To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

Notice

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

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

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

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

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

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

User's Manual

16

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.

H8/3687 Group

Hardware Manual

Renesas 16-Bit Single-Chip Microcomputer H8 Family/H8/300H Tiny Series

H8/3687N	HD64N3687G, HD6483687G,
H8/3687F	HD64F3687, HD64F3687G,
H8/3687	HD6433687, HD6433687G,
H8/3686	HD6433686, HD6433686G,
H8/3685	HD6433685, HD6433685G,
H8/3684F	HD64F3684, HD64F3684G,
H8/3684	HD6433684, HD6433684G,
H8/3683	HD6433683, HD6433683G,
H8/3682	HD6433682, HD6433682G

Renesas Electronics
www.renesas.com

Rev.5.00 2005.11

Keep safety first in your circuit designs!

1. Renesas Technology Corp. 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 Corp. 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 Corp. or a third party.
2. Renesas Technology Corp. 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 Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. 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 Corp. 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 Corp. by various means, including the Renesas Technology Corp. 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 Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. 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 Corp. or an authorized Renesas Technology Corp. 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 Corp. 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 Corp. for further details on these materials or the products contained therein.

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 be 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. List of Registers
8. Electrical Characteristics
9. Appendix
10. Main Revisions and Additions in this Edition (only for revised versions)

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

11. Index

Preface

The H8/3687 Group are single-chip microcomputers made up of the high-speed H8/300H CPU employing Renesas Technology original architecture as their cores, and the peripheral functions required to configure a system. The H8/300H CPU has an instruction set that is compatible with the H8/300 CPU.

Target Users: This manual was written for users who will be using the H8/3687 Group in the design of application systems. Target users are expected to understand the fundamentals of electrical circuits, logical circuits, and microcomputers.

Objective: This manual was written to explain the hardware functions and electrical characteristics of the H8/3687 Group to the target users.
Refer to the H8/300H Series Software Manual for a detailed description of the instruction set.

Notes on reading this manual:

- In order to understand the overall functions of the chip
  Read the manual according to the contents. This manual can be roughly categorized into 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 H8/300H Series Software 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 section 22, List of Registers.

Example:	Register name:	The following notation is used for cases when the same or a similar function, e.g. serial communication interface, 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.

Notes:

When using the on-chip emulator (E7, E8) for H8/3687 program development and debugging, the following restrictions must be noted.

1. The $\overline{\text{NMI}}$ pin is reserved for the E7 or E8, and cannot be used.
2. Pins P85, P86, and P87 cannot be used. In order to use these pins, additional hardware must be provided on the user board.
3. Area H'D000 to H'DFFF is used by the E7 or E8, and is not available to the user.
4. Area H'F780 to H'FB7F must on no account be accessed.
5. When the E7 or E8 is used, address breaks can be set as either available to the user or for use by the E7 or E8. If address breaks are set as being used by the E7 or E8, the address break control registers must not be accessed.
6. When the E7 or E8 is used, $\overline{\text{NMI}}$ is an input/output pin (open-drain in output mode), P85 and P87 are input pins, and P86 is an output pin.
7. Use channel 1 of the SCI3 (P21/RXD, P22/TXD) in on-board programming mode by boot mode.

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.renesas.com/

H8/3687 Group manuals:

Document Title	Document No.
H8/3687 Group Hardware Manual	This manual
H8/300H Series Software Manual	REJ09B0213

User's manuals for development tools:

Document Title	Document No.
H8S, H8/300 Series C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual	REJ10B0058
Microcomputer Development Environment System H8S, H8/300 Series Simulator/Debugger User's Manual	ADE-702-282
H8S, H8/300 Series High-Performance Embedded Workshop 3, Tutorial	REJ10B0024
H8S, H8/300 Series High-Performance Embedded Workshop 3, User's Manual	REJ10B0026

Application notes:

Document Title	Document No.
H8S, H8/300 Series C/C++ Compiler Package Application Note	REJ05B0464
Single Power Supply F-ZTAT™ On-Board Programming	ADE-502-055

Contents

Section 1 Overview 1
1.1 Features 1
1.2 Internal Block Diagram 3
1.3 Pin Arrangement 5
1.4 Pin Functions 7

