0 is masked. |
| 14  | IMR6     | 1             | R/W | Bus Off Interrupt Mask<br>When this bit is cleared to 0, ERS0 (interrupt request by IRR6) is enabled. When set to 1, ERS0 is masked.                         |
| 13  | IMR5     | 1             | R/W | Error Passive Interrupt Mask<br>When this bit is cleared to 0, ERS0 (interrupt request by IRR5) is enabled. When set to 1, ERS0 is masked.                   |
| 12  | IMR4     | 1             | R/W | Receive Overload Warning Interrupt Mask<br>When this bit is cleared to 0, OVR0 (interrupt request by IRR4) is enabled. When set to 1, OVR0 is masked.        |
| 11  | IMR3     | 1             | R/W | Transmit Overload Warning Interrupt Mask<br>When this bit is cleared to 0, OVR0 (interrupt request by IRR3) is enabled. When set to 1, OVR0 is masked.       |
| 10  | IMR2     | 1             | R/W | Remote Frame Request Interrupt Mask<br>When this bit is cleared to 0, OVR0 (interrupt request by IRR2) is enabled. When set to 1, OVR0 is masked.            |
| 9   | IMR1     | 1             | R/W | Receive Message Interrupt Mask<br>When this bit is cleared to 0, RM1 (interrupt request by IRR1) is enabled. When set to 1, RMI is masked.                   |
| 8   | —        | 0             | R   | Reserved<br>This bit is always read as 0. Only 0 should be written to this bit.                                                                              |
| 7   | —        | 1             | R   | Reserved                                                                                                                                                     |
| 6   | —        | 1             | R   | These bits are always read as 1. Only 1 should be written to these bits.                                                                                     |
| 5   | —        | 1             | R   |                                                                                                                                                              |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                 |
|-----|----------|---------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------|
| 4   | IMR12    | 1             | R/W | Bus Operation Interrupt Mask<br>When this bit is cleared to 0, OVR0 (interrupt request by IRR12) is enabled. When set to 1, OVR0 is masked. |
| 3   | —        | 1             | R   | Reserved                                                                                                                                    |
| 2   | —        | 1             | R   | These bits are always read as 1. Only 1 should be written to these bits.                                                                    |
| 1   | IMR9     | 1             | R/W | Unread Interrupt Mask<br>When this bit is cleared to 0, OVR0 (interrupt request by IRR9) is enabled. When set to 1, OVR0 is masked.         |
| 0   | IMR8     | 1             | R/W | Mailbox Empty Interrupt Mask<br>When this bit is cleared to 0, SLE0 (interrupt request by IRR8) is enabled. When set to 1, SLE0 is masked.  |

#### 15.3.14 Receive Error Counter (REC)

The receive error counter (REC) is an 8-bit read-only register that functions as a counter indicating the number of receive message errors on the CAN bus. The count value is stipulated in the CAN protocol.

#### 15.3.15 Transmit Error Counter (TEC)

The transmit error counter (TEC) is an 8-bit read-only register that functions as a counter indicating the number of transmit message errors on the CAN bus. The count value is stipulated in the CAN protocol.

### 15.3.16 Unread Message Status Register (UMSR)

The unread message status register (UMSR) is a 16-bit register containing status flags that indicate, for individual mailboxes (buffers), that a received message has been overwritten by a new receive message before being read. When overwritten by a new message, data in the unread receive message is lost.

| Bit | Bit Name | Initial Value | R/W    | Description                                           |
|-----|----------|---------------|--------|-------------------------------------------------------|
| 15  | UMSR7    | 0             | R/W    | [Setting condition]                                   |
| 14  | UMSR6    | 0             | R/W    | When a new message is received before RXPR is cleared |
| 13  | UMSR5    | 0             | R/W    |                                                       |
| 12  | UMSR4    | 0             | R/W    | [Clearing condition]                                  |
| 11  | UMSR3    | 0             | R/W    | Writing 1                                             |
| 10  | UMSR2    | 0             | R/W    |                                                       |
| 9   | UMSR1    | 0             | R/W    |                                                       |
| 8   | UMSR0    | 0             | R/W    |                                                       |
| 7   | UMSR15   | 0             | R/W    |                                                       |
| 6   | UMSR14   | 0             | R/W    |                                                       |
| 5   | UMSR13   | 0             | R/W    |                                                       |
| 4   | UMSR12   | 0             | R/W    |                                                       |
| 3   | UMSR11   | 0             | R/W    |                                                       |
| 2   | UMSR10   | 0             | R/(W)* |                                                       |
| 1   | UMSR9    | 0             | R/(W)* |                                                       |
| 0   | UMSR8    | 0             | R/(W)* |                                                       |

Note: \* Only 1 can be written to clear the flag.

### 15.3.17 Local Acceptance Filter Masks (LAFML, LAFMH)

The local acceptance filter masks (LAFML and LAFMH) are 16-bit registers that individually set the identifier bits of the message to be stored in mailbox 0 as Don't Care. For details, refer to section 15.4.4, Message Reception. The relationship between the identifier bits and mask bits are shown in the following.

#### LAFML

| Bit | Bit Name | Initial Value | R/W | Description                                                                         |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------|
| 15  | LAFML7   | 0             | R/W | When this bit is set to 1, ID-7 of the receive message identifier is not compared.  |
| 14  | LAFML6   | 0             | R/W | When this bit is set to 1, ID-6 of the receive message identifier is not compared.  |
| 13  | LAFML5   | 0             | R/W | When this bit is set to 1, ID-5 of the receive message identifier is not compared.  |
| 12  | LAFML4   | 0             | R/W | When this bit is set to 1, ID-4 of the receive message identifier is not compared.  |
| 11  | LAFML3   | 0             | R/W | When this bit is set to 1, ID-3 of the receive message identifier is not compared.  |
| 10  | LAFML2   | 0             | R/W | When this bit is set to 1, ID-2 of the receive message identifier is not compared.  |
| 9   | LAFML1   | 0             | R/W | When this bit is set to 1, ID-1 of the receive message identifier is not compared.  |
| 8   | LAFML0   | 0             | R/W | When this bit is set to 1, ID-0 of the receive message identifier is not compared.  |
| 7   | LAFML15  | 0             | R/W | When this bit is set to 1, ID-15 of the receive message identifier is not compared. |
| 6   | LAFML14  | 0             | R/W | When this bit is set to 1, ID-14 of the receive message identifier is not compared. |
| 5   | LAFML13  | 0             | R/W | When this bit is set to 1, ID-13 of the receive message identifier is not compared. |
| 4   | LAFML12  | 0             | R/W | When this bit is set to 1, ID-12 of the receive message identifier is not compared. |
| 3   | LAFML11  | 0             | R/W | When this bit is set to 1, ID-11 of the receive message identifier is not compared. |
| 2   | LAFML10  | 0             | R/W | When this bit is set to 1, ID-10 of the receive message identifier is not compared. |
| 1   | LAFML9   | 0             | R/W | When this bit is set to 1, ID-9 of the receive message identifier is not compared.  |
| 0   | LAFML8   | 0             | R/W | When this bit is set to 1, ID-8 of the receive message identifier is not compared.  |

## LAFMH

| Bit | Bit Name | Initial Value | R/W | Description                                                                         |
|-----|----------|---------------|-----|-------------------------------------------------------------------------------------|
| 15  | LAFMH7   | 0             | R/W | When this bit is set to 1, ID-20 of the receive message identifier is not compared. |
| 14  | LAFMH6   | 0             | R/W | When this bit is set to 1, ID-19 of the receive message identifier is not compared. |
| 13  | LAFMH5   | 0             | R/W | When this bit is set to 1, ID-18 of the receive message identifier is not compared. |
| 12  | —        | 0             | R   | Reserved                                                                            |
| 11  | —        | 0             | R   | These bits are always read as 0. Only 0 should be written to these bits.            |
| 10  | —        | 0             | R   |                                                                                     |
| 9   | LAFMH1   | 0             | R/W | When this bit is set to 1, ID-17 of the receive message identifier is not compared. |
| 8   | LAFMH0   | 0             | R/W | When this bit is set to 1, ID-16 of the receive message identifier is not compared. |
| 7   | LAFMH15  | 0             | R/W | When this bit is set to 1, ID-28 of the receive message identifier is not compared. |
| 6   | LAFMH14  | 0             | R/W | When this bit is set to 1, ID-27 of the receive message identifier is not compared. |
| 5   | LAFMH13  | 0             | R/W | When this bit is set to 1, ID-26 of the receive message identifier is not compared. |
| 4   | LAFMH12  | 0             | R/W | When this bit is set to 1, ID-25 of the receive message identifier is not compared. |
| 3   | LAFMH11  | 0             | R/W | When this bit is set to 1, ID-24 of the receive message identifier is not compared. |
| 2   | LAFMH10  | 0             | R/W | When this bit is set to 1, ID-23 of the receive message identifier is not compared. |
| 1   | LAFMH9   | 0             | R/W | When this bit is set to 1, ID-22 of the receive message identifier is not compared. |
| 0   | LAFMH8   | 0             | R/W | When this bit is set to 1, ID-21 of the receive message identifier is not compared. |

15.3.18 Message Control (MC0 to MC15)

The message control register sets consist of eight 8-bit registers for one mailbox. The HCAN has 16 sets of these registers. Because message control registers are in RAM, their initial values after power-on are undefined. Be sure to initialize them by writing 0 or 1. Figure 15-2 shows the register names for each mailbox.

|             |         |         |         |         |         |         |         |         |
|-------------|---------|---------|---------|---------|---------|---------|---------|---------|
| Mail box 0  | MC0[1]  | MC0[2]  | MC0[3]  | MC0[4]  | MC0[5]  | MC0[6]  | MC0[7]  | MC0[8]  |
| Mail box 1  | MC1[1]  | MC1[2]  | MC1[3]  | MC1[4]  | MC1[5]  | MC1[6]  | MC1[7]  | MC1[8]  |
| Mail box 2  | MC2[1]  | MC2[2]  | MC2[3]  | MC2[4]  | MC2[5]  | MC2[6]  | MC2[7]  | MC2[8]  |
| Mail box 3  | MC3[1]  | MC3[2]  | MC3[3]  | MC3[4]  | MC3[5]  | MC3[6]  | MC3[7]  | MC3[8]  |
|             |         |         |         |         |         |         |         |         |
| Mail box 15 | MC15[1] | MC15[2] | MC15[3] | MC15[4] | MC15[5] | MC15[6] | MC15[7] | MC15[8] |

Figure 15-2 Message Control Register Configuration

The setting of message control registers are shown in the following. Figures 15.3 and 15.4 show the correspondence between the identifiers and register bit names.

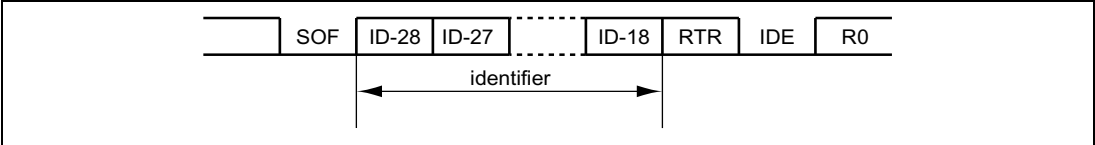


Figure 15-3 Standard Format

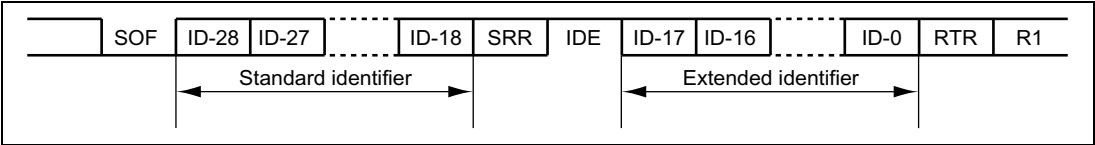


Figure 15-4 Extended Format

| Register Name | Bit    | Bit Name       | R/W | Description                                                                                                                                                                                                                                                                                                               |
|---------------|--------|----------------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MCx[1]        | 7 to 4 | —              | R/W | The initial value of these bits is undefined; they must be initialized (by writing 0 or 1).                                                                                                                                                                                                                               |
|               | 3 to 0 | DLC3 to DLC0   | R/W | Data Length Code<br>Set the data length of a data frame or the data length requested in a remote frame within the range of 0 to 8 bits.<br>0000: 0 byte<br>0001: 1 byte<br>0010: 2 bytes<br>0011: 3 bytes<br>0100: 4 bytes<br>0101: 5 bytes<br>0110: 6 bytes<br>0111: 7 bytes<br>1000: 8 bytes<br>:<br>:<br>1111: 8 bytes |
| MCx[2]        | 7 to 0 | —              | R/W | The initial value of these bits is undefined; they must be initialized (by writing 0 or 1).                                                                                                                                                                                                                               |
| MCx[3]        | 7 to 0 | —              | R/W |                                                                                                                                                                                                                                                                                                                           |
| MCx[4]        | 7 to 0 | —              | R/W |                                                                                                                                                                                                                                                                                                                           |
| MCx[5]        | 7 to 5 | ID-20 to ID-18 | R/W | Sets ID-20 to ID-18 in the identifier.                                                                                                                                                                                                                                                                                    |
|               | 4      | RTR            | R/W | Remote Transmission Request<br>Used to distinguish between data frames and remote frames.<br>0: Data frame<br>1: Remote frame                                                                                                                                                                                             |
|               | 3      | IDE            | R/W | Identifier Extension<br>Used to distinguish between the standard format and extended format of data frames and remote frames.<br>0: Standard format<br>1: Extended format                                                                                                                                                 |
|               | 2      | —              | R/W | The initial value of this bit is undefined. It must be initialized by writing 0 or 1.                                                                                                                                                                                                                                     |
|               | 1 to 0 | ID-17 to ID-16 | R/W | Sets ID-17 and ID-16 in the identifier.                                                                                                                                                                                                                                                                                   |
| MCx[6]        | 7 to 0 | ID-28 to ID-21 | R/W | Sets ID-28 to ID-21 in the identifier.                                                                                                                                                                                                                                                                                    |
| MCx[7]        | 7 to 0 | ID-7 to ID-0   | R/W | Sets ID-7 to ID-0 in the identifier.                                                                                                                                                                                                                                                                                      |
| MCx[8]        | 7 to 0 | ID-15 to ID-8  | R/W | Sets ID-15 to ID-8 in the identifier.                                                                                                                                                                                                                                                                                     |

Note: x: Mailbox number

### 15.3.19 Message Data (MD0 to MD15)

The message data register sets consist of eight 8-bit registers for one mailbox. The HCAN has 16 sets of these registers. Because message data registers are in RAM, their initial values after power-on are undefined. Be sure to initialize them by writing 0 or 1. Figure 15-5 shows the register names for each mailbox.

|             |         |         |         |         |         |         |         |         |
|-------------|---------|---------|---------|---------|---------|---------|---------|---------|
| Mail box 0  | MD0[1]  | MD0[2]  | MD0[3]  | MD0[4]  | MD0[5]  | MD0[6]  | MD0[7]  | MD0[8]  |
| Mail box 1  | MD1[1]  | MD1[2]  | MD1[3]  | MD1[4]  | MD1[5]  | MD1[6]  | MD1[7]  | MD1[8]  |
| Mail box 2  | MD2[1]  | MD2[2]  | MD2[3]  | MD2[4]  | MD2[5]  | MD2[6]  | MD2[7]  | MD2[8]  |
| Mail box 3  | MD3[1]  | MD3[2]  | MD3[3]  | MD3[4]  | MD3[5]  | MD3[6]  | MD3[7]  | MD3[8]  |
|             |         |         |         |         |         |         |         |         |
| Mail box 15 | MD15[1] | MD15[2] | MD15[3] | MD15[4] | MD15[5] | MD15[6] | MD15[7] | MD15[8] |

**Figure 15-5 Message Data Configuration**

### 15.3.20 HCAN Monitor Register (HCANMON)

The HCAN monitor register (HCANMON) is an 8-bit read-only register that reflects the states of the HCAN pins. This register cannot be modified.

| Bit | Bit Name | Initial Value | R/W | Description                                                                            |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------|
| 7   | —        | Undefined     | —   |                                                                                        |
| 6   | —        | Undefined     | —   |                                                                                        |
| 5   | —        | Undefined     | —   |                                                                                        |
| 4   | —        | Undefined     | —   |                                                                                        |
| 3   | —        | Undefined     | —   |                                                                                        |
| 2   | —        | Undefined     | —   |                                                                                        |
| 1   | TxD      | Undefined     | R   | Transmission pin<br>The state of the HTxD pin is read.<br>This bit cannot be modified. |
| 0   | RxD      | Undefined     | R   | Reception pin<br>The state of the HRxD pin is read.<br>This bit cannot be modified.    |

## 15.4 Operation

### 15.4.1 Hardware and Software Resets

The HCAN can be reset by a hardware reset or software reset.

- **Hardware Reset**

At power-on reset, or in hardware or software standby mode, the HCAN is initialized by automatically setting the MCR reset request bit (MCR0) in MCR and the reset state bit (GSR3) in GSR. At the same time, all internal registers, except for message control and message data registers, are initialized by a hardware reset.

- **Software Reset**

The HCAN can be reset by setting the MCR reset request bit (MCR0) in MCR via software. In a software reset, the error counters (TEC and REC) are initialized, however other registers are not. If the MCR0 bit is set while the CAN controller is performing a communication operation (transmission or reception), the initialization state is not entered until message transfer has been completed. The reset status bit (GSR3) in GSR is set on completion of initialization.

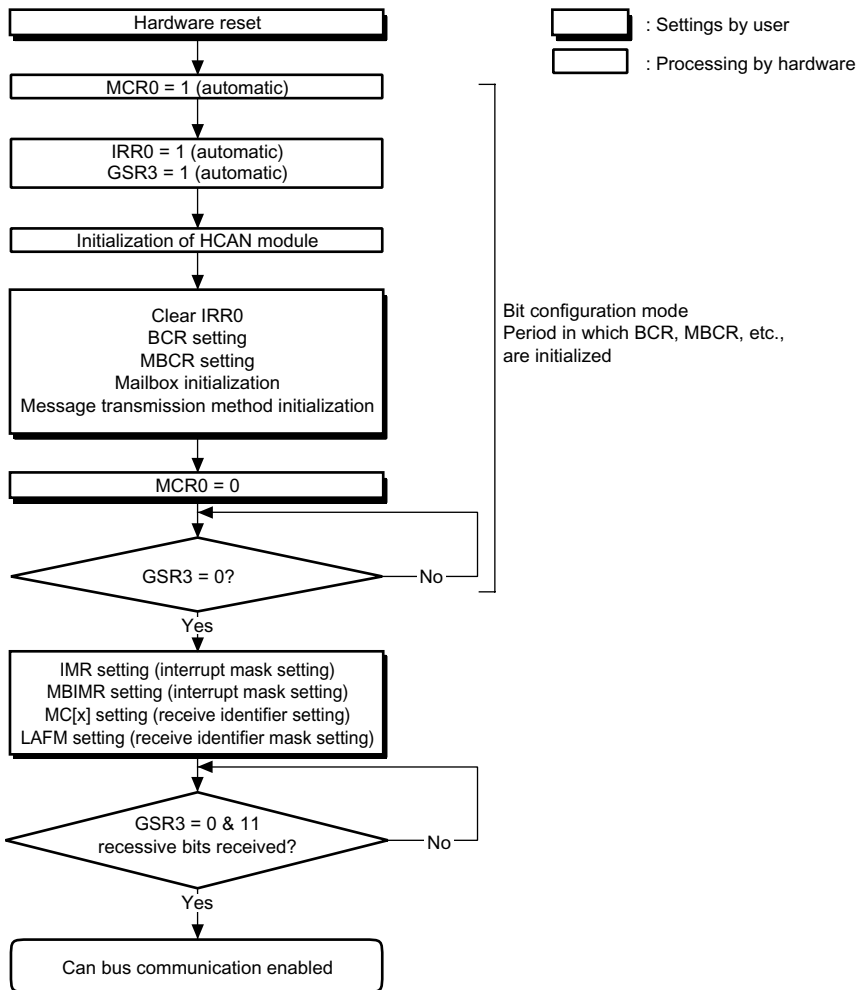
### 15.4.2 Initialization after Hardware Reset

After a hardware reset, the following initialization processing should be carried out:

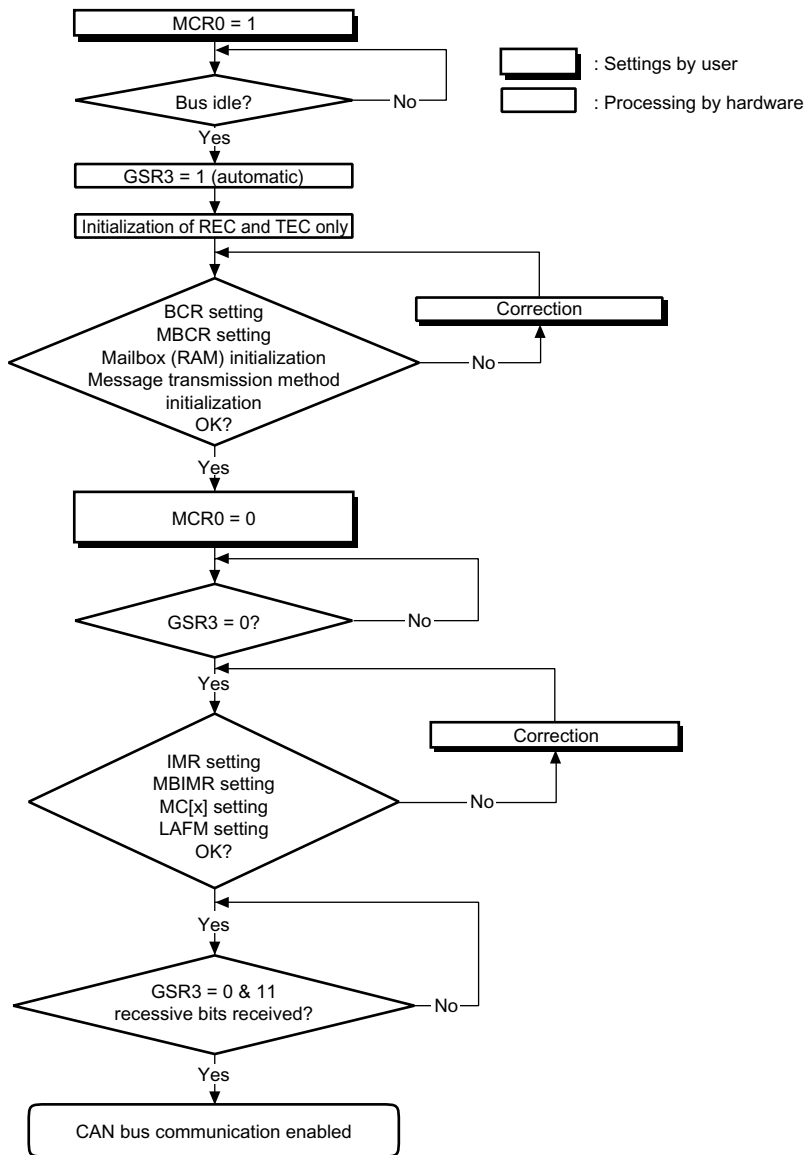
1. Clearing of IRR0 bit in the interrupt register (IRR)
2. Bit rate setting
3. Mailbox transmit/receive settings
4. Mailbox (RAM) initialization
5. Message transmission method setting

These initial settings must be made while the HCAN is in bit configuration mode. Configuration mode is a state in which the GSR3 bit in GSR is set to 1 by a reset. Configuration mode is exited by clearing the MCR0 bit in MCR to 0; when the MCR0 bit is cleared to 0, the HCAN automatically clears the GSR3 bit in GSR. There is a delay between clearing the MCR0 bit and clearing the GSR3 bit because the HCAN needs time to be internally reset, there is a delay between clearing of the MCR0 bit and GSR3 bit. After the HCAN exits configuration mode, the power-up sequence begins, and communication with the CAN bus is possible as soon as 11 consecutive recessive bits have been detected.

**IRR0 Clearing:** The reset interrupt flag (IRR0) is always set after a power-on reset or recovery from software standby mode. As an HCAN interrupt is initiated immediately when interrupts are enabled, IRR0 should be cleared.

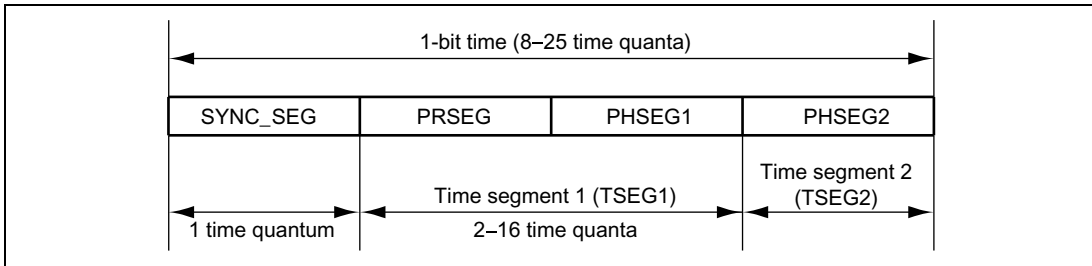


**Figure 15-6 Hardware Reset Flowchart**



**Figure 15-7 Software Reset Flowchart**

**Bit Rate and Bit Timing Settings:** The bit rate and bit timing settings are made in the bit configuration register (BCR). Settings should be made such that all CAN controllers connected to the CAN bus have the same baud rate and bit width. The 1-bit time consists of the total of the settable time quantum (tq).



**Figure 15-8 Detailed Description of One Bit**

SYNC\_SEG is a segment for establishing the synchronization of nodes on the CAN bus. Normal bit edge transitions occur in this segment. PRSEG is a segment for compensating for the physical delay between networks. PHSEG1 is a buffer segment for correcting phase drift (positive). This segment is extended when synchronization (resynchronization) is established. PHSEG2 is a buffer segment for correcting phase drift (negative). This segment is shortened when synchronization (resynchronization) is established. Limits on the settable value (TSEG1, TSEG2, BRP, sample point, and SJW) are shown in table 15-2.

**Table 15-2 Limits for Settable Value**

| Name                          | Abbreviation      | Min Value            | Max Value |
|-------------------------------|-------------------|----------------------|-----------|
| Time segment 1                | TSEG1             | B'0011 <sup>*2</sup> | B'1111    |
| Time segment 2                | TSEG2             | B'001 <sup>*3</sup>  | B'111     |
| Baud rate prescaler           | BRP               | B'000000             | B'111111  |
| Bit sample point              | BSP               | B'0                  | B'1       |
| Re-synchronization jump width | SJW <sup>*1</sup> | B'00                 | B'11      |

Notes: \*1 SJW is stipulated in the CAN specifications:

$$3 \geq \text{SJW} \geq 0$$

\*2 The minimum value of TSEG2 is stipulated in the CAN specifications:

$$\text{TSEG2} \geq \text{SJW}$$

\*3 The minimum value of TSEG1 is stipulated in the CAN specifications:

$$\text{TSEG1} > \text{TSEG2}$$

Time Quanta (TQ) is an integer multiple of the number of system clocks, and is determined by the baud rate prescaler (BRP) as follows.  $f_{CLK}$  is the system clock frequency.

$$TQ = 2 \times (\text{BPR setting} + 1) / f_{CLK}$$

The following formula is used to calculate the 1-bit time and bit rate.

$$\text{1-bit time} = TQ \times (3 + \text{TSEG1} + \text{TSEG2})$$

$$\text{Bit rate} = 1 / \text{Bit time}$$

$$= f_{CLK} / \{2 \times (\text{BPR setting} + 1) \times (3 + \text{TSEG1} + \text{TSEG2})\}$$

Note:  $f_{CLK} = \phi$  (system clock)

A BCR value is used for BRP, TSEG1, and TSEG2.

Example: With a system clock of 20 MHz, a BRP setting of B'000000, a TSEG1 setting of B'0100, and a TSEG2 setting of B'011:

$$\text{Bit rate} = 20 / \{2 \times (0 + 1) \times (3 + 4 + 3)\} = 1 \text{ Mbps}$$

**Table 15-3 Setting Range for TSEG1 and TSEG2 in BCR**

|                      |      | TSEG2 (BCR[14:12]) |     |     |     |     |     |     |
|----------------------|------|--------------------|-----|-----|-----|-----|-----|-----|
|                      |      | 001                | 010 | 011 | 100 | 101 | 110 | 111 |
|                      |      | 2                  | 3   | 4   | 5   | 6   | 7   | 8   |
| TSEG1<br>(BCR[11:8]) | 0011 | No                 | Yes | No  | No  | No  | No  | No  |
|                      | 0100 | Yes*               | Yes | Yes | No  | No  | No  | No  |
|                      | 0101 | Yes*               | Yes | Yes | Yes | No  | No  | No  |
|                      | 0110 | Yes*               | Yes | Yes | Yes | Yes | No  | No  |
|                      | 0111 | Yes*               | Yes | Yes | Yes | Yes | Yes | No  |
|                      | 1000 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1001 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1010 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1011 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1100 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1101 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1110 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |
|                      | 1111 | Yes*               | Yes | Yes | Yes | Yes | Yes | Yes |

Notes: The time quantum value for TSEG1 and TSEG2 is the TSEG value + 1.

\* Only a value other than BRP[13:8] = B'000000 can be set.

**Mailbox Transmit/Receive Settings:** The HCAN has 16 mailboxes. Mailbox 0 is receive-only, while mailboxes 1 to 15 can be set for transmission or reception. The Initial status of mailboxes 1 to 15 is for transmission. Mailbox transmit/receive settings are not initialized by a software reset.

Clearing a bit to 0 in the mailbox configuration register (MBCR) designates the corresponding mailbox for transmission use, whereas a setting of 1 in MBCR designates the corresponding mailbox for reception use. When setting mailboxes for reception, in order to improve message reception efficiency, high-priority messages should be set in low-to-high mailbox order.

**Mailbox (Message Control/Data) Initial Settings:** Message control/data are held in RAM, and so their initial values are undefined after power is supplied. Initial values must therefore be set in all the mailboxes (by writing 0s or 1s).

**Setting the Message Transmission Method:** The following two kinds of message transmission methods are available.

- Transmission order determined by message identifier priority
- Transmission order determined by mailbox number priority

Either of the message transmission methods can be selected with the message transmission method bit (MCR2) in the master control register (MCR): When messages are set to be transmitted according to the message identifier priority, if several messages are designated as waiting for transmission (TXPR = 1), the message with the highest priority in the message identifier is stored in the transmit buffer. CAN bus arbitration is then carried out for the message stored in the transmit buffer, and the message is transmitted when the transmission right is acquired. When the TXPR bit is set, the highest-priority message is found and stored in the transmit buffer.

When messages are set to be transmitted according to the mailbox number priority, if several messages are designated as waiting for transmission (TXPR = 1), messages are stored in the transmit buffer in low-to-high mailbox order. CAN bus arbitration is then carried out for the message stored in the transmit buffer, and the message is transmitted when the transmission right is acquired.

### 15.4.3 Message Transmission

Messages are transmitted using mailboxes 1 to 15. The transmission procedure after initial settings is described below, and a transmission flowchart is shown in figure 15-9.

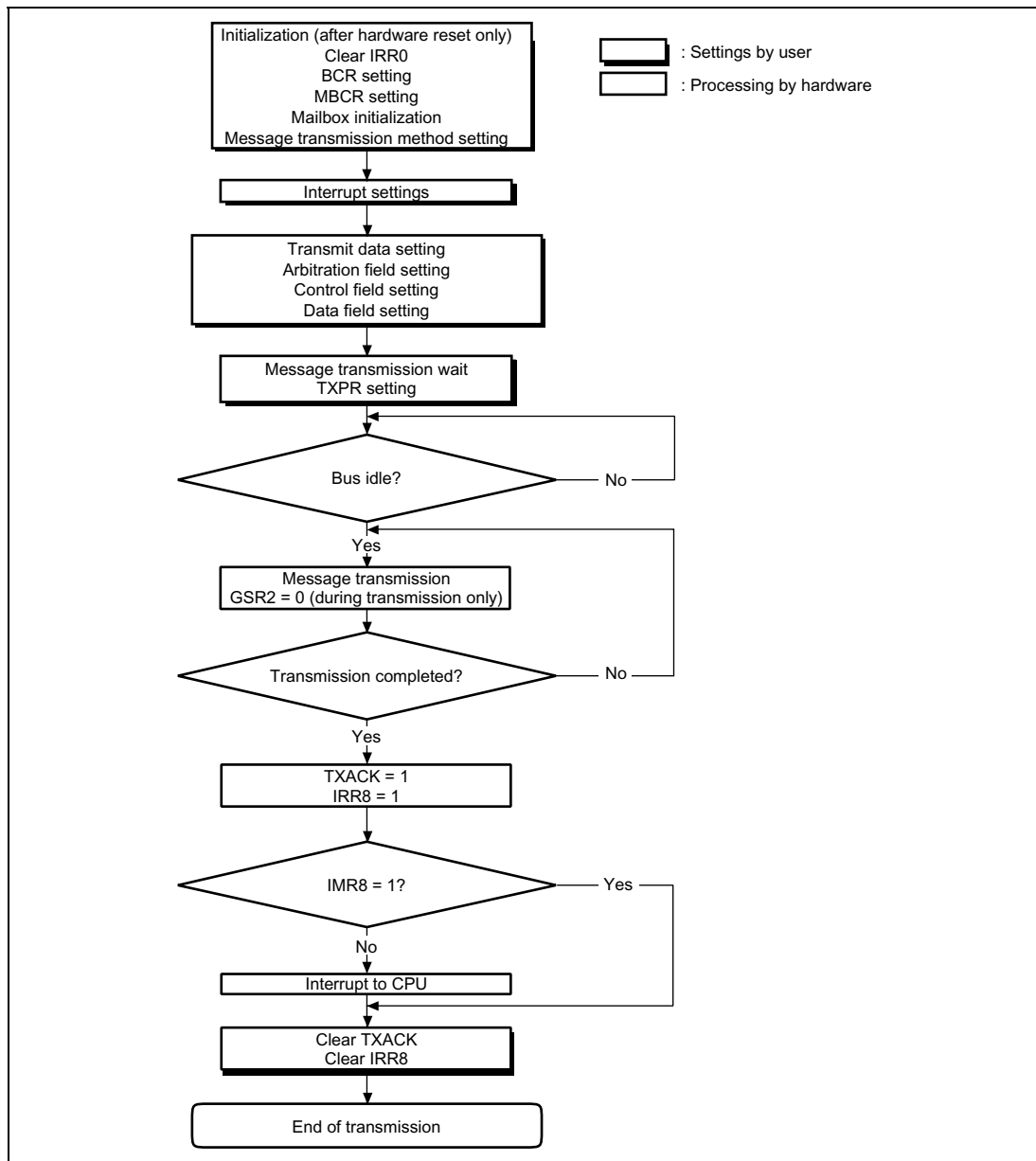


Figure 15-9 Transmission Flowchart

**CPU Interrupt Source Settings:** The CPU interrupt source is set by the interrupt mask register (IMR) and mailbox interrupt mask register (MBIMR). Transmission acknowledge and transmission abort acknowledge interrupts can be generated for individual mailboxes in the mailbox interrupt mask register (MBIMR).

**Arbitration Field Setting:** The arbitration field is set by message control registers MCx[5]–MCx[8] in a transmit mailbox. For a standard format, an 11-bit identifier (ID-28 to ID-18) and the RTR bit are set, and the IDE bit is cleared to 0. For an extended format, a 29-bit identifier (ID-28 to ID-0) and the RTR bit are set, and the IDE bit is set to 1.

**Control Field Setting:** In the control field, the byte length of the data to be transmitted is set within the range of zero to eight bytes. The register to be set is the message control register MCx[1] in a transmit mailbox.

**Data Field Setting:** In the data field, the data to be transmitted is set within the range zero to eight. The registers to be set are the message data registers MDx[1]–MDx[8]. The byte length of the data to be transmitted is determined by the data length code in the control field. Even if data exceeding the value set in the control field is set in the data field, up to the byte length set in the control field will actually be transmitted.

**Message Transmission:** If the corresponding mailbox transmit wait bit (TXPR1–TXPR15) in the transmit wait register (TXPR) is set to 1 after message control and message data registers have been set, the message enters transmit wait state. If the message is transmitted error-free, the corresponding acknowledge bit (TXACK1–TXACK15) in the transmit acknowledge register (TXACK) is set to 1, and the corresponding transmit wait bit (TXPR1–TXPR15) in the transmit wait register (TXPR) is automatically cleared to 0. Also, if the corresponding bit (MBIMR1–MBIMR15) in the mailbox interrupt mask register (MBIMR) and the mailbox empty interrupt bit (IRR8) in the interrupt mask register (IMR) are both simultaneously set to enable interrupts, interrupts may be sent to the CPU.

If transmission of a transmit message is aborted in the following cases, the message is retransmitted automatically:

- CAN bus arbitration failure (failure to acquire the bus)
- Error during transmission (bit error, stuff error, CRC error, frame error, or ACK error)

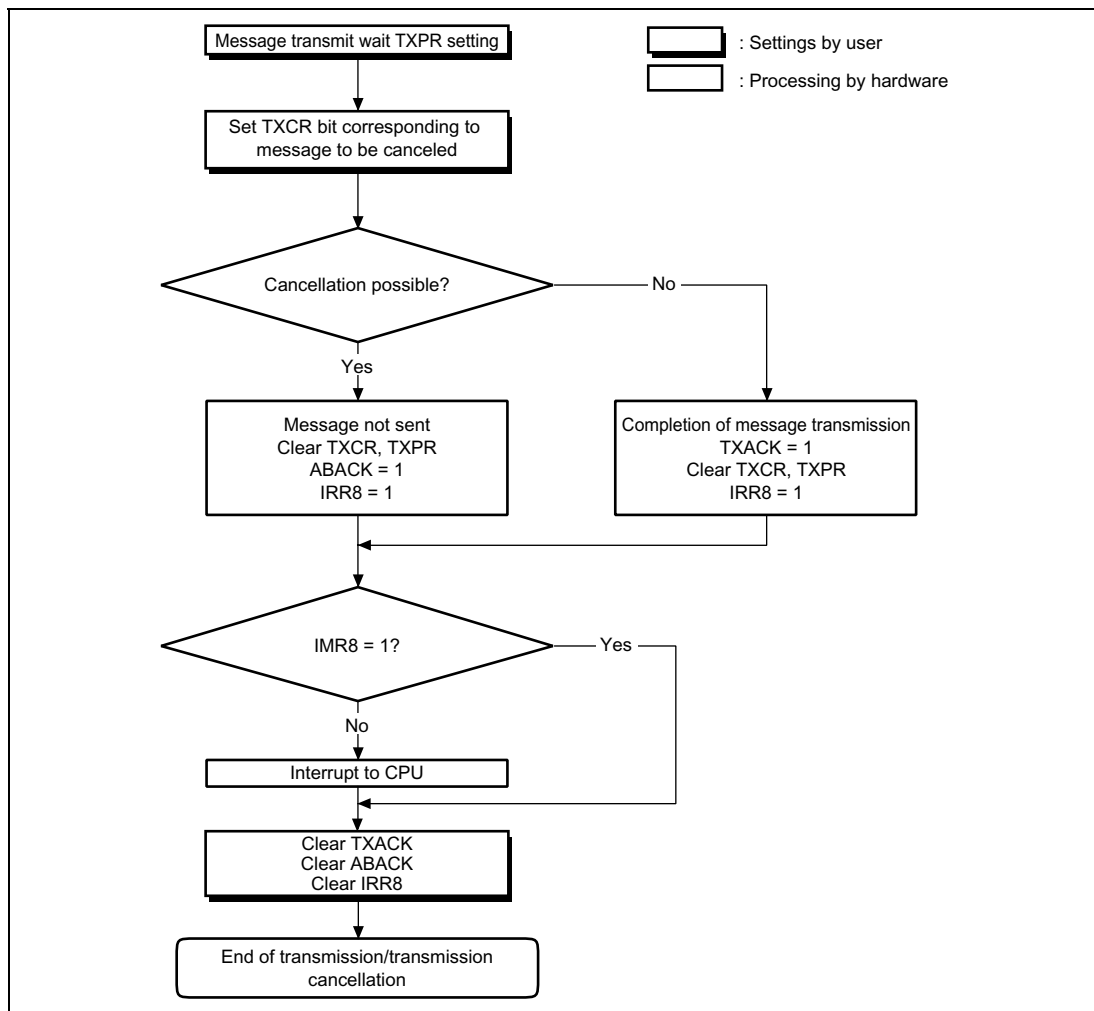
**Message Transmission Cancellation:** Transmission cancellation can be specified for a message stored in a mailbox as a transmit wait message. A transmit wait message is canceled by setting the bit for the corresponding mailbox (TXCR1–TXCR15) to 1 in the transmit cancel register (TXCR). Clearing the transmit wait register (TXPR) does not cancel transmission. When cancellation is executed, the transmit wait register (TXPR) is automatically reset, and the corresponding bit is set to 1 in the abort acknowledge register (ABACK). An interrupt to the CPU can be requested, and if the mailbox empty interrupt (IRR8) is enabled for the bits (MBIMR1–MBIMR15) corresponding

to the mailbox interrupt mask register (MBIMR) and interrupt mask register (IMR), interrupts may be sent to the CPU.