Section 2 CPU 11
2.1 Address Space and Memory Map 12
2.2 Register Configuration 15
2.2.1 General Registers 16
2.2.2 Program Counter (PC) 17
2.2.3 Condition-Code Register (CCR) 17
2.3 Data Formats 19
2.3.1 General Register Data Formats 19
2.3.2 Memory Data Formats 21
2.4 Instruction Set 22
2.4.1 Table of Instructions Classified by Function 22
2.4.2 Basic Instruction Formats 32
2.5 Addressing Modes and Effective Address Calculation 33
2.5.1 Addressing Modes 33
2.5.2 Effective Address Calculation 36
2.6 Basic Bus Cycle 38
2.6.1 Access to On-Chip Memory (RAM, ROM) 38
2.6.2 On-Chip Peripheral Modules 39
2.7 CPU States 40
2.8 Usage Notes 41
2.8.1 Notes on Data Access to Empty Areas 41
2.8.2 EEPMOV Instruction 41
2.8.3 Bit-Manipulation Instruction 41

Section 3 Exception Handling 47
3.1 Exception Sources and Vector Address 48
3.2 Register Descriptions 49
3.2.1 Interrupt Edge Select Register 1 (IEGR1) 50
3.2.2 Interrupt Edge Select Register 2 (IEGR2) 51
3.2.3 Interrupt Enable Register 1 (IENR1) 52
3.2.4 Interrupt Enable Register 2 (IENR2) 53

3.2.5 Interrupt Flag Register 1 (IRR1) ... 53
3.2.6 Interrupt Flag Register 2 (IRR2) ... 55
3.2.7 Wakeup Interrupt Flag Register (IWPR) ... 55
3.3 Reset Exception Handling ... 57
3.4 Interrupt Exception Handling ... 57
3.4.1 External Interrupts ... 57
3.4.2 Internal Interrupts ... 59
3.4.3 Interrupt Handling Sequence ... 59
3.4.4 Interrupt Response Time ... 60
3.5 Usage Notes ... 62
3.5.1 Interrupts after Reset ... 62
3.5.2 Notes on Stack Area Use ... 62
3.5.3 Notes on Rewriting Port Mode Registers ... 62

Section 4 Address Break ... 63
4.1 Register Descriptions ... 63
4.1.1 Address Break Control Register (ABRKCR) ... 64
4.1.2 Address Break Status Register (ABRKSR) ... 65
4.1.3 Break Address Registers (BARH, BARL) ... 65
4.1.4 Break Data Registers (BDRH, BDRL) ... 66
4.2 Operation ... 66

Section 5 Clock Pulse Generators ... 69
5.1 System Clock Generator ... 70
5.1.1 Connecting Crystal Resonator ... 70
5.1.2 Connecting Ceramic Resonator ... 71
5.1.3 External Clock Input Method ... 71
5.2 Subclock Generator ... 72
5.2.1 Connecting 32.768-kHz Crystal Resonator ... 72
5.2.2 Pin Connection when Not Using Subclock ... 73
5.3 Prescalers ... 73
5.3.1 Prescaler S ... 73
5.3.2 Prescaler W ... 73
5.4 Usage Notes ... 74
5.4.1 Note on Resonators ... 74
5.4.2 Notes on Board Design ... 74

Section 6 Power-Down Modes ... 75
6.1 Register Descriptions ... 75
6.1.1 System Control Register 1 (SYSCR1) ... 76

6.1.2 System Control Register 2 (SYSCR2) 78
6.1.3 Module Standby Control Register 1 (MSTCR1) 79
6.1.4 Module Standby Control Register 2 (MSTCR2) 80
6.2 Mode Transitions and States of LSI 80
6.2.1 Sleep Mode 84
6.2.2 Standby Mode 84
6.2.3 Subsleep Mode 84
6.2.4 Subactive Mode 85
6.3 Operating Frequency in Active Mode 85
6.4 Direct Transition 86
6.4.1 Direct Transition from Active Mode to Subactive Mode 86
6.4.2 Direct Transition from Subactive Mode to Active Mode 86
6.5 Module Standby Function 87

Section 7 ROM 89
7.1 Block Configuration 89
7.2 Register Descriptions 91
7.2.1 Flash Memory Control Register 1 (FLMCR1) 91
7.2.2 Flash Memory Control Register 2 (FLMCR2) 92
7.2.3 Erase Block Register 1 (EBR1) 93
7.2.4 Flash Memory Power Control Register (FLPWCR) 94
7.2.5 Flash Memory Enable Register (FENR) 94
7.3 On-Board Programming Modes 95
7.3.1 Boot Mode 95
7.3.2 Programming/Erasing in User Program Mode 98
7.4 Flash Memory Programming/Erasing 100
7.4.1 Program/Program-Verify 100
7.4.2 Erase/Erase-Verify 102
7.4.3 Interrupt Handling when Programming/Erasing Flash Memory 103
7.5 Program/Erase Protection 105
7.5.1 Hardware Protection 105
7.5.2 Software Protection 105
7.5.3 Error Protection 105
7.6 Programmer Mode 106
7.7 Power-Down States for Flash Memory 106

Section 8 RAM 107

Section 9 I/O Ports 109
9.1 Port 1 109

9.1.1 Port Mode Register 1 (PMR1) 110
9.1.2 Port Control Register 1 (PCR1) 111
9.1.3 Port Data Register 1 (PDR1) 111
9.1.4 Port Pull-Up Control Register 1 (PUCR1) 112
9.1.5 Pin Functions 112
9.2 Port 2 115
9.2.1 Port Control Register 2 (PCR2) 115
9.2.2 Port Data Register 2 (PDR2) 116
9.2.3 Port Mode Register 3 (PMR3) 116
9.2.4 Pin Functions 117
9.3 Port 3 119
9.3.1 Port Control Register 3 (PCR3) 119
9.3.2 Port Data Register 3 (PDR3) 120
9.3.3 Pin Functions 120
9.4 Port 5 122
9.4.1 Port Mode Register 5 (PMR5) 123
9.4.2 Port Control Register 5 (PCR5) 124
9.4.3 Port Data Register 5 (PDR5) 124
9.4.4 Port Pull-Up Control Register 5 (PUCR5) 125
9.4.5 Pin Functions 125
9.5 Port 6 128
9.5.1 Port Control Register 6 (PCR6) 128
9.5.2 Port Data Register 6 (PDR6) 129
9.5.3 Pin Functions 129
9.6 Port 7 134
9.6.1 Port Control Register 7 (PCR7) 134
9.6.2 Port Data Register 7 (PDR7) 135
9.6.3 Pin Functions 135
9.7 Port 8 137
9.7.1 Port Control Register 8 (PCR8) 137
9.7.2 Port Data Register 8 (PDR8) 137
9.7.3 Pin Functions 138
9.8 Port B 139
9.8.1 Port Data Register B (PDRB) 139

Section 10 Realtime Clock (RTC) 141
10.1 Features 141
10.2 Input/Output Pin 142
10.3 Register Descriptions 143
10.3.1 Second Data Register/Free Running Counter Data Register (RSECDR) 143

10.3.2 Minute Data Register (RMINDR) 144
10.3.3 Hour Data Register (RHRDR) 145
10.3.4 Day-of-Week Data Register (RWKDR) 146
10.3.5 RTC Control Register 1 (RTCCR1) 147
10.3.6 RTC Control Register 2 (RTCCR2) 148
10.3.7 Clock Source Select Register (RTCCSR) 149
10.4 Operation 150
10.4.1 Initial Settings of Registers after Power-On 150
10.4.2 Initial Setting Procedure 150
10.4.3 Data Reading Procedure 151
10.5 Interrupt Source 152

Section 11 Timer B1 153
11.1 Features 153
11.2 Input/Output Pin 154
11.3 Register Descriptions 154
11.3.1 Timer Mode Register B1 (TMB1) 155
11.3.2 Timer Counter B1 (TCB1) 155
11.3.3 Timer Load Register B1 (TLB1) 156
11.4 Operation 156
11.4.1 Interval Timer Operation 156
11.4.2 Auto-Reload Timer Operation 156
11.4.3 Event Counter Operation 157
11.5 Timer B1 Operating Modes 157