However, a transmit wait message cannot be canceled at the following times:

- During internal arbitration or CAN bus arbitration
- During data frame or remote frame transmission

Figure 15-10 shows a flowchart for transmit message cancellation.



**Figure 15-10 Transmit Message Cancellation Flowchart**

## 15.4.4 Message Reception

The reception procedure after initial settings is described below. A reception flowchart is shown in figure 15-11.

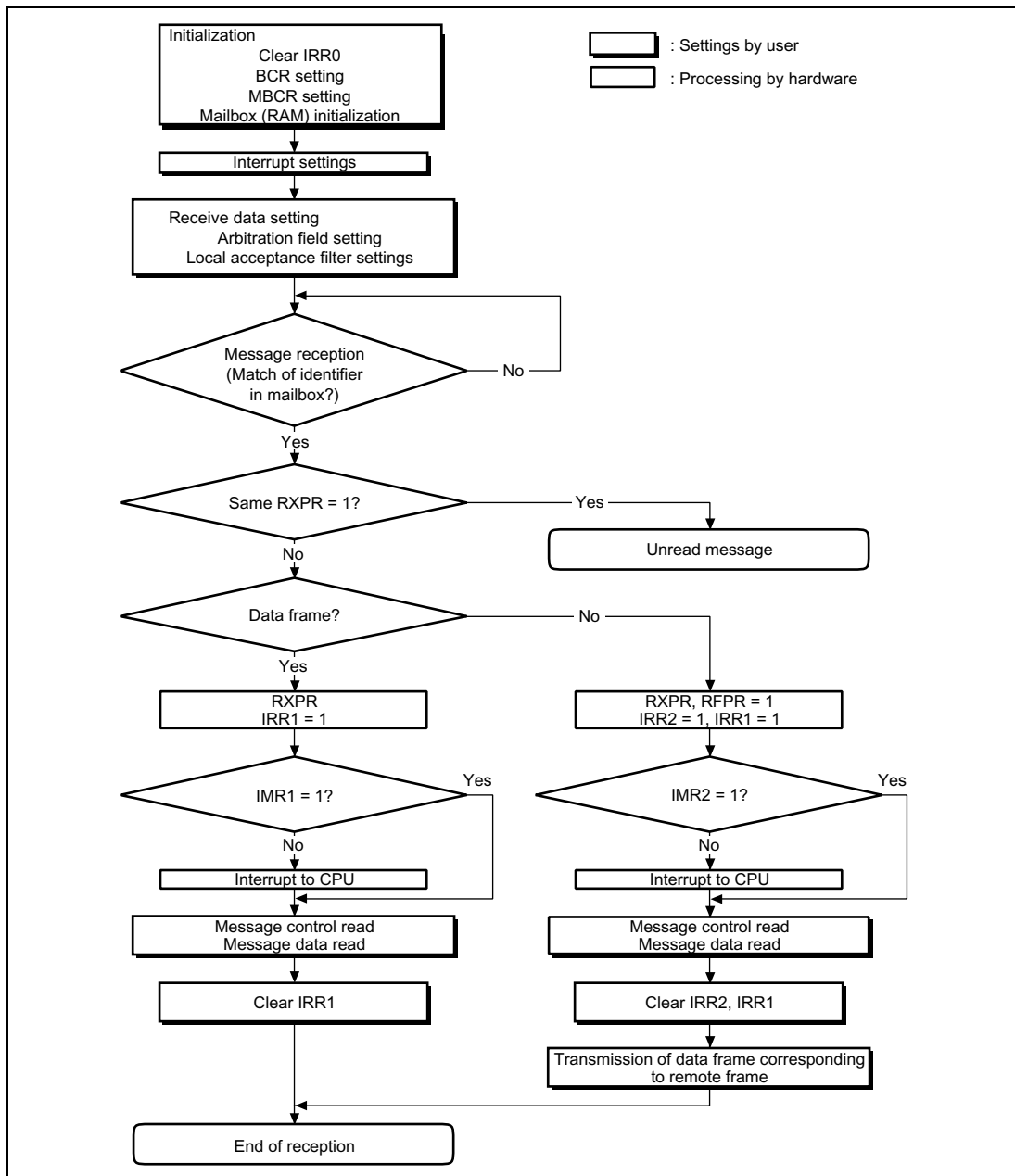


Figure 15-11 Reception Flowchart

**CPU Interrupt Source Settings:** CPU interrupt source settings are made in the interrupt mask register (IMR) and mailbox interrupt register (MBIMR). The message to be received is also specified. Data frame and remote frame receive wait interrupt requests can be generated for individual mailboxes in the MBIMR.

**Arbitration Field Setting:** To receive a message, the message identifier must be set in advance in the message control registers (MCx[1]–MCx[8]) for the receiving mailbox. When a message is received, all the bits in the receive message identifier are compared with those in each message control register identifier, and if a 100% match is found, the message is stored in the matching mailbox. Mailbox 0 has a local acceptance filter mask (LAFM) that allows Don't care settings to be made. The LAFM setting can be made only for mailbox 0. By making the Don't care setting for all the bits in the receive message identifier, messages of multiple identifiers can be received.

Examples:

- When the identifier of mailbox 1 is 010\_1010\_1010 (standard format), only one kind of message identifier can be received by mailbox 1:  
Identifier 1:       010\_1010\_1010
- When the identifier of mailbox 0 is 010\_1010\_1010 (standard format) and the LAFM setting is 000\_0000\_0011 (0: Care, 1: Don't care), a total of four kinds of message identifiers can be received by mailbox 0:  
Identifier 1:       010\_1010\_1000  
Identifier 2:       010\_1010\_1001  
Identifier 3:       010\_1010\_1010  
Identifier 4:       010\_1010\_1011

**Message Reception:** When a message is received, a CRC check is performed automatically. If the result of the CRC check is normal, ACK is transmitted in the ACK field irrespective of whether the message can be received or not.

- Data frame reception

If the received message is confirmed to be error-free by the CRC check, the identifier in the mailbox (and also LAFM in the case of mailbox 0 only) and the identifier of the receive message, are compared. If a complete match is found, the message is stored in the mailbox. The message identifier comparison is carried out on each mailbox in turn, starting with mailbox 0 and ending with mailbox 15. If a complete match is found, the comparison ends at that point, the message is stored in the matching mailbox, and the corresponding receive complete bit (RXPR0–RXPR15) is set in the receive complete register (RXPR). However, when a mailbox 0 LAFM comparison is carried out, even if the identifier matches, the mailbox comparison sequence does not end at that point, but continues with mailbox 1 and then the remaining mailboxes. It is therefore possible for a message matching mailbox 0 to be received by another mailbox. Note that the same message cannot be stored in more than one of mailboxes 1 to 15. On receiving a message, a CPU interrupt request may be generated

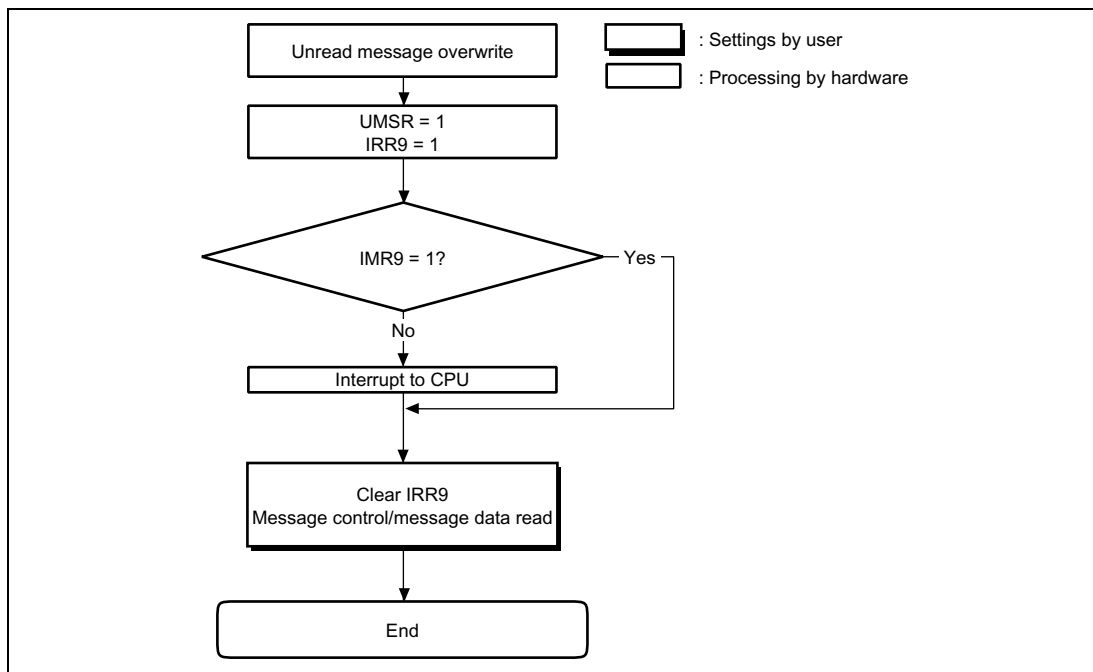
depending on the mailbox interrupt mask register (MBIMR) and interrupt mask register (IMR) settings.

- Remote frame reception

Two kinds of messages—data frames and remote frames—can be stored in mailboxes. A remote frame differs from a data frame in that the remote transmission request bit (RTR) in the message control register and the data field are 0 bytes long. The data length to be returned in a data frame must be stored in the data length code (DLC) in the control field.

When a remote frame (RTR = recessive) is received, the corresponding bit is set in the remote request wait register (RFPR). If the corresponding bit (MBIMR0–MBIMR15) in the mailbox interrupt mask register (MBIMR) and the remote frame request interrupt mask (IRR2) in the interrupt mask register (IMR) are set to the interrupt enable value at this time, an interrupt can be sent to the CPU.

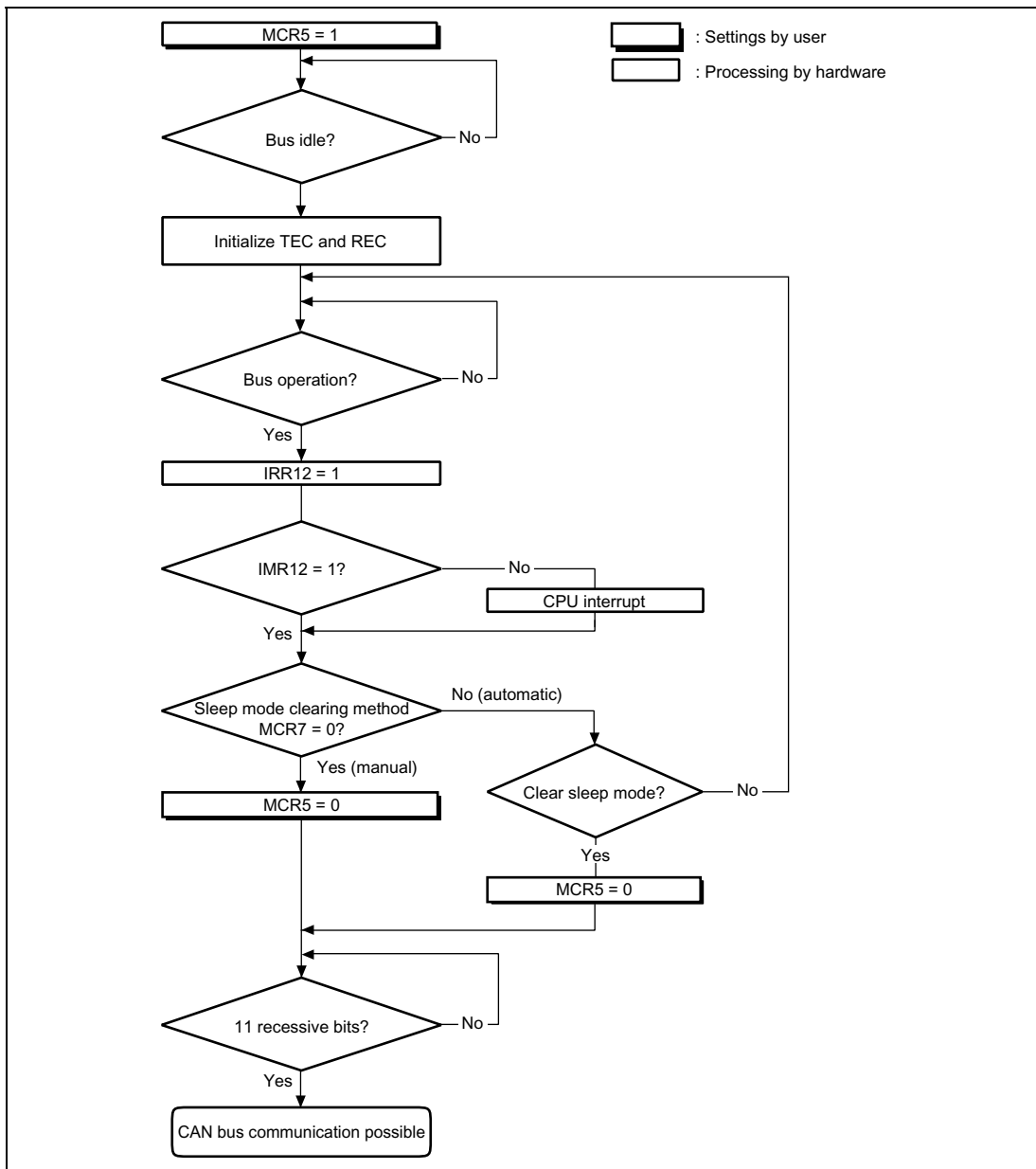
**Unread Message Overwrite:** If the receive message identifier matches the mailbox identifier, the receive message is stored in the mailbox regardless of whether the mailbox contains an unread message or not. If a message overwrite occurs, the corresponding bit (UMSR0–UMSR15) is set in the unread message register (UMSR). In overwriting an unread message, when a new message is received before the corresponding bit in the receive complete register (RXPR) has been cleared, the unread message register (UMSR) is set. If the unread interrupt flag (IRR9) in the interrupt mask register (IMR) is set to the interrupt enable value at this time, an interrupt can be sent to the CPU. Figure 15-12 shows a flowchart for unread message overwriting.



**Figure 15-12 Unread Message Overwrite Flowchart**

### 15.4.5 HCAN Sleep Mode

The HCAN is provided with an HCAN sleep mode that places the HCAN module in the sleep state in order to reduce current dissipation. Figure 15-13 shows a flowchart of the HCAN sleep mode.



**Figure 15-13 HCAN Sleep Mode Flowchart**

HCAN sleep mode is entered by setting the HCAN sleep mode bit (MCR5) to 1 in the master control register (MCR). If the CAN bus is operating, the transition to HCAN sleep mode is delayed until the bus becomes idle.

Either of the following methods of clearing HCAN sleep mode can be selected:

- Clearing by software
- Clearing by CAN bus operation

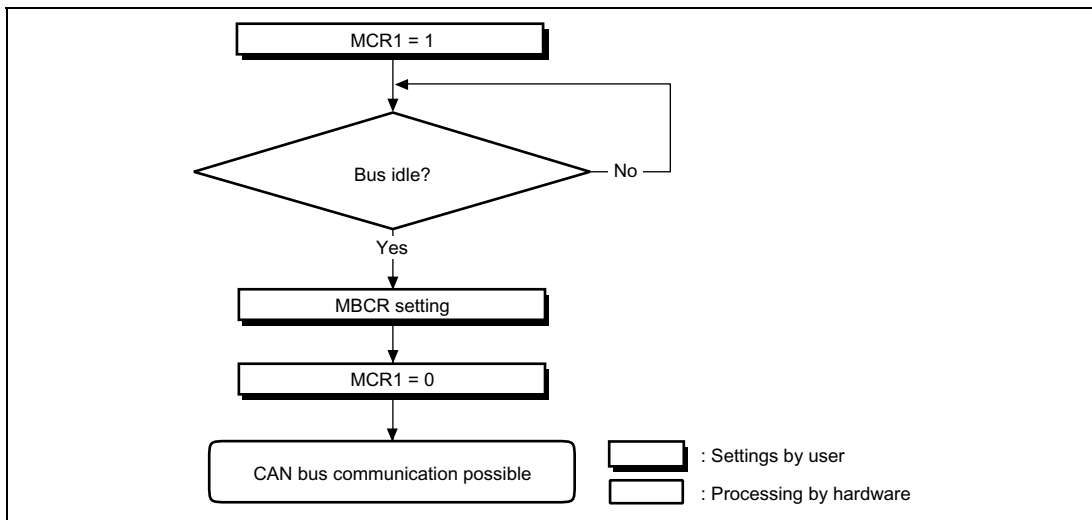
Eleven recessive bits must be received after HCAN sleep mode is cleared before CAN bus communication is re-enabled.

**Clearing by Software:** HCAN sleep mode is cleared by writing a 0 to MCR5 from the CPU.

**Clearing by CAN Bus Operation:** The cancellation method is selected by the MCR7 bit setting in MCR. Clearing by CAN bus operation occurs automatically when the CAN bus performs an operation and this change is detected. In this case, the first message is not stored in a mailbox; messages will be received normally from the second message onward. When a change is detected on the CAN bus in HCAN sleep mode, the bus operation interrupt flag (IRR12) is set in the interrupt register (IRR). If the bus interrupt mask (IMR12) in the interrupt mask register (IMR) is set to the interrupt enable value at this time, an interrupt can be sent to the CPU.

## 15.4.6 HCAN Halt Mode

The HCAN halt mode is provided to enable mailbox settings to be changed without performing an HCAN hardware or software reset. Figure 15-14 shows a flowchart of the HCAN halt mode.



**Figure 15-14 HCAN Halt Mode Flowchart**

HCAN halt mode is entered by setting the halt request bit (MCR1) to 1 in the master control register (MCR). If the CAN bus is operating, the transition to HCAN halt mode is delayed until the bus becomes idle.

HCAN halt mode is cleared by clearing MCR1 to 0.

## 15.5 Interrupts

Table 15-4 lists the HCAN interrupt sources. With the exception of the reset processing vector (IRR0), these sources can be masked. Masking is implemented using the mailbox interrupt mask register (MBIMR) and interrupt mask register (IMR). For details on the interrupt vector of each interrupt source, refer to section 5, Interrupt Controller.

**Table 15-4 HCAN Interrupt Sources**

| Name      | Description                                                  | Interrupt Flag | DTC Activation |
|-----------|--------------------------------------------------------------|----------------|----------------|
| ERS0/OVR0 | Error passive interrupt ( $TEC \geq 128$ or $REC \geq 128$ ) | IRR5           | Not possible   |
|           | Bus off interrupt ( $TEC \geq 256$ )                         | IRR6           |                |
|           | Reset process interrupt by power-on reset                    | IRR0           |                |
|           | Remote frame reception                                       | IRR2           |                |
|           | Error warning interrupt ( $TEC \geq 96$ )                    | IRR3           |                |
|           | Error warning interrupt ( $REC \geq 96$ )                    | IRR4           |                |
|           | Overload frame transmission interrupt                        | IRR7           |                |
|           | Unread message overwrite                                     | IRR9           |                |
|           | Detection of CAN bus operation in HCAN sleep mode            | IRR12          |                |
| RM0       | Mailbox 0 message reception                                  | IRR1           | Possible       |
| RM1       | Mailboxes 1-15 message reception                             | IRR1           | Not possible   |
| SLE0      | Message transmission/cancellation                            | IRR8           | Not possible   |

## 15.6 DTC Interface

The DTC can be activated by the reception of a message in HCAN mailbox 0. When DTC transfer ends after DTC activation has been set, the RXPR0 and RFPR0 flags are acknowledge automatically. An interrupt request due to a receive interrupt from the HCAN cannot be sent to the CPU in this case. Figure 15-15 shows a DTC transfer flowchart.

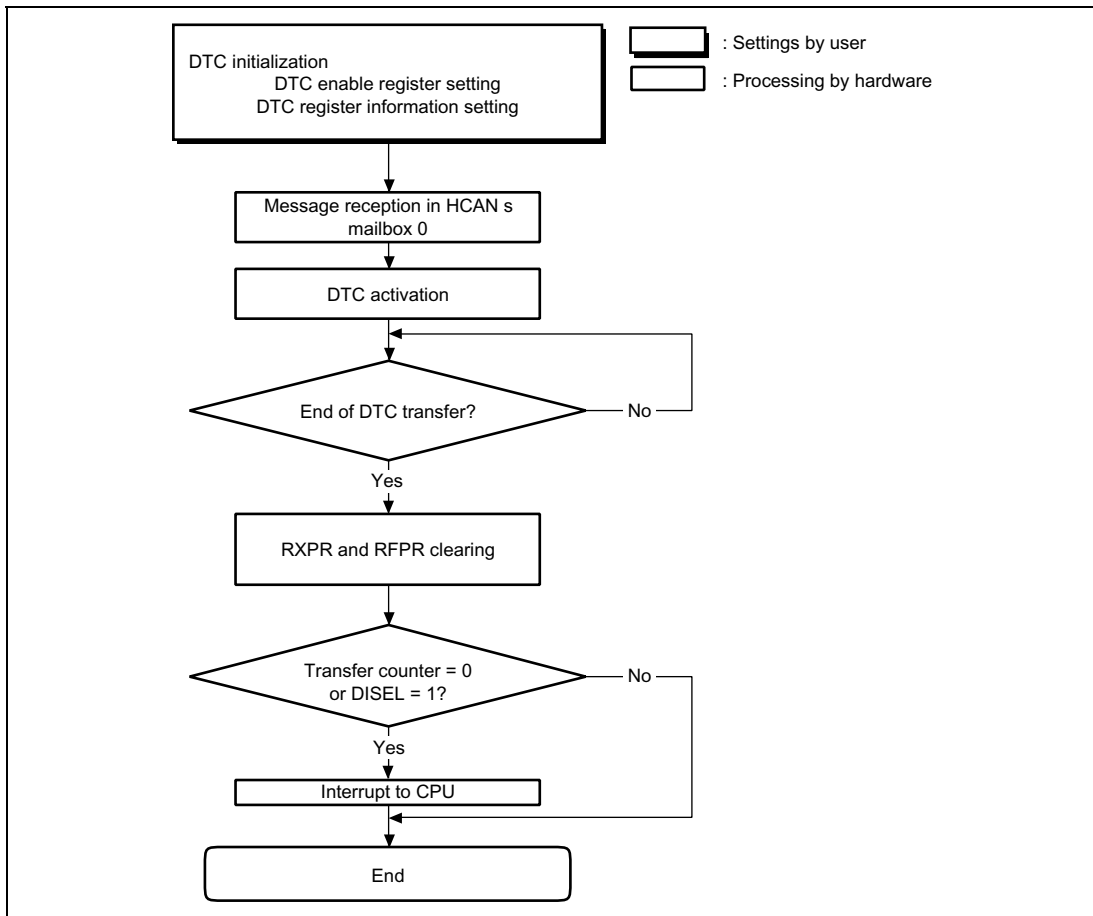
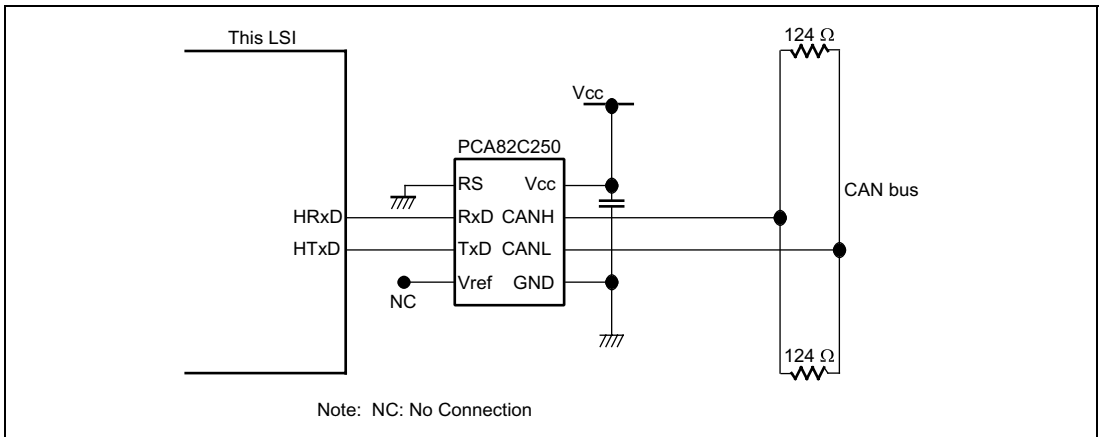


Figure 15-15 DTC Transfer Flowchart

## 15.7 CAN Bus Interface

A bus transceiver IC is necessary to connect this chip to a CAN bus. A Philips PCA82C250 transceiver IC is recommended. Any other product must be compatible with the PCA82C250. Figure 15-16 shows a sample connection diagram.



**Figure 15-16 High-Speed Interface Using PCA82C250**

## 15.8 Usage Notes

### 15.8.1 Module Stop Mode Setting

HCAN operation can be disabled or enabled using the module stop control register. The initial setting is for HCAN operation to be halted. Register access is enabled by clearing module stop mode. For details, refer to section 20, Power-Down Modes.

### 15.8.2 Reset

The HCAN is reset by a power-on reset, in hardware standby mode, and in software standby mode. All the registers are initialized in a reset, however mailboxes (message control (MCx[x])/message data (MDx[x])) are not. After power-on, mailboxes (message control (MCx[x])/message data (MDx[x])) are un-initialized, and their values are undefined. Therefore, mailbox initialization must always be carried out after a power-on reset, a transition to hardware standby mode, or software standby mode. The reset interrupt flag (IRR0) is always set after a power-on reset or recovery from software standby mode. As this bit cannot be masked in the interrupt mask register (IMR), if HCAN interrupt enabling is set in the interrupt controller without clearing the flag, an HCAN interrupt will be initiated immediately. IRR0 should therefore be cleared during initialization.

### **15.8.3 HCAN Sleep Mode**

The bus operation interrupt flag (IRR12) in the interrupt register (IRR) is set by CAN bus operation in HCAN sleep mode. Therefore, this flag is not used by the HCAN to indicate sleep mode release. Note that the reset status bit (GSR3) in the general status register (GSR) is set in sleep mode.

### **15.8.4 Interrupts**

When the mailbox interrupt mask register (MBIMR) is set, the interrupt register (IRR8, 2, 1) is not set by reception completion, transmission completion, or transmission cancellation for the set mailboxes.

### **15.8.5 Error Counters**

In the case of error active and error passive, REC and TEC normally count up and down. In the bus-off state, 11-bit recessive sequences are counted (REC + 1) using REC. If REC reaches 96 during the count, IRR4 and GSR1 are set.

### **15.8.6 Register Access**

Byte or word access can be used on all HCAN registers. Longword access cannot be used.

### **15.8.7 HCAN Medium-Speed Mode**

In medium-speed mode, neither read nor write is possible for the HCAN registers.

### **15.8.8 Register Hold in Standby Modes**

All HCAN registers are initialized in hardware standby mode and software standby mode.

### **15.8.9 Usage of Bit Manipulation Instructions**

The HCAN status flags are cleared by writing 1, so do not use a bit manipulation instruction to clear a flag.

When clearing a flag, use the MOV instruction to write 1 to only the bit that is to be cleared.

### 15.8.10 HCAN TXCR Operation

1. When the transmit wait cancel register (TXCR) is used to cancel a transmit wait message in a transmit wait mailbox, the corresponding bit to TXCR and the transmit wait register (TXPR) may not be cleared even if transmission is canceled. This occurs when the following conditions are all satisfied.
  - The HRxD pin is stacked to 1 because of a CAN bus error, etc.
  - There is at least one mailbox waiting for transmission or being transmitted.
  - The message transmission in a mailbox being transmitted is canceled by TXCR.

If this occurs, transmission is canceled. However, since TXPR and TXCR states are indicated wrongly that a message is being cancelled, transmission cannot be restarted even if the stack state of the HRxD pin is canceled and the CAN bus recovers the normal state. If there are at least two transmission messages, a message which is not being transmitted is canceled and a message being transmitted retains its state.

To avoid this, one of the following countermeasures must be executed.

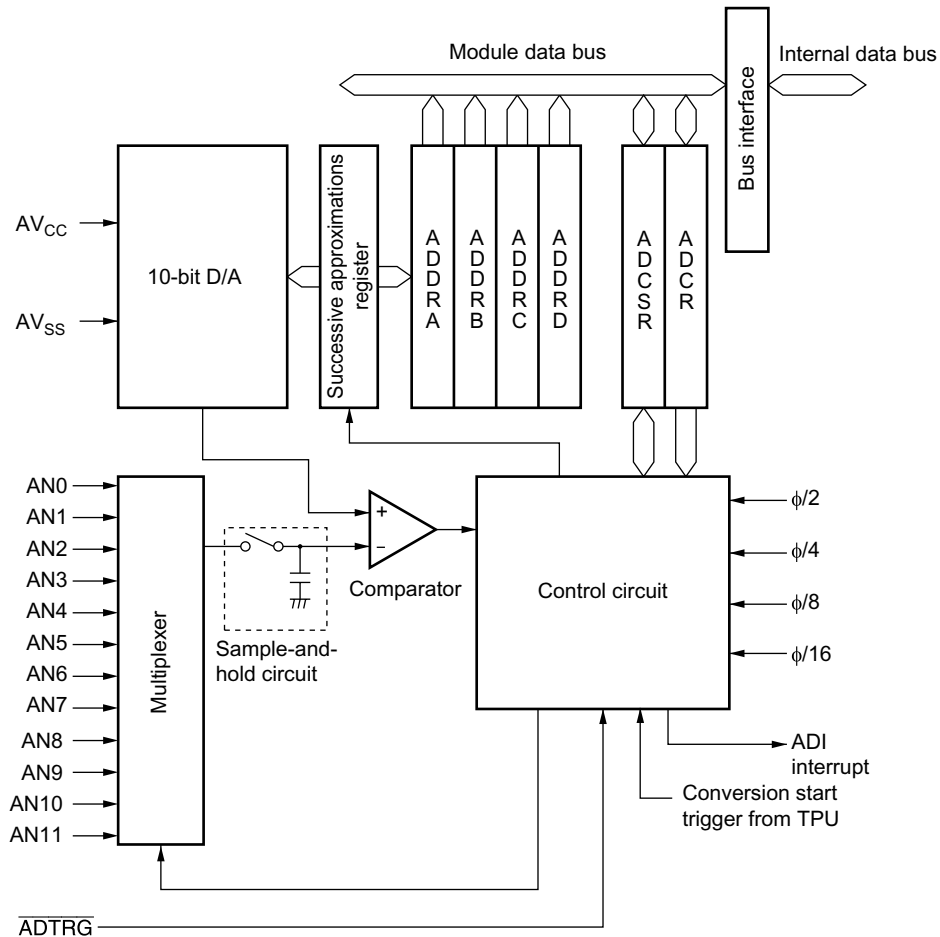
- Transmission must not be canceled by TXCR. When transmission is normally completed after the CAN bus has recovered, TXPR is cleared and the HCAN recovers the normal state.
  - To cancel transmission, the corresponding bit to TXCR must be written to 1 continuously until the bit becomes 0. TXPR and TXCR are cleared and the HCAN recovers the normal state.
2. When the bus-off state is entered while TXPR is set and the transmit wait state is entered, the internal state machine does not operate even if TXCR is set during the bus-off state. Therefore transmission cannot be canceled. The message can be canceled when one message is transmitted or a transmission error occurs after the bus-off state is recovered. To clear a message after the bus-off state is recovered, the following countermeasure must be executed.
    - A transmit wait message must be cleared by resetting the HCAN during the bus-off period. To reset the HCAN, the module stop bit (MSTPC3 in MSTPCRC) must be set or cleared. In this case, the HCAN is entirely reset. Therefore the initial settings must be made again.

## Section 16 A/D Converter

This LSI includes a successive approximation type 10-bit A/D converter that allows up to twelve analog input channels to be selected. The Block diagram of the A/D converter is shown in figure 16-1.

### 16.1 Features

- 10-bit resolution
- Twelve input channels
- Conversion time: 13.3  $\mu$ s per channel (at 20 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
  - 16-bit timer pulse unit (TPU) 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



#### Legend

ADCR : A/D control register  
 ADCSR : A/D control/status register  
 ADDR A : A/D data register A  
 ADDR B : A/D data register B  
 ADDR C : A/D data register C  
 ADDR D : A/D data register D

**Figure 16-1 Block Diagram of A/D Converter**

## 16.2 Input/Output Pins

Table 16-1 summarizes the input pins used by the A/D converter. The 12 analog input pins are divided into four channel sets and three groups; analog input pins 0 to 3 (AN0 to AN3) comprising group 0, analog input pins 4 to 7 (AN4 to AN7) comprising group 1, and analog input pins 8 to 11 (AN8 to AN11) comprising group 2. The AVcc and AVss pins are the power supply pins for the analog block in the A/D converter.

**Table 16-1 Pin Configuration**

| Pin Name                       | Symbol           | I/O   | Function                                               |
|--------------------------------|------------------|-------|--------------------------------------------------------|
| Analog power supply pin        | AV <sub>CC</sub> | Input | Analog block power supply and reference voltage        |
| Analog ground pin              | AV <sub>SS</sub> | Input | Analog block ground and reference voltage              |
| 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 |                                                        |
| Analog input pin 8             | AN8              | Input | Group 2 analog input pins                              |
| Analog input pin 9             | AN9              | Input |                                                        |
| Analog input pin 10            | AN10             | Input |                                                        |
| Analog input pin 11            | AN11             | Input |                                                        |
| A/D external trigger input pin | ADTRG            | Input | External trigger input pin for starting A/D conversion |

## 16.3 Register Description

The A/D converter has the following registers. For details on register addresses, refer to Appendix A, On-Chip I/O Register. The MSTPA1 bit in the module stop control register (MSTPCRA) specifies the modes of this module as module stop mode. For details on the module stop control register A (MSTPCRA), refer to section 20.1.2, Module Stop Control Register 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)

### 16.3.1 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 16-2.

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. The temporary register contents are transferred from the ADDR when the upper byte data is read. When reading the ADDR, read the upper byte before the lower byte, or read in word unit. When only the lower byte is read, the contents are not guaranteed.

**Table 16-2 Analog Input Channels and Corresponding ADDR Registers**

| Analog Input Channel |                      |                      |                    | A/D Data Register to Be Stored the Results of A/D Conversion |
|----------------------|----------------------|----------------------|--------------------|--------------------------------------------------------------|
| CH3 = 0              |                      | CH3 = 1              |                    |                                                              |
| Group 0<br>(CH2 = 0) | Group 1<br>(CH2 = 1) | Group 2<br>(CH2 = 0) | —<br>(CH2 = 1)     |                                                              |
| AN0                  | AN4                  | AN8                  | Setting prohibited | ADDRA                                                        |
| AN1                  | AN5                  | AN9                  | Setting prohibited | ADDRB                                                        |
| AN2                  | AN6                  | AN10                 | Setting prohibited | ADDRC                                                        |
| AN3                  | AN7                  | AN11                 | Setting prohibited | ADDRD                                                        |

### 16.3.2 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"> <li>When A/D conversion ends</li> <li>When A/D conversion ends on all specified channels</li> </ul> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When 0 is written after reading ADF = 1</li> <li>When the DTC is activated by an ADI interrupt and ADDR is read</li> </ul> |
| 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>                                                                                                                                                                                                                                                                                                                                                  |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 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, or a transition to software standby mode, hardware standby mode or module stop mode.</p> |
| 4   | SCAN     | 0             | R/W | <p>Scan Mode</p> <p>Selects single mode or scan mode as the A/D conversion operating mode.</p> <p>0: Single mode</p> <p>1: Scan mode</p>                                                                                                                                                                                                                                                                                                                                                               |
| 3   | CH3      | 0             | R/W | Channel Select 0 to 3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| 2   | CH2      | 0             | R/W | Select analog input channels.                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 1   | CH1      | 0             | R/W | When SCAN = 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| 0   | CH0      | 0             | R/W | When SCAN = 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|     |          |               |     | 0000: AN0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0001: AN1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0010: AN2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0011: AN3                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0100: AN4                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0101: AN5                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0110: AN6                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 0111: AN7                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 1000: AN8                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 1001: AN9                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|     |          |               |     | 1010: AN10                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|     |          |               |     | 1011: AN11                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|     |          |               |     | 1100: Setting prohibited                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|     |          |               |     | 1101: Setting prohibited                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|     |          |               |     | 1110: Setting prohibited                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|     |          |               |     | 1111: Setting prohibited                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

### 16.3.3 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 0 and 1                                                                                                                                                                                                                                                                                                                                |
| 6   | TRGS0    | 0             | R/W | Enables the start of A/D conversion by a trigger signal. Only set bits TRGS0 and TRGS1 while conversion is stopped (ADST = 0).<br>00: A/D conversion start by software is enabled<br>01: A/D conversion start by TPU conversion start trigger is enabled<br>10: Setting prohibited<br>11: A/D conversion start by external trigger pin (ADTRG) is enabled   |
| 5   | —        | 1             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                    |
| 4   | —        | 1             | —   | These bits are always read as 1.                                                                                                                                                                                                                                                                                                                            |
| 3   | CKS1     | 0             | R/W | Clock Select 0 and 1                                                                                                                                                                                                                                                                                                                                        |
| 2   | CKS0     | 0             | R/W | 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 21-7.<br>00: Conversion time = 530 states (max.)<br>01: Conversion time = 266 states (max.)<br>10: Conversion time = 134 states (max.)<br>11: Conversion time = 68 states (max.) |
| 1   | —        | 1             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                    |
| 0   | —        | 1             | —   | These bits are always read as 1.                                                                                                                                                                                                                                                                                                                            |

## 16.4 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.

### 16.4.1 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 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.

### 16.4.2 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 or external trigger input, A/D conversion starts on the first channel in the group (AN0 when CH3 and CH2 = 00, AN4 when CH3 and CH2 = 01, or AN8 when CH3 and CH2 = 10).
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.