Section 12 Timer V 159
12.1 Features 159
12.2 Input/Output Pins 161
12.3 Register Descriptions 161
12.3.1 Timer Counter V (TCNTV) 161
12.3.2 Time Constant Registers A and B (TCORA, TCORB) 161
12.3.3 Timer Control Register V0 (TCRV0) 162
12.3.4 Timer Control/Status Register V (TCSRV) 164
12.3.5 Timer Control Register V1 (TCRV1) 165
12.4 Operation 166
12.4.1 Timer V Operation 166
12.5 Timer V Application Examples 170
12.5.1 Pulse Output with Arbitrary Duty Cycle 170
12.5.2 Pulse Output with Arbitrary Pulse Width and Delay from TRGV Input 171
12.6 Usage Notes 172

Section 13 Timer Z 175
13.1 Features 175
13.2 Input/Output Pins 180
13.3 Register Descriptions 181
13.3.1 Timer Start Register (TSTR) 182
13.3.2 Timer Mode Register (TMDR) 183
13.3.3 Timer PWM Mode Register (TPMR) 184
13.3.4 Timer Function Control Register (TFCR) 185
13.3.5 Timer Output Master Enable Register (TOER) 187
13.3.6 Timer Output Control Register (TOCR) 188
13.3.7 Timer Counter (TCNT) 189
13.3.8 General Registers A, B, C, and D (GRA, GRB, GRC, and GRD) 190
13.3.9 Timer Control Register (TCR) 191
13.3.10 Timer I/O Control Register (TIORA and TIORC) 192
13.3.11 Timer Status Register (TSR) 194
13.3.12 Timer Interrupt Enable Register (TIER) 196
13.3.13 PWM Mode Output Level Control Register (POCR) 197
13.3.14 Interface with CPU 197
13.4 Operation 199
13.4.1 Counter Operation 199
13.4.2 Waveform Output by Compare Match 203
13.4.3 Input Capture Function 207
13.4.4 Synchronous Operation 210
13.4.5 PWM Mode 211
13.4.6 Reset Synchronous PWM Mode 217
13.4.7 Complementary PWM Mode 221
13.4.8 Buffer Operation 230
13.4.9 Timer Z Output Timing 237
13.5 Interrupts 240
13.5.1 Status Flag Set Timing 240
13.5.2 Status Flag Clearing Timing 242
13.6 Usage Notes 242

Section 14 Watchdog Timer 251
14.1 Features 251
14.2 Register Descriptions 251
14.2.1 Timer Control/Status Register WD (TCSRWD) 252
14.2.2 Timer Counter WD (TCWD) 253
14.2.3 Timer Mode Register WD (TMWD) 253
14.3 Operation 254

Section 15 14-Bit PWM 255
15.1 Features 255
15.2 Input/Output Pin 256
15.3 Register Descriptions 256
15.3.1 PWM Control Register (PWCR) 256
15.3.2 PWM Data Registers U and L (PWDRU, PWDRL) 257
15.4 Operation 257

Section 16 Serial Communication Interface 3 (SCI3) 259
16.1 Features 259
16.2 Input/Output Pins 262
16.3 Register Descriptions 262
16.3.1 Receive Shift Register (RSR) 263
16.3.2 Receive Data Register (RDR) 263
16.3.3 Transmit Shift Register (TSR) 263
16.3.4 Transmit Data Register (TDR) 263
16.3.5 Serial Mode Register (SMR) 264
16.3.6 Serial Control Register 3 (SCR3) 265
16.3.7 Serial Status Register (SSR) 267
16.3.8 Bit Rate Register (BRR) 269
16.4 Operation in Asynchronous Mode 276
16.4.1 Clock 276
16.4.2 SCI3 Initialization 277
16.4.3 Data Transmission 278
16.4.4 Serial Data Reception 280
16.5 Operation in Clocked Synchronous Mode 284
16.5.1 Clock 284
16.5.2 SCI3 Initialization 284
16.5.3 Serial Data Transmission 285
16.5.4 Serial Data Reception (Clocked Synchronous Mode) 288
16.5.5 Simultaneous Serial Data Transmission and Reception 290
16.6 Multiprocessor Communication Function 292
16.6.1 Multiprocessor Serial Data Transmission 294
16.6.2 Multiprocessor Serial Data Reception 295
16.7 Interrupts 299
16.8 Usage Notes 300
16.8.1 Break Detection and Processing 300
16.8.2 Mark State and Break Sending 300
16.8.3 Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only) 300

16.8.4 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode 301

Section 17 I²C Bus Interface 2 (IIC2) 303
17.1 Features 303
17.2 Input/Output Pins 305
17.3 Register Descriptions 305
17.3.1 I²C Bus Control Register 1 (ICCR1) 306
17.3.2 I²C Bus Control Register 2 (ICCR2) 308
17.3.3 I²C Bus Mode Register (ICMR) 309
17.3.4 I²C Bus Interrupt Enable Register (ICIER) 311
17.3.5 I²C Bus Status Register (ICSR) 313
17.3.6 Slave Address Register (SAR) 316
17.3.7 I²C Bus Transmit Data Register (ICDRT) 317
17.3.8 I²C Bus Receive Data Register (ICDRR) 317
17.3.9 I²C Bus Shift Register (ICDRS) 317
17.4 Operation 318
17.4.1 I²C Bus Format 318
17.4.2 Master Transmit Operation 319
17.4.3 Master Receive Operation 321
17.4.4 Slave Transmit Operation 323
17.4.5 Slave Receive Operation 325
17.4.6 Clocked Synchronous Serial Format 327
17.4.7 Noise Canceler 329
17.4.8 Example of Use 330
17.5 Interrupt Request 334
17.6 Bit Synchronous Circuit 335
17.7 Usage Notes 336
17.7.1 Issue (Retransmission) of Start/Stop Conditions 336
17.7.2 WAIT Setting in I²C Bus Mode Register (ICMR) 336

Section 18 A/D Converter 337
18.1 Features 337
18.2 Input/Output Pins 339
18.3 Register Descriptions 340
18.3.1 A/D Data Registers A to D (ADDRA to ADDRD) 340
18.3.2 A/D Control/Status Register (ADCSR) 341
18.3.3 A/D Control Register (ADCR) 342
18.4 Operation 343
18.4.1 Single Mode 343

18.4.2 Scan Mode 343
18.4.3 Input Sampling and A/D Conversion Time 344
18.4.4 External Trigger Input Timing 345
18.5 A/D Conversion Accuracy Definitions 346
18.6 Usage Notes 348
18.6.1 Permissible Signal Source Impedance 348
18.6.2 Influences on Absolute Accuracy 348

Section 19 EEPROM 349
19.1 Features 349
19.2 Input/Output Pins 351
19.3 Register Description 351
19.3.1 EEPROM Key Register (EKR) 351
19.4 Operation 352
19.4.1 EEPROM Interface 352
19.4.2 Bus Format and Timing 352
19.4.3 Start Condition 352
19.4.4 Stop Condition 353
19.4.5 Acknowledge 353
19.4.6 Slave Addressing 353
19.4.7 Write Operations 354
19.4.8 Acknowledge Polling 356
19.4.9 Read Operation 356
19.5 Usage Notes 359
19.5.1 Data Protection at V_{CC} On/Off 359
19.5.2 Write/Erase Endurance 359
19.5.3 Noise Suppression Time 359

Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional) .. 361
20.1 Features 361
20.2 Register Descriptions 362
20.2.1 Low-Voltage-Detection Control Register (LVDCR) 362
20.2.2 Low-Voltage-Detection Status Register (LVDSR) 364
20.3 Operation 365
20.3.1 Power-On Reset Circuit 365
20.3.2 Low-Voltage Detection Circuit 366

Section 21 Power Supply Circuit 371
21.1 When Using Internal Power Supply Step-Down Circuit 371
21.2 When Not Using Internal Power Supply Step-Down Circuit 372

Section 22 List of Registers 373
22.1 Register Addresses (Address Order) 374
22.2 Register Bits 381
22.3 Registers States in Each Operating Mode 386