### 16.4.3 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 ( $t_D$ ) has passed after the ADST bit is set to 1, then starts conversion. Figure 16-2 shows the A/D conversion timing. Table 16-3 shows the A/D conversion time.

As indicated in figure 16-2, the A/D conversion time ( $t_{CONV}$ ) includes  $t_D$  and the input sampling time ( $t_{SPL}$ ). The length of  $t_D$  varies depending on the timing of the write access to ADCSR. The total conversion time therefore varies within the ranges indicated in table 16-3.

In scan mode, the values given in table 16-3 apply to the first conversion time. The values given in table 16-4 apply to the second and subsequent conversions. In both cases, set bits CKS1 and CKS0 in ADCR to give an A/D conversion time within the range shown in table 21-7.

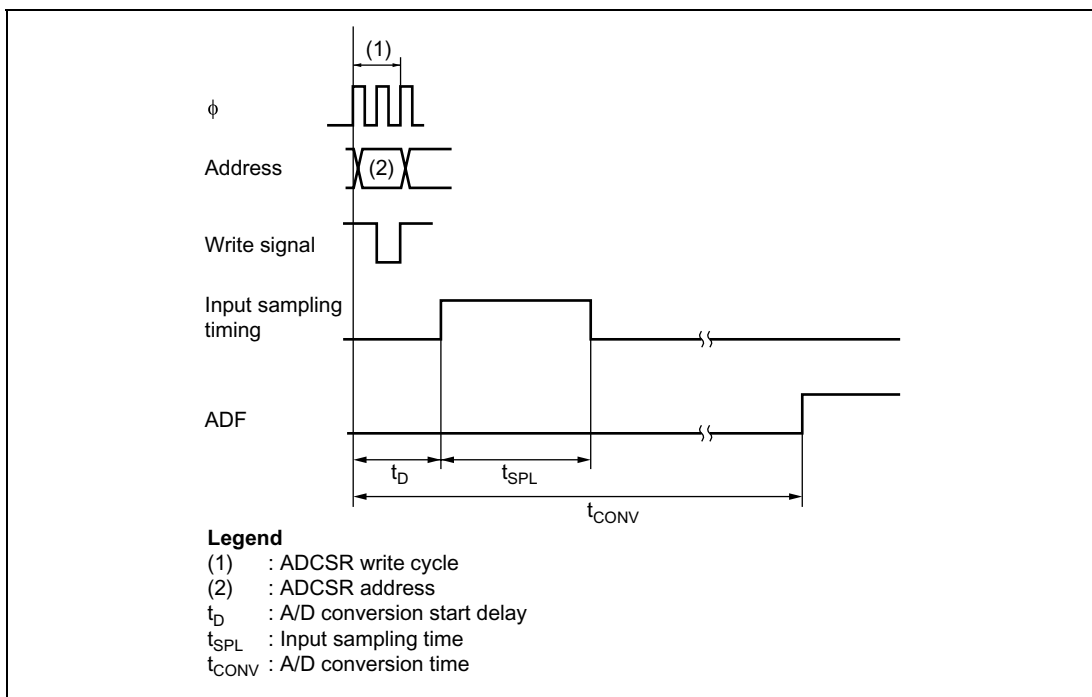


Figure 16-2 A/D Conversion Timing

**Table 16-3 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 16-4 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)              |

### 16.4.4 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  $\overline{\text{ADTRG}}$  pin. A falling edge at the  $\overline{\text{ADTRG}}$  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 16-3 shows the timing.

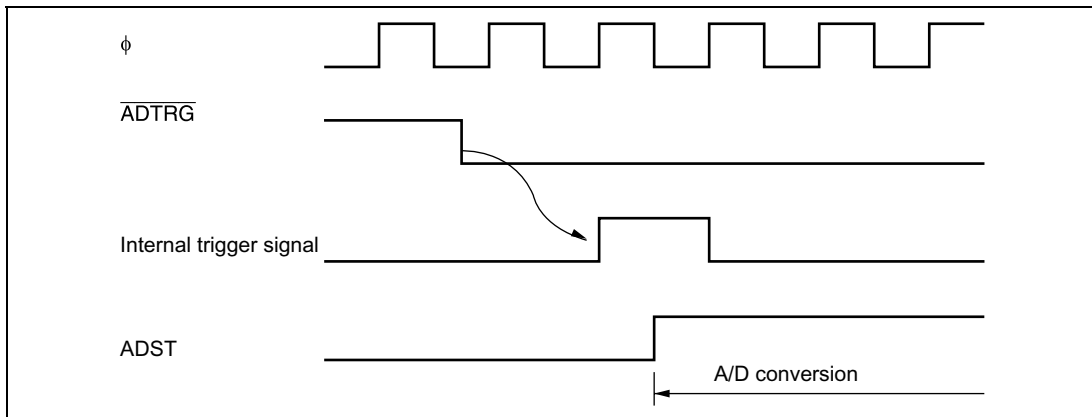


Figure 16-3 External Trigger Input Timing

## 16.5 Interrupts

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 DTC can be activated by an ADI interrupt. Having the converted data read by the DTC in response to an ADI interrupt enables continuous conversion without imposing a load on software.

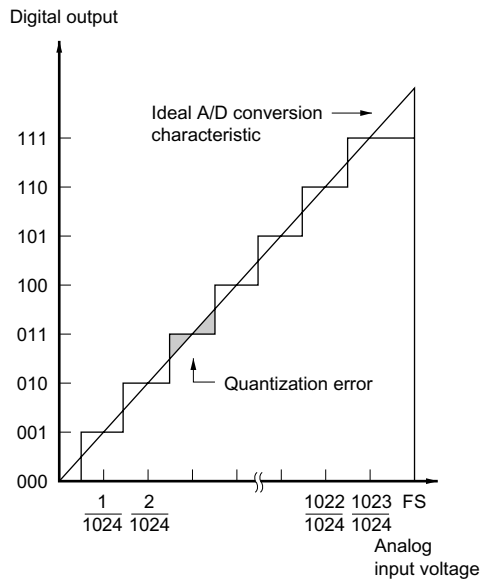
Table 16-5 A/D Converter Interrupt Source

| Name | Interrupt Source         | Interrupt Source Flag | DTC Activation |
|------|--------------------------|-----------------------|----------------|
| ADI  | A/D conversion completed | ADF                   | Possible       |

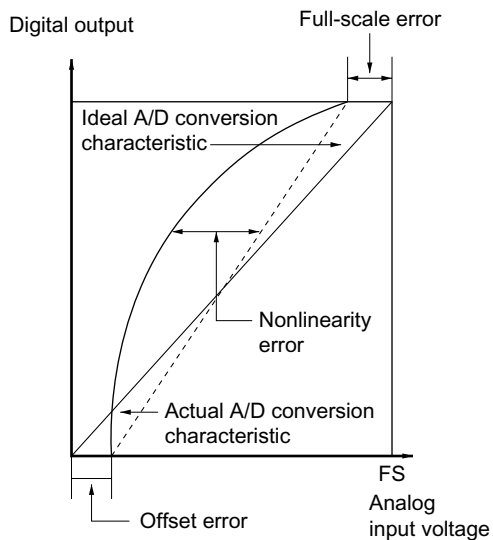
## 16.6 A/D Conversion Precision Definitions

This LSI's A/D conversion precision 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 16-4).
- 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'00) to B'0000000001 (H'01) (see figure 16-5).
- 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'3E) to B'1111111111 (H'3F) (see figure 16-5).
- 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 16-5).
- Absolute precision  
The deviation between the digital value and the analog input value. Includes offset error, full-scale error, quantization error, and nonlinearity error.



**Figure 16-4 A/D Conversion Precision Definitions**



**Figure 16-5 A/D Conversion Precision Definitions**

## 16.7 Usage Notes

### 16.7.1 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 20, Power-Down Modes.

### 16.7.2 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion precision 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 precision. 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 16-6). When converting a high-speed analog signal, a low-impedance buffer should be inserted.

### 16.7.3 Influences on Absolute Precision

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute precision. 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).

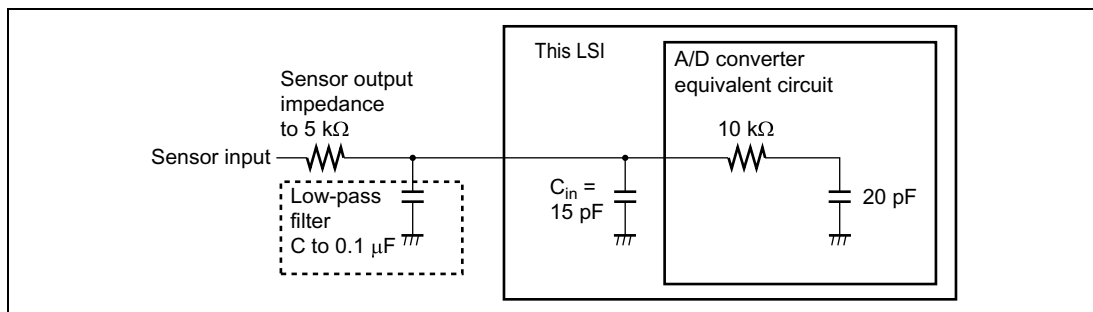


Figure 16-6 Example of Analog Input Circuit

#### 16.7.4 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  $AV_{SS} \leq V_{AN} \leq AV_{CC}$ .

- Relationship between AVcc, AVss and Vcc, Vss

Set  $AV_{SS} = V_{SS}$  as the relationship between AVcc, AVss and Vcc, Vss. If the A/D converter is not used, the AVcc and AVss pins must not be left open.

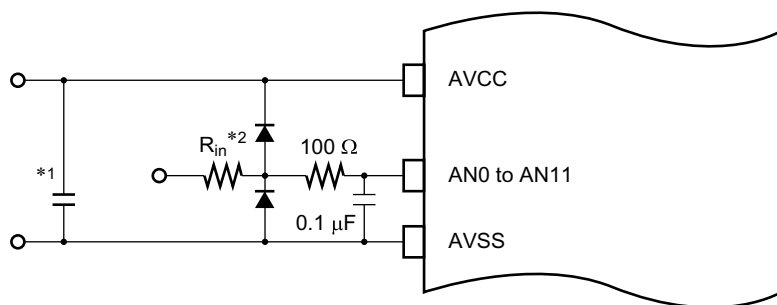
#### 16.7.5 Notes on Board Design

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 AN11), and analog power supply (AVcc) by the analog ground (AVss). Also, the analog ground (AVss) should be connected at one point to a stable digital ground (Vss) on the board.

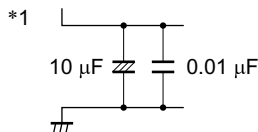
#### 16.7.6 Notes on Noise Countermeasures

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 AN11), between AVcc and AVss, as shown in figure 16-7. Also, the bypass capacitors connected to AVcc and the filter capacitor connected to AN0 to AN11 must be connected to AVss.

If a filter capacitor is connected, the input currents at the analog input pins (AN0 to AN11) 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.

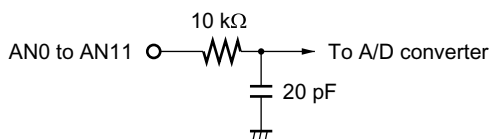


\*2  $R_{in}$ : Input impedance

**Figure 16-7 Example of Analog Input Protection Circuit**

**Table 16-6 Analog Pin Specifications**

| Item                                | Min | Max | Unit |
|-------------------------------------|-----|-----|------|
| Analog input capacitance            | —   | 20  | pF   |
| Permissible signal source impedance | —   | 5   | kΩ   |



Note: Values are reference values.

**Figure 16-8 Analog Input Pin Equivalent Circuit**

## Section 17 RAM

This LSI has 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.

The on-chip RAM can be enabled or disabled by means of the RAME bit in the system control register (SYSCR). For details on the system control register (SYSCR), refer to section 3.2.2, System Control Register (SYSCR).

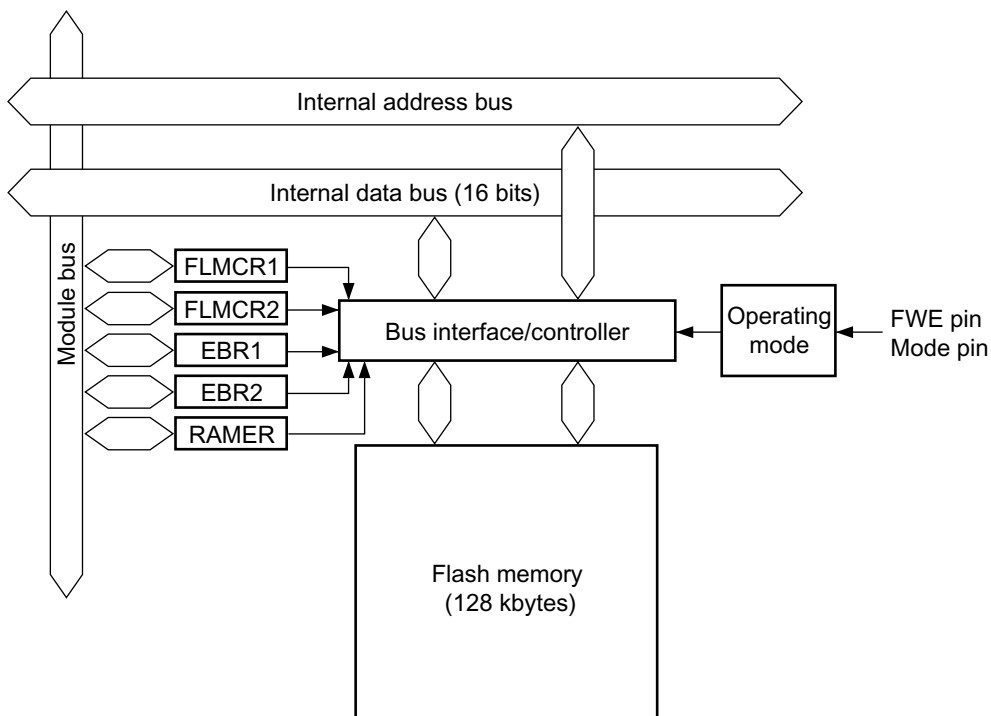


The features of the flash memory are summarized below.

The block diagram of the flash memory is shown in figure 18-1.

## 18.1 Features

- Size: 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 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 up to 100 times.
- Three programming modes
  - Boot mode
  - User mode
  - Programmer 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.
- Programmer mode
  - Flash memory can be programmed/erased in programmer mode using a PROM programmer, as well as in on-board programming mode.
- 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
  - Sets software protection against flash memory programming/erasing.



#### Legend

FLMCR1: Flash memory control register 1  
 FLMCR2: Flash memory control register 2  
 EBR1: Erase block register 1  
 EBR2: Erase block register 2  
 RAMER: RAM emulation register

**Figure 18-1 Block Diagram of Flash Memory**

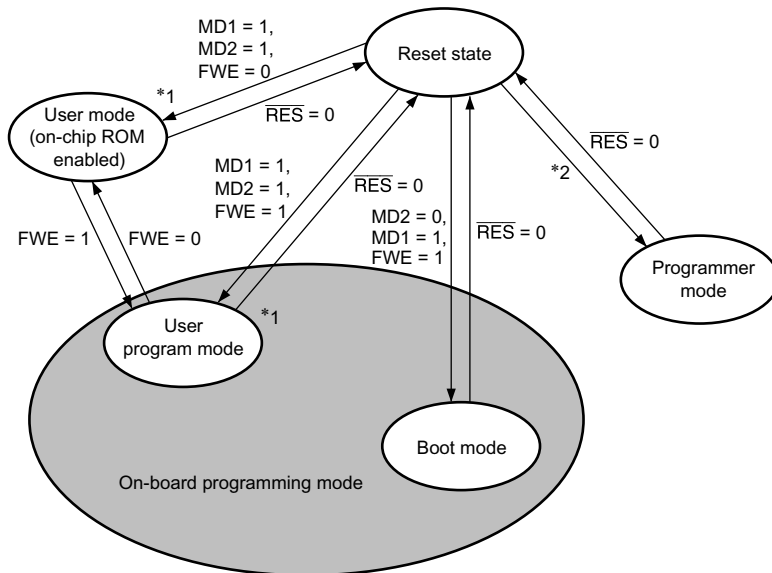
## 18.2 Mode Transitions

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 18-2. 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 18-1.

Figure 18-3 shows the operation flow for boot mode and figure 18-4 shows that for user program mode.



Notes: Only make a transition between user mode and user program mode when the CPU is not accessing the flash memory.

\*1 RAM emulation possible

\*2 This LSI transits to programmer mode by using the dedicated PROM programmer.

**Figure 18-2 Flash Memory State Transitions**

**Table 18-1 Differences between Boot Mode and User Program Mode**

|                              | Boot Mode | User Program Mode |
|------------------------------|-----------|-------------------|
| Total erase                  | Yes       | Yes               |
| Block erase                  | No        | Yes               |
| Programming control program* | (2)       | (1) (2) (3)       |

(1) Erase/erase-verify

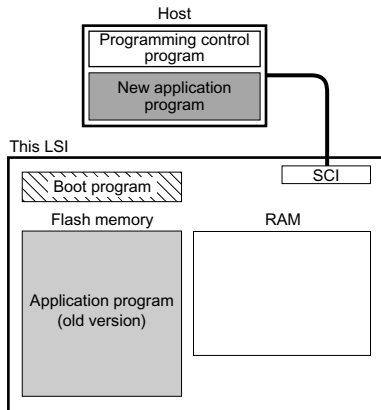
(2) Program/program-verify

(3) Emulation

Note: \* To be provided by the user, in accordance with the recommended algorithm.

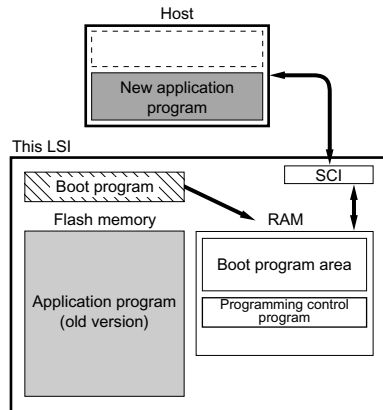
### 1. Initial state

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.



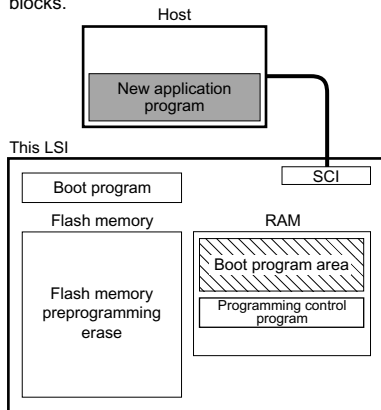
### 2. Programming control program transfer

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.



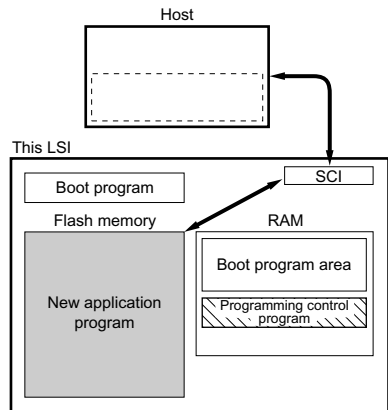
### 3. Flash memory initialization

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.

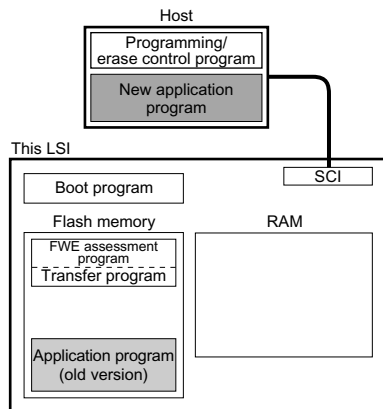


Program execution state

**Figure 18-3 Boot Mode**

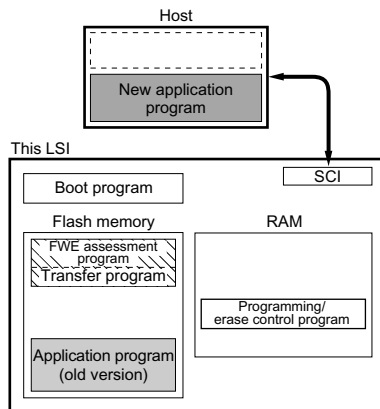
1. Initial state

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.



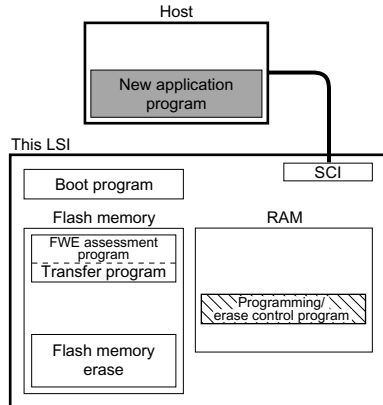
2. Programming/erase control program transfer

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.



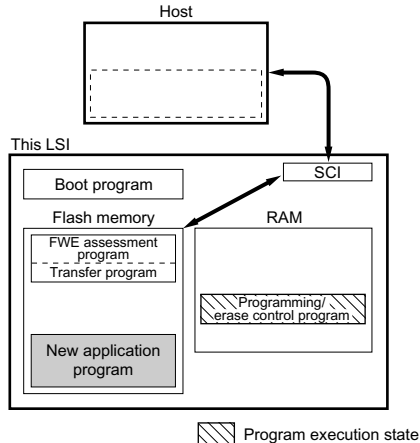
3. Flash memory initialization

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 18-4 User Program Mode

## 18.3 Block Configuration

Figure 18-5 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 flash memory is divided into 32 kbytes (2 blocks), 28 kbytes (1 block), 16 kbytes (1 block), 8 kbytes (2 blocks), and 1 kbyte (4 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<br>Erase unit<br>1 kbyte   | H'000000 | H'000001 | H'000002 | ← Programming unit: 128 bytes → | H'00007F |
|                                |          |          |          |                                 |          |
|                                | H'000380 | H'000381 | H'000382 |                                 | H'0003FF |
|                                |          |          |          |                                 |          |
| EB1<br>Erase unit<br>1 kbyte   | H'000400 | H'000401 | H'000402 | ← Programming unit: 128 bytes → | H'00047F |
|                                |          |          |          |                                 |          |
|                                | H'000780 | H'000781 | H'000782 |                                 | H'0007FF |
|                                |          |          |          |                                 |          |
| EB2<br>Erase unit<br>1 kbyte   | H'000800 | H'000801 | H'000802 | ← Programming unit: 128 bytes → | H'00087F |
|                                |          |          |          |                                 |          |
|                                | H'000B80 | H'000B81 | H'000B82 |                                 | H'000BFF |
|                                |          |          |          |                                 |          |
| EB3<br>Erase unit<br>1 kbyte   | H'000C00 | H'000C01 | H'000C02 | ← Programming unit: 128 bytes → | H'000C7F |
|                                |          |          |          |                                 |          |
|                                | H'000F80 | H'000F81 | H'000F82 |                                 | H'000FFF |
|                                |          |          |          |                                 |          |
| EB4<br>Erase unit<br>28 kbytes | H'001000 | H'001001 | H'001002 | ← Programming unit: 128 bytes → | H'00107F |
|                                |          |          |          |                                 |          |
|                                | H'007F80 | H'007F81 | H'007F82 |                                 | H'007FFF |
|                                |          |          |          |                                 |          |
| EB5<br>Erase unit<br>16 kbytes | H'008000 | H'008001 | H'008002 | ← Programming unit: 128 bytes → | H'00807F |
|                                |          |          |          |                                 |          |
|                                | H'00BF80 | H'00BF81 | H'00BF82 |                                 | H'00BFFF |
|                                |          |          |          |                                 |          |
| EB6<br>Erase unit<br>8 kbytes  | H'00C000 | H'00C001 | H'00C002 | ← Programming unit: 128 bytes → | H'00C07F |
|                                |          |          |          |                                 |          |
|                                | H'00DF80 | H'00DF81 | H'00DF82 |                                 | H'00DFFF |
|                                |          |          |          |                                 |          |
| EB7<br>Erase unit<br>8 kbytes  | H'00E000 | H'00E001 | H'00E002 | ← Programming unit: 128 bytes → | H'00E07F |
|                                |          |          |          |                                 |          |
|                                | H'00FF80 | H'00FF81 | H'00FF82 |                                 | H'00FFFF |
|                                |          |          |          |                                 |          |
| EB8<br>Erase unit<br>32 kbytes | H'010000 | H'010001 | H'010002 | ← Programming unit: 128 bytes → | H'01007F |
|                                |          |          |          |                                 |          |
|                                | H'017F80 | H'017F81 | H'017F82 |                                 | H'017FFF |
|                                |          |          |          |                                 |          |
| EB9<br>Erase unit<br>32 kbytes | H'018000 | H'018001 | H'018002 | ← Programming unit: 128 bytes → | H'01807F |
|                                |          |          |          |                                 |          |
|                                | H'01FF80 | H'01FF81 | H'01FF82 |                                 | H'01FFFF |
|                                |          |          |          |                                 |          |

Figure 18-5 Flash Memory Block Configuration

## 18.4 Input/Output Pins

The flash memory is controlled by means of the pins shown in table 18-2.

**Table 18-2 Pin Configuration**

| Pin Name                | I/O    | Function                                   |
|-------------------------|--------|--------------------------------------------|
| $\overline{\text{RES}}$ | 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             |
| TxD2                    | Output | Serial transmit data output                |
| RxD2                    | Input  | Serial receive data input                  |

## 18.5 Register Descriptions

The flash memory has the following registers. For details on register addresses and register states during each processing, refer to Appendix A, On-Chip I/O Register.

- 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)

### 18.5.1 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 18.8, Flash Memory Programming/Erasing.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | FWE      | —             | R   | Reflects the input level at the FWE pin. It is cleared to 0 when a low level is input to the FWE pin, and set to 1 when a high level is input.                                                                             |
| 6   | SWE      | 0             | R/W | Software Write Enable Bit<br>When this bit is set to 1, flash memory programming/erasing is enabled. When this bit is cleared to 0, other FLMCR1 register bits and all EBR1 and EBR2 bits cannot be set.                   |
| 5   | ESU1     | 0             | R/W | Erase Setup Bit<br>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.                                                              |
| 4   | PSU1     | 0             | R/W | Program Setup Bit<br>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. |
| 3   | EV1      | 0             | R/W | Erase-Verify<br>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.                                                                         |
| 2   | PV1      | 0             | R/W | Program-Verify<br>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.                                                                   |
| 1   | E1       | 0             | R/W | Erase<br>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.                                                      |
| 0   | P1       | 0             | R/W | Program<br>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.                                                |

### 18.5.2 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   | 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.<br><br>See section 18.9.3, Error Protection, for details. |
| 6   | —        | 0             | —   | Reserved                                                                                                                                                                                                                             |
| 5   |          |               |     | These bits are always read as 0.                                                                                                                                                                                                     |
| 4   |          |               |     |                                                                                                                                                                                                                                      |
| 3   |          |               |     |                                                                                                                                                                                                                                      |
| 2   |          |               |     |                                                                                                                                                                                                                                      |
| 1   |          |               |     |                                                                                                                                                                                                                                      |
| 0   |          |               |     |                                                                                                                                                                                                                                      |

### 18.5.3 Erase Block Register 1 (EBR1)

EBR1 specifies the flash memory erase area block. EBR1 is initialized to H'00 when the SWE bit in FLMCR1 is 0. Do not set more than one bit at a time, as this will cause all the bits in EBR1 to be automatically cleared to 0.

| 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.   |

### 18.5.4 Erase Block Register 2 (EBR2)

EBR2 specifies the flash memory erase area block. EBR2 is initialized to H'00 when the SWE bit in FLMCR1 is 0. Do not set more than one bit at a time, as this will cause all the bits in EBR2 to be automatically cleared to 0.

| Bit | Bit Name | Initial Value | R/W | Description                                                                        |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------|
| 7   | —        | 0             | —   | Reserved                                                                           |
| 6   |          |               |     | These bits are always read as 0.                                                   |
| 5   |          |               |     |                                                                                    |
| 4   |          |               |     |                                                                                    |
| 3   |          |               |     |                                                                                    |
| 2   |          |               |     |                                                                                    |
| 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. |

### 18.5.5 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   | —        | 0             | —   | Reserved                                                                                                                                                                                       |
| 6   |          |               |     | These bits are always read as 0.                                                                                                                                                               |
| 5   | —        | 0             | R/W |                                                                                                                                                                                                |
| 4   |          |               |     | Reserved                                                                                                                                                                                       |
|     |          |               |     | Only 0 should be written to these bits.                                                                                                                                                        |
| 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 | When the RAMS bit is set to 1, one of the following flash memory areas are selected to overlap the RAM area of H'FFE000 to H'FFE3FF. The areas correspond with 1-kbyte erase blocks.<br><br>00X: H'000000 to H'0003FF (EB0)<br><br>01X: H'000400 to H'0007FF (EB1)<br><br>10X: H'000800 to H'000BFF (EB2)<br><br>11X: H'000C00 to H'000FFF (EB3)<br><br>Note: X: Don't care |
| 0   | RAM0     | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                             |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                             |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                             |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                             |

## 18.6 On-Board Programming Modes

There are two modes for programming/erasing of the flash memory; boot mode, which enables on-board programming/erasing, and programmer mode, in which programming/erasing is performed with a PROM programmer. On-board programming/erasing can also be performed in user program mode. At reset-start in reset mode, this LSI changes to a mode depending on the MD pin settings and FWE pin setting, as shown in table 18-3. The input level of each pin must be defined four states before the reset ends.

When changing to boot mode, the boot program built into this LSI is initiated. The boot program transfers the programming control program from the externally-connected host to on-chip RAM via SCI\_2. After erasing the entire flash memory, the programming control program is executed. This can be used for programming initial values in the on-board state or for a forcible return when programming/erasing can no longer be done in user program mode. In user program mode, individual blocks can be erased and programmed by branching to the user program/erase control program prepared by the user.

**Table 18-3 Setting On-Board Programming Modes**

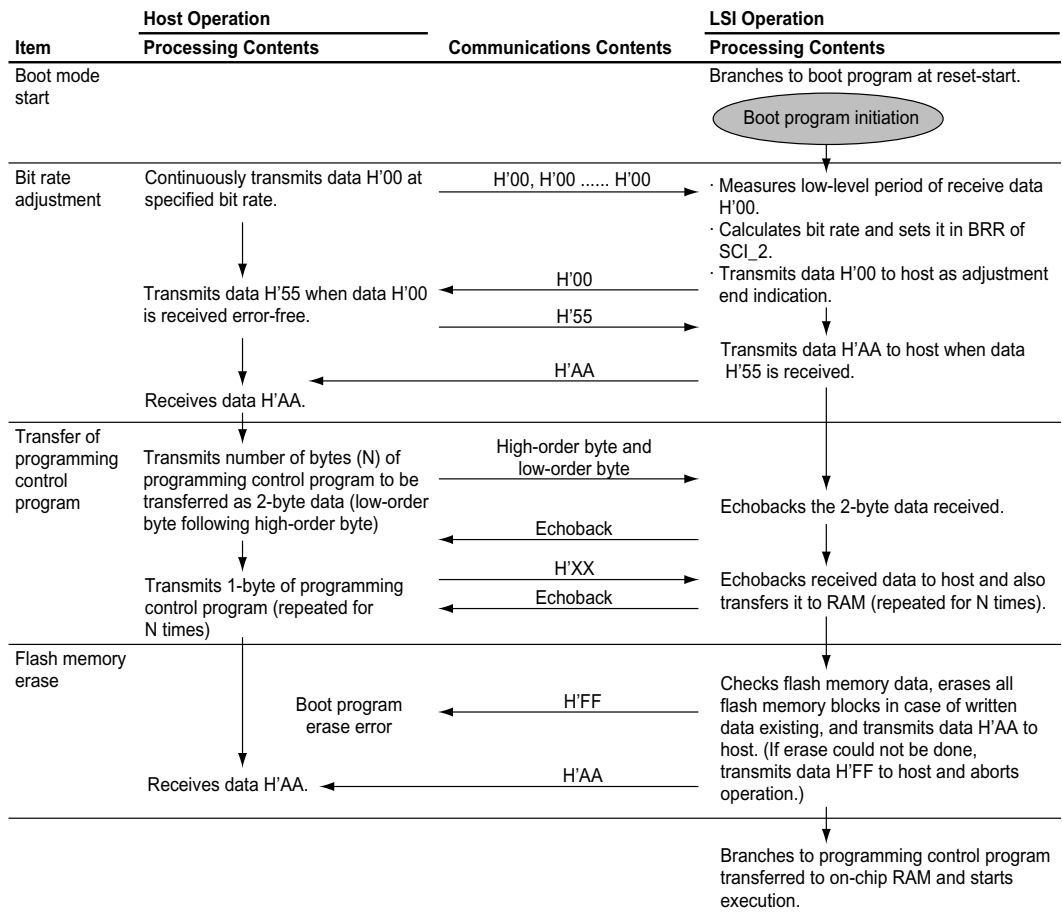
| MD2 | MD1 | MD0 | FWE | LSI State after Reset End |
|-----|-----|-----|-----|---------------------------|
| 1   | 1   | 1   | 1   | User Mode                 |
| 0   | 1   | 1   | 1   | Boot Mode                 |

### 18.6.1 Boot Mode

Table 18-4 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 18.8, Flash Memory Programming/Erasing.
2. SCI\_2 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\_2 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 18-5.
5. In boot mode, a part of the on-chip RAM area is used by the boot program. The area H'FFE800 to H'FFEFBF 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\_2 (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 a reset. End the reset after driving the reset pin low, waiting at least 20 states, and then setting the mode (MD) pins. Boot mode is also cleared when a WDT overflow occurs.
8. Do not change the MD pin input levels in boot mode.
9. All interrupts are disabled during programming or erasing of the flash memory.

**Table 18-4 Boot Mode Operation**

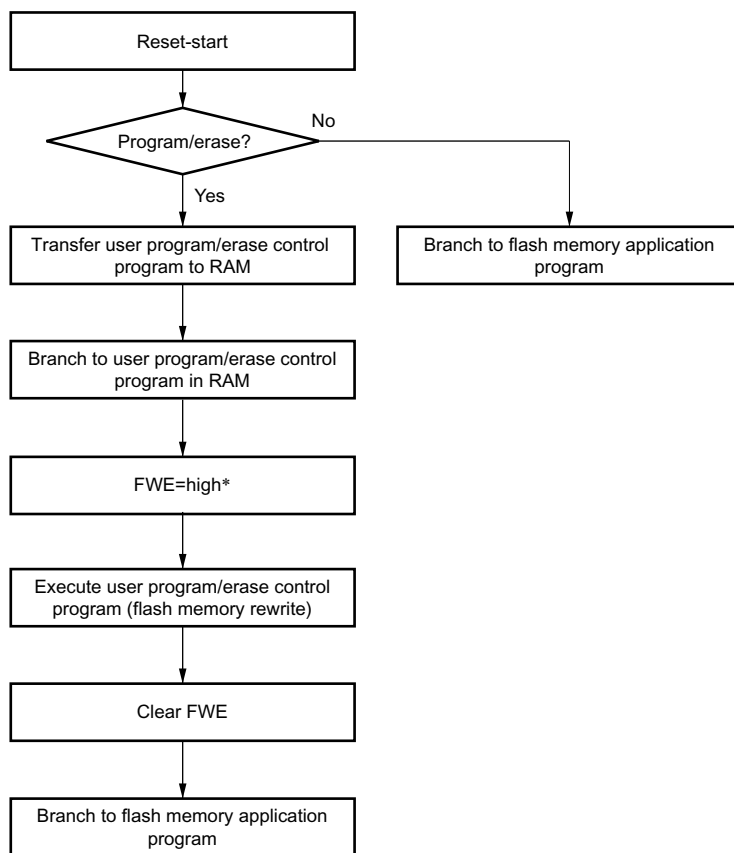


**Table 18-5 System Clock Frequencies for which Automatic Adjustment of LSI Bit Rate is Possible**

| Host Bit Rate | System Clock Frequency Range of LSI |
|---------------|-------------------------------------|
| 19,200 bps    | 20 MHz                              |
| 9,600 bps     | 8 to 20 MHz                         |
| 4,800 bps     | 4 to 20 MHz                         |

## 18.6.2 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 set branching conditions and provide on-board means of supplying programming data. 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 18-6 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 18.8, Flash Memory Programming/Erasing.



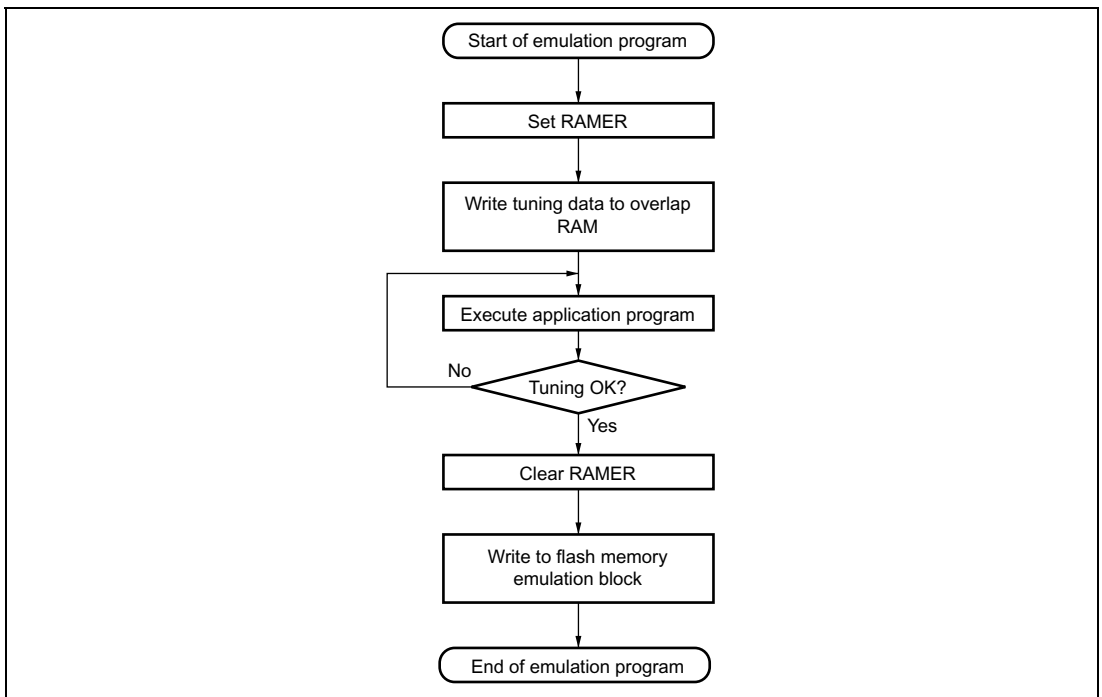
Note: \* Do not constantly apply a high level to the FWE pin. Only apply a high level to the FWE pin when programming or erasing the flash memory. To prevent excessive programming or excessive erasing, while a high level is being applied to the FWE pin, activate the watchdog timer in case of handling CPU runaways.

**Figure 18-6 Programming/Erasing Flowchart Example in User Program Mode**

## 18.7 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 18-7 shows an example of emulation of real-time flash memory programming.