Section 23 Electrical Characteristics 391
23.1 Absolute Maximum Ratings 391
23.2 Electrical Characteristics
(F-ZTAT™ Version, EEPROM Stacked F-ZTAT™ Version) 391
23.2.1 Power Supply Voltage and Operating Ranges 391
23.2.2 DC Characteristics 394
23.2.3 AC Characteristics 400
23.2.4 A/D Converter Characteristics 404
23.2.5 Watchdog Timer Characteristics 405
23.2.6 Flash Memory Characteristics 406
23.2.7 EEPROM Characteristics 408
23.2.8 Power-Supply-Voltage Detection Circuit Characteristics (Optional) 409
23.2.9 Power-On Reset Circuit Characteristics (Optional) 409
23.3 Electrical Characteristics
(Mask-ROM Version, EEPROM Stacked Mask-ROM Version) 410
23.3.1 Power Supply Voltage and Operating Ranges 410
23.3.2 DC Characteristics 413
23.3.3 AC Characteristics 420
23.3.4 A/D Converter Characteristics 424
23.3.5 Watchdog Timer Characteristics 425
23.3.6 EEPROM Characteristics 426
23.3.7 Power-Supply-Voltage Detection Circuit Characteristics (Optional) 427
23.3.8 Power-On Reset Circuit Characteristics (Optional) 428
23.4 Operation Timing 428
23.5 Output Load Condition 431

Appendix A Instruction Set 433
A.1 Instruction List 433
A.2 Operation Code Map 448
A.3 Number of Execution States 451
A.4 Combinations of Instructions and Addressing Modes 462

Appendix B I/O Port Block Diagrams 463
B.1 I/O Port Block Diagrams 463
B.2 Port States in Each Operating State 485

Appendix C Product Code Lineup 486

Appendix D Package Dimensions 488

Appendix E EEPROM Stacked-Structure Cross-Sectional View 490

Main Revisions and Additions in this Edition 491

Index 497

Figures

Section 1 Overview

Figure 1.1 Internal Block Diagram of H8/3687 Group of F-ZTAT™ and Mask-ROM Versions 3
Figure 1.2 Internal Block Diagram of H8/3687N (EEPROM Stacked Version) 4
Figure 1.3 Pin Arrangement of H8/3687 Group of F-ZTAT™ and Mask-ROM Versions (FP-64E, FP-64A) 5
Figure 1.4 Pin Arrangement of H8/3687N (EEPROM Stacked Version) (FP-64E) 6

Section 2 CPU

Figure 2.1 Memory Map (1) 12
Figure 2.1 Memory Map (2) 13
Figure 2.1 Memory Map (3) 14
Figure 2.2 CPU Registers 15
Figure 2.3 Usage of General Registers 16
Figure 2.4 Relationship between Stack Pointer and Stack Area 17
Figure 2.5 General Register Data Formats (1) 19
Figure 2.5 General Register Data Formats (2) 20
Figure 2.6 Memory Data Formats 21
Figure 2.7 Instruction Formats 32
Figure 2.8 Branch Address Specification in Memory Indirect Mode 36
Figure 2.9 On-Chip Memory Access Cycle 38
Figure 2.10 On-Chip Peripheral Module Access Cycle (3-State Access) 39
Figure 2.11 CPU Operation States 40
Figure 2.12 State Transitions 41
Figure 2.13 Example of Timer Configuration with Two Registers Allocated to Same Address 42

Section 3 Exception Handling

Figure 3.1 Reset Sequence 58
Figure 3.2 Stack Status after Exception Handling 60
Figure 3.3 Interrupt Sequence 61
Figure 3.4 Port Mode Register Setting and Interrupt Request Flag Clearing Procedure 62

Section 4 Address Break

Figure 4.1 Block Diagram of Address Break 63
Figure 4.2 Address Break Interrupt Operation Example (1) 66
Figure 4.2 Address Break Interrupt Operation Example (2) 67

Section 5 Clock Pulse Generators
Figure 5.1 Block Diagram of Clock Pulse Generators 69
Figure 5.2 Block Diagram of System Clock Generator 70
Figure 5.3 Typical Connection to Crystal Resonator 70
Figure 5.4 Equivalent Circuit of Crystal Resonator 70
Figure 5.5 Typical Connection to Ceramic Resonator 71
Figure 5.6 Example of External Clock Input 71
Figure 5.7 Block Diagram of Subclock Generator 72
Figure 5.8 Typical Connection to 32.768-kHz Crystal Resonator 72
Figure 5.9 Equivalent Circuit of 32.768-kHz Crystal Resonator 72
Figure 5.10 Pin Connection when not Using Subclock 73
Figure 5.11 Example of Incorrect Board Design 74