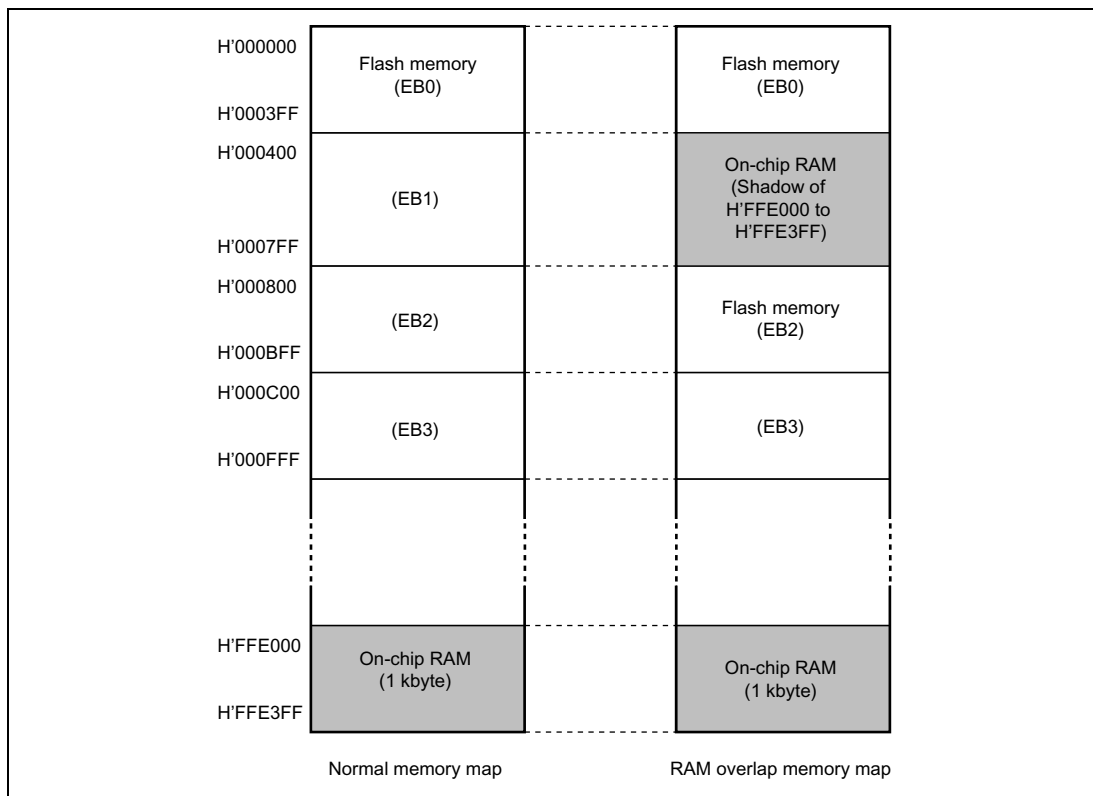
1. Set RAMER to overlap part of RAM onto the area for which real-time programming is required.
2. Emulation is performed using the overlapping RAM.
3. After the program data has been confirmed, the RAMS bit is cleared, thus releasing the RAM overlap.
4. The data written in the overlapping RAM is written into the flash memory space (EB0).



**Figure 18-7 Flowchart for Flash Memory Emulation in RAM**

An example in which flash memory block area EB0 is overlapped is shown in figure 18-8.

1. The RAM area to be overlapped is fixed at a 1-kbyte area in the range H'FFE000 to H'FFE3FF.
2. The flash memory area to overlap is selected by RAMER from a 1-kbyte area of the EB0 to EB3 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 18-8 Example of RAM Overlap Operation**

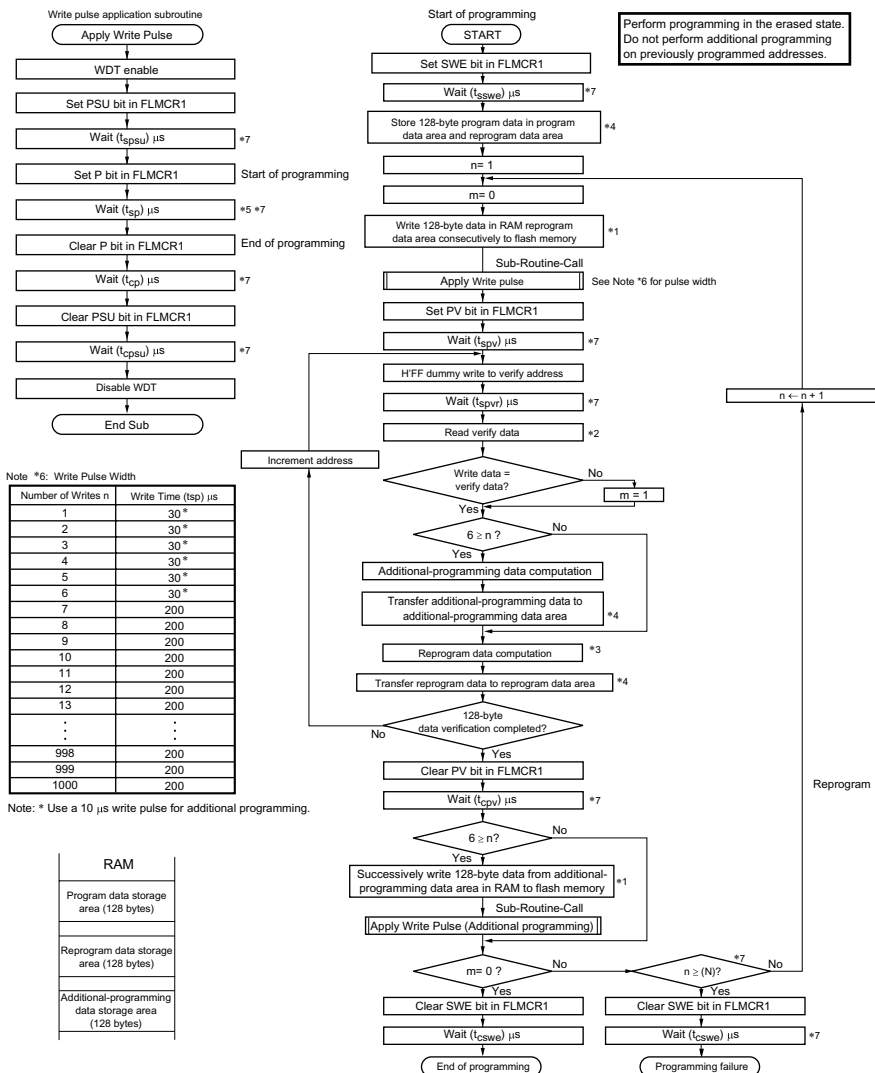
## 18.8 Flash Memory Programming/Erasing

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 18.8.1, Program/Program-Verify and section 18.8.2, Erase/Erase-Verify, respectively.

### 18.8.1 Program/Program-Verify

When writing data or programs to the flash memory, the program/program-verify flowchart shown in Figure 18-9 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 18-9.
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 P bit is set to 1 is the programming time. Figure 18-9 shows the allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runaway, etc. An overflow cycle of approximately 6.6 ms is allowed.
7. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lower 2 bits are B'00. Verify data can be read in longwords 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 1,000.



- Notes: \*1 Data transfer is performed by byte transfer. The lower 8 bits of the first address written to must be H'00 or H'80.  
 \*2 A 128-byte data transfer must be performed even if writing fewer than 128 bytes; in this case, HFF data must be written to the extra addresses.  
 \*3 Verify data is read in 16-bit (word) units.  
 \*4 Reprogram data is determined by the operation shown in the table below (comparison between the data stored in the program data area and the verify data). Bits for which the reprogram data is 0 are programmed in the next reprogramming loop. Therefore, even bits for which programming has been completed will be subjected to programming once again if the result of the subsequent verify operation is NG.  
 \*5 A 128-byte area for storing program data, a 128-byte area for storing reprogram data, and a 128-byte area for storing additional data must be provided in RAM. The contents of the reprogram data area and additional data area are modified as programming proceeds.  
 \*6 A write pulse of 30 μs or 200 μs is applied according to the progress of the programming operation. See Note \*6 for details of the pulse widths. When writing of additional-programming data is executed, a 10 μs write pulse should be applied. Reprogram data 'X' means reprogram data when the write pulse is applied.  
 \*7 The wait times and value of N are shown in section 21.5, Flash Memory characteristics.

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 18-9 Program/Program-Verify Flowchart

### 18.8.2 Erase/Erase-Verify

When erasing flash memory, the erase/erase-verify flowchart shown in figure 18-10 should be followed.

1. Prewriting (setting erase block data to all 0s) is not necessary.
2. Erasing is performed in block units. Make only a single-bit specification in the erase block registers (EBR1 and EBR2). To erase multiple blocks, each block must be erased in turn.
3. The time during which the E bit is set to 1 is the flash memory erase time.
4. The watchdog timer (WDT) is set to prevent overerasing due to program runaway, etc. An overflow cycle of approximately 19.8 ms is allowed.
5. For a dummy write to a verify address, write 1-byte data H'FF to an address whose lower two bits are B'00. Verify data can be read in longwords from the address to which a dummy write was performed.
6. 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 100.

### 18.8.3 Interrupt Handling when Programming/Erasing Flash Memory

All interrupts, including the  $\overline{\text{NMI}}$  interrupt, are disabled while flash memory is being programmed or erased, or while the boot program is executing, for the following three reasons:

1. Interrupt during programming/erasing may cause a violation of the programming or erasing algorithm, with the result that normal operation cannot be assured.
2. If interrupt exception handling starts before the vector address is written or during programming/erasing, a correct vector cannot be fetched and the CPU malfunctions.
3. If an interrupt occurs during boot program execution, normal boot mode sequence cannot be carried out.

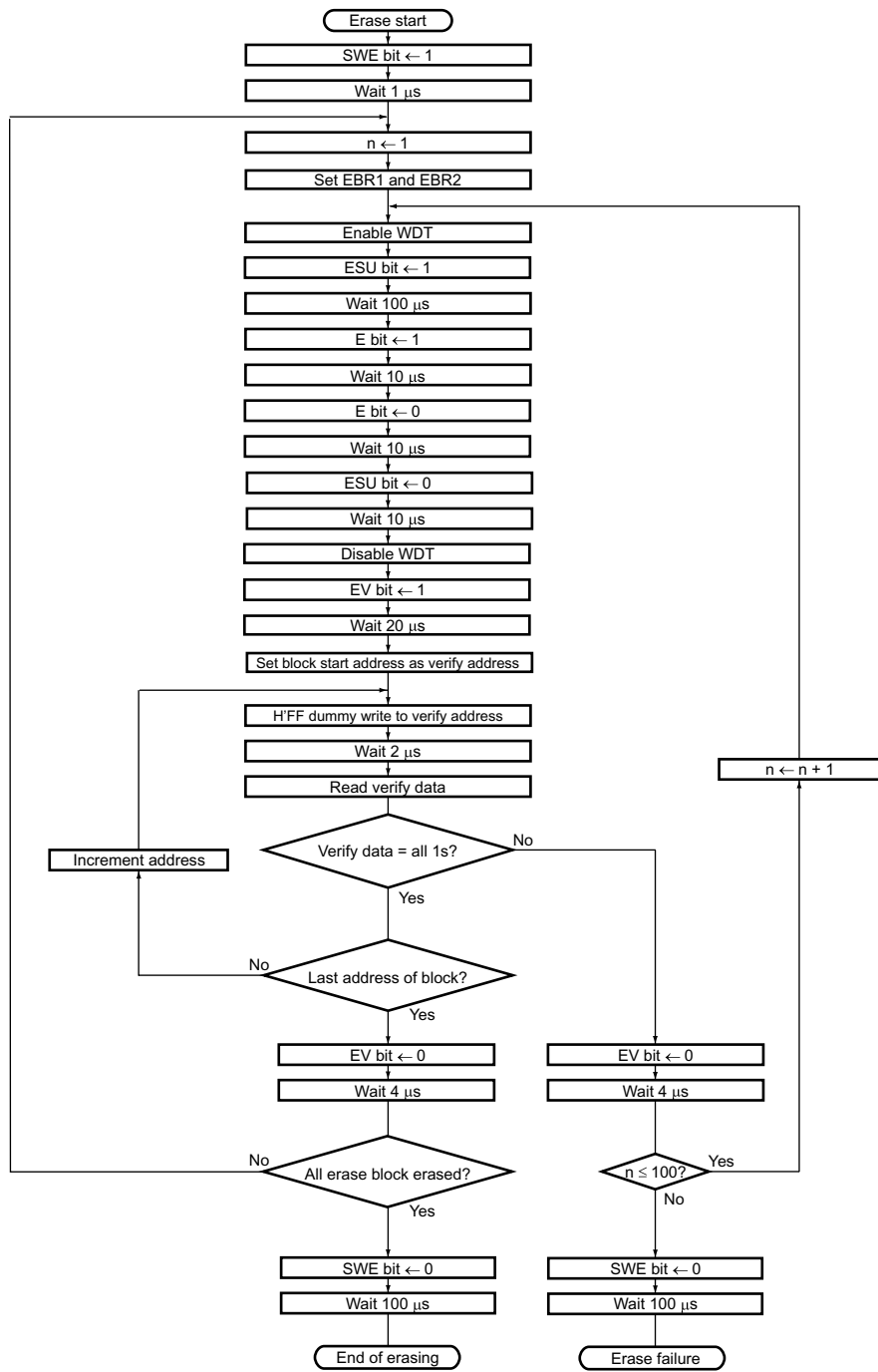


Figure 18-10 Erase/Eraser-Verify Flowchart

## 18.9 Program/Erase Protection

There are three kinds of flash memory program/erase protection; hardware protection, software protection, and error protection.

### 18.9.1 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), and erase block register 1 (EBR1) 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.

### 18.9.2 Software Protection

Software protection can be implemented against programming/erasing of all flash memory blocks by clearing the SWE 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 (EBR1), erase protection can be set for individual blocks. When EBR1 is set to H'00, erase protection is set for all blocks.

### 18.9.3 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

The FLMCR1, FLMCR2, and EBR1 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, PV and EV bit setting is enabled, and a transition can be made to verify mode. Error protection can be cleared only by a power-on reset.

## 18.10 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 that supports the Hitachi 128-kbyte flash memory on-chip MCU device type (FZTAT128V5A).

## 18.11 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.
- Standby mode  
All flash memory circuits are halted.

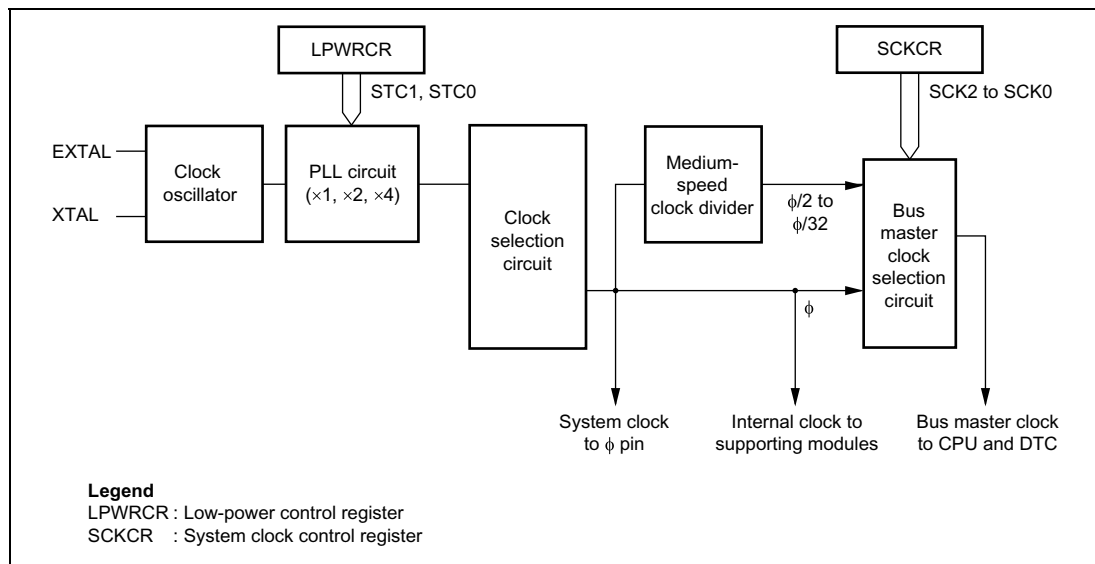
Table 18-6 shows the correspondence between the operating modes of the H8S/2612 Series and the flash memory. When the flash memory returns to its normal operating state from standby mode, a period to stabilize the power supply circuits that were stopped is needed. When the flash memory returns to its normal operating state, bits STS2 to STS0 in SBYCR must be set to provide a wait time of at least 20  $\mu$ s, even when the external clock is being used.

**Table 18-6 Flash Memory Operating States**

| LSI Operating State | Flash Memory Operating State |
|---------------------|------------------------------|
| Active mode         | Normal operating mode        |
| Standby mode        | Standby mode                 |

## Section 19 Clock Pulse Generator

This LSI has an on-chip clock pulse generator that generates the system clock ( $\phi$ ), the bus master clock, and internal clocks. The clock pulse generator consists of an oscillator, PLL circuit, clock selection circuit, medium-speed clock divider, and bus master clock selection circuit. A block diagram of the clock pulse generator is shown in figure 19-1.



**Figure 19-1 Block Diagram of Clock Pulse Generator**

The frequency can be changed by means of the PLL circuit. Frequency changes are performed by software by settings in the low-power control register (LPWRCR) and system clock control register (SCKCR).

## 19.1 Register Descriptions

The on-chip clock pulse generator has the following registers. For details on register addresses, refer to Appendix A, On-Chip I/O Register.

- System clock control register (SCKCR)
- Low-power control register (LPWRCR)

### 19.1.1 System Clock Control Register (SCKCR)

SCKCR performs  $\phi$  clock output control, selection of operation when the PLL circuit frequency multiplication factor is changed, and medium-speed mode control.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                   |
|-----|----------|---------------|-----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | PSTOP    | 0             | R/W | $\phi$ Clock Output Disable<br>Controls $\phi$ output.<br>High-speed Mode, Medium-Speed Mode<br>0: $\phi$ output<br>1: Fixed high<br>Sleep Mode<br>0: $\phi$ output<br>1: Fixed high<br>Software Standby Mode<br>0: Fixed high<br>1: Fixed high<br>Hardware Standby Mode<br>0: High impedance<br>1: High impedance                            |
| 6   | —        | 0             | —   | Reserved                                                                                                                                                                                                                                                                                                                                      |
| 5   | —        | 0             | —   | These bits are always read as 0.                                                                                                                                                                                                                                                                                                              |
| 4   | —        | 0             | —   |                                                                                                                                                                                                                                                                                                                                               |
| 3   | STCS     | 0             | R/W | Frequency Multiplication Factor Switching Mode Select<br>Selects the operation when the PLL circuit frequency multiplication factor is changed.<br>0: Specified multiplication factor is valid after transition to software standby mode<br>1: Specified multiplication factor is valid immediately after STC1 bit and STC0 bit are rewritten |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                          |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 2   | SCK2     | 0             | R/W | System Clock Select 0 to 2                                                                                                                                                                                                                           |
| 1   | SCK1     | 0             | R/W | These bits select the bus master clock.                                                                                                                                                                                                              |
| 0   | SCK0     | 0             | R/W | 000: High-speed mode<br>001: Medium-speed clock is $\phi/2$<br>010: Medium-speed clock is $\phi/4$<br>011: Medium-speed clock is $\phi/8$<br>100: Medium-speed clock is $\phi/16$<br>101: Medium-speed clock is $\phi/32$<br>11X: Setting prohibited |

#### Legend

X: Don't care

### 19.1.2 Low-Power Control Register (LPWRCR)

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                  |
|-----|----------|---------------|-----|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | —        | 0             | —   | Reserved                                                                                                                                                     |
| 6   | —        | 0             | —   | Only 0 should be written to these bits.                                                                                                                      |
| 5   | —        | 0             | —   |                                                                                                                                                              |
| 4   | —        | 0             | —   |                                                                                                                                                              |
| 3   | —        | 0             | R/W | These bits can be read and write, but should not be set to 1.                                                                                                |
| 2   | —        | 0             | R/W |                                                                                                                                                              |
| 1   | STC1     | 0             | R/W | Frequency Multiplication Factor                                                                                                                              |
| 0   | STC0     | 0             | R/W | The STC bits specify the frequency multiplication factor of the PLL circuit.<br>00: $\times 1$<br>01: $\times 2$<br>10: $\times 4$<br>11: Setting prohibited |

# 19.2 Oscillator

Clock pulses can be supplied by connecting a crystal resonator, or by input of an external clock. In either case, the input clock should not exceed 20 MHz.

## 19.2.1 Connecting a Crystal Resonator

Circuit Configuration: A crystal resonator can be connected as shown in the example in figure 19-2. Select the damping resistance  $R_d$  according to table 19-1. An AT-cut parallel-resonance crystal should be used.

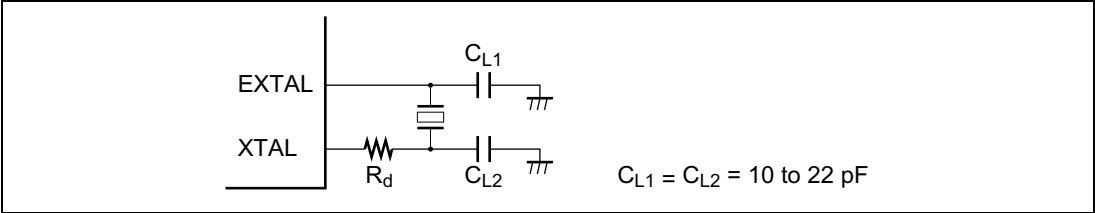


Figure 19-2 Connection of Crystal Resonator (Example)

Table 19-1 Damping Resistance Value

| Frequency (MHz)    | 4   | 8   | 12 | 16 | 20 |
|--------------------|-----|-----|----|----|----|
| $R_d$ ( $\Omega$ ) | 500 | 200 | 0  | 0  | 0  |

Figure 19-3 shows the equivalent circuit of the crystal resonator. Use a crystal resonator that has the characteristics shown in table 19-2.

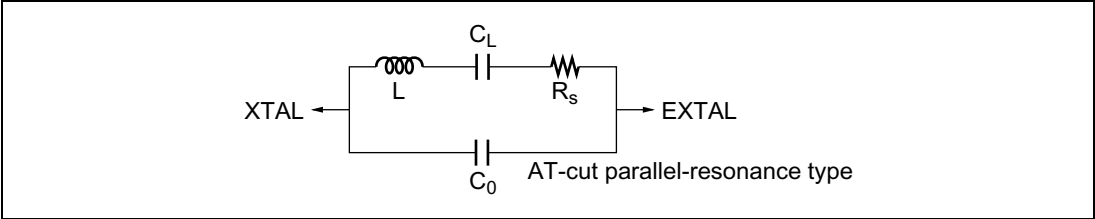


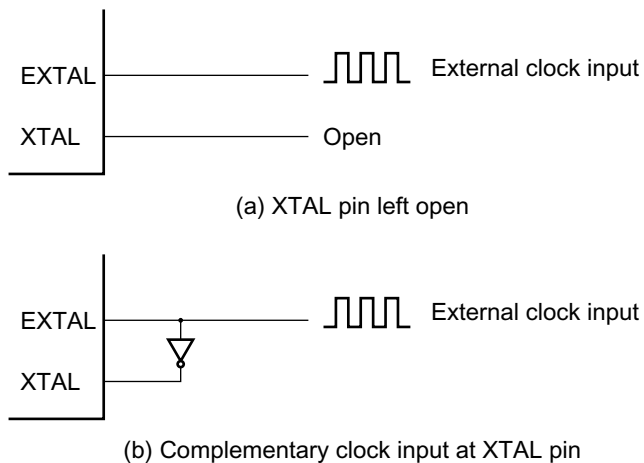
Figure 19-3 Crystal Resonator Equivalent Circuit

Table 19-2 Crystal Resonator Characteristics

| Frequency (MHz)        | 4   | 8  | 12 | 16 | 20 |
|------------------------|-----|----|----|----|----|
| $R_s$ max ( $\Omega$ ) | 120 | 80 | 60 | 50 | 40 |
| $C_0$ max (pF)         | 7   | 7  | 7  | 7  | 7  |

## 19.2.2 External Clock Input

Circuit Configuration: An external clock signal can be input as shown in the examples in figure 19-4. 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.

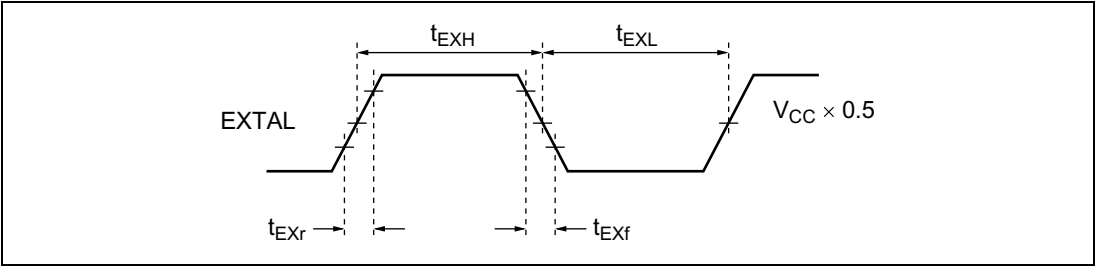


**Figure 19-4 External Clock Input (Examples)**

Table 19-3 shows the input conditions for the external clock.

**Table 19-3 External Clock Input Conditions**

| Item                                  | Symbol    | $V_{CC} = 5.0\text{ V} \pm 10\%$ |     | Unit      | Test Conditions          |             |
|---------------------------------------|-----------|----------------------------------|-----|-----------|--------------------------|-------------|
|                                       |           | Min                              | Max |           |                          |             |
| External clock input low pulse width  | $t_{EXL}$ | 15                               | —   | ns        | Figure 19-5              |             |
| External clock input high pulse width | $t_{EXH}$ | 15                               | —   | ns        |                          |             |
| External clock rise time              | $t_{EXr}$ | —                                | 5   | ns        |                          |             |
| External clock fall time              | $t_{EXf}$ | —                                | 5   | ns        |                          |             |
| Clock low pulse width level           | $t_{CL}$  | 0.4                              | 0.6 | $t_{cyc}$ | $\phi \geq 5\text{ MHz}$ | Figure 21-2 |
|                                       |           | 80                               | —   | ns        | $\phi < 5\text{ MHz}$    |             |
| Clock high pulse width level          | $t_{CH}$  | 0.4                              | 0.6 | $t_{cyc}$ | $\phi \geq 5\text{ MHz}$ |             |
|                                       |           | 80                               | —   | ns        | $\phi < 5\text{ MHz}$    |             |



**Figure 19-5 External Clock Input Timing**

## 19.3 PLL Circuit

The PLL circuit multiplies the frequency of the clock from the oscillator by a factor of 1, 2, or 4. The multiplication factor is set by the STC0 bit and the STC1 bit in LPWRCR. The phase of the rising edge of the internal clock is controlled so as to match that at the EXTAL pin.

When the multiplication factor of the PLL circuit is changed, the operation varies according to the setting of the STCS bit in SCKCR.

When STCS = 0, the setting becomes valid after a transition to software standby mode. The transition time count is performed in accordance with the setting of bits STS0 to STS2 in SBYCR. For details on SBYCR, refer to section 20.1.1, Standby Control Register (SBYCR).

1. The initial PLL circuit multiplication factor is 1.
2. STS0 to STS2 are set to give the specified transition time.
3. The target value is set in STC0 and STC1, and a transition is made to software standby mode.
4. The clock pulse generator stops and the value set in STC0 and STC1 becomes valid.
5. Software standby mode is cleared, and a transition time is secured in accordance with the setting in STS0 to STS2.
6. After the set transition time has elapsed, this LSI resumes operation using the target multiplication factor.

If a PC break is set for the SLEEP instruction, software standby mode is entered and break exception handling is executed after the oscillation stabilization time. In this case, the instruction following the SLEEP instruction is executed after execution of the RTE instruction. When STCS = 1, this LSI operates on the changed multiplication factor immediately after bits STC0 and STC1 are rewritten.

## 19.4 Medium-Speed Clock Divider

The medium-speed clock divider divides the system clock to generate  $\phi/2$ ,  $\phi/4$ ,  $\phi/8$ ,  $\phi/16$ , and  $\phi/32$ .

## 19.5 Bus Master Clock Selection Circuit

The bus master clock selection circuit selects the clock supplied to the bus master by setting the bits SCK 2 to SCK 0 in SCKCR. The bus master clock can be selected from high-speed mode, or medium-speed clocks ( $\phi/2$ ,  $\phi/4$ ,  $\phi/8$ ,  $\phi/16$ ,  $\phi/32$ ).

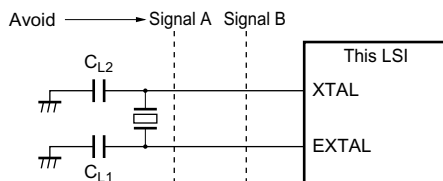
## 19.6 Usage Notes

### 19.6.1 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.

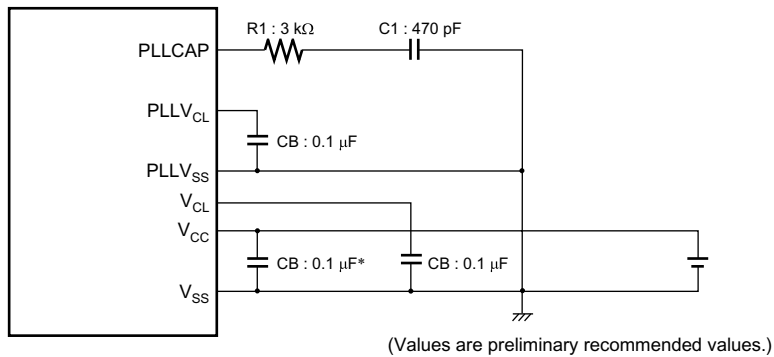
### 19.6.2 Note on Board Design

When designing the board, place the crystal resonator and its load capacitors as close as possible to the XTAL and EXTAL pins. Other signal lines should be routed away from the oscillator circuit, as shown in figure 19-6. This is to prevent induction from interfering with correct oscillation.



**Figure 19-6 Note on Board Design of Oscillator Circuit**

Figure 19-7 shows external circuitry recommended to be provided around the PLL circuit. Place oscillation stabilization capacitor C1 and resistor R1 close to the PLLCAP pin, and ensure that no other signal lines cross this line. Separate PLLV<sub>CL</sub> and PLLV<sub>SS</sub> from the other V<sub>CC</sub> and V<sub>SS</sub> lines at the board power supply source, and be sure to insert bypass capacitors CB close to the pins.



Note: \* CB are laminated ceramic.

**Figure 19-7 External Circuitry Recommended for PLL Circuit**



## Section 20 Power-Down Modes

In addition to the normal program execution state, this LSI has five 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 supporting modules, and so on.

This LSI's operating modes are as follows:

- (1) High-speed mode
- (2) Medium-speed mode
- (3) Sleep mode
- (4) Module stop mode
- (5) Software standby mode
- (6) Hardware standby mode

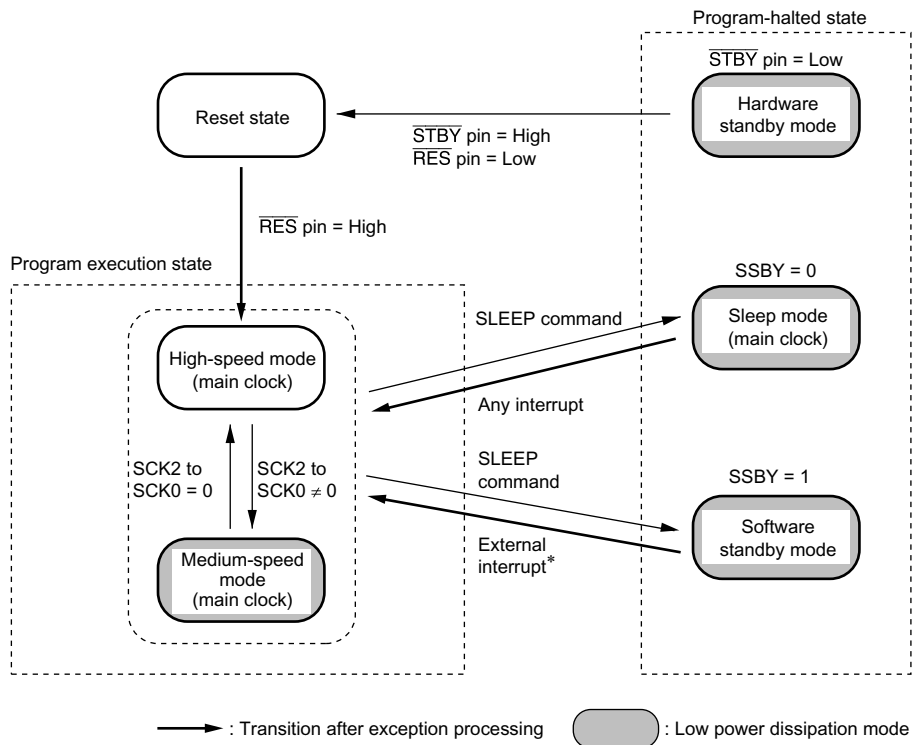
(2) to (6) are power-down modes. Sleep mode is a CPU state, medium-speed mode is a CPU and bus master state, and module stop mode is an internal peripheral function (including bus masters other than the CPU) state. Some of these states can be combined.

After a reset, the LSI is in high-speed mode.

Figure 20-1 show a mode transition. Table 20-1 shows the conditions of transition between modes when executing the SLEEP instruction and the state after transition back from low power mode due to an interrupt. Table 20-2 shows the internal state of the LSI in each mode.

**Table 20-1 Low Power Dissipation Mode Transition Conditions**

| Pre-Transition State        | Status of Control Bit at Transition | State after Transition Invoked by SLEEP Command | State after Transition Back from Low Power Mode Invoked by Interrupt |
|-----------------------------|-------------------------------------|-------------------------------------------------|----------------------------------------------------------------------|
|                             | SSBY                                |                                                 |                                                                      |
| High-speed/<br>Medium-speed | 0                                   | Sleep                                           | High-speed/Medium-speed                                              |
|                             | 1                                   | Software standby                                | High-speed/Medium-speed                                              |



Note: \* NMI and IRQ0 to IRQ5

- 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  $\overline{RES}$  is driven low.
- From any state, a transition to hardware standby mode occurs when  $\overline{STBY}$  is driven low.

**Figure 20-1 Mode Transition Diagram**

**Table 20-2 LSI Internal States in Each Mode**

| Function                     |              | High-Speed  | Medium-Speed           | Sleep             | Module Stop                 | Software Standby  | Hardware Standby   |
|------------------------------|--------------|-------------|------------------------|-------------------|-----------------------------|-------------------|--------------------|
| System clock pulse generator |              | Functioning | Functioning            | Functioning       | Functioning                 | Halted            | Halted             |
| CPU                          | Instructions | Functioning | Medium-speed operation | Halted (retained) | High/medium-speed operation | Halted (retained) | Halted (undefined) |
|                              | Registers    |             |                        |                   |                             |                   |                    |
| External interrupts          | NMI          | Functioning | Functioning            | Functioning       | Functioning                 | Functioning       | Halted             |
|                              | IRQ0–IRQ5    |             |                        |                   |                             |                   |                    |
| Peripheral functions         | PBC          | Functioning | Medium-speed operation | Functioning       | Halted (retained)           | Halted (retained) | Halted (reset)     |
|                              | DTC          |             |                        |                   |                             |                   |                    |
|                              | I/O          | Functioning | Functioning            | Functioning       | Functioning                 | Retained          | High impedance     |
|                              | TPU          | Functioning | Functioning            | Functioning       | Halted (retained)           | Halted (retained) | Halted (reset)     |
|                              | MMT/POE      |             |                        |                   |                             |                   |                    |
|                              | PPG          |             |                        |                   |                             |                   |                    |
|                              | WDT          | Functioning | Functioning            | Functioning       | Functioning                 | Halted (retained) | Halted (reset)     |
|                              | SCI          | Functioning | Functioning            | Functioning       | Halted (reset)              | Halted (reset)    | Halted (reset)     |
|                              | HCAN         |             |                        |                   |                             |                   |                    |
|                              | A/D          |             |                        |                   |                             |                   |                    |
|                              | RAM          | Functioning | Medium-speed operation | Functioning (DTC) | Functioning                 | Retained          | Retained           |

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).

## 20.1 Register Descriptions

Registers related to the power down mode are shown below. For details on the system clock control register (SCKCR), refer to section 19.1.1, System Clock Control Register (SCKCR). For details on register addresses and register states during each process, refer to Appendix A, On-Chip I/O Register.

- System clock control register (SCKCR)
- Standby control register (SBYCR)
- Module stop control register A (MSTPCRA)
- Module stop control register B (MSTPCRB)
- Module stop control register C (MSTPCRC)

### 20.1.1 Standby Control Register (SBYCR)

SBYCR is an 8-bit readable/writable register that performs software standby mode control.

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----|----------|---------------|-----|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 7   | SSBY     | 0             | R/W | <p>Software Standby</p> <p>This bit specifies the transition mode after executing the SLEEP instruction</p> <p>0: Shifts to sleep mode when the SLEEP instruction is executed</p> <p>1: Shifts to software standby mode when the SLEEP instruction is executed</p> <p>This bit does not change when clearing the software standby mode by using external interrupts and shifting to normal operation. This bit should be written with 0 when clearing.</p> |

| Bit | Bit Name | Initial Value | R/W | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|-----|----------|---------------|-----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 6   | STS2     | 0             | R/W | Standby Timer Select 0 to 2                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| 5   | STS1     | 0             | R/W | These bits select the MCU wait time for clock stabilization when software standby mode is cancelled by an external interrupt. With a crystal oscillator (table 20-3), select a wait time of 8ms (oscillation stabilization time) or more, depending on the operating frequency. With an external clock, select a wait time of 2 ms or more.<br><br>000: Standby time = 8192 states<br>001: Standby time = 16384 states<br>010: Standby time = 32768 states<br>011: Standby time = 65536 states<br>100: Standby time = 131072 states<br>101: Standby time = 262144 states<br>110: Reserved<br>111: Standby time = 16 states |
| 4   | STS0     | 0             | R/W |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|     |          |               |     |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| 3   | —        | 1             | R/W | Reserved<br><br>Only 1 should be written to this bit.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| 2   | —        | 0             | —   | Reserved                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| 1   | —        | 0             | —   | These bits are always read as 0 and cannot be modified.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| 0   | —        | 0             | —   |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |

### 20.1.2 Module Stop Control Registers A to C (MSTPCRA to MSTPCRC)

MSTPCR is comprised of three 8-bit readable/writable registers, and performs module stop mode control. Setting a bit to 1 causes the corresponding module to enter module stop mode. Clearing the bit to 0 clears the module stop mode.

#### MSTPCRA

| Bit | Bit Name | Initial Value | R/W | Module                             |
|-----|----------|---------------|-----|------------------------------------|
| 7   | MSTPA7*  | 0             | R/W |                                    |
| 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 |                                    |
| 3   | MSTPA3   | 1             | R/W | Programmable pulse generator (PPG) |
| 2   | MSTPA2*  | 1             | R/W |                                    |
| 1   | MSTPA1   | 1             | R/W | A/D converter                      |
| 0   | MSTPA0*  | 1             | R/W |                                    |

#### MSTPCRB

| Bit | Bit Name | Initial Value | R/W | Module                                  |
|-----|----------|---------------|-----|-----------------------------------------|
| 7   | MSTPB7   | 1             | R/W | Serial communication interface 0 (SCI0) |
| 6   | MSTPB6   | 1             | R/W | Serial communication interface 1 (SCI1) |
| 5   | MSTPB5   | 1             | R/W | Serial communication interface 2 (SCI2) |
| 4   | MSTPB4*  | 1             | R/W |                                         |
| 3   | MSTPB3*  | 1             | R/W |                                         |
| 2   | MSTPB2*  | 1             | R/W |                                         |
| 1   | MSTPB1*  | 1             | R/W |                                         |
| 0   | MSTPB0*  | 1             | R/W |                                         |

## MSTPCRC

| Bit | Bit Name | Initial Value | R/W | Module                                 |
|-----|----------|---------------|-----|----------------------------------------|
| 7   | MSTPC7*  | 1             | R/W |                                        |
| 6   | MSTPC6*  | 1             | R/W |                                        |
| 5   | MSTPC5*  | 1             | R/W |                                        |
| 4   | MSTPC4   | 1             | R/W | PC break controller (PBC)              |
| 3   | MSTPC3   | 1             | R/W | Hitachi Controller Area Network (HCAN) |
| 2   | MSTPC2   | 1             | R/W | Motor Management Timer (MMT)           |
| 1   | MSTPC1*  | 1             | R/W |                                        |
| 0   | MSTPC0*  | 1             | R/W |                                        |

Note: \*MSTPA7 is a readable/writable bit with an initial value of 0 and should always be written with 0.

MSTPA4, MSTPA2, MSTPA0, MSTPB4 to MSTPB0, MSTPC7 to MSTPC5, MSTPC1, and MSTPC0 are readable/writable bits with an initial value of 1 and should always be written with 1.

## 20.2 Medium-Speed Mode

When the SCK0 to SCK2 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 SCK0 to SCK2 bits. Bus masters (DTC) other than the CPU also operate in medium-speed mode. On-chip supporting modules other than bus masters always operate on the high-speed clock ( $\phi$ ).

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 SCK0 to SCK2 to 0. A transition is made to high-speed mode and medium-speed mode is cleared at the end of the current bus cycle.

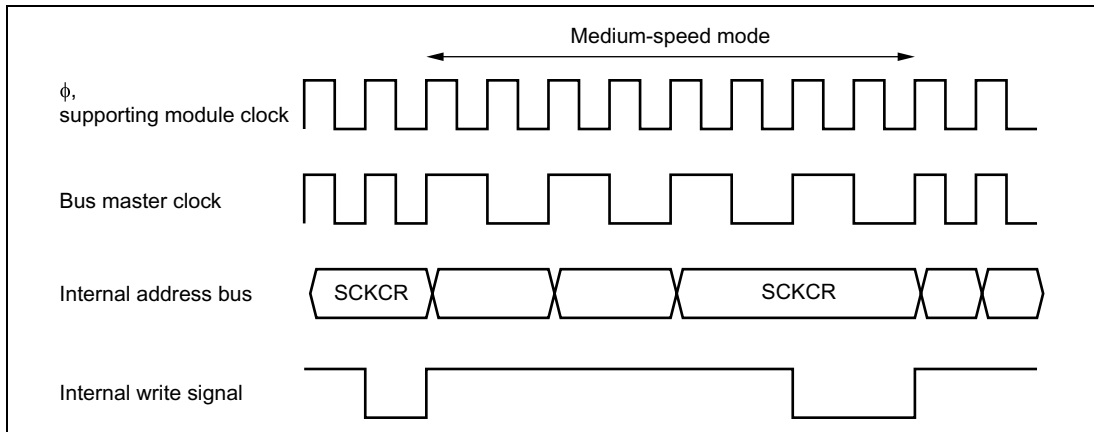
If a SLEEP instruction is executed when the SSBY bit in SBYCR is cleared to 0, a transition is made to sleep mode. When sleep mode is cleared by an interrupt, medium-speed mode is restored.

When the SLEEP instruction is executed with the SSBY bit = 1, operation shifts to the software standby mode. When software standby mode is cleared by an external interrupt, medium-speed mode is restored.

When the  $\overline{\text{RES}}$  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 20-2 shows the timing for transition to and clearance of medium-speed mode.



**Figure 20-2 Medium-Speed Mode Transition and Clearance Timing**

## 20.3 Sleep Mode

### 20.3.1 Transition to Sleep Mode

If SLEEP instruction is executed when the SBYCR SSBY bit = 0, the CPU enters the sleep mode. In sleep mode, CPU operation stops, however the contents of the CPU's internal registers are retained. Other supporting modules do not stop.

### 20.3.2 Clearing Sleep Mode

Sleep mode is cleared by any interrupt, or signals at the  $\overline{\text{RES}}$ , or  $\overline{\text{STBY}}$  pins.

- **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 if interrupts other than NMI are masked by the CPU.
- **Exiting Sleep Mode by  $\overline{\text{RES}}$  pin:**  
Setting the  $\overline{\text{RES}}$  pin Low selects the reset state. After the stipulated reset input duration, driving the  $\overline{\text{RES}}$  pin High restart 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.

## 20.4 Software Standby Mode

### 20.4.1 Transition to Software Standby Mode

A transition is made to software standby mode if the SLEEP instruction is executed when the SBYCR SSBY bit is set to 1. In this mode, the CPU, on-chip supporting modules, and oscillator, all stop. However, the contents of the CPU's internal registers, on-chip RAM data, and the states of on-chip supporting modules other than the SCI, 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.

### 20.4.2 Clearing Software Standby Mode

Software standby mode is cleared by an external interrupt (NMI pin, or pins  $\overline{\text{IRQ0}}$  to  $\overline{\text{IRQ5}}$ ), or by means of the  $\overline{\text{RES}}$  pin or  $\overline{\text{STBY}}$  pin.

- Clearing with an interrupt:

When an NMI or IRQ0 to IRQ5 interrupt request signal is input, clock oscillation starts, and after the time set in bits STS0 to STS2 in SBYCR has elapsed, stable clocks are supplied to the entire chip, software standby mode is cleared, and interrupt exception handling is started.

When clearing software standby mode with an IRQ0 to IRQ5 interrupt, set the corresponding enable bit to 1 and ensure that no interrupt with a higher priority than interrupts IRQ0 to IRQ5 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:

When the  $\overline{\text{RES}}$  pin is driven low, clock oscillation is started. At the same time as clock oscillation starts, clocks are supplied to the entire chip. Note that the  $\overline{\text{RES}}$  pin must be held low until clock oscillation stabilizes. When the  $\overline{\text{RES}}$  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.

### 20.4.3 Setting Oscillation Stabilization Time after Clearing Software Standby Mode

Bits STS2 to STS0 in SBYCR should be set as described below.

- Using a Crystal Oscillator:  
Set bits STS0 to STS2 so that the standby time is at least 8 ms (the oscillation stabilization time).  
Table 20-3 shows the standby times for different operating frequencies and settings of bits STS0 to STS2.
- Using an External Clock:  
The PLL circuit requires a time for stabilization. Set bits STS0 to STS2 so that the standby time is at least 2 ms.

**Table 20-3 Oscillation Stabilization Time Settings**

| STS2 | STS1 | STS0 | Standby Time  | 20 MHz | 16 MHz | 12 MHz | 10 MHz | 8 MHz | 6 MHz | 4 MHz | Unit |
|------|------|------|---------------|--------|--------|--------|--------|-------|-------|-------|------|
| 0    | 0    | 0    | 8192 states   | 0.41   | 0.51   | 0.68   | 0.8    | 1.0   | 1.3   | 2.0   | ms   |
|      |      | 1    | 16384 states  | 0.82   | 1.0    | 1.3    | 1.6    | 2.0   | 2.7   | 4.1   |      |
|      | 1    | 0    | 32768 states  | 1.6    | 2.0    | 2.7    | 3.3    | 4.1   | 5.5   | 8.2   |      |
|      |      | 1    | 65536 states  | 3.3    | 4.1    | 5.5    | 6.6    | 8.2   | 10.9  | 16.4  |      |
| 1    | 0    | 0    | 131072 states | 6.6    | 8.2    | 10.9   | 13.1   | 16.4  | 21.8  | 32.8  |      |
|      |      | 1    | 262144 states | 13.1   | 16.4   | 21.8   | 26.2   | 32.8  | 43.6  | 65.6  |      |
|      | 1    | 0    | Reserved      | —      | —      | —      | —      | —     | —     | —     | —    |
|      |      | 1    | 16 states*    | 0.8    | 1.0    | 1.3    | 1.6    | 2.0   | 1.7   | 4.0   |      |

: Recommended time setting

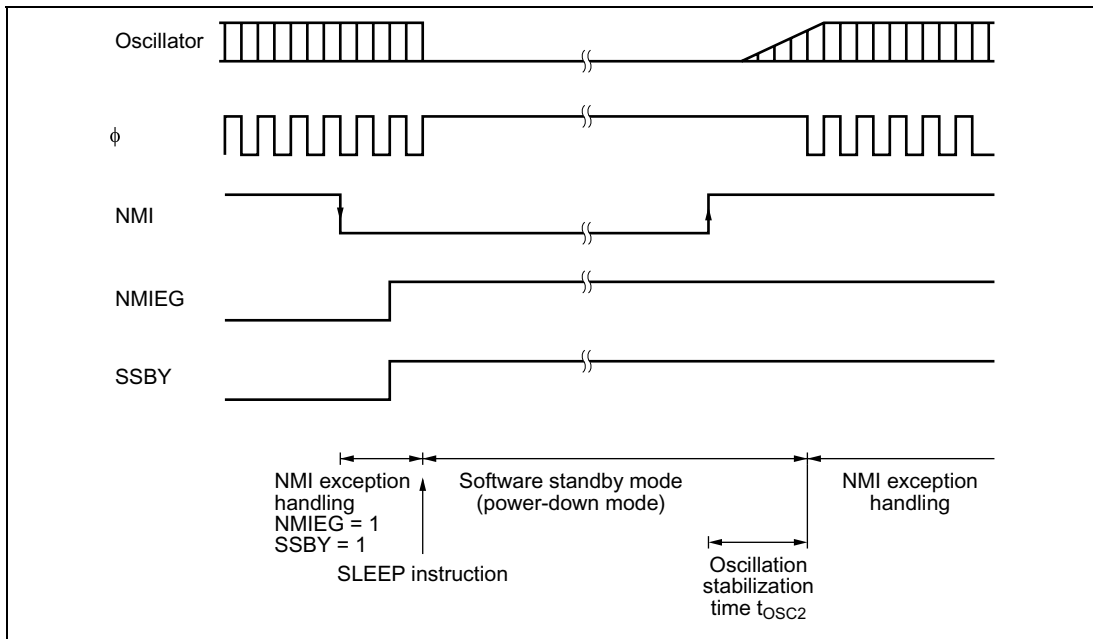
Note: \* Setting prohibited

## 20.4.4 Software Standby Mode Application Example

Figure 20-3 shows an example in which a transition is made to software standby mode at a falling edge on the NMI pin, and software standby mode is cleared at a 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 20-3 Software Standby Mode Application Example**

## 20.5 Hardware Standby Mode

### 20.5.1 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.

In order to retain on-chip RAM data, the RAME bit in SYSCR should be cleared to 0 before driving the  $\overline{\text{STBY}}$  pin low.

Do not change the state of the mode pins (MD0 to MD2) while this LSI is in hardware standby mode.

### 20.5.2 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 stabilizes (at least 8 ms—the oscillation stabilization 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.

### 20.5.3 Hardware Standby Mode Timings

#### Timing of Transition to Hardware Standby Mode

1. To retain RAM contents with the RAME bit set to 1 in SYSCR

Drive the  $\overline{\text{RES}}$  signal low at least 10 states before the  $\overline{\text{STBY}}$  signal goes low, as shown in figure 20-4. After  $\overline{\text{STBY}}$  has gone low,  $\overline{\text{RES}}$  has to wait for at least 0 ns before becoming high.

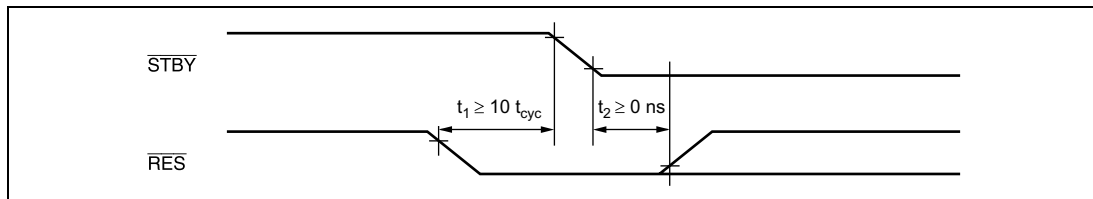


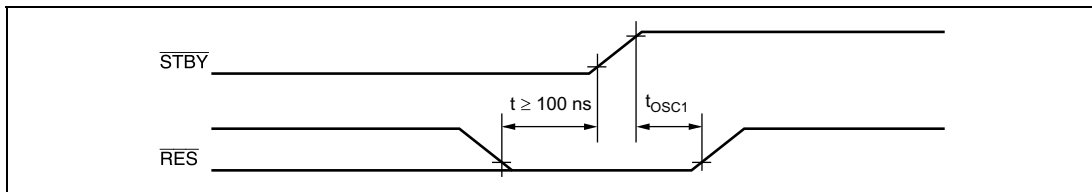
Figure 20-4 Timing of Transition to Hardware Standby Mode

2. To retain RAM contents with the RAME bit cleared to 0 in SYSCR, or when RAM contents do not need to be retained

$\overline{\text{RES}}$  does not have to be driven low as in the above case.

## Timing of Recovery from Hardware Standby Mode

Drive the  $\overline{\text{RES}}$  signal low approximately 100 ns or more before  $\overline{\text{STBY}}$  goes high to execute a power-on reset.



**Figure 20-5 Timing of Recovery from Hardware Standby Mode**

## 20.6 Module Stop Mode

### 20.6.1 Module Stop Mode

Module stop mode can be set for individual on-chip supporting 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 the SCI\*, HCAN, and A/D converter are retained.

After reset clearance, all modules other than DTC are in module stop mode.

When an on-chip supporting module is in module stop mode, read/write access to its registers is disabled.

Note: \* The internal states of some SCI registers are retained.

## 20.7 $\phi$ Clock Output Disabling Function

The output of the  $\phi$  clock can be controlled by means of the PSTOP bit in SCKCR, and DDR for the corresponding port. When the PSTOP bit is set to 1, the  $\phi$  clock stops at the end of the bus cycle, and  $\phi$  output goes high.  $\phi$  clock output is enabled when the PSTOP bit is cleared to 0. When DDR for the corresponding port is cleared to 0,  $\phi$  clock output is disabled and input port mode is set. Table 20-4 shows the state of the  $\phi$  pin in each processing state.

**Table 20-4     $\phi$  Pin State in Each Processing State**

| <b>Register Settings</b> |              | <b>Normal Mode</b> | <b>Sleep Mode</b> | <b>Software Standby Mode</b> | <b>Hardware Standby Mode</b> |
|--------------------------|--------------|--------------------|-------------------|------------------------------|------------------------------|
| <b>DDR</b>               | <b>PSTOP</b> |                    |                   |                              |                              |
| 0                        | X            | High impedance     | High impedance    | High impedance               | High impedance               |
| 1                        | 0            | $\phi$ output      | $\phi$ output     | Fixed high                   | High impedance               |
| 1                        | 1            | Fixed high         | Fixed high        | Fixed high                   | High impedance               |

**Legend**

X: Don't care

## **20.8 Usage Notes**

### **20.8.1 I/O Port Status**

In software standby 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.

### **20.8.2 Current Dissipation during Oscillation Stabilization Wait Period**

Current dissipation increases during the oscillation stabilization wait period.

### **20.8.3 DTC Module Stop**

Depending on the operating status of the DTC, MSTPA6 bit may not be set to 1. Setting of the DTC module stop mode should be carried out only when the respective module is not activated.

For details, refer to section 8, Data Transfer Controller (DTC).

### **20.8.4 On-Chip Supporting Module Interrupt**

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 DTC activation source.

Interrupts should therefore be disabled before entering module stop mode.

### **20.8.5 Writing to MSTPCR**

MSTPCR should only be written to by the CPU.



## 21.1 Absolute Maximum Ratings

Table 21-1 lists the absolute maximum ratings.

**Table 21-1 Absolute Maximum Ratings**

| Item                                              | Symbol    | Value                                 | Unit |
|---------------------------------------------------|-----------|---------------------------------------|------|
| Power supply voltage                              | $V_{CC}$  | −0.3 to +7.0                          | V    |
| Input voltage (XTAL, EXTAL)                       | $V_{in}$  | −0.3 to $V_{CC} + 0.3$                | V    |
| Input voltage (ports 4 and 9)                     | $V_{in}$  | −0.3 to $AV_{CC} + 0.3$               | V    |
| Input voltage (except XTAL, EXTAL, ports 4 and 9) | $V_{in}$  | −0.3 to $V_{CC} + 0.3$                | 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.

## 21.2 DC Characteristics

Table 21-2 lists the DC characteristics. Table 21-3 lists the permissible output currents.

**Table 21-2 DC Characteristics**

Conditions:  $V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ ,  $AV_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ ,  $V_{SS} = AV_{SS} = 0 \text{ V}$ ,  
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$  (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 IRQ5                     | $V_T^-$         | $V_{CC} \times 0.2$  | —   | —                    | V             |                                                     |
|                               |                                  | $V_T^+$         | —                    | —   | $V_{CC} \times 0.7$  | V             |                                                     |
|                               |                                  | $V_T^+ - V_T^-$ | $V_{CC} \times 0.05$ | —   | —                    | V             |                                                     |
| Input high voltage            | RES, STBY, NMI, MD2 to MD0, FWE  | $V_{IH}$        | $V_{CC} \times 0.9$  | —   | $V_{CC} + 0.3$       | V             |                                                     |
|                               | EXTAL                            |                 | $V_{CC} \times 0.7$  | —   | $V_{CC} + 0.3$       | V             |                                                     |
|                               | Ports 1, A to D, F, HRxD         |                 | $V_{CC} \times 0.7$  | —   | $V_{CC} + 0.3$       | V             |                                                     |
|                               | Port 4 and 9                     |                 | $AV_{CC} \times 0.7$ | —   | $AV_{CC} + 0.3$      | V             |                                                     |
| Input low voltage             | RES, STBY, NMI, MD2 to MD0, FWE  | $V_{IL}$        | -0.3                 | —   | $V_{CC} \times 0.1$  | V             |                                                     |
|                               | EXTAL                            |                 | -0.3                 | —   | $V_{CC} \times 0.2$  | V             |                                                     |
|                               | Ports 1, A to D, F, HRxD         |                 | -0.3                 | —   | $V_{CC} \times 0.2$  | V             |                                                     |
|                               | Ports 4 and 9                    |                 | -0.3                 | —   | $AV_{CC} \times 0.2$ | 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             | $I_{OH} = -1 \text{ mA}$                            |
| Output low voltage            | All output pins                  | $V_{OL}$        | —                    | —   | 0.4                  | V             | $I_{OL} = 1.6 \text{ mA}$                           |
| Input leakage current         | RES                              | $ I_{in} $      | —                    | —   | 1.0                  | $\mu\text{A}$ | $V_{in} = 0.5 \text{ V to } V_{CC} - 0.5 \text{ V}$ |
|                               | STBY, NMI, MD2 to MD0, FWE, HRxD |                 | —                    | —   | 1.0                  | $\mu\text{A}$ |                                                     |
|                               | Ports 4 and 9                    |                 | —                    | —   | 1.0                  | $\mu\text{A}$ |                                                     |

| Item                              |                                                | Symbol        | Min | Typ                    | Max                    | Unit    | Test Conditions                                          |
|-----------------------------------|------------------------------------------------|---------------|-----|------------------------|------------------------|---------|----------------------------------------------------------|
| MOS input pull-up current         | Ports A to D                                   | $-I_P$        | 30  | —                      | 300                    | $\mu A$ | $V_{in} = 0 V$                                           |
| Input capacitance                 | $\overline{RES}$                               | $C_{in}$      | —   | —                      | 30                     | pF      | $V_{in} = 0 V$                                           |
|                                   | NMI                                            |               | —   | —                      | 30                     | pF      | $f = 1 MHz$                                              |
|                                   | All input pins except $\overline{RES}$ and NMI |               | —   | —                      | 15                     | pF      | $T_a = 25^\circ C$                                       |
| Current dissipation <sup>*2</sup> | Normal operation                               | $I_{CC}^{*3}$ | —   | 65<br>$V_{CC} = 5.0 V$ | 75<br>$V_{CC} = 5.5 V$ | mA      | $f = 20 MHz$                                             |
|                                   | Sleep mode                                     |               | —   | 50<br>$V_{CC} = 5.0 V$ | 60<br>$V_{CC} = 5.5 V$ | mA      |                                                          |
|                                   | All modules stopped                            |               | —   | 40                     | —                      | mA      | $f = 20 MHz$ ,<br>$V_{CC} = 5.0 V$<br>(reference values) |
|                                   | Medium-speed mode ( $\phi/32$ )                |               | —   | 45                     | —                      | mA      |                                                          |
|                                   | Standby mode                                   |               | —   | 2.0                    | 5.0                    | $\mu A$ | $T_a \leq 50^\circ C$                                    |
| Analog power supply current       | During A/D conversion                          | $AI_{CC}$     | —   | 2.5                    | 4.0                    | mA      | $AV_{CC} = 5.0 V$                                        |
|                                   | Idle                                           |               | —   | —                      | 5.0                    | $\mu A$ | $50^\circ C < T_a$                                       |
| RAM standby voltage               |                                                | $V_{RAM}$     | 2.0 | —                      | —                      | V       |                                                          |

Notes: <sup>\*1</sup> If the A/D converter is not used, do not leave the  $AV_{CC}$ , and  $AV_{SS}$  pins open. Apply a voltage between 4.5 V and 5.5 V to the  $AV_{CC}$  pin by connecting them to  $V_{CC}$ , for instance.

<sup>\*2</sup> Current dissipation values are for  $V_{IH} = V_{CC}$  (EXTAL),  $AV_{CC}$  (ports 4 and 9), or  $V_{CC}$  (other), and  $V_{IL} = 0 V$ , with all output pins unloaded and the on-chip MOS pull-up transistors in the off state.

<sup>\*3</sup>  $I_{CC}$  depends on  $V_{CC}$  and  $f$  as follows:

$I_{CC} (\max) = 27 + (0.435 \times V_{CC} \times f)$  (normal operation)

$I_{CC} (\max) = 27 + (0.3 \times V_{CC} \times f)$  (sleep mode)

**Table 21-3 Permissible Output Currents**

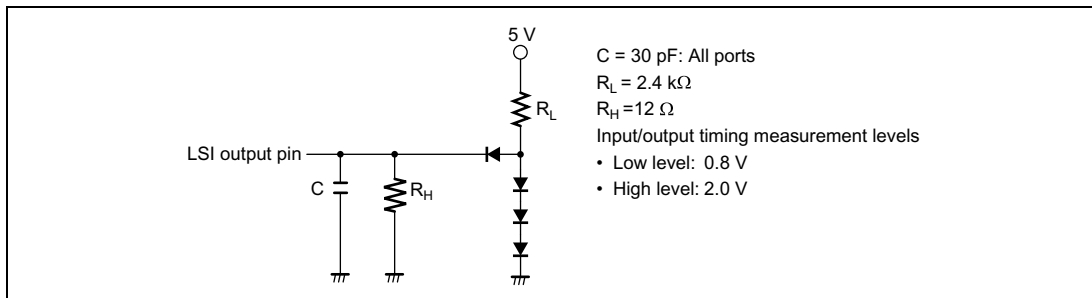
Conditions:  $V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ ,  $AV_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ ,  $V_{SS} = AV_{SS} = 0 \text{ V}$ ,  
 $T_a = -20^\circ\text{C to } +75^\circ\text{C}$  (regular specifications),  $T_a = -40^\circ\text{C to } +85^\circ\text{C}$  (wide-range specifications)\*

| Item                                      |                          |                                            | Symbol           | Min | Typ | Max | Unit |
|-------------------------------------------|--------------------------|--------------------------------------------|------------------|-----|-----|-----|------|
| Permissible output low current (per pin)  | All output pins          | $V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ | $I_{OL}$         | —   | —   | 10  | mA   |
| Permissible output low current (total)    | Total of all output pins | $V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ | $\Sigma I_{OL}$  | —   | —   | 100 | mA   |
| Permissible output high current (per pin) | All output pins          | $V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ | $-I_{OH}$        | —   | —   | 2.0 | mA   |
| Permissible output high current (total)   | Total of all output pins | $V_{CC} = 4.5 \text{ V to } 5.5 \text{ V}$ | $\Sigma -I_{OH}$ | —   | —   | 30  | mA   |

Note: \* To protect chip reliability, do not exceed the output current values in table 21-3.

## 21.3 AC Characteristics

Figure 21-1 show, the test conditions for the AC characteristics.

**Figure 21-1 Output Load Circuit**

21.3.1 Clock Timing

Table 21-4 lists the clock timing

Table 21-4 Clock Timing

Conditions :  $V_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $AV_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $V_{SS} = AV_{SS} = 0\text{ V}$ ,  $\phi = 4\text{ MHz to }20\text{ MHz}$ ,  $T_a = -20^{\circ}\text{C to }+75^{\circ}\text{C}$  (regular specifications),  $T_a = -40^{\circ}\text{C to }+85^{\circ}\text{C}$  (wide-range specifications)

| Item                                                         | Symbol     | Min | Max | Unit | Test Conditions |
|--------------------------------------------------------------|------------|-----|-----|------|-----------------|
| Clock cycle time                                             | $t_{cyc}$  | 50  | 250 | ns   | Figure 21-2     |
| Clock high pulse width                                       | $t_{CH}$   | 15  | —   | ns   |                 |
| Clock low pulse width                                        | $t_{CL}$   | 15  | —   | ns   |                 |
| Clock rise time                                              | $t_{Cr}$   | —   | 10  | ns   |                 |
| Clock fall time                                              | $t_{Cf}$   | —   | 10  | ns   |                 |
| Oscillation stabilization time at reset (crystal)            | $t_{OSC1}$ | 20  | —   | ms   | Figure 21-3     |
| Oscillation stabilization time in software standby (crystal) | $t_{OSC2}$ | 8   | —   | ms   | Figure 20-2     |
| External clock output stabilization delay time               | $t_{DEXT}$ | 2   | —   | ms   | Figure 21-3     |

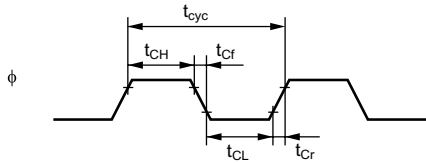
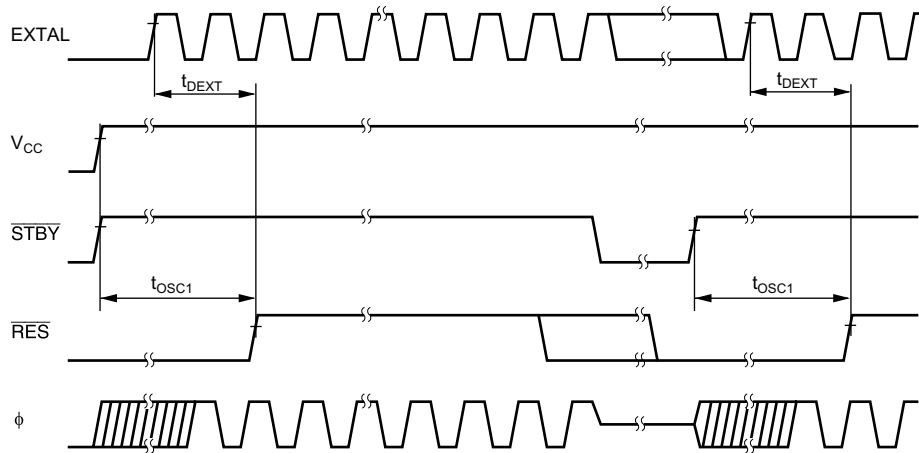


Figure 21-2 System Clock Timing



**Figure 21-3 Oscillation Stabilization Timing**

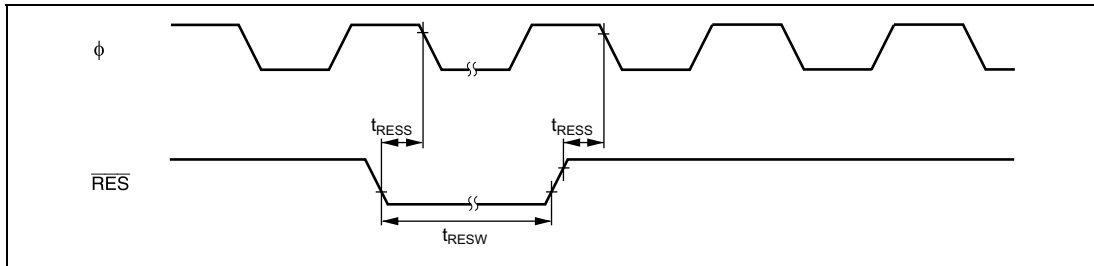
### 21.3.2 Control Signal Timing

Table 21-5 lists the control signal timing.

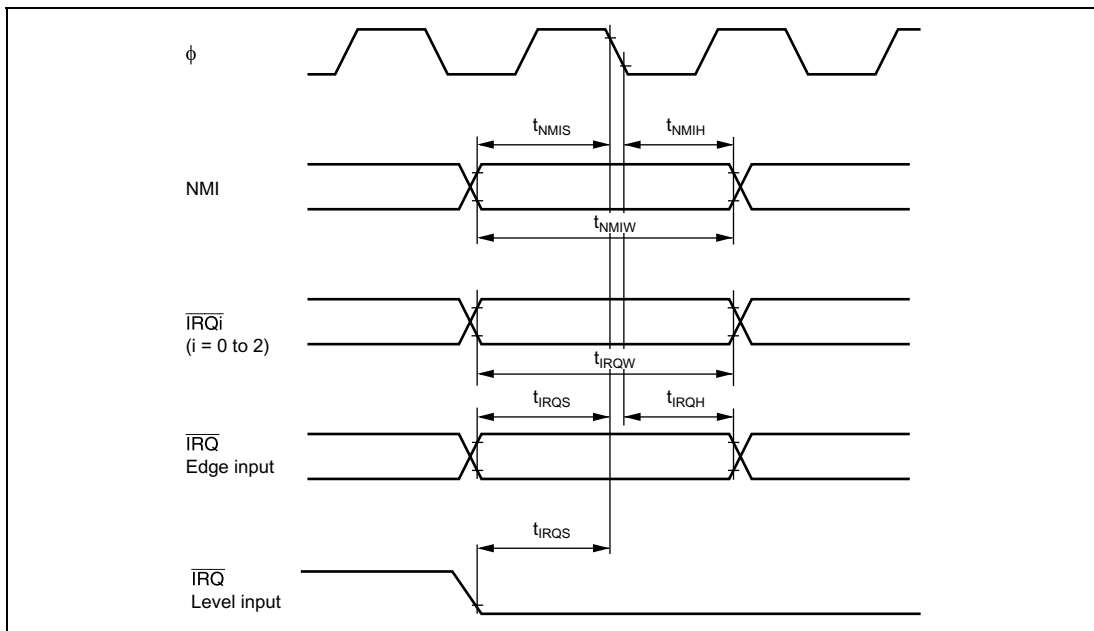
**Table 21-5 Control Signal Timing**

Conditions:  $V_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $AV_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $V_{SS} = AV_{SS} = 0\text{ V}$ ,  $\phi = 4\text{ MHz to }20\text{ MHz}$ ,  $T_a = -20^\circ\text{C to }+75^\circ\text{C}$  (regular specifications),  $T_a = -40^\circ\text{C to }+85^\circ\text{C}$  (wide-range specifications)

| Item                                            | Symbol     | Min | Max | Unit      | Test Conditions |
|-------------------------------------------------|------------|-----|-----|-----------|-----------------|
| RES setup time                                  | $t_{RESS}$ | 200 | —   | ns        | Figure 21-4     |
| RES pulse width                                 | $t_{RESW}$ | 20  | —   | $t_{cyc}$ |                 |
| NMI setup time                                  | $t_{NMIS}$ | 150 | —   | ns        | Figure 21-5     |
| NMI hold time                                   | $t_{NMIH}$ | 10  | —   | ns        |                 |
| NMI pulse width (exiting software standby mode) | $t_{NMIW}$ | 200 | —   | ns        |                 |
| IRQ setup time                                  | $t_{IRQS}$ | 150 | —   | ns        |                 |
| IRQ hold time                                   | $t_{IRQH}$ | 10  | —   | ns        |                 |
| IRQ pulse width (exiting software standby mode) | $t_{IRQW}$ | 200 | —   | ns        |                 |



**Figure 21-4 Reset Input Timing**



**Figure 21-5 Interrupt Input Timing**

### 21.3.3 Timing of On-Chip Supporting Modules

Table 21-6 lists the timing of on-chip supporting modules.

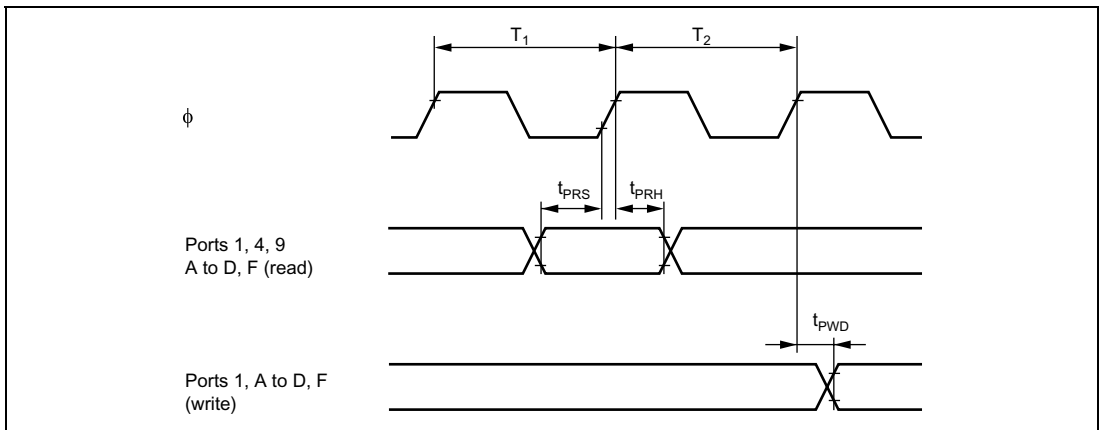
**Table 21-6 Timing of On-Chip Supporting Modules**

Conditions:  $V_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $AV_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $V_{SS} = AV_{SS} = 0$ ,  $\phi = 4\text{ MHz to }20\text{ MHz}$ ,  $T_a = -20^\circ\text{C to }+75^\circ\text{C}$  (regular specifications),  $T_a = -40^\circ\text{C to }+85^\circ\text{C}$  (wide-range specifications)

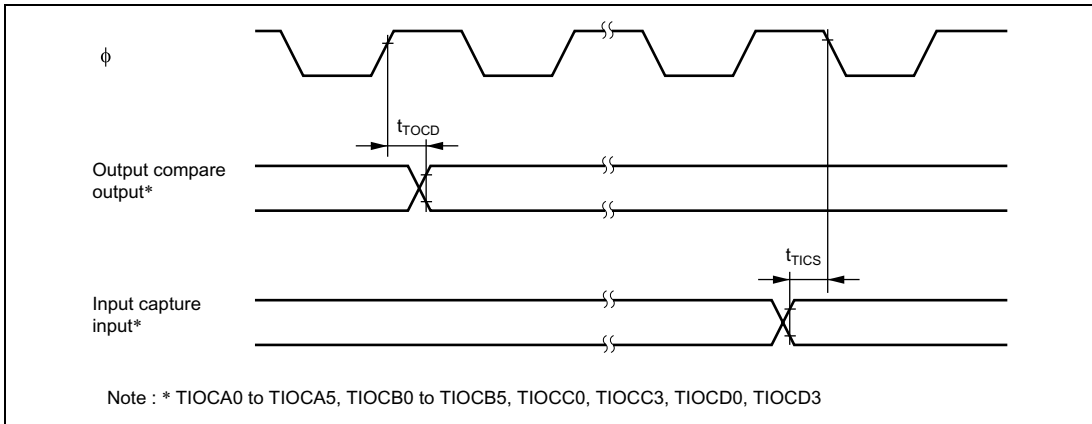
| Item     |                                       | Symbol                     | Min | Max | Unit       | Test Conditions |
|----------|---------------------------------------|----------------------------|-----|-----|------------|-----------------|
| I/O port | Output data delay time                | $t_{PWD}$                  | —   | 50  | ns         | Figure 21-6     |
|          | Input data setup time                 | $t_{PRS}$                  | 30  | —   |            |                 |
|          | Input data hold time                  | $t_{PRH}$                  | 30  | —   |            |                 |
| TPU      | Timer output delay time               | $t_{TOCD}$                 | —   | 50  | ns         | Figure 21-7     |
|          | Timer input setup time                | $t_{TICS}$                 | 30  | —   |            |                 |
|          | Timer clock input setup time          | $t_{TCKS}$                 | 30  | —   | ns         | Figure 21-8     |
|          | Timer clock pulse width               | Single edge<br>$t_{TCKWH}$ | 1.5 | —   | $t_{cyc}$  |                 |
|          |                                       | Both edges<br>$t_{TCKWL}$  | 2.5 | —   |            |                 |
| SCI      | Input clock cycle                     | Asynchronous<br>$t_{Scyc}$ | 4   | —   | $t_{cyc}$  | Figure 21-9     |
|          |                                       | Synchronous                | 6   | —   |            |                 |
|          | Input clock pulse width               | $t_{SCKW}$                 | 0.4 | 0.6 | $t_{Scyc}$ | Figure 21-10    |
|          | Input clock rise time                 | $t_{SCKr}$                 | —   | 1.5 | $t_{cyc}$  |                 |
|          | Input clock fall time                 | $t_{SCKf}$                 | —   | 1.5 |            |                 |
|          | Transmit data delay time              | $t_{TXD}$                  | —   | 50  | ns         |                 |
|          | Receive data setup time (synchronous) | $t_{RXS}$                  | 50  | —   |            | Figure 21-10    |
|          | Receive data hold time (synchronous)  | $t_{RXH}$                  | 50  | —   |            |                 |

| Item          |                                    | Symbol     | Min | Max | Unit      | Test Conditions |
|---------------|------------------------------------|------------|-----|-----|-----------|-----------------|
| A/D converter | Trigger input setup time           | $t_{TRGS}$ | 30  | —   | ns        | Figure 21-11    |
| HCAN*         | Transmit data delay time           | $t_{HTXD}$ | —   | 100 | ns        | Figure 21-12    |
|               | Transmit data setup time           | $t_{HRXS}$ | 100 | —   |           |                 |
|               | Transmit data hold time            | $t_{HRXH}$ | 100 | —   |           |                 |
| PPG           | Pulse output delay time            | $t_{POD}$  | —   | 50  | ns        | Figure 21-13    |
| MMT           | MMT output delay time              | $t_{MTOD}$ | —   | 50  | ns        | Figure 21-14    |
|               | $\overline{PCI}$ input setup time  | $t_{PCIS}$ | 35  | —   |           |                 |
|               | $\overline{PCI}$ input pulse width | $t_{PCIW}$ | 1.5 | —   | $t_{cyc}$ |                 |
|               | $\overline{POE}$ Input setup time  | $t_{POES}$ | 35  | —   | ns        |                 |
|               | $\overline{POE}$ Input pulse width | $t_{POEW}$ | 1.5 | —   | $t_{cyc}$ |                 |

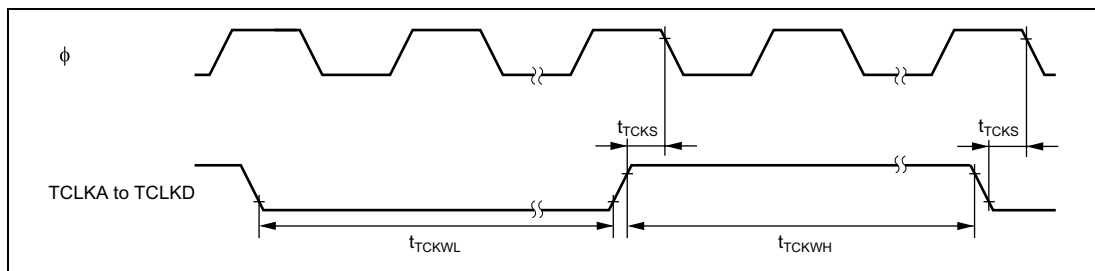
Note: \* The HCAN input signal is asynchronous. However, its state is judged to have changed at the rising-edge (two clock cycles) of the CK clock signal shown in figure 21-12. The HCAN output signal is also asynchronous. Its state changes based on the rising-edge (two clock cycles) of the CK clock signal shown in figure 21-12.