Section 6 Power-Down Modes
Figure 6.1 Mode Transition Diagram 81

Section 7 ROM
Figure 7.1 Flash Memory Block Configuration 90
Figure 7.2 Programming/Erasing Flowchart Example in User Program Mode 99
Figure 7.3 Program/Program-Verify Flowchart 101
Figure 7.4 Erase/Erase-Verify Flowchart 104

Section 9 I/O Ports
Figure 9.1 Port 1 Pin Configuration 109
Figure 9.2 Port 2 Pin Configuration 115
Figure 9.3 Port 3 Pin Configuration 119
Figure 9.4 Port 5 Pin Configuration 122
Figure 9.5 Port 6 Pin Configuration 128
Figure 9.6 Port 7 Pin Configuration 134
Figure 9.7 Port 8 Pin Configuration 137
Figure 9.8 Port B Pin Configuration 139

Section 10 Realtime Clock (RTC)
Figure 10.1 Block Diagram of RTC 142
Figure 10.2 Definition of Time Expression 147
Figure 10.3 Initial Setting Procedure 150
Figure 10.4 Example: Reading of Inaccurate Time Data 151

Section 11 Timer B1
Figure 11.1 Block Diagram of Timer B1 153

Section 12 Timer V
Figure 12.1 Block Diagram of Timer V 160

Figure 12.2 Increment Timing with Internal Clock 167
Figure 12.3 Increment Timing with External Clock 167
Figure 12.4 OVF Set Timing 167
Figure 12.5 CMFA and CMFB Set Timing 168
Figure 12.6 TMOV Output Timing 168
Figure 12.7 Clear Timing by Compare Match 168
Figure 12.8 Clear Timing by TMRIV Input 169
Figure 12.9 Pulse Output Example 170
Figure 12.10 Example of Pulse Output Synchronized to TRGV Input 171
Figure 12.11 Contention between TCNTV Write and Clear 172
Figure 12.12 Contention between TCORA Write and Compare Match 173
Figure 12.13 Internal Clock Switching and TCNTV Operation 173

Section 13 Timer Z

Figure 13.1 Timer Z Block Diagram 177
Figure 13.2 Timer Z (Channel 0) Block Diagram 178
Figure 13.3 Timer Z (Channel 1) Block Diagram 179
Figure 13.4 Example of Outputs in Reset Synchronous PWM Mode and Complementary PWM Mode 186
Figure 13.5 Accessing Operation of 16-Bit Register (between CPU and TCNT (16 bits)) 197
Figure 13.6 Accessing Operation of 8-Bit Register (between CPU and TSTR (8 bits)) 198
Figure 13.7 Example of Counter Operation Setting Procedure 199
Figure 13.8 Free-Running Counter Operation 200
Figure 13.9 Periodic Counter Operation 201
Figure 13.10 Count Timing at Internal Clock Operation 201
Figure 13.11 Count Timing at External Clock Operation (Both Edges Detected) 202
Figure 13.12 Example of Setting Procedure for Waveform Output by Compare Match 203
Figure 13.13 Example of 0 Output/1 Output Operation 204
Figure 13.14 Example of Toggle Output Operation 205
Figure 13.15 Output Compare Timing 206
Figure 13.16 Example of Input Capture Operation Setting Procedure 207
Figure 13.17 Example of Input Capture Operation 208
Figure 13.18 Input Capture Signal Timing 209
Figure 13.19 Example of Synchronous Operation Setting Procedure 210
Figure 13.20 Example of Synchronous Operation 211
Figure 13.21 Example of PWM Mode Setting Procedure 212
Figure 13.22 Example of PWM Mode Operation (1) 213
Figure 13.23 Example of PWM Mode Operation (2) 214
Figure 13.24 Example of PWM Mode Operation (3) 215
Figure 13.25 Example of PWM Mode