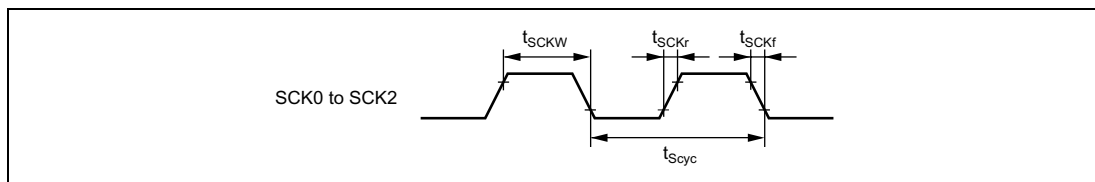
**Figure 21-6 I/O Port Input/Output Timing**



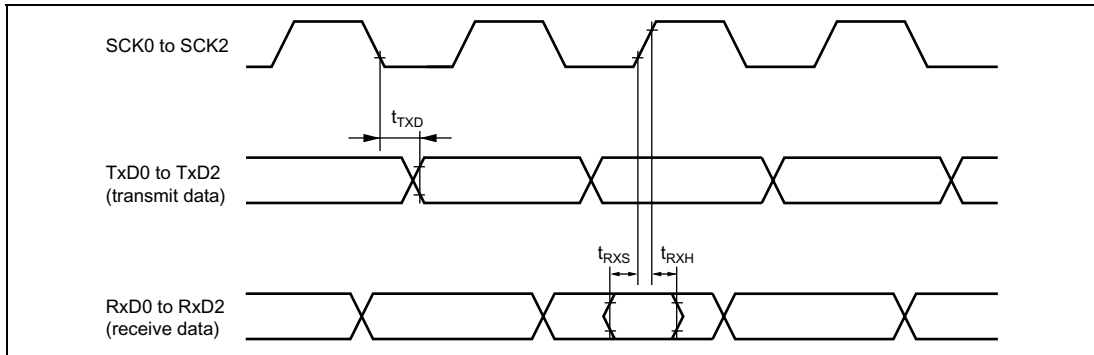
**Figure 21-7 TPU Input/Output Timing**



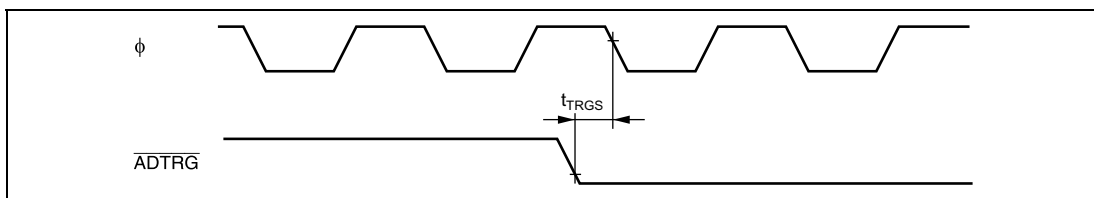
**Figure 21-8 TPU Clock Input Timing**



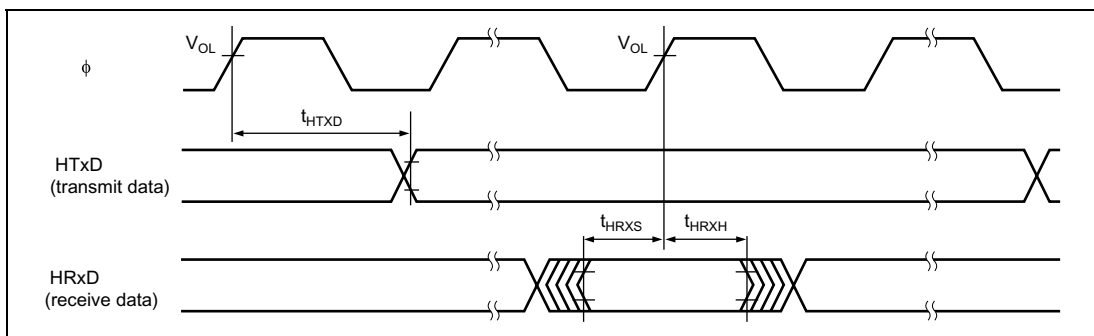
**Figure 21-9 SCK Clock Input Timing**



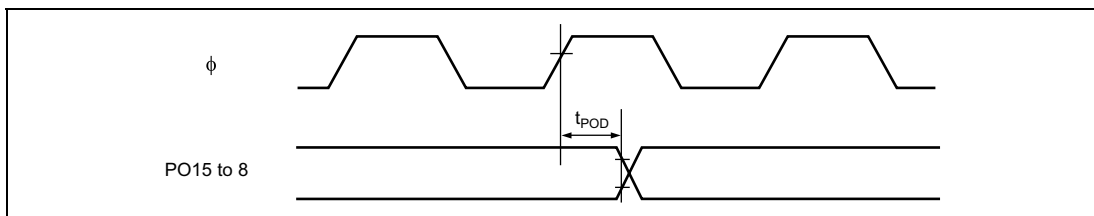
**Figure 21-10 SCI Input/Output Timing (Clock Synchronous Mode)**



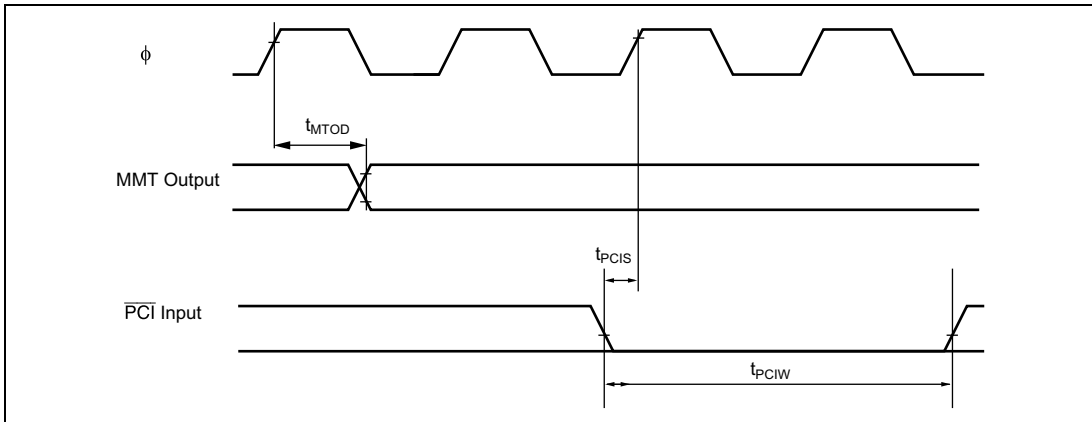
**Figure 21-11 A/D Converter External Trigger Input Timing**



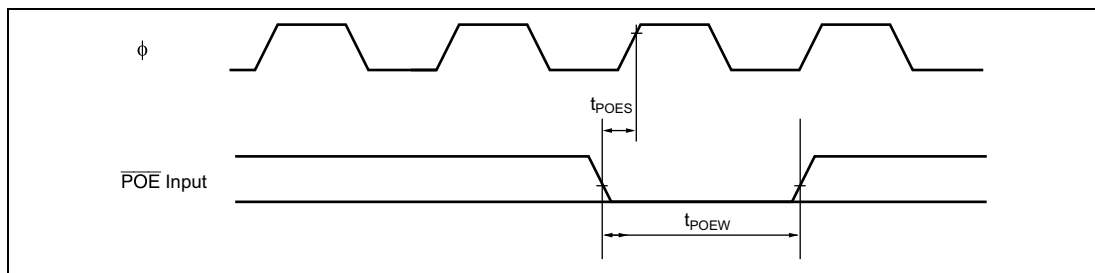
**Figure 21-12 HCAN Input/Output Timing**



**Figure 21-13 PPG Output Timing**



**Figure 21-14 MMT Input/Output Timing**



**Figure 21-15 POE Input/Output Timing**

## 21.4 A/D Conversion Characteristics

Table 21-7 lists the A/D conversion characteristics.

**Table 21-7 A/D Conversion Characteristics**

Conditions:  $V_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $AV_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $V_{SS} = AV_{SS} = 0\text{ V}$ ,  $\phi = 4\text{ MHz to }20\text{ MHz}$ ,  $T_a = -20^\circ\text{C to }+75^\circ\text{C}$  (regular specifications),  $T_a = -40^\circ\text{C to }+85^\circ\text{C}$  (wide-range specifications)

| Item                                | Min | Typ       | Max       | Unit          |
|-------------------------------------|-----|-----------|-----------|---------------|
| Resolution                          | 10  | 10        | 10        | bits          |
| Conversion time                     | 10  | —         | 200       | $\mu\text{s}$ |
| Analog input capacitance            | —   | —         | 20        | pF            |
| Permissible signal-source impedance | —   | —         | 5         | k $\Omega$    |
| Nonlinearity error                  | —   | —         | $\pm 3.5$ | LSB           |
| Offset error                        | —   | —         | $\pm 3.5$ | LSB           |
| Full-scale error                    | —   | —         | $\pm 3.5$ | LSB           |
| Quantization                        | —   | $\pm 0.5$ | —         | LSB           |
| Absolute accuracy                   | —   | —         | $\pm 4.0$ | LSB           |

## 21.5 Flash Memory Characteristics

Table 21-8 lists the flash memory characteristics.

**Table 21-8 Flash Memory Characteristics**

Conditions:  $V_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,  $AV_{CC} = 4.5\text{ V to }5.5\text{ V}$ ,

$V_{SS} = PLLV_{SS} = AV_{SS} = 0\text{ V}$ ,

$T_a = 0\text{ to }+75^{\circ}\text{C}$  (Programming / erasing operating temperature range)

| Item                                 | Symbol                                         | Min         | Typ | Max  | Unit             | Test Condition                   |
|--------------------------------------|------------------------------------------------|-------------|-----|------|------------------|----------------------------------|
| Programming time <sup>*1 *2 *4</sup> | $t_P$                                          | —           | 10  | 200  | ms/<br>128 bytes |                                  |
| Erase time <sup>*1 *3 *5</sup>       | $t_E$                                          | —           | 100 | 1200 | ms/block         |                                  |
| Reprogramming count                  | $N_{WEC}$                                      | —           | —   | 100  | times            |                                  |
| Programming                          | Wait time after SWE bit setting <sup>*1</sup>  | $t_{sswe}$  | 1   | 1    | —                | $\mu\text{s}$                    |
|                                      | Wait time after PSU bit setting <sup>*1</sup>  | $t_{psu}$   | 50  | 50   | —                | $\mu\text{s}$                    |
|                                      | Wait time after P bit setting <sup>*1 *4</sup> | $t_{sp30}$  | 28  | 30   | 32               | $\mu\text{s}$                    |
|                                      |                                                |             |     |      |                  | Programming time wait            |
|                                      |                                                | $t_{sp200}$ | 198 | 200  | 202              | $\mu\text{s}$                    |
|                                      |                                                |             |     |      |                  | Programming time wait            |
|                                      |                                                | $t_{sp10}$  | 8   | 10   | 12               | $\mu\text{s}$                    |
|                                      |                                                |             |     |      |                  | Additional-programming time wait |
|                                      | Wait time after P bit clear <sup>*1</sup>      | $t_{cp}$    | 5   | 5    | —                | $\mu\text{s}$                    |
|                                      | Wait time after PSU bit clear <sup>*1</sup>    | $t_{cpsu}$  | 5   | 5    | —                | $\mu\text{s}$                    |
|                                      | Wait time after PV bit setting <sup>*1</sup>   | $t_{spv}$   | 4   | 4    | —                | $\mu\text{s}$                    |
|                                      | Wait time after H'FF dummy write <sup>*1</sup> | $t_{spvr}$  | 2   | 2    | —                | $\mu\text{s}$                    |
|                                      | Wait time after PV bit clear <sup>*1</sup>     | $t_{cpv}$   | 2   | 2    | —                | $\mu\text{s}$                    |
|                                      | Wait time after SWE bit clear <sup>*1</sup>    | $t_{cswe}$  | 100 | 100  | —                | $\mu\text{s}$                    |
|                                      | Maximum programming count <sup>*1 *4</sup>     | $N$         | —   | —    | 1000             | times                            |
| Erase                                | Wait time after SWE bit setting <sup>*1</sup>  | $t_{sswe}$  | 1   | 1    | —                | $\mu\text{s}$                    |
|                                      | Wait time after ESU bit setting <sup>*1</sup>  | $t_{sesu}$  | 100 | 100  | —                | $\mu\text{s}$                    |
|                                      | Wait time after E bit setting <sup>*1 *5</sup> | $t_{se}$    | 10  | 10   | 100              | ms                               |
|                                      |                                                |             |     |      |                  | Erase time wait                  |
|                                      | Wait time after E bit clear <sup>*1</sup>      | $t_{ce}$    | 10  | 10   | —                | $\mu\text{s}$                    |
|                                      | Wait time after ESU bit clear <sup>*1</sup>    | $t_{cesu}$  | 10  | 10   | —                | $\mu\text{s}$                    |
|                                      | Wait time after EV bit setting <sup>*1</sup>   | $t_{sev}$   | 20  | 20   | —                | $\mu\text{s}$                    |
|                                      | Wait time after H'FF dummy write <sup>*1</sup> | $t_{sevr}$  | 2   | 2    | —                | $\mu\text{s}$                    |
|                                      | Wait time after EV bit clear <sup>*1</sup>     | $t_{cev}$   | 4   | 4    | —                | $\mu\text{s}$                    |
|                                      | Wait time after SWE bit clear <sup>*1</sup>    | $t_{cswe}$  | 100 | 100  | —                | $\mu\text{s}$                    |
|                                      | Maximum erase count <sup>*1 *5</sup>           | $N$         | 12  | —    | 120              | 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 P-bit in the flash memory control register (FLMCR1) is set. It does not include the programming verification time.)
- \*3 Block erase time (Shows the total period for which the E-bit FLMCR1 is set. It does not include the erase verification time.)
- \*4 To specify the maximum programming time value ( $t_p(\max)$ ) in the 128-bytes programming algorithm, set the max. value (1000) for the maximum programming count (N).  
 The wait time after P bit setting should be changed as follows according to the value of the programming counter (n).  
 Programming counter (n) = 1 to 6:  $t_{sp30} = 30 \mu s$   
 Programming counter (n) = 7 to 1000:  $t_{sp200} = 200 \mu s$   
 [In additional programming]  
 Programming counter (n) = 1 to 6:  $t_{sp10} = 10 \mu s$
- \*5 For the maximum erase time ( $t_E(\max)$ ), the following relationship applies between the wait time after E bit setting ( $t_{se}$ ) and the maximum erase count (N):  

$$t_E(\max) = \text{Wait time after E bit setting } (t_{se}) \times \text{maximum erase count } (N)$$
  
 To set the maximum erase time, the values of ( $t_{se}$ ) and (N) should be set so as to satisfy the above formula.  
 Examples: When  $t_{se} = 100 \text{ ms}$ ,  $N = 12 \text{ times}$   
 When  $t_{se} = 10 \text{ ms}$ ,  $N = 120 \text{ times}$



# Appendix

## A. On-Chip I/O Register

### A.1 Register Addresses

| Register Name                   | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|---------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Master control register         | MCR               | 8       | H'F800   | HCAN   | 16            | 4               |
| General status register         | GSR               | 8       | H'F801   | HCAN   | 16            | 4               |
| Bit configuration register      | BCR               | 16      | H'F802   | HCAN   | 16            | 4               |
| Mailbox configuration register  | MBCR              | 16      | H'F804   | HCAN   | 16            | 4               |
| Transmit wait register          | TXPR              | 16      | H'F806   | HCAN   | 16            | 4               |
| Transmit wait cancel register   | TXCR              | 16      | H'F808   | HCAN   | 16            | 4               |
| Transmit acknowledge register   | TXACK             | 16      | H'F80A   | HCAN   | 16            | 4               |
| Abort acknowledge register      | ABACK             | 16      | H'F80C   | HCAN   | 16            | 4               |
| Receive complete register       | RXPR              | 16      | H'F80E   | HCAN   | 16            | 4               |
| Remote request register         | RFPR              | 16      | H'F810   | HCAN   | 16            | 4               |
| Interrupt register              | IRR               | 16      | H'F812   | HCAN   | 16            | 4               |
| Mailbox interrupt mask register | MBIMR             | 16      | H'F814   | HCAN   | 16            | 4               |
| Interrupt mask register         | IMR               | 16      | H'F816   | HCAN   | 16            | 4               |
| Receive error counter           | REC               | 8       | H'F818   | HCAN   | 16            | 4               |
| Transmit error counter          | TEC               | 8       | H'F819   | HCAN   | 16            | 4               |
| Unread message status register  | UMSR              | 16      | H'F81A   | HCAN   | 16            | 4               |
| Local acceptance filter mask L  | LAFML             | 16      | H'F81C   | HCAN   | 16            | 4               |
| Local acceptance filter mask H  | LAFMH             | 16      | H'F81E   | HCAN   | 16            | 4               |
| Message control 0[1]            | MC0[1]            | 8       | H'F820   | HCAN   | 16            | 4               |
| Message control 0[2]            | MC0[2]            | 8       | H'F821   | HCAN   | 16            | 4               |
| Message control 0[3]            | MC0[3]            | 8       | H'F822   | HCAN   | 16            | 4               |
| Message control 0[4]            | MC0[4]            | 8       | H'F823   | HCAN   | 16            | 4               |
| Message control 0[5]            | MC0[5]            | 8       | H'F824   | HCAN   | 16            | 4               |
| Message control 0[6]            | MC0[6]            | 8       | H'F825   | HCAN   | 16            | 4               |
| Message control 0[7]            | MC0[7]            | 8       | H'F826   | HCAN   | 16            | 4               |
| Message control 0[8]            | MC0[8]            | 8       | H'F827   | HCAN   | 16            | 4               |
| Message control 1[1]            | MC1[1]            | 8       | H'F828   | HCAN   | 16            | 4               |

| Register Name        | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|----------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message control 1[2] | MC1[2]            | 8       | H'F829   | HCAN   | 16            | 4               |
| Message control 1[3] | MC1[3]            | 8       | H'F82A   | HCAN   | 16            | 4               |
| Message control 1[4] | MC1[4]            | 8       | H'F82B   | HCAN   | 16            | 4               |
| Message control 1[5] | MC1[5]            | 8       | H'F82C   | HCAN   | 16            | 4               |
| Message control 1[6] | MC1[6]            | 8       | H'F82D   | HCAN   | 16            | 4               |
| Message control 1[7] | MC1[7]            | 8       | H'F82E   | HCAN   | 16            | 4               |
| Message control 1[8] | MC1[8]            | 8       | H'F82F   | HCAN   | 16            | 4               |
| Message control 2[1] | MC2[1]            | 8       | H'F830   | HCAN   | 16            | 4               |
| Message control 2[2] | MC2[2]            | 8       | H'F831   | HCAN   | 16            | 4               |
| Message control 2[3] | MC2[3]            | 8       | H'F832   | HCAN   | 16            | 4               |
| Message control 2[4] | MC2[4]            | 8       | H'F833   | HCAN   | 16            | 4               |
| Message control 2[5] | MC2[5]            | 8       | H'F834   | HCAN   | 16            | 4               |
| Message control 2[6] | MC2[6]            | 8       | H'F835   | HCAN   | 16            | 4               |
| Message control 2[7] | MC2[7]            | 8       | H'F836   | HCAN   | 16            | 4               |
| Message control 2[8] | MC2[8]            | 8       | H'F837   | HCAN   | 16            | 4               |
| Message control 3[1] | MC3[1]            | 8       | H'F838   | HCAN   | 16            | 4               |
| Message control 3[2] | MC3[2]            | 8       | H'F839   | HCAN   | 16            | 4               |
| Message control 3[3] | MC3[3]            | 8       | H'F83A   | HCAN   | 16            | 4               |
| Message control 3[4] | MC3[4]            | 8       | H'F83B   | HCAN   | 16            | 4               |
| Message control 3[5] | MC3[5]            | 8       | H'F83C   | HCAN   | 16            | 4               |
| Message control 3[6] | MC3[6]            | 8       | H'F83D   | HCAN   | 16            | 4               |
| Message control 3[7] | MC3[7]            | 8       | H'F83E   | HCAN   | 16            | 4               |
| Message control 3[8] | MC3[8]            | 8       | H'F83F   | HCAN   | 16            | 4               |
| Message control 4[1] | MC4[1]            | 8       | H'F840   | HCAN   | 16            | 4               |
| Message control 4[2] | MC4[2]            | 8       | H'F841   | HCAN   | 16            | 4               |
| Message control 4[3] | MC4[3]            | 8       | H'F842   | HCAN   | 16            | 4               |
| Message control 4[4] | MC4[4]            | 8       | H'F843   | HCAN   | 16            | 4               |
| Message control 4[5] | MC4[5]            | 8       | H'F844   | HCAN   | 16            | 4               |
| Message control 4[6] | MC4[6]            | 8       | H'F845   | HCAN   | 16            | 4               |
| Message control 4[7] | MC4[7]            | 8       | H'F846   | HCAN   | 16            | 4               |
| Message control 4[8] | MC4[8]            | 8       | H'F847   | HCAN   | 16            | 4               |
| Message control 5[1] | MC5[1]            | 8       | H'F848   | HCAN   | 16            | 4               |
| Message control 5[2] | MC5[2]            | 8       | H'F849   | HCAN   | 16            | 4               |

| Register Name        | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|----------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message control 5[3] | MC5[3]            | 8       | H'F84A   | HCAN   | 16            | 4               |
| Message control 5[4] | MC5[4]            | 8       | H'F84B   | HCAN   | 16            | 4               |
| Message control 5[5] | MC5[5]            | 8       | H'F84C   | HCAN   | 16            | 4               |
| Message control 5[6] | MC5[6]            | 8       | H'F84D   | HCAN   | 16            | 4               |
| Message control 5[7] | MC5[7]            | 8       | H'F84E   | HCAN   | 16            | 4               |
| Message control 5[8] | MC5[8]            | 8       | H'F84F   | HCAN   | 16            | 4               |
| Message control 6[1] | MC6[1]            | 8       | H'F850   | HCAN   | 16            | 4               |
| Message control 6[2] | MC6[2]            | 8       | H'F851   | HCAN   | 16            | 4               |
| Message control 6[3] | MC6[3]            | 8       | H'F852   | HCAN   | 16            | 4               |
| Message control 6[4] | MC6[4]            | 8       | H'F853   | HCAN   | 16            | 4               |
| Message control 6[5] | MC6[5]            | 8       | H'F854   | HCAN   | 16            | 4               |
| Message control 6[6] | MC6[6]            | 8       | H'F855   | HCAN   | 16            | 4               |
| Message control 6[7] | MC6[7]            | 8       | H'F856   | HCAN   | 16            | 4               |
| Message control 6[8] | MC6[8]            | 8       | H'F857   | HCAN   | 16            | 4               |
| Message control 7[1] | MC7[1]            | 8       | H'F858   | HCAN   | 16            | 4               |
| Message control 7[2] | MC7[2]            | 8       | H'F859   | HCAN   | 16            | 4               |
| Message control 7[3] | MC7[3]            | 8       | H'F85A   | HCAN   | 16            | 4               |
| Message control 7[4] | MC7[4]            | 8       | H'F85B   | HCAN   | 16            | 4               |
| Message control 7[5] | MC7[5]            | 8       | H'F85C   | HCAN   | 16            | 4               |
| Message control 7[6] | MC7[6]            | 8       | H'F85D   | HCAN   | 16            | 4               |
| Message control 7[7] | MC7[7]            | 8       | H'F85E   | HCAN   | 16            | 4               |
| Message control 7[8] | MC7[8]            | 8       | H'F85F   | HCAN   | 16            | 4               |
| Message control 8[1] | MC8[1]            | 8       | H'F860   | HCAN   | 16            | 4               |
| Message control 8[2] | MC8[2]            | 8       | H'F861   | HCAN   | 16            | 4               |
| Message control 8[3] | MC8[3]            | 8       | H'F862   | HCAN   | 16            | 4               |
| Message control 8[4] | MC8[4]            | 8       | H'F863   | HCAN   | 16            | 4               |
| Message control 8[5] | MC8[5]            | 8       | H'F864   | HCAN   | 16            | 4               |
| Message control 8[6] | MC8[6]            | 8       | H'F865   | HCAN   | 16            | 4               |
| Message control 8[7] | MC8[7]            | 8       | H'F866   | HCAN   | 16            | 4               |
| Message control 8[8] | MC8[8]            | 8       | H'F867   | HCAN   | 16            | 4               |
| Message control 9[1] | MC9[1]            | 8       | H'F868   | HCAN   | 16            | 4               |
| Message control 9[2] | MC9[2]            | 8       | H'F869   | HCAN   | 16            | 4               |
| Message control 9[3] | MC9[3]            | 8       | H'F86A   | HCAN   | 16            | 4               |

| Register Name         | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-----------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message control 9[4]  | MC9[4]            | 8       | H'F86B   | HCAN   | 16            | 4               |
| Message control 9[5]  | MC9[5]            | 8       | H'F86C   | HCAN   | 16            | 4               |
| Message control 9[6]  | MC9[6]            | 8       | H'F86D   | HCAN   | 16            | 4               |
| Message control 9[7]  | MC9[7]            | 8       | H'F86E   | HCAN   | 16            | 4               |
| Message control 9[8]  | MC9[8]            | 8       | H'F86F   | HCAN   | 16            | 4               |
| Message control 10[1] | MC10[1]           | 8       | H'F870   | HCAN   | 16            | 4               |
| Message control 10[2] | MC10[2]           | 8       | H'F871   | HCAN   | 16            | 4               |
| Message control 10[3] | MC10[3]           | 8       | H'F872   | HCAN   | 16            | 4               |
| Message control 10[4] | MC10[4]           | 8       | H'F873   | HCAN   | 16            | 4               |
| Message control 10[5] | MC10[5]           | 8       | H'F874   | HCAN   | 16            | 4               |
| Message control 10[6] | MC10[6]           | 8       | H'F875   | HCAN   | 16            | 4               |
| Message control 10[7] | MC10[7]           | 8       | H'F876   | HCAN   | 16            | 4               |
| Message control 10[8] | MC10[8]           | 8       | H'F877   | HCAN   | 16            | 4               |
| Message control 11[1] | MC11[1]           | 8       | H'F878   | HCAN   | 16            | 4               |
| Message control 11[2] | MC11[2]           | 8       | H'F879   | HCAN   | 16            | 4               |
| Message control 11[3] | MC11[3]           | 8       | H'F87A   | HCAN   | 16            | 4               |
| Message control 11[4] | MC11[4]           | 8       | H'F87B   | HCAN   | 16            | 4               |
| Message control 11[5] | MC11[5]           | 8       | H'F87C   | HCAN   | 16            | 4               |
| Message control 11[6] | MC11[6]           | 8       | H'F87D   | HCAN   | 16            | 4               |
| Message control 11[7] | MC11[7]           | 8       | H'F87E   | HCAN   | 16            | 4               |
| Message control 11[8] | MC11[8]           | 8       | H'F87F   | HCAN   | 16            | 4               |
| Message control 12[1] | MC12[1]           | 8       | H'F880   | HCAN   | 16            | 4               |
| Message control 12[2] | MC12[2]           | 8       | H'F881   | HCAN   | 16            | 4               |
| Message control 12[3] | MC12[3]           | 8       | H'F882   | HCAN   | 16            | 4               |
| Message control 12[4] | MC12[4]           | 8       | H'F883   | HCAN   | 16            | 4               |
| Message control 12[5] | MC12[5]           | 8       | H'F884   | HCAN   | 16            | 4               |
| Message control 12[6] | MC12[6]           | 8       | H'F885   | HCAN   | 16            | 4               |
| Message control 12[7] | MC12[7]           | 8       | H'F886   | HCAN   | 16            | 4               |
| Message control 12[8] | MC12[8]           | 8       | H'F887   | HCAN   | 16            | 4               |
| Message control 13[1] | MC13[1]           | 8       | H'F888   | HCAN   | 16            | 4               |
| Message control 13[2] | MC13[2]           | 8       | H'F889   | HCAN   | 16            | 4               |
| Message control 13[3] | MC13[3]           | 8       | H'F88A   | HCAN   | 16            | 4               |
| Message control 13[4] | MC13[4]           | 8       | H'F88B   | HCAN   | 16            | 4               |

| Register Name         | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-----------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message control 13[5] | MC13[5]           | 8       | H'F88C   | HCAN   | 16            | 4               |
| Message control 13[6] | MC13[6]           | 8       | H'F88D   | HCAN   | 16            | 4               |
| Message control 13[7] | MC13[7]           | 8       | H'F88E   | HCAN   | 16            | 4               |
| Message control 13[8] | MC13[8]           | 8       | H'F88F   | HCAN   | 16            | 4               |
| Message control 14[1] | MC14[1]           | 8       | H'F890   | HCAN   | 16            | 4               |
| Message control 14[2] | MC14[2]           | 8       | H'F891   | HCAN   | 16            | 4               |
| Message control 14[3] | MC14[3]           | 8       | H'F892   | HCAN   | 16            | 4               |
| Message control 14[4] | MC14[4]           | 8       | H'F893   | HCAN   | 16            | 4               |
| Message control 14[5] | MC14[5]           | 8       | H'F894   | HCAN   | 16            | 4               |
| Message control 14[6] | MC14[6]           | 8       | H'F895   | HCAN   | 16            | 4               |
| Message control 14[7] | MC14[7]           | 8       | H'F896   | HCAN   | 16            | 4               |
| Message control 14[8] | MC14[8]           | 8       | H'F897   | HCAN   | 16            | 4               |
| Message control 15[1] | MC15[1]           | 8       | H'F898   | HCAN   | 16            | 4               |
| Message control 15[2] | MC15[2]           | 8       | H'F899   | HCAN   | 16            | 4               |
| Message control 15[3] | MC15[3]           | 8       | H'F89A   | HCAN   | 16            | 4               |
| Message control 15[4] | MC15[4]           | 8       | H'F89B   | HCAN   | 16            | 4               |
| Message control 15[5] | MC15[5]           | 8       | H'F89C   | HCAN   | 16            | 4               |
| Message control 15[6] | MC15[6]           | 8       | H'F89D   | HCAN   | 16            | 4               |
| Message control 15[7] | MC15[7]           | 8       | H'F89E   | HCAN   | 16            | 4               |
| Message control 15[8] | MC15[8]           | 8       | H'F89F   | HCAN   | 16            | 4               |
| Message data 0[1]     | MD0[1]            | 8       | H'F8B0   | HCAN   | 16            | 4               |
| Message data 0[2]     | MD0[2]            | 8       | H'F8B1   | HCAN   | 16            | 4               |
| Message data 0[3]     | MD0[3]            | 8       | H'F8B2   | HCAN   | 16            | 4               |
| Message data 0[4]     | MD0[4]            | 8       | H'F8B3   | HCAN   | 16            | 4               |
| Message data 0[5]     | MD0[5]            | 8       | H'F8B4   | HCAN   | 16            | 4               |
| Message data 0[6]     | MD0[6]            | 8       | H'F8B5   | HCAN   | 16            | 4               |
| Message data 0[7]     | MD0[7]            | 8       | H'F8B6   | HCAN   | 16            | 4               |
| Message data 0[8]     | MD0[8]            | 8       | H'F8B7   | HCAN   | 16            | 4               |
| Message data 1[1]     | MD1[1]            | 8       | H'F8B8   | HCAN   | 16            | 4               |
| Message data 1[2]     | MD1[2]            | 8       | H'F8B9   | HCAN   | 16            | 4               |
| Message data 1[3]     | MD1[3]            | 8       | H'F8BA   | HCAN   | 16            | 4               |
| Message data 1[4]     | MD1[4]            | 8       | H'F8BB   | HCAN   | 16            | 4               |
| Message data 1[5]     | MD1[5]            | 8       | H'F8BC   | HCAN   | 16            | 4               |

| Register Name     | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message data 1[6] | MD1[6]            | 8       | H'F8BD   | HCAN   | 16            | 4               |
| Message data 1[7] | MD1[7]            | 8       | H'F8BE   | HCAN   | 16            | 4               |
| Message data 1[8] | MD1[8]            | 8       | H'F8BF   | HCAN   | 16            | 4               |
| Message data 2[1] | MD2[1]            | 8       | H'F8C0   | HCAN   | 16            | 4               |
| Message data 2[2] | MD2[2]            | 8       | H'F8C1   | HCAN   | 16            | 4               |
| Message data 2[3] | MD2[3]            | 8       | H'F8C2   | HCAN   | 16            | 4               |
| Message data 2[4] | MD2[4]            | 8       | H'F8C3   | HCAN   | 16            | 4               |
| Message data 2[5] | MD2[5]            | 8       | H'F8C4   | HCAN   | 16            | 4               |
| Message data 2[6] | MD2[6]            | 8       | H'F8C5   | HCAN   | 16            | 4               |
| Message data 2[7] | MD2[7]            | 8       | H'F8C6   | HCAN   | 16            | 4               |
| Message data 2[8] | MD2[8]            | 8       | H'F8C7   | HCAN   | 16            | 4               |
| Message data 3[1] | MD3[1]            | 8       | H'F8C8   | HCAN   | 16            | 4               |
| Message data 3[2] | MD3[2]            | 8       | H'F8C9   | HCAN   | 16            | 4               |
| Message data 3[3] | MD3[3]            | 8       | H'F8CA   | HCAN   | 16            | 4               |
| Message data 3[4] | MD3[4]            | 8       | H'F8CB   | HCAN   | 16            | 4               |
| Message data 3[5] | MD3[5]            | 8       | H'F8CC   | HCAN   | 16            | 4               |
| Message data 3[6] | MD3[6]            | 8       | H'F8CD   | HCAN   | 16            | 4               |
| Message data 3[7] | MD3[7]            | 8       | H'F8CE   | HCAN   | 16            | 4               |
| Message data 3[8] | MD3[8]            | 8       | H'F8CF   | HCAN   | 16            | 4               |
| Message data 4[1] | MD4[1]            | 8       | H'F8D0   | HCAN   | 16            | 4               |
| Message data 4[2] | MD4[2]            | 8       | H'F8D1   | HCAN   | 16            | 4               |
| Message data 4[3] | MD4[3]            | 8       | H'F8D2   | HCAN   | 16            | 4               |
| Message data 4[4] | MD4[4]            | 8       | H'F8D3   | HCAN   | 16            | 4               |
| Message data 4[5] | MD4[5]            | 8       | H'F8D4   | HCAN   | 16            | 4               |
| Message data 4[6] | MD4[6]            | 8       | H'F8D5   | HCAN   | 16            | 4               |
| Message data 4[7] | MD4[7]            | 8       | H'F8D6   | HCAN   | 16            | 4               |
| Message data 4[8] | MD4[8]            | 8       | H'F8D7   | HCAN   | 16            | 4               |
| Message data 5[1] | MD5[1]            | 8       | H'F8D8   | HCAN   | 16            | 4               |
| Message data 5[2] | MD5[2]            | 8       | H'F8D9   | HCAN   | 16            | 4               |
| Message data 5[3] | MD5[3]            | 8       | H'F8DA   | HCAN   | 16            | 4               |
| Message data 5[4] | MD5[4]            | 8       | H'F8DB   | HCAN   | 16            | 4               |
| Message data 5[5] | MD5[5]            | 8       | H'F8DC   | HCAN   | 16            | 4               |
| Message data 5[6] | MD5[6]            | 8       | H'F8DD   | HCAN   | 16            | 4               |

| Register Name     | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message data 5[7] | MD5[7]            | 8       | H'F8DE   | HCAN   | 16            | 4               |
| Message data 5[8] | MD5[8]            | 8       | H'F8DF   | HCAN   | 16            | 4               |
| Message data 6[1] | MD6[1]            | 8       | H'F8E0   | HCAN   | 16            | 4               |
| Message data 6[2] | MD6[2]            | 8       | H'F8E1   | HCAN   | 16            | 4               |
| Message data 6[3] | MD6[3]            | 8       | H'F8E2   | HCAN   | 16            | 4               |
| Message data 6[4] | MD6[4]            | 8       | H'F8E3   | HCAN   | 16            | 4               |
| Message data 6[5] | MD6[5]            | 8       | H'F8E4   | HCAN   | 16            | 4               |
| Message data 6[6] | MD6[6]            | 8       | H'F8E5   | HCAN   | 16            | 4               |
| Message data 6[7] | MD6[7]            | 8       | H'F8E6   | HCAN   | 16            | 4               |
| Message data 6[8] | MD6[8]            | 8       | H'F8E7   | HCAN   | 16            | 4               |
| Message data 7[1] | MD7[1]            | 8       | H'F8E8   | HCAN   | 16            | 4               |
| Message data 7[2] | MD7[2]            | 8       | H'F8E9   | HCAN   | 16            | 4               |
| Message data 7[3] | MD7[3]            | 8       | H'F8EA   | HCAN   | 16            | 4               |
| Message data 7[4] | MD7[4]            | 8       | H'F8EB   | HCAN   | 16            | 4               |
| Message data 7[5] | MD7[5]            | 8       | H'F8EC   | HCAN   | 16            | 4               |
| Message data 7[6] | MD7[6]            | 8       | H'F8ED   | HCAN   | 16            | 4               |
| Message data 7[7] | MD7[7]            | 8       | H'F8EE   | HCAN   | 16            | 4               |
| Message data 7[8] | MD7[8]            | 8       | H'F8EF   | HCAN   | 16            | 4               |
| Message data 8[1] | MD8[1]            | 8       | H'F8F0   | HCAN   | 16            | 4               |
| Message data 8[2] | MD8[2]            | 8       | H'F8F1   | HCAN   | 16            | 4               |
| Message data 8[3] | MD8[3]            | 8       | H'F8F2   | HCAN   | 16            | 4               |
| Message data 8[4] | MD8[4]            | 8       | H'F8F3   | HCAN   | 16            | 4               |
| Message data 8[5] | MD8[5]            | 8       | H'F8F4   | HCAN   | 16            | 4               |
| Message data 8[6] | MD8[6]            | 8       | H'F8F5   | HCAN   | 16            | 4               |
| Message data 8[7] | MD8[7]            | 8       | H'F8F6   | HCAN   | 16            | 4               |
| Message data 8[8] | MD8[8]            | 8       | H'F8F7   | HCAN   | 16            | 4               |
| Message data 9[1] | MD9[1]            | 8       | H'F8F8   | HCAN   | 16            | 4               |
| Message data 9[2] | MD9[2]            | 8       | H'F8F9   | HCAN   | 16            | 4               |
| Message data 9[3] | MD9[3]            | 8       | H'F8FA   | HCAN   | 16            | 4               |
| Message data 9[4] | MD9[4]            | 8       | H'F8FB   | HCAN   | 16            | 4               |
| Message data 9[5] | MD9[5]            | 8       | H'F8FC   | HCAN   | 16            | 4               |
| Message data 9[6] | MD9[6]            | 8       | H'F8FD   | HCAN   | 16            | 4               |
| Message data 9[7] | MD9[7]            | 8       | H'F8FE   | HCAN   | 16            | 4               |
| Message data 9[8] | MD9[8]            | 8       | H'F8FF   | HCAN   | 16            | 4               |

| Register Name      | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|--------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message data 10[1] | MD10[1]           | 8       | H'F900   | HCAN   | 16            | 4               |
| Message data 10[2] | MD10[2]           | 8       | H'F901   | HCAN   | 16            | 4               |
| Message data 10[3] | MD10[3]           | 8       | H'F902   | HCAN   | 16            | 4               |
| Message data 10[4] | MD10[4]           | 8       | H'F903   | HCAN   | 16            | 4               |
| Message data 10[5] | MD10[5]           | 8       | H'F904   | HCAN   | 16            | 4               |
| Message data 10[6] | MD10[6]           | 8       | H'F905   | HCAN   | 16            | 4               |
| Message data 10[7] | MD10[7]           | 8       | H'F906   | HCAN   | 16            | 4               |
| Message data 10[8] | MD10[8]           | 8       | H'F907   | HCAN   | 16            | 4               |
| Message data 11[1] | MD11[1]           | 8       | H'F908   | HCAN   | 16            | 4               |
| Message data 11[2] | MD11[2]           | 8       | H'F909   | HCAN   | 16            | 4               |
| Message data 11[3] | MD11[3]           | 8       | H'F90A   | HCAN   | 16            | 4               |
| Message data 11[4] | MD11[4]           | 8       | H'F90B   | HCAN   | 16            | 4               |
| Message data 11[5] | MD11[5]           | 8       | H'F90C   | HCAN   | 16            | 4               |
| Message data 11[6] | MD11[6]           | 8       | H'F90D   | HCAN   | 16            | 4               |
| Message data 11[7] | MD11[7]           | 8       | H'F90E   | HCAN   | 16            | 4               |
| Message data 11[8] | MD11[8]           | 8       | H'F90F   | HCAN   | 16            | 4               |
| Message data 12[1] | MD12[1]           | 8       | H'F910   | HCAN   | 16            | 4               |
| Message data 12[2] | MD12[2]           | 8       | H'F911   | HCAN   | 16            | 4               |
| Message data 12[3] | MD12[3]           | 8       | H'F912   | HCAN   | 16            | 4               |
| Message data 12[4] | MD12[4]           | 8       | H'F913   | HCAN   | 16            | 4               |
| Message data 12[5] | MD12[5]           | 8       | H'F914   | HCAN   | 16            | 4               |
| Message data 12[6] | MD12[6]           | 8       | H'F915   | HCAN   | 16            | 4               |
| Message data 12[7] | MD12[7]           | 8       | H'F916   | HCAN   | 16            | 4               |
| Message data 12[8] | MD12[8]           | 8       | H'F917   | HCAN   | 16            | 4               |
| Message data 13[1] | MD13[1]           | 8       | H'F918   | HCAN   | 16            | 4               |
| Message data 13[2] | MD13[2]           | 8       | H'F919   | HCAN   | 16            | 4               |
| Message data 13[3] | MD13[3]           | 8       | H'F91A   | HCAN   | 16            | 4               |
| Message data 13[4] | MD13[4]           | 8       | H'F91B   | HCAN   | 16            | 4               |
| Message data 13[5] | MD13[5]           | 8       | H'F91C   | HCAN   | 16            | 4               |
| Message data 13[6] | MD13[6]           | 8       | H'F91D   | HCAN   | 16            | 4               |
| Message data 13[7] | MD13[7]           | 8       | H'F91E   | HCAN   | 16            | 4               |
| Message data 13[8] | MD13[8]           | 8       | H'F91F   | HCAN   | 16            | 4               |
| Message data 14[1] | MD14[1]           | 8       | H'F920   | HCAN   | 16            | 4               |

| Register Name                                       | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-----------------------------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Message data 14[2]                                  | MD14[2]           | 8       | H'F921   | HCAN   | 16            | 4               |
| Message data 14[3]                                  | MD14[3]           | 8       | H'F922   | HCAN   | 16            | 4               |
| Message data 14[4]                                  | MD14[4]           | 8       | H'F923   | HCAN   | 16            | 4               |
| Message data 14[5]                                  | MD14[5]           | 8       | H'F924   | HCAN   | 16            | 4               |
| Message data 14[6]                                  | MD14[6]           | 8       | H'F925   | HCAN   | 16            | 4               |
| Message data 14[7]                                  | MD14[7]           | 8       | H'F926   | HCAN   | 16            | 4               |
| Message data 14[8]                                  | MD14[8]           | 8       | H'F927   | HCAN   | 16            | 4               |
| Message data 15[1]                                  | MD15[1]           | 8       | H'F928   | HCAN   | 16            | 4               |
| Message data 15[2]                                  | MD15[2]           | 8       | H'F929   | HCAN   | 16            | 4               |
| Message data 15[3]                                  | MD15[3]           | 8       | H'F92A   | HCAN   | 16            | 4               |
| Message data 15[4]                                  | MD15[4]           | 8       | H'F92B   | HCAN   | 16            | 4               |
| Message data 15[5]                                  | MD15[5]           | 8       | H'F92C   | HCAN   | 16            | 4               |
| Message data 15[6]                                  | MD15[6]           | 8       | H'F92D   | HCAN   | 16            | 4               |
| Message data 15[7]                                  | MD15[7]           | 8       | H'F92E   | HCAN   | 16            | 4               |
| Message data 15[8]                                  | MD15[8]           | 8       | H'F92F   | HCAN   | 16            | 4               |
| HCAN monitor register                               | HCANMON           | 8       | H'FA00   | HCAN   | 16            | 4               |
| Timer mode register                                 | TMDR              | 8       | H'FB00   | MMT    | 16            | 3               |
| Timer control register                              | TCNR              | 8       | H'FB02   | MMT    | 16            | 3               |
| Timer status register                               | TSR               | 8       | H'FB04   | MMT    | 16            | 3               |
| Timer counter                                       | TCNT              | 16      | H'FB06   | MMT    | 16            | 3               |
| Timer period data register                          | TPDR              | 16      | H'FB08   | MMT    | 16            | 3               |
| Timer period buffer register                        | TPBR              | 16      | H'FB0A   | MMT    | 16            | 3               |
| Timer dead time data register                       | TDDR              | 16      | H'FB0C   | MMT    | 16            | 3               |
| MMT pin control register                            | MMTPC             | 8       | H'FB0E   | MMT    | 16            | 3               |
| Timer buffer register U<br>(for a buffer operation) | TBRU              | 16      | H'FB10   | MMT    | 16            | 3               |
| Timer general register UU                           | TGRUU             | 16      | H'FB12   | MMT    | 16            | 3               |
| Timer general register U                            | TGRU              | 16      | H'FB14   | MMT    | 16            | 3               |
| Timer general register UD                           | TGRUD             | 16      | H'FB16   | MMT    | 16            | 3               |
| Timer dead time counter 0                           | TDCNT0            | 16      | H'FB18   | MMT    | 16            | 3               |
| Timer dead time counter 1                           | TDCNT1            | 16      | H'FB1A   | MMT    | 16            | 3               |
| Timer buffer register U<br>(for a free operation)   | TBRU              | 16      | H'FB1C   | MMT    | 16            | 3               |
| Timer buffer register V<br>(for a buffer operation) | TBRV              | 16      | H'FB20   | MMT    | 16            | 3               |

| Register Name                                       | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-----------------------------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Timer general register VU                           | TGRVU             | 16      | H'FB22   | MMT    | 16            | 3               |
| Timer general register V                            | TGRV              | 16      | H'FB24   | MMT    | 16            | 3               |
| Timer general register VD                           | TGRVD             | 16      | H'FB26   | MMT    | 16            | 3               |
| Timer dead time counter 2                           | TDCNT2            | 16      | H'FB28   | MMT    | 16            | 3               |
| Timer dead time counter 3                           | TDCNT3            | 16      | H'FB2A   | MMT    | 16            | 3               |
| Timer buffer register V<br>(for a free operation)   | TBRV              | 16      | H'FB2C   | MMT    | 16            | 3               |
| Timer buffer register W<br>(for a buffer operation) | TBRW              | 16      | H'FB30   | MMT    | 16            | 3               |
| Timer general register WU                           | TGRWU             | 16      | H'FB32   | MMT    | 16            | 3               |
| Timer general register W                            | TGRW              | 16      | H'FB34   | MMT    | 16            | 3               |
| Timer general register WD                           | TGRWD             | 16      | H'FB36   | MMT    | 16            | 3               |
| Timer dead time counter4                            | TDCNT4            | 16      | H'FB38   | MMT    | 16            | 3               |
| Timer dead time counter5                            | TDCNT5            | 16      | H'FB3A   | MMT    | 16            | 3               |
| Timer buffer register W<br>(for a free operation)   | TBRW              | 16      | H'FB3C   | MMT    | 16            | 3               |
| Input level control/status register                 | ICSR              | 16      | H'FC00   | POE    | 16            | 3               |
| POE pin control register                            | POEPC             | 8       | H'FC02   | POE    | 16            | 3               |
| Standby control register                            | SBYCR             | 8       | H'FDE4   | SYSTEM | 8             | 2               |
| System control register                             | SYSCR             | 8       | H'FDE5   | SYSTEM | 8             | 2               |
| 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               |
| Low-power control register                          | LPWRCR            | 8       | H'FDEC   | SYSTEM | 8             | 2               |
| Break address register A                            | BARA              | 32      | H'FE00   | PBC    | 32            | 2               |
| Break address register B                            | BARB              | 32      | H'FE04   | PBC    | 32            | 2               |
| Break control register A                            | BCRA              | 8       | H'FE08   | PBC    | 8             | 2               |
| Break control register B                            | BCRB              | 8       | H'FE09   | PBC    | 8             | 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               |

| Register Name                       | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-------------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| 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 G               | DTCERG            | 8       | H'FE1C   | DTC    | 8             | 2               |
| DTC vector register                 | DTVECR            | 8       | H'FE1F   | DTC    | 8             | 2               |
| PPG output control register         | PCR               | 8       | H'FE26   | PPG    | 8             | 2               |
| PPG output mode register            | PMR               | 8       | H'FE27   | PPG    | 8             | 2               |
| Next data enable register H         | NDERH             | 8       | H'FE28   | PPG    | 8             | 2               |
| Next data enable register L         | NDERL             | 8       | H'FE29   | PPG    | 8             | 2               |
| Output data register H              | PODRH             | 8       | H'FE2A   | PPG    | 8             | 2               |
| Output data register L              | PODRL             | 8       | H'FE2B   | PPG    | 8             | 2               |
| Next data register H                | NDRH              | 8       | H'FE2C   | PPG    | 8             | 2               |
| Next data register L                | NDRL              | 8       | H'FE2D   | PPG    | 8             | 2               |
| Next data register H                | NDRH              | 8       | H'FE2E   | PPG    | 8             | 2               |
| Next data register L                | NDRL              | 8       | H'FE2F   | PPG    | 8             | 2               |
| Port 1 data direction register      | P1DDR             | 8       | H'FE30   | 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 F data direction register      | PFDDR             | 8       | H'FE3E   | 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 A open drain control register  | PAODR             | 8       | H'FE47   | PORT   | 8             | 2               |
| Port B open drain control register  | PBODR             | 8       | H'FE48   | PORT   | 8             | 2               |
| Port C open drain control register  | PCODR             | 8       | H'FE49   | PORT   | 8             | 2               |

| Register Name                     | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-----------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Timer control register_3          | TCR_3             | 8       | H'FE80   | TPU_3  | 16            | 2               |
| Timer mode register_3             | TMDR_3            | 8       | H'FE81   | TPU_3  | 16            | 2               |
| Timer I/O control register H_3    | TIORH_3           | 8       | H'FE82   | TPU_3  | 16            | 2               |
| Timer I/O control register L_3    | TIORL_3           | 8       | H'FE83   | TPU_3  | 16            | 2               |
| Timer interrupt enable register_3 | TIER_3            | 8       | H'FE84   | TPU_3  | 16            | 2               |
| Timer status register_3           | TSR_3             | 8       | H'FE85   | TPU_3  | 16            | 2               |
| Timer counter H_3                 | TCNTH_3           | 8       | H'FE86   | TPU_3  | 16            | 2               |
| Timer counter L_3                 | TCNTL_3           | 8       | H'FE87   | TPU_3  | 16            | 2               |
| Timer general register AH_3       | TGRAH_3           | 8       | H'FE88   | TPU_3  | 16            | 2               |
| Timer general register AL_3       | TGRAL_3           | 8       | H'FE89   | TPU_3  | 16            | 2               |
| Timer general register BH_3       | TGRBH_3           | 8       | H'FE8A   | TPU_3  | 16            | 2               |
| Timer general register BL_3       | TGRBL_3           | 8       | H'FE8B   | TPU_3  | 16            | 2               |
| Timer general register CH_3       | TGRCH_3           | 8       | H'FE8C   | TPU_3  | 16            | 2               |
| Timer general register CL_3       | TGRCL_3           | 8       | H'FE8D   | TPU_3  | 16            | 2               |
| Timer general register DH_3       | TGRDH_3           | 8       | H'FE8E   | TPU_3  | 16            | 2               |
| Timer general register DL_3       | TGRDL_3           | 8       | H'FE8F   | TPU_3  | 16            | 2               |
| Timer control register_4          | TCR_4             | 8       | H'FE90   | TPU_4  | 16            | 2               |
| Timer mode register_4             | TMDR_4            | 8       | H'FE91   | TPU_4  | 16            | 2               |
| Timer I/O control register_4      | TIOR_4            | 8       | H'FE92   | TPU_4  | 16            | 2               |
| Timer interrupt enable register_4 | TIER_4            | 8       | H'FE94   | TPU_4  | 16            | 2               |
| Timer status register_4           | TSR_4             | 8       | H'FE95   | TPU_4  | 16            | 2               |
| Timer counter H_4                 | TCNTH_4           | 8       | H'FE96   | TPU_4  | 16            | 2               |
| Timer counter L_4                 | TCNTL_4           | 8       | H'FE97   | TPU_4  | 16            | 2               |
| Timer general register AH_4       | TGRAH_4           | 8       | H'FE98   | TPU_4  | 16            | 2               |
| Timer general register AL_4       | TGRAL_4           | 8       | H'FE99   | TPU_4  | 16            | 2               |
| Timer general register BH_4       | TGRBH_4           | 8       | H'FE9A   | TPU_4  | 16            | 2               |
| Timer general register BL_4       | TGRBL_4           | 8       | H'FE9B   | TPU_4  | 16            | 2               |
| Timer control register_5          | TCR_5             | 8       | H'FEA0   | TPU_5  | 16            | 2               |
| Timer mode register_5             | TMDR_5            | 8       | H'FEA1   | TPU_5  | 16            | 2               |
| Timer I/O control register_5      | TIOR_5            | 8       | H'FEA2   | TPU_5  | 16            | 2               |
| Timer interrupt enable register_5 | TIER_5            | 8       | H'FEA4   | TPU_5  | 16            | 2               |
| Timer status register_5           | TSR_5             | 8       | H'FEA5   | TPU_5  | 16            | 2               |

| Register Name                     | Abbrevia-<br>tion | Bit No. | Address* | Module        | Data<br>Width | Access<br>State |
|-----------------------------------|-------------------|---------|----------|---------------|---------------|-----------------|
| Timer counter H_5                 | TCNTH_5           | 8       | H'FEA6   | TPU_5         | 16            | 2               |
| Timer counter L_5                 | TCNTL_5           | 8       | H'FEA7   | TPU_5         | 16            | 2               |
| Timer general register AH_5       | TGRAH_5           | 8       | H'FEA8   | TPU_5         | 16            | 2               |
| Timer general register AL_5       | TGRAL_5           | 8       | H'FEA9   | TPU_5         | 16            | 2               |
| Timer general register BH_5       | TGRBH_5           | 8       | H'FEAA   | TPU_5         | 16            | 2               |
| Timer general register BL_5       | TGRBL_5           | 8       | H'FEAB   | TPU_5         | 16            | 2               |
| Timer start register              | TSTR              | 8       | H'FEB0   | TPU<br>common | 16            | 2               |
| Timer synchro register            | TSYR              | 8       | H'FEB1   | TPU<br>common | 16            | 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 J     | IPRJ              | 8       | H'FEC9   | INT           | 8             | 2               |
| Interrupt priority register K     | IPRK              | 8       | H'FECA   | INT           | 8             | 2               |
| Interrupt priority register M     | IPRM              | 8       | H'FECC   | INT           | 8             | 2               |
| RAM emulation register            | RAMER             | 8       | H'FEDB   | ROM           | 8             | 2               |
| Port 1 data register              | P1DR              | 8       | H'FF00   | 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 F data register              | PFDR              | 8       | H'FF0E   | PORT          | 8             | 2               |
| Timer control register_0          | TCR_0             | 8       | H'FF10   | TPU_0         | 16            | 2               |
| Timer mode register_0             | TMDR_0            | 8       | H'FF11   | TPU_0         | 16            | 2               |
| Timer I/O control register H_0    | TIORH_0           | 8       | H'FF12   | TPU_0         | 16            | 2               |
| Timer I/O control register L_0    | TIORL_0           | 8       | H'FF13   | TPU_0         | 16            | 2               |
| Timer interrupt enable register_0 | TIER_0            | 8       | H'FF14   | TPU_0         | 16            | 2               |
| Timer status register_0           | TSR_0             | 8       | H'FF15   | TPU_0         | 16            | 2               |

| Register Name                     | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-----------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Timer counter H_0                 | TCNTH_0           | 8       | H'FF16   | TPU_0  | 16            | 2               |
| Timer counter L_0                 | TCNTL_0           | 8       | H'FF17   | TPU_0  | 16            | 2               |
| Timer general register AH_0       | TGRAH_0           | 8       | H'FF18   | TPU_0  | 16            | 2               |
| Timer general register AL_0       | TGRAL_0           | 8       | H'FF19   | TPU_0  | 16            | 2               |
| Timer general register BH_0       | TGRBH_0           | 8       | H'FF1A   | TPU_0  | 16            | 2               |
| Timer general register BL_0       | TGRBL_0           | 8       | H'FF1B   | TPU_0  | 16            | 2               |
| Timer general register CH_0       | TGRCH_0           | 8       | H'FF1C   | TPU_0  | 16            | 2               |
| Timer general register CL_0       | TGRCL_0           | 8       | H'FF1D   | TPU_0  | 16            | 2               |
| Timer general register DH_0       | TGRDH_0           | 8       | H'FF1E   | TPU_0  | 16            | 2               |
| Timer general register DL_0       | TGRDL_0           | 8       | H'FF1F   | TPU_0  | 16            | 2               |
| Timer control register_1          | TCR_1             | 8       | H'FF20   | TPU_1  | 16            | 2               |
| Timer mode register_1             | TMDR_1            | 8       | H'FF21   | TPU_1  | 16            | 2               |
| Timer I/O control register_1      | TIOR_1            | 8       | H'FF22   | TPU_1  | 16            | 2               |
| Timer interrupt enable register_1 | TIER_1            | 8       | H'FF24   | TPU_1  | 16            | 2               |
| Timer status register_1           | TSR_1             | 8       | H'FF25   | TPU_1  | 16            | 2               |
| Timer counter H_1                 | TCNTH_1           | 8       | H'FF26   | TPU_1  | 16            | 2               |
| Timer counter L_1                 | TCNTL_1           | 8       | H'FF27   | TPU_1  | 16            | 2               |
| Timer general register AH_1       | TGRAH_1           | 8       | H'FF28   | TPU_1  | 16            | 2               |
| Timer general register AL_1       | TGRAL_1           | 8       | H'FF29   | TPU_1  | 16            | 2               |
| Timer general register BH_1       | TGRBH_1           | 8       | H'FF2A   | TPU_1  | 16            | 2               |
| Timer general register BL_1       | TGRBL_1           | 8       | H'FF2B   | TPU_1  | 16            | 2               |
| Timer control register_2          | TCR_2             | 8       | H'FF30   | TPU_2  | 16            | 2               |
| Timer mode register_2             | TMDR_2            | 8       | H'FF31   | TPU_2  | 16            | 2               |
| Timer I/O control register_2      | TIOR_2            | 8       | H'FF32   | TPU_2  | 16            | 2               |
| Timer interrupt enable register_2 | TIER_2            | 8       | H'FF34   | TPU_2  | 16            | 2               |
| Timer status register_2           | TSR_2             | 8       | H'FF35   | TPU_2  | 16            | 2               |
| Timer counter H_2                 | TCNTH_2           | 8       | H'FF36   | TPU_2  | 16            | 2               |
| Timer counter L_2                 | TCNTL_2           | 8       | H'FF37   | TPU_2  | 16            | 2               |
| Timer general register AH_2       | TGRAH_2           | 8       | H'FF38   | TPU_2  | 16            | 2               |
| Timer general register AL_2       | TGRAL_2           | 8       | H'FF39   | TPU_2  | 16            | 2               |
| Timer general register BH_2       | TGRBH_2           | 8       | H'FF3A   | TPU_2  | 16            | 2               |
| Timer general register BL_2       | TGRBL_2           | 8       | H'FF3B   | TPU_2  | 16            | 2               |

| Register Name                 | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|-------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Timer control/status register | TCSR              | 8       | H'FF74   | WDT    | 16            | 2               |
| Timer counter                 | TCNT              | 8       | H'FF75   | WDT    | 16            | 2               |
| Reset control/status register | RSTCSR            | 8       | H'FF77   | WDT    | 16            | 2               |
| Serial mode register_0        | SMR_0             | 8       | H'FF78   | SCI_0  | 8             | 2               |
| Bit rate register_0           | BRR_0             | 8       | H'FF79   | SCI_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   | SCI_0  | 8             | 2               |
| Serial mode register_1        | SMR_1             | 8       | H'FF80   | SCI_1  | 8             | 2               |
| Bit rate register_1           | BRR_1             | 8       | H'FF81   | SCI_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   | SCI_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               |

| Register Name                   | Abbrevia-<br>tion | Bit No. | Address* | Module | Data<br>Width | Access<br>State |
|---------------------------------|-------------------|---------|----------|--------|---------------|-----------------|
| Flash memory control register 1 | FLMCR1            | 8       | H'FFA8   | ROM    | 8             | 2               |
| Flash memory control registe 2  | FLMCR2            | 8       | H'FFA9   | ROM    | 8             | 2               |
| Erase block register 1          | EBR1              | 8       | H'FFAA   | ROM    | 8             | 2               |
| Erase block register 2          | EBR2              | 8       | H'FFAB   | ROM    | 8             | 2               |
| Port 1 register                 | PORT1             | 8       | H'FFB0   | PORT   | 8             | 2               |
| Port 4 register                 | PORT4             | 8       | H'FFB3   | 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 F register                 | PORTF             | 8       | H'FFBE   | PORT   | 8             | 2               |

Note: \* Lower 16 bits of the address.

## A.2 Register Bits

| Register Name | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1  | Bit 0  | Module |
|---------------|---------|---------|---------|---------|---------|---------|--------|--------|--------|
| MCR           | MCR7    | —       | MCR5    | —       | —       | MCR2    | MCR1   | MCR0   | HCAN   |
| GSR           | —       | —       | —       | —       | GSR3    | GSR2    | GSR1   | GSR0   |        |
| BCR           | BCR7    | BCR6    | BCR5    | BCR4    | BCR3    | BCR2    | BCR1   | BCR0   |        |
|               | BCR15   | BCR14   | BCR13   | BCR12   | BCR11   | BCR10   | BCR9   | BCR8   |        |
| MBCR          | MBCR7   | MBCR6   | MBCR5   | MBCR4   | MBCR3   | MBCR2   | MBCR1  | —      |        |
|               | MBCR15  | MBCR14  | MBCR13  | MBCR12  | MBCR11  | MBCR10  | MBCR9  | MBCR8  |        |
| TXPR          | TXPR7   | TXPR6   | TXPR5   | TXPR4   | TXPR3   | TXPR2   | TXPR1  | —      |        |
|               | TXPR15  | TXPR14  | TXPR13  | TXPR12  | TXPR11  | TXPR10  | TXPR9  | TXPR8  |        |
| TXCR          | TXCR7   | TXCR6   | TXCR5   | TXCR4   | TXCR3   | TXCR2   | TXCR1  | —      |        |
|               | TXCR15  | TXCR14  | TXCR13  | TXCR12  | TXCR11  | TXCR10  | TXCR9  | TXCR8  |        |
| TXACK         | TXACK7  | TXACK6  | TXACK5  | TXACK4  | TXACK3  | TXACK2  | TXACK1 | —      |        |
|               | TXACK15 | TXACK14 | TXACK13 | TXACK12 | TXACK11 | TXACK10 | TXACK9 | TXACK8 |        |
| ABACK         | ABACK7  | ABACK6  | ABACK5  | ABACK4  | ABACK3  | ABACK2  | ABACK1 | —      |        |
|               | ABACK15 | ABACK14 | ABACK13 | ABACK12 | ABACK11 | ABACK10 | ABACK9 | ABACK8 |        |
| RXPR          | RXPR7   | RXPR6   | RXPR5   | RXPR4   | RXPR3   | RXPR2   | RXPR1  | RXPR0  |        |
|               | RXPR15  | RXPR14  | RXPR13  | RXPR12  | RXPR11  | RXPR10  | RXPR9  | RXPR8  |        |
| RFPR          | RFPR7   | RFPR6   | RFPR5   | RFPR4   | RFPR3   | RFPR2   | RFPR1  | RFPR0  |        |
|               | RFPR15  | RFPR14  | RFPR13  | RFPR12  | RFPR11  | RFPR10  | RFPR9  | RFPR8  |        |
| IRR           | IRR7    | IRR6    | IRR5    | IRR4    | IRR3    | IRR2    | IRR1   | IRR0   |        |
|               | —       | —       | —       | IRR12   | —       | —       | IRR9   | IRR8   |        |
| MBIMR         | MBIMR7  | MBIMR6  | MBIMR5  | MBIMR4  | MBIMR3  | MBIMR2  | MBIMR1 | MBIMR0 |        |
|               | MBIMR15 | MBIMR14 | MBIMR13 | MBIMR12 | MBIMR11 | MBIMR10 | MBIMR9 | MBIMR8 |        |
| IMR           | IMR7    | IMR6    | IMR5    | IMR4    | IMR3    | IMR2    | IMR1   | —      |        |
|               | —       | —       | —       | IMR12   | —       | —       | IMR9   | IMR8   |        |
| REC           | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1  | Bit 0  |        |
| TEC           | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1  | Bit 0  |        |
| UMSR          | UMSR7   | UMSR6   | UMSR5   | UMSR4   | UMSR3   | UMSR2   | UMSR1  | UMSR0  |        |
|               | UMSR15  | UMSR14  | UMSR13  | UMSR12  | UMSR11  | UMSR10  | UMSR9  | UMSR8  |        |
| LAFML         | LAFML7  | LAFML6  | LAFML5  | LAFML4  | LAFML3  | LAFML2  | LAFML1 | LAFML0 |        |
|               | LAFML15 | LAFML14 | LAFML13 | LAFML12 | LAFML11 | LAFML10 | LAFML9 | LAFML8 |        |
| LAFMH         | LAFMH7  | LAFMH6  | LAFMH5  | —       | —       | —       | LAFMH1 | LAFMH0 |        |
|               | LAFMH15 | LAFMH14 | LAFMH13 | LAFMH12 | LAFMH11 | LAFMH10 | LAFMH9 | LAFMH8 |        |
| MC0[1]        | —       | —       | —       | —       | DLC3    | DLC2    | DLC1   | DLC0   |        |
| MC0[2]        | —       | —       | —       | —       | —       | —       | —      | —      |        |
| MC0[3]        | —       | —       | —       | —       | —       | —       | —      | —      |        |
| MC0[4]        | —       | —       | —       | —       | —       | —       | —      | —      |        |
| MC0[5]        | ID-20   | ID-19   | ID-18   | RTR     | IDE     | —       | ID-17  | ID-16  |        |
| MC0[6]        | ID-28   | ID-27   | ID-26   | ID-25   | ID-24   | ID-23   | ID-22  | ID-21  |        |
| MC0[7]        | ID-7    | ID-6    | ID-5    | ID-4    | ID-3    | ID-2    | ID-1   | ID-0   |        |

| Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| MC0[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  | HCAN   |
| MC1[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC1[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC1[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC1[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC1[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC1[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC1[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC1[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC2[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC2[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC2[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC2[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC2[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC2[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC2[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC2[8]        | ID-15 | ID-14 | ID-13 | ID12  | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC3[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC3[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC3[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC3[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC3[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC3[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC3[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC3[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC4[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC4[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC4[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC4[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC4[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC4[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC4[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC4[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC5[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC5[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC5[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC5[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC5[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC5[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC5[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC5[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC6[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |

| Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| MC6[2]        | —     | —     | —     | —     | —     | —     | —     | —     | HCAN   |
| MC6[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC6[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC6[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC6[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC6[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC6[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC7[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC7[2]        | —     | —     | —     | —     | —     | —     | —     | —     | HCAN   |
| MC7[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC7[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC7[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC7[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC7[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC7[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC8[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  | HCAN   |
| MC8[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC8[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC8[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC8[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC8[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC8[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC8[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC9[1]        | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  | HCAN   |
| MC9[2]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC9[3]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC9[4]        | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC9[5]        | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC9[6]        | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC9[7]        | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC9[8]        | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC10[1]       | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  | HCAN   |
| MC10[2]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC10[3]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC10[4]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC10[5]       | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC10[6]       | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC10[7]       | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC10[8]       | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC11[1]       | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  | HCAN   |
| MC11[2]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC11[3]       | —     | —     | —     | —     | —     | —     | —     | —     |        |

| Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| MC11[4]       | —     | —     | —     | —     | —     | —     | —     | —     | HCAN   |
| MC11[5]       | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC11[6]       | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC11[7]       | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC11[8]       | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC12[1]       | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC12[2]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC12[3]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC12[4]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC12[5]       | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC12[6]       | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC12[7]       | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC12[8]       | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC13[1]       | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC13[2]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC13[3]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC13[4]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC13[5]       | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC13[6]       | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC13[7]       | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC13[8]       | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC14[1]       | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC14[2]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC14[3]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC14[4]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC14[5]       | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC14[6]       | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC14[7]       | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC14[8]       | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MC15[1]       | —     | —     | —     | —     | DLC3  | DLC2  | DLC1  | DLC0  |        |
| MC15[2]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC15[3]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC15[4]       | —     | —     | —     | —     | —     | —     | —     | —     |        |
| MC15[5]       | ID-20 | ID-19 | ID-18 | RTR   | IDE   | —     | ID-17 | ID-16 |        |
| MC15[6]       | ID-28 | ID-27 | ID-26 | ID-25 | ID-24 | ID-23 | ID-22 | ID-21 |        |
| MC15[7]       | ID-7  | ID-6  | ID-5  | ID-4  | ID-3  | ID-2  | ID-1  | ID-0  |        |
| MC15[8]       | ID-15 | ID-14 | ID-13 | ID-12 | ID-11 | ID-10 | ID-9  | ID-8  |        |
| MD0[1]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD0[2]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD0[3]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD0[4]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD0[5]        | 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 |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| MD5[8]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | HCAN   |
| MD6[1]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[2]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[3]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[4]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[5]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[6]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[7]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD6[8]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[1]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[2]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[3]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[4]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[5]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[6]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[7]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD7[8]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[1]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[2]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[3]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[4]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[5]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[6]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[7]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD8[8]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[1]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[2]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[3]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[4]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[5]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[6]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[7]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD9[8]        | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[1]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[2]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[3]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[4]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[5]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[6]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[7]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD10[8]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD11[1]       | 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 |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| MD11[2]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | HCAN   |
| MD11[3]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD11[4]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD11[5]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD11[6]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD11[7]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD11[8]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[1]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[2]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[3]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[4]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[5]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[6]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[7]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD12[8]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[1]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[2]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[3]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[4]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[5]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[6]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[7]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD13[8]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[1]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[2]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[3]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[4]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[5]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[6]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[7]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD14[8]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[1]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[2]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[3]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[4]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[5]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[6]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[7]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| MD15[8]       | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |        |
| HCANMON       | —     | —     | —     | —     | —     | —     | TxD   | RxD   |        |

| Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| TMDR          | —     | —     | —     | —     | OLSN  | OLSP  | MD1   | MD0   | MMT    |
| TCNR          | —     | CST   | RPRO  | —     | —     | —     | TGIEN | TGIEM |        |
| TSR           | TCFG  | —     | —     | —     | —     | —     | TGFN  | TGFM  |        |
| TCNT          | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TPDR          | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TPBR          | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TDDR          | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| MMTPC         | PWOBE | PWOAE | PVOBE | PVOAE | PUOBE | PUOAE | PCOE  | PCIE  |        |
| TBRU*1        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRUU         | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRU          | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRUD         | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TDCNT0        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TDCNT1        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TBRU*2        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TBRV*1        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRVU         | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRV          | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRVD         | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TDCNT2        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TDCNT3        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TBRV*2        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  | MMT    |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TBRW*1        | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
|               | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |

| Register Name | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1   | Bit 0   | Module |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|--------|
| TGRWU         | Bit15   | Bit14   | Bit13   | Bit12   | Bit11   | Bit10   | Bit9    | Bit8    | MMT    |
|               | Bit7    | Bit6    | Bit5    | Bit4    | Bit3    | Bit2    | Bit1    | Bit0    |        |
| TGRW          | Bit15   | Bit14   | Bit13   | Bit12   | Bit11   | Bit10   | Bit9    | Bit8    |        |
|               | Bit7    | Bit6    | Bit5    | Bit4    | Bit3    | Bit2    | Bit1    | Bit0    |        |
| TGRWD         | Bit15   | Bit14   | Bit13   | Bit12   | Bit11   | Bit10   | Bit9    | Bit8    |        |
|               | Bit7    | Bit6    | Bit5    | Bit4    | Bit3    | Bit2    | Bit1    | Bit0    |        |
| TDCNT4        | Bit15   | Bit14   | Bit13   | Bit12   | Bit11   | Bit10   | Bit9    | Bit8    |        |
|               | Bit7    | Bit6    | Bit5    | Bit4    | Bit3    | Bit2    | Bit1    | Bit0    |        |
| TDCNT5        | Bit15   | Bit14   | Bit13   | Bit12   | Bit11   | Bit10   | Bit9    | Bit8    |        |
|               | Bit7    | Bit6    | Bit5    | Bit4    | Bit3    | Bit2    | Bit1    | Bit0    |        |
| TBRW*2        | Bit15   | Bit14   | Bit13   | Bit12   | Bit11   | Bit10   | Bit9    | Bit8    |        |
|               | Bit7    | Bit6    | Bit5    | Bit4    | Bit3    | Bit2    | Bit1    | Bit0    |        |
| ICSR          | POE3F   | POE2F   | POE1F   | POE0F   | —       | —       | —       | PIE     | POE    |
|               | POE3M1  | POE3M0  | POE2M1  | POE2M0  | POE1M1  | POE1M0  | POE0M1  | POE0M0  |        |
| POEPC         | POE3E   | POE2E   | POE1E   | POE0E   | —       | —       | —       | —       | SYSTEM |
| SBYCR         | SSBY    | STS2    | STS1    | STS0    | —       | —       | —       | —       |        |
| SYSCR         | MACS    | —       | INTM1   | INTM0   | NMIEG   | —       | —       | RAME    |        |
| SCKCR         | PSTOP   | —       | —       | —       | STCS    | 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  |        |
| LPWRCR        | —       | —       | —       | —       | —       | —       | STC1    | STC0    | PBC    |
| BARA          | —       | —       | —       | —       | —       | —       | —       | —       |        |
|               | 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    | INT    |
| BCRB          | CMFB    | CDB     | BAMRB2  | BAMRB1  | BAMRB0  | CSELB1  | CSELB0  | BIEB    |        |
| ISCRH         | —       | —       | —       | —       | IRQ5SCB | IRQ5SCA | IRQ4SCB | IRQ4SCA |        |
| ISCR L        | IRQ3SCB | IRQ3SCA | IRQ2SCB | IRQ2SCA | IRQ1SCB | IRQ1SCA | IRQ0SCB | IRQ0SCA |        |
| IER           | —       | —       | IRQ5E   | IRQ4E   | IRQ3E   | IRQ2E   | IRQ1E   | IRQ0E   |        |
| ISR           | —       | —       | IRQ5F   | IRQ4F   | IRQ3F   | IRQ2F   | IRQ1F   | IRQ0F   |        |

| Register Name | Bit 7  | Bit 6  | Bit 5  | Bit 4  | Bit 3  | Bit 2  | Bit 1  | Bit 0  | Module |
|---------------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| DTCEA7        | DTCEA7 | DTCEA6 | DTCEA5 | DTCEA4 | DTCEA3 | DTCEA2 | DTCEA1 | DTCEA0 | DTC    |
| DTCEB7        | DTCEB7 | DTCEB6 | DTCEB5 | DTCEB4 | DTCEB3 | DTCEB2 | DTCEB1 | DTCEB0 |        |
| DTCEC7        | DTCEC7 | DTCEC6 | DTCEC5 | DTCEC4 | DTCEC3 | DTCEC2 | DTCEC1 | DTCEC0 |        |
| DTCED7        | DTCED7 | DTCED6 | DTCED5 | DTCED4 | DTCED3 | DTCED2 | DTCED1 | DTCED0 |        |
| DTCEE7        | DTCEE7 | DTCEE6 | DTCEE5 | DTCEE4 | DTCEE3 | DTCEE2 | DTCEE1 | DTCEE0 |        |
| DTCEF7        | DTCEF7 | DTCEF6 | DTCEF5 | DTCEF4 | DTCEF3 | DTCEF2 | DTCEF1 | DTCEF0 |        |
| DTCEG7        | DTCEG7 | DTCEG6 | DTCEG5 | DTCEG4 | DTCEG3 | DTCEG2 | DTCEG1 | DTCEG0 |        |
| DTVEC7        | DTVEC7 | DTVEC6 | DTVEC5 | DTVEC4 | DTVEC3 | DTVEC2 | DTVEC1 | DTVEC0 |        |
| PCR           | G3CMS1 | G3CMS0 | G2CMS1 | G2CMS0 | G1CMS1 | G1CMS0 | G0CMS1 | G0CMS0 |        |
| PMR           | G3INV  | G2INV  | —      | —      | G3NOV  | G2NOV  | —      | —      |        |
| NDERH         | NDER15 | NDER14 | NDER13 | NDER12 | NDER11 | NDER10 | NDER9  | NDER8  | PPG    |
| NDERL         | NDER7  | NDER6  | NDER5  | NDER4  | NDER3  | NDER2  | NDER1  | NDER0  |        |
| PODRH         | POD15  | POD14  | POD13  | POD12  | POD11  | POD10  | POD9   | POD8   |        |
| PODRL         | POD7   | POD6   | POD5   | POD4   | POD3   | POD2   | POD1   | POD0   |        |
| NDRH          | NDR15  | NDR14  | NDR13  | NDR12  | NDR11  | NDR10  | NDR9   | NDR8   |        |
| NDRL          | NDR7   | NDR6   | NDR5   | NDR4   | NDR3   | NDR2   | NDR1   | NDR0   |        |
| NDRH          | —      | —      | —      | —      | NDR11  | NDR10  | NDR9   | NDR8   |        |
| NDRL          | —      | —      | —      | —      | NDR3   | NDR2   | NDR1   | NDR0   |        |
| P1DDR         | P17DDR | P16DDR | P15DDR | P14DDR | P13DDR | P12DDR | P11DDR | P10DDR |        |
| PADDR         | —      | —      | —      | —      | PA3DDR | PA2DDR | PA1DDR | PA0DDR |        |
| PBDDR         | PB7DDR | PB6DDR | PB5DDR | PB4DDR | PB3DDR | PB2DDR | PB1DDR | PB0DDR | PORT   |
| PCDDR         | PC7DDR | PC6DDR | PC5DDR | PC4DDR | PC3DDR | PC2DDR | PC1DDR | PC0DDR |        |
| PDDDR         | PD7DDR | PD6DDR | PD5DDR | PD4DDR | PD3DDR | PD2DDR | PD1DDR | PD0DDR |        |
| PFDDR         | PF7DDR | PF6DDR | PF5DDR | PF4DDR | PF3DDR | PF2DDR | PF1DDR | PF0DDR |        |
| 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 |        |
| PAODR         | —      | —      | —      | —      | PA3ODR | PA2ODR | PA1ODR | PA0ODR |        |
| PBODR         | PB7ODR | PB6ODR | PB5ODR | PB4ODR | PB3ODR | PB2ODR | PB1ODR | PB0ODR |        |
| PCODR         | PC7ODR | PC6ODR | PC5ODR | PC4ODR | PC3ODR | PC2ODR | PC1ODR | PC0ODR |        |
| 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   |        |
| TCNTH_3       | Bit15  | Bit14  | Bit13  | Bit12  | Bit11  | Bit10  | Bit9   | Bit8   |        |
| TCNTL_3       | Bit7   | Bit6   | Bit5   | Bit4   | Bit3   | Bit2   | Bit1   | Bit0   |        |

| Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module     |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|------------|
| TGRAH_3       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  | TPU3       |
| TGRAL_3       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRBH_3       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TGRBL_3       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRCH_3       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TGRCL_3       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRDH_3       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TGRDL_3       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TCR_4         | —     | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU4       |
| 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  |            |
| TCNTH_4       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TCNTL_4       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRAH_4       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TGRAL_4       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRBH_4       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  | TPU5       |
| TGRBL_4       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TCR_5         | —     | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 |            |
| TMDR_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  |            |
| TCNTH_5       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TCNTL_5       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRAH_5       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  | TPU common |
| TGRAL_5       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TGRBH_5       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |            |
| TGRBL_5       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |            |
| TSTR          | —     | —     | CST5  | CST4  | CST3  | CST2  | CST1  | CST0  |            |
| 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  |            |
| IPRF          | —     | IPR6  | IPR5  | IPR4  | —     | IPR2  | IPR1  | IPR0  |            |
| IPRG          | —     | IPR6  | IPR5  | IPR4  | —     | IPR2  | IPR1  | IPR0  |            |
| IPRH          | —     | IPR6  | IPR5  | IPR4  | —     | IPR2  | IPR1  | IPR0  |            |
| IPRJ          | —     | IPR6  | IPR5  | IPR4  | —     | IPR2  | IPR1  | IPR0  |            |
| IPRK          | —     | IPR6  | IPR5  | IPR4  | —     | IPR2  | IPR1  | IPR0  |            |
| IPRM          | —     | 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 |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| RAMER         | —     | —     | —     | —     | RAMS  | RAM2  | RAM1  | RAM0  | ROM    |
| P1DR          | P17DR | P16DR | P15DR | P14DR | P13DR | P12DR | P11DR | P10DR | PORT   |
| 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 |        |
| PFDR          | PF7DR | PF6DR | PF5DR | PF4DR | PF3DR | PF2DR | PF1DR | PF0DR |        |
| TCR_0         | CCLR2 | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU0   |
| 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 |        |
| TSR_0         | —     | —     | —     | TCFV  | TGFD  | TGFC  | TGFB  | TGFA  |        |
| TCNTH_0       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TCNTL_0       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRAH_0       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TGRAL_0       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRBH_0       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TGRBL_0       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRCH_0       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TGRCL_0       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRDH_0       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TGRDL_0       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TCR_1         | —     | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU1   |
| 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  |        |
| TCNTH_1       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TCNTL_1       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRAH_1       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TGRAL_1       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TGRBH_1       | Bit15 | Bit14 | Bit13 | Bit12 | Bit11 | Bit10 | Bit9  | Bit8  |        |
| TGRBL_1       | Bit7  | Bit6  | Bit5  | Bit4  | Bit3  | Bit2  | Bit1  | Bit0  |        |
| TCR_2         | —     | CCLR1 | CCLR0 | CKEG1 | CKEG0 | TPSC2 | TPSC1 | TPSC0 | TPU2   |
| 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  |        |

| Register Name                                 | Bit 7                | Bit 6          | Bit 5          | Bit 4                           | Bit 3          | Bit 2          | Bit 1          | Bit 0          | Module |
|-----------------------------------------------|----------------------|----------------|----------------|---------------------------------|----------------|----------------|----------------|----------------|--------|
| TCNTH_2                                       | Bit15                | Bit14          | Bit13          | Bit12                           | Bit11          | Bit10          | Bit9           | Bit8           | TPU_2  |
| TCNTL_2                                       | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| TGRAH_2                                       | Bit15                | Bit14          | Bit13          | Bit12                           | Bit11          | Bit10          | Bit9           | Bit8           |        |
| TGRAL_2                                       | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| TGRBH_2                                       | Bit15                | Bit14          | Bit13          | Bit12                           | Bit11          | Bit10          | Bit9           | Bit8           | WDT    |
| TGRBL_2                                       | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| TCSR                                          | OVF                  | WT/IT          | TME            | —                               | —              | CKS2           | CKS1           | CKS0           |        |
| TCNT                                          | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| RSTCSR                                        | WOVF                 | RSTE           | RSTS           | —                               | —              | —              | —              | —              | SCI_0  |
| SMR_0* <sup>3</sup><br>(SMR_0* <sup>4</sup> ) | C/ $\bar{A}$<br>(GM) | CHR<br>(BLK)   | PE<br>(PE)     | O/ $\bar{E}$<br>(O/ $\bar{E}$ ) | STOP<br>(BCP1) | MP<br>(BCP0)   | CKS1<br>(CKS1) | CKS0<br>(CKS0) |        |
| BRR_0                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| SCR_0                                         | TIE                  | RIE            | TE             | RE                              | MPIE           | TEIE           | CKE1           | CKE0           |        |
| TDR_0                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           | SCI1   |
| SSR_0* <sup>3</sup><br>(SSR_0* <sup>4</sup> ) | TDRE<br>(TDRE)       | RDRF<br>(RDRF) | ORER<br>(ORER) | FER<br>(ERS)                    | PER<br>(PER)   | TEND<br>(TEND) | MPB<br>(MPB)   | MPBT<br>(MPBT) |        |
| RDR_0                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| SCMR_0                                        | —                    | —              | —              | —                               | SDIR           | SINV           | —              | SMIF           |        |
| SMR_1* <sup>3</sup><br>(SMR_1* <sup>4</sup> ) | C/ $\bar{A}$<br>(GM) | CHR<br>(BLK)   | PE<br>(PE)     | O/ $\bar{E}$<br>(O/ $\bar{E}$ ) | STOP<br>(BCP1) | MP<br>(BCP0)   | CKS1<br>(CKS1) | CKS0<br>(CKS0) | SCI2   |
| BRR_1                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| SCR_1                                         | TIE                  | RIE            | TE             | RE                              | MPIE           | TEIE           | CKE1           | CKE0           |        |
| TDR_1                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| SSR_1* <sup>3</sup><br>(SSR_1* <sup>4</sup> ) | TDRE<br>(TDRE)       | RDRF<br>(RDRF) | ORER<br>(ORER) | FER<br>(ERS)                    | PER<br>(PER)   | TEND<br>(TEND) | MPB<br>(MPB)   | MPBT<br>(MPBT) | A/DC   |
| RDR_1                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| SCMR_1                                        | —                    | —              | —              | —                               | SDIR           | SINV           | —              | SMIF           |        |
| SMR_2* <sup>3</sup><br>(SMR_2* <sup>4</sup> ) | C/ $\bar{A}$<br>(GM) | CHR<br>(BLK)   | PE<br>(PE)     | O/ $\bar{E}$<br>(O/ $\bar{E}$ ) | STOP<br>(BCP1) | MP<br>(BCP0)   | CKS1<br>(CKS1) | CKS0<br>(CKS0) |        |
| BRR_2                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           | A/DC   |
| SCR_2                                         | TIE                  | RIE            | TE             | RE                              | MPIE           | TEIE           | CKE1           | CKE0           |        |
| TDR_2                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           |        |
| SSR_2* <sup>3</sup><br>(SSR_2* <sup>4</sup> ) | TDRE<br>(TDRE)       | RDRF<br>(RDRF) | ORER<br>(ORER) | FER<br>(ERS)                    | PER<br>(PER)   | TEND<br>(TEND) | MPB<br>(MPB)   | MPBT<br>(MPBT) |        |
| RDR_2                                         | Bit7                 | Bit6           | Bit5           | Bit4                            | Bit3           | Bit2           | Bit1           | Bit0           | A/DC   |
| SCMR_2                                        | —                    | —              | —              | —                               | SDIR           | SINV           | —              | SMIF           |        |
| ADDRAH                                        | AD9                  | AD8            | AD7            | AD6                             | AD5            | AD4            | AD3            | AD2            |        |
| ADDRAL                                        | AD1                  | AD0            | —              | —                               | —              | —              | —              | —              |        |
| ADDRBH                                        | AD9                  | AD8            | AD7            | AD6                             | AD5            | AD4            | AD3            | AD2            | A/DC   |
| ADDRBL                                        | AD1                  | AD0            | —              | —                               | —              | —              | —              | —              |        |
| ADDRCH                                        | AD9                  | AD8            | AD7            | AD6                             | AD5            | AD4            | AD3            | AD2            |        |
| ADDRCL                                        | AD1                  | AD0            | —              | —                               | —              | —              | —              | —              |        |
| ADDRDH                                        | AD9                  | AD8            | AD7            | AD6                             | AD5            | AD4            | AD3            | AD2            |        |

| Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module |
|---------------|-------|-------|-------|-------|-------|-------|-------|-------|--------|
| ADDRDL        | AD1   | AD0   | —     | —     | —     | —     | —     | —     | A/DC   |
| ADCSR         | ADF   | ADIE  | ADST  | SCAN  | CH3   | CH2   | CH1   | CH0   |        |
| ADCR          | TRGS1 | TRGS0 | —     | —     | CKS1  | CKS0  | —     | —     |        |
| FLMCR1        | FWE   | SWE   | ESU1  | PSU1  | EV1   | PV1   | E1    | P1    | ROM    |
| FLMCR2        | FLER  | —     | —     | —     | —     | —     | —     | —     |        |
| EBR1          | EB7   | EB6   | EB5   | EB4   | EB3   | EB2   | EB1   | EB0   |        |
| EBR2          | —     | —     | —     | —     | —     | —     | EB9   | EB8   | PORT   |
| PORT1         | P17   | P16   | P15   | P14   | P13   | P12   | P11   | P10   |        |
| PORT4         | P47   | P46   | P45   | P44   | P43   | P42   | P41   | P40   |        |
| PORT9         | —     | —     | —     | —     | P93   | P92   | P91   | P90   |        |
| 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   |        |
| PORTF         | PF7   | PF6   | PF5   | PF4   | PF3   | PF2   | PF1   | PF0   |        |

Notes: \*1 For buffer operation.

\*2 For free operation.

\*3 Normal serial communication interface mode.

\*4 Smart Card interface mode.

Some bit functions of SMR differ in normal serial communication interface mode and Smart Card interface mode.

## A.3 Register States in Each Operating Mode

| Register Name | Reset       | High-speed | Medium-speed | Sleep | Module Stop | Software Standby | Hardware Standby | Module |
|---------------|-------------|------------|--------------|-------|-------------|------------------|------------------|--------|
| MCR           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | HCAN   |
| GSR           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| BCR           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MBCR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TXPR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TXCR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TXACK         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ABACK         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| RXPR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| RFPR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| IRR           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MBIMR         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| IMR           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| REC           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TEC           | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| UMSR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| LAFML         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| LAFMH         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[1]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[2]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[3]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[4]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[5]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[6]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[7]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC0[8]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[1]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[2]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[3]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[4]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[5]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[6]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[7]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC1[8]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[1]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[2]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[3]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[4]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[5]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[6]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[7]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC2[8]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC3[1]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC3[2]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC3[3]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC3[4]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC3[5]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MC3[6]        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |









| Register Name | Reset       | High-speed | Medium-speed | Sleep | Module Stop | Software Standby | Hardware Standby | Module |
|---------------|-------------|------------|--------------|-------|-------------|------------------|------------------|--------|
| MD13[3]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | HCAN   |
| MD13[4]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD13[5]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD13[6]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD13[7]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD13[8]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[1]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[2]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[3]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[4]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[5]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[6]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[7]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD14[8]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[1]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[2]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[3]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | MMT    |
| MD15[4]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[5]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[6]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[7]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| MD15[8]       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| HCANMON       | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TMDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TSR           | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNT          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TPDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TPBR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| MMTPC         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TBRU          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRUU         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRU          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRUD         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDCNT0        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDCNT1        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TBRU          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TBRV          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRVU         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRV          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRVD         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDCNT2        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDCNT3        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TBRV          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TBRW          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRWU         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRW          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRWD         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDCNT4        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TDCNT5        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |

| Register Name | Reset       | High-speed | Medium-speed | Sleep | Module Stop | Software Standby | Hardware Standby | Module |
|---------------|-------------|------------|--------------|-------|-------------|------------------|------------------|--------|
| TBRW          | Initialized | –          | –            | –     | –           | –                | Initialized      | MMT    |
| ICSR          | Initialized | –          | –            | –     | –           | –                | Initialized      | POE    |
| POEPC         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| SBYCR         | Initialized | –          | –            | –     | –           | –                | Initialized      | SYSTEM |
| SYSCR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| SCKCR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| MDCR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| MSTPCRA       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| MSTPCRB       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| MSTPCRC       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| LPWRCR        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| BARA          | Initialized | –          | –            | –     | –           | –                | Initialized      | PBC    |
| BARB          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| BCRA          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| BCRB          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| ISCRH         | Initialized | –          | –            | –     | –           | –                | Initialized      | INT    |
| ISCR_L        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| IER           | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| ISR           | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTCERA        | Initialized | –          | –            | –     | –           | –                | Initialized      | DTC    |
| DTCERB        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTCERC        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTCERD        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTCERE        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTCERF        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTCERG        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| DTVECR        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PCR           | Initialized | –          | –            | –     | –           | –                | Initialized      | PPG    |
| PMR           | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| NDERH         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| NDERL         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PODRH         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PODR_L        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| NDRH          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| NDRL          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| NDRH          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| NDRL          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| P1DDR         | Initialized | –          | –            | –     | –           | –                | Initialized      | PORT   |
| PADDR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PBDDR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PCDDR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PDDDR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PFDDR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PAPCR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PBPCR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PCPCR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PDPCR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PAODR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PBODR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PCODR         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCR_3         | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_3  |

| Register Name | Reset       | High-speed | Medium-speed | Sleep | Module Stop | Software Standby | Hardware Standby | Module     |
|---------------|-------------|------------|--------------|-------|-------------|------------------|------------------|------------|
| TMDR_3        | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_3      |
| TIORH_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TIORL_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TIER_3        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TSR_3         | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCNTH_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCNTL_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRAH_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRAL_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRBH_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRBL_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRCH_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRCL_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRDH_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRDL_3       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCR_4         | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_4      |
| TMDR_4        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TIOR_4        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TIER_4        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TSR_4         | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCNTH_4       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCNTL_4       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRAH_4       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRAL_4       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRBH_4       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRBL_4       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCR_5         | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_5      |
| TMDR_5        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TIOR_5        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TIER_5        | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TSR_5         | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCNTH_5       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TCNTL_5       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRAH_5       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRAL_5       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRBH_5       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TGRBL_5       | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| TSTR          | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU common |
| TSYR          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRA          | Initialized | –          | –            | –     | –           | –                | Initialized      | INT        |
| IPRB          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRC          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRD          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRE          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRF          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRG          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRH          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRJ          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRK          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| IPRM          | Initialized | –          | –            | –     | –           | –                | Initialized      |            |
| RAMER         | Initialized | –          | –            | –     | –           | –                | Initialized      | ROM        |

| Register Name | Reset       | High-speed | Medium-speed | Sleep | Module Stop | Software Standby | Hardware Standby | Module |
|---------------|-------------|------------|--------------|-------|-------------|------------------|------------------|--------|
| P1DR          | Initialized | –          | –            | –     | –           | –                | Initialized      | PORT   |
| PADR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PBDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PCDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PDDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PFDR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCR_0         | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_0  |
| TMDR_0        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIORH_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIORL_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIER_0        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TSR_0         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNTH_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNTL_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRAH_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRAL_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRBH_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRBL_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRCH_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRCL_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRDH_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRDL_0       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCR_1         | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_1  |
| TMDR_1        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIOR_1        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIER_1        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TSR_1         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNTH_1       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNTL_1       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRAH_1       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRAL_1       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRBH_1       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRBL_1       | Initialized | –          | –            | –     | –           | –                | Initialized      | TPU_2  |
| TCR_2         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TMDR_2        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIOR_2        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TIER_2        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TSR_2         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNTH_2       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNTL_2       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRAH_2       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRAL_2       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TGRBH_2       | Initialized | –          | –            | –     | –           | –                | Initialized      | WDT    |
| TGRBL_2       | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCSR          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| TCNT          | Initialized | –          | –            | –     | –           | –                | Initialized      | SCI_0  |
| RSTCSR        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| SMR_0         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| BRR_0         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SCR_0         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TDR_0         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |

| Register Name | Reset       | High-speed | Medium-speed | Sleep | Module Stop | Software Standby | Hardware Standby | Module |
|---------------|-------------|------------|--------------|-------|-------------|------------------|------------------|--------|
| SSR_0         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | SCI_0  |
| RDR_0         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SCMR_0        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SMR_1         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | SCI_1  |
| BRR_1         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SCR_1         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| TDR_1         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SSR_1         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| RDR_1         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SCMR_1        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SMR_2         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| BRR_2         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SCR_2         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | SCI_2  |
| TDR_2         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SSR_2         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| RDR_2         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| SCMR_2        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRAH        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRAL        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      | A/DC   |
| ADDRBH        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRBL        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRCH        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRCL        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRDH        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADDRDL        | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADCSR         | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| ADCR          | Initialized | –          | –            | –     | Initialized | Initialized      | Initialized      |        |
| FLMCR1        | Initialized | –          | –            | –     | –           | –                | Initialized      | ROM    |
| FLMCR2        | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| EBR1          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| EBR2          | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORT1         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORT4         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORT9         | Initialized | –          | –            | –     | –           | –                | Initialized      | PORT   |
| PORTA         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORTB         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORTC         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORTD         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |
| PORTF         | Initialized | –          | –            | –     | –           | –                | Initialized      |        |

Note: – is not initialized.

## B. I/O Port States in Each Pin State

| Port Name | MCU<br>Operating<br>Mode | Reset | Hardware<br>Standby<br>Mode | Software<br>Standby<br>Mode      | Program<br>Execution<br>State Sleep<br>Mode |
|-----------|--------------------------|-------|-----------------------------|----------------------------------|---------------------------------------------|
| Port 1    | 7                        | T     | T                           | Keep                             | I/O port                                    |
| Port 4    | 7                        | T     | T                           | T                                | Input port                                  |
| Port 9    | 7                        | T     | T                           | T                                | Input port                                  |
| Port A    | 7                        | T     | T                           | Keep                             | I/O port                                    |
| Port B    | 7                        | T     | T                           | Keep                             | I/O port                                    |
| Port C    | 7                        | T     | T                           | Keep                             | I/O port                                    |
| Port D    | 7                        | T     | T                           | Keep                             | I/O port                                    |
| PF7       | 7                        | T     | T                           | [DDR = 0]<br>T<br>[DDR = 1]<br>H | [DDR = 0]<br>T<br>[DDR = 1]<br>Clock output |
| PF6       | 7                        | T     | T                           | Keep                             | I/O port                                    |
| PF5       |                          |       |                             |                                  |                                             |
| PF4       |                          |       |                             |                                  |                                             |
| PF3       |                          |       |                             |                                  |                                             |
| PF2       |                          |       |                             |                                  |                                             |
| PF1       |                          |       |                             |                                  |                                             |
| PF0       |                          |       |                             |                                  |                                             |
| HTxD      | 7                        | H     | T                           | H                                | Output                                      |
| HRxD      | 7                        | Input | T                           | T                                | Input                                       |

### Legend:

H: High level

T: High impedance

Keep: Input port becomes high-impedance, output port retains state

C. Product Code Lineup

|              |                    |                  | Package (Hitachi<br>Package Code) |
|--------------|--------------------|------------------|-----------------------------------|
| Product Type |                    |                  | QFP-80 (FP-80Q)                   |
| H8S/2612     | F-ZTAT version     | Standard product | HD64F2612                         |
|              | Mask ROM version * | Standard product | HD6432612                         |
|              |                    | Standard product | HD6432611                         |

Note: \* In planning

D. Package Dimensions

The package dimension that is shown in the Hitachi Semiconductor Package Data Book has priority.

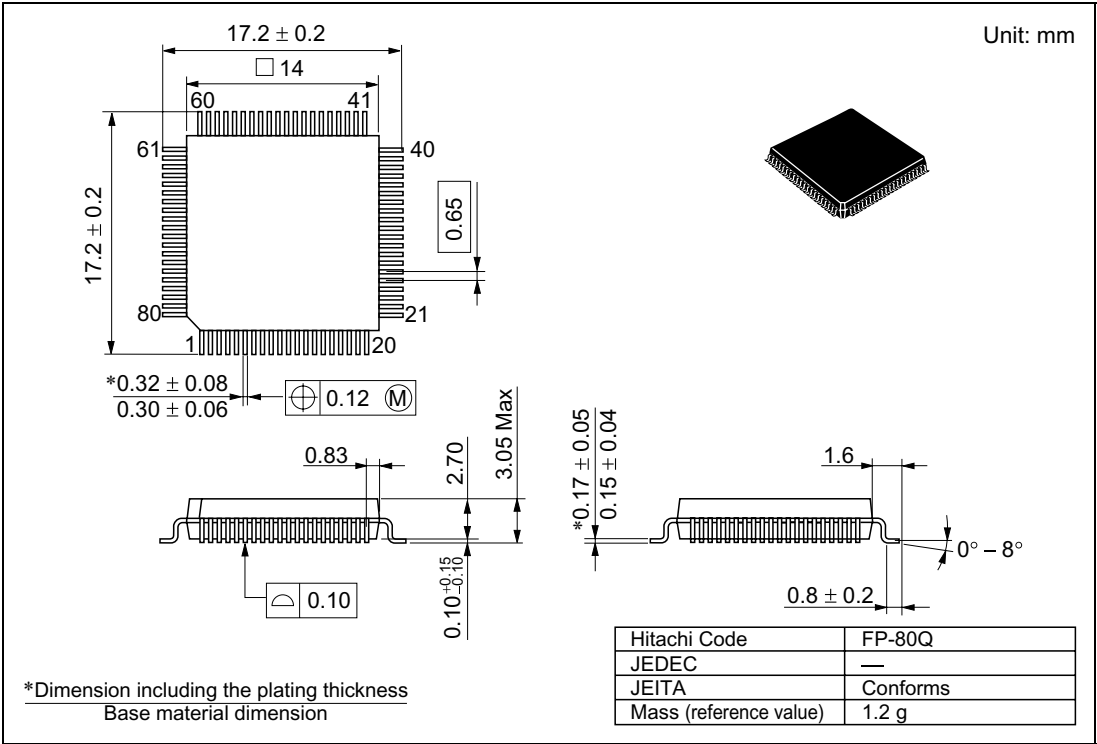


Figure D-1 FP-80Q Package Dimensions

# Index

|                                             |     |                                           |          |
|---------------------------------------------|-----|-------------------------------------------|----------|
| 16-Bit Timer Pulse Unit (TPU) .....         | 149 | Bcc.....                                  | 26, 34   |
| Buffer Operation .....                      | 194 | Bit Rate.....                             | 385      |
| Cascaded Operation.....                     | 198 | break address .....                       | 83, 86   |
| Free-running count operation .....          | 188 | break conditions .....                    | 86       |
| Input Capture.....                          | 191 | Bus Arbitration .....                     | 94       |
| periodic count operation .....              | 188 | bus cycle .....                           | 91       |
| Phase Counting Mode .....                   | 204 | Bus Masters .....                         | 94       |
| PWM Modes .....                             | 199 | Clock Pulse Generator .....               | 443      |
| Synchronous Operation .....                 | 193 | Condition Field.....                      | 37       |
| toggle output.....                          | 189 | Condition-Code Register (CCR).....        | 20       |
| Waveform Output by Compare Match ...        | 189 | CPU Operating Modes .....                 | 12       |
| A/D Converter.....                          | 403 | Advanced Mode.....                        | 13       |
| A/D Converter Activation .....              | 213 | Normal Mode.....                          | 12       |
| A/D trigger input .....                     | 146 | data direction register .....             | 119      |
| Conversion Time .....                       | 411 | data register .....                       | 119      |
| External Trigger .....                      | 413 | Data Transfer Controller.....             | 95       |
| Scan Mode.....                              | 410 | Activation by Software.....               | 114      |
| Single Mode .....                           | 410 | Block Transfer Mode.....                  | 109      |
| Address Map .....                           | 50  | Chain Transfer .....                      | 110, 115 |
| Address Space .....                         | 16  | DTC Vector Table .....                    | 102      |
| Addressing Modes.....                       | 38  | Normal Mode.....                          | 107, 114 |
| Absolute Address .....                      | 39  | Register Information.....                 | 102      |
| Immediate.....                              | 40  | Repeat Mode.....                          | 108      |
| Memory Indirect.....                        | 40  | software activation.....                  | 111      |
| Program-Counter Relative.....               | 40  | Software Activation.....                  | 116      |
| Register Direct.....                        | 38  | vector number for the software activation |          |
| Register Indirect .....                     | 38  | interrupt .....                           | 100      |
| Register Indirect with Displacement .....   | 38  | Effective Address .....                   | 38, 41   |
| Register Indirect with Post-Increment ..... | 39  | Effective Address Extension .....         | 37       |
| Register Indirect with Pre-Decrement .....  | 39  | Exception Handling .....                  | 51       |
|                                             |     | Interrupts .....                          | 56       |
|                                             |     | Reset Exception Handling .....            | 53       |
|                                             |     | Stack Status .....                        | 58       |
|                                             |     | Traces .....                              | 56       |
|                                             |     | Trap Instruction .....                    | 57       |
|                                             |     | Exception Handling Vector Table .....     | 52       |
|                                             |     | Extended Control Register (EXR) .....     | 19       |
|                                             |     | Flash Memory                              |          |
|                                             |     | Boot Mode .....                           | 432      |
|                                             |     | Emulation .....                           | 435      |

|                                                  |               |                                                |        |
|--------------------------------------------------|---------------|------------------------------------------------|--------|
| Erase/Erase-Verify.....                          | 439           | Interrupt Control Modes .....                  | 73     |
| erasing units.....                               | 425           | Interrupt Controller.....                      | 61     |
| Power-Down States .....                          | 442           | Interrupt Exception Handling Vector Table..... | 70     |
| Program/Program-Verify.....                      | 437           | Interrupt Mask Bit .....                       | 21     |
| programming units.....                           | 425           | interrupt mask level .....                     | 19     |
| Programming/Erasing in User Program<br>Mode..... | 434           | interrupt priority register (IPR).....         | 61     |
| General Registers.....                           | 18            | Interrupts                                     |        |
| HCAN .....                                       | 93, 355       | ADI .....                                      | 413    |
| 11 consecutive recessive bits .....              | 383           | ERS0/OVR0 .....                                | 398    |
| Arbitration field .....                          | 390, 393      | NMI .....                                      | 69, 80 |
| buffer segment.....                              | 386           | RM0 .....                                      | 398    |
| Configuration mode.....                          | 383           | RM1 .....                                      | 398    |
| Control field .....                              | 390           | SLE0 .....                                     | 398    |
| Data field .....                                 | 390           | SWDTEND.....                                   | 111    |
| Data frame .....                                 | 393           | TCIU_1 .....                                   | 212    |
| DTC Interface.....                               | 399           | TCIU_2 .....                                   | 212    |
| HCAN Halt Mode.....                              | 397           | TCIU_4.....                                    | 212    |
| HCAN Sleep Mode.....                             | 395           | TCIU_5.....                                    | 212    |
| mailbox .....                                    | 380, 382, 388 | TCIV_0.....                                    | 212    |
| Message Control (MC0 to MC15).....               | 380           | TCIV_1 .....                                   | 212    |
| Message Data (MD0 to MD15).....                  | 382           | TCIV_2.....                                    | 212    |
| Message transmission cancellation.....           | 390           | TCIV_3.....                                    | 212    |
| Message Transmission Method .....                | 388           | TCIV_4.....                                    | 212    |
| Remote frame .....                               | 394           | TCIV_5.....                                    | 212    |
| remote transmission request bit .....            | 394           | TGIA_0.....                                    | 212    |
| Unread message overwrite.....                    | 394           | TGIA_1.....                                    | 212    |
| input pull-up MOS.....                           | 119           | TGIA_2.....                                    | 212    |
| Instruction Set .....                            | 26            | TGIA_3.....                                    | 212    |
| Arithmetic Operations Instructions.....          | 29            | TGIA_4.....                                    | 212    |
| Bit Manipulation Instructions .....              | 32            | TGIA_5.....                                    | 212    |
| Block Data Transfer Instructions .....           | 36            | TGIB_0.....                                    | 212    |
| Branch Instructions.....                         | 34            | TGIB_1 .....                                   | 212    |
| Data Transfer Instructions .....                 | 28            | TGIB_2 .....                                   | 212    |
| Logic Operations Instructions.....               | 31            | TGIB_3.....                                    | 212    |
| Shift Instructions .....                         | 31            | TGIB_4.....                                    | 212    |
| System Control Instructions .....                | 35            | TGIB_5.....                                    | 212    |
|                                                  |               | TGIC_0.....                                    | 212    |
|                                                  |               | TGIC_3.....                                    | 212    |
|                                                  |               | TGID_0.....                                    | 212    |
|                                                  |               | TGID_3.....                                    | 212    |
|                                                  |               | TGIMN .....                                    | 247    |
|                                                  |               | TGINN.....                                     | 247    |
|                                                  |               | WOVI .....                                     | 288    |

|                                         |                    |               |                    |
|-----------------------------------------|--------------------|---------------|--------------------|
| MAC instruction.....                    | 48                 | GSR .....     | 359, 485, 501, 515 |
| memory cycle .....                      | 91                 | HCANMON ..... | 382, 493, 507, 520 |
| MMT .....                               | 93                 | ICSR .....    | 257, 494, 509, 521 |
| Motor Management Timer .....            | 231                | IER .....     | 65, 494, 509, 521  |
| buffer registers.....                   | 237                | IMR .....     | 375, 485, 501, 515 |
| dead time .....                         | 237, 243           | IPR.....      | 64, 497, 511, 522  |
| non-overlap time.....                   | 237                | IRR .....     | 370, 485, 501, 515 |
| Multiply-Accumulate Register (MAC)..... | 22                 | ISCR .....    | 65, 494, 509, 521  |
| On-Board Programming .....              | 431                | ISR.....      | 67, 494, 509, 521  |
| open-drain control register .....       | 119                | LAFMH .....   | 378, 485, 501, 515 |
| Operating Mode Selection.....           | 47                 | LAFML.....    | 378, 485, 501, 515 |
| Operation Field .....                   | 37                 | LPWRCR.....   | 445, 494, 509, 521 |
| PC Break Controller .....               | 83                 | MBCR .....    | 363, 485, 501, 515 |
| Pin Arrangement .....                   | 3                  | MBIMR .....   | 374, 485, 501, 515 |
| PLL Circuit .....                       | 449                | MC.....       | 380, 485, 501, 515 |
| port register .....                     | 119                | MCR .....     | 358, 485, 501, 515 |
| Program Counter (PC).....               | 19                 | MD .....      | 382, 489, 504, 517 |
| Program/Erase Protection.....           | 441                | MDCR .....    | 47, 494, 509, 521  |
| Programmable Pulse Generator .....      | 263                | MMTPC.....    | 238, 493, 508, 520 |
| Non-Overlapping Pulse Output .....      | 276                | MRA.....      | 98                 |
| output trigger .....                    | 270                | MRB .....     | 99                 |
| Programmer Mode .....                   | 442                | MSTPCR .....  | 458, 494, 509, 521 |
| Register Field .....                    | 37                 | NDER .....    | 266, 495, 510, 521 |
| Registers                               |                    | NDR.....      | 268, 495, 510, 521 |
| ABACK.....                              | 367, 485, 501, 515 | PIDDR .....   | 122, 495, 510, 521 |
| ADCR.....                               | 409, 499, 514, 524 | PIDR .....    | 123, 497, 512, 523 |
| ADCSR .....                             | 407, 499, 514, 524 | PADDR .....   | 128, 495, 510, 521 |
| ADDR .....                              | 406, 499, 513, 524 | PADR .....    | 129, 497, 512, 523 |
| BARA.....                               | 84, 494, 509, 521  | PAODR .....   | 130, 495, 510, 521 |
| BARB.....                               | 85, 494, 509, 521  | PAPCR .....   | 130, 495, 510, 521 |
| BCR.....                                | 361, 485, 501, 515 | PBDDR.....    | 132, 495, 510, 521 |
| BCRA.....                               | 85, 494, 509, 521  | PBDR .....    | 133, 497, 512, 523 |
| BCRB.....                               | 86, 494, 509, 521  | PBODR.....    | 134, 495, 510, 521 |
| BRR.....                                | 308, 499, 513, 523 | PBPCR.....    | 134, 495, 510, 521 |
| CRA.....                                | 99                 | PCDDR.....    | 137, 495, 510, 521 |
| CRB.....                                | 100                | PCDR .....    | 138, 497, 512, 523 |
| DAR .....                               | 99                 | PCODR.....    | 140, 495, 510, 521 |
| DTCER.....                              | 100, 495, 510, 521 | PCPCR.....    | 139, 495, 510, 521 |
| DTVECR.....                             | 100, 495, 510, 521 | PCR .....     | 270, 495, 510, 521 |
| EBR1 .....                              | 429, 500, 514, 524 | PDDDR .....   | 142, 495, 510, 521 |
| EBR2 .....                              | 430, 500, 514, 524 | PDDR .....    | 143, 497, 512, 523 |
| FLMCR1 .....                            | 428, 500, 514, 524 | PDPCR .....   | 143, 495, 510, 521 |
| FLMCR2 .....                            | 429, 500, 514, 524 | PFDDR .....   | 144, 495, 510, 521 |

|             |                                                      |                                      |                                                      |
|-------------|------------------------------------------------------|--------------------------------------|------------------------------------------------------|
| PFDR .....  | 145, 497, 512, 523                                   | TDR .....                            | 294, 499, 513, 523                                   |
| PMR .....   | 271, 495, 510, 521                                   | TEC.....                             | 376, 485, 501, 515                                   |
| PODR .....  | 267, 495, 510, 521                                   | TGR .....                            | 185, 195, 237, 493,<br>..... 498, 508, 512, 520, 523 |
| POEPC ..... | 260, 494, 509, 521                                   | TIER .....                           | 179, 497, 512, 523                                   |
| PORT1.....  | 123, 500, 514, 524                                   | TIOR.....                            | 163, 497, 512, 523                                   |
| PORT4.....  | 126, 500, 514, 524                                   | TMDR.....                            | 160, 234, 493, 497,<br>..... 508, 512, 520, 523      |
| PORT9.....  | 127, 500, 514, 524                                   | TPBR .....                           | 237, 493, 508, 520                                   |
| PORTA.....  | 129, 500, 514, 524                                   | TPDR.....                            | 237, 493, 508, 520                                   |
| PORTB ..... | 133, 500, 514, 524                                   | TSR.....                             | 181, 236, 294, 493,<br>..... 497, 508, 512, 520, 523 |
| PORTC ..... | 139, 500, 514, 524                                   | TSTR .....                           | 185, 497, 511, 522                                   |
| PORTD ..... | 143, 500, 514, 524                                   | TSYR.....                            | 186, 497, 511, 522                                   |
| PORTF ..... | 145, 500, 514, 524                                   | TXACK.....                           | 366, 485, 501, 515                                   |
| RAMER.....  | 430, 497, 512, 522                                   | TXCR.....                            | 365, 485, 501, 515                                   |
| RDR.....    | 294, 499, 513, 524                                   | TXPR.....                            | 364, 485, 501, 515                                   |
| REC .....   | 376, 485, 501, 515                                   | UMSR.....                            | 377, 485, 501, 515                                   |
| RFPR .....  | 369, 485, 501, 515                                   | Reset .....                          | 53                                                   |
| RSR .....   | 294                                                  | Serial Communication Interface ..... | 291                                                  |
| RSTCSR..... | 286, 499, 513, 523                                   | Asynchronous Mode .....              | 315                                                  |
| RXPR .....  | 368, 485, 501, 515                                   | bit rate .....                       | 308                                                  |
| SAR .....   | 99                                                   | Break.....                           | 354                                                  |
| SBYCR.....  | 456, 494, 509, 521                                   | framing error .....                  | 322                                                  |
| SCKCR.....  | 444, 494, 509, 521                                   | Mark State.....                      | 354                                                  |
| SCMR.....   | 307, 499, 513, 524                                   | overrun error .....                  | 322                                                  |
| SCR .....   | 299, 499, 513, 523                                   | parity error .....                   | 322                                                  |
| SMR .....   | 295, 499, 513, 523                                   | stack pointer (SP).....              | 18                                                   |
| SSR.....    | 302, 499, 513, 524                                   | Time Quanta (TQ) .....               | 387                                                  |
| SYSCR ..... | 48, 494, 509, 521                                    | Trace Bit .....                      | 19                                                   |
| TBR .....   | 237, 493, 508, 520                                   | TRAPA instruction .....              | 40, 57                                               |
| TCNR .....  | 235, 493, 508, 520                                   | Watchdog Timer .....                 | 283                                                  |
| TCNT.....   | 184, 236, 284, 493,<br>..... 499, 508, 513, 520, 523 | Interval Timer Mode .....            | 287                                                  |
| TCR .....   | 155, 497, 512, 523                                   | overflows .....                      | 287                                                  |
| TCSR.....   | 284, 499, 513, 523                                   |                                      |                                                      |
| TDCNT.....  | 237, 493, 508, 520                                   |                                      |                                                      |
| TDDR .....  | 237, 493, 508, 520                                   |                                      |                                                      |

---

## **H8S/2612 Series Hardware Manual**

Publication Date: 1st Edition, September 2000  
4th Edition, March 2003

Published by: Business Operation Division  
Semiconductor & Integrated Circuits  
Hitachi, Ltd.

Edited by: Technical Documentation Group  
Hitachi Kodaira Semiconductor Co., Ltd.

Copyright © Hitachi, Ltd., 2000. All rights reserved. Printed in Japan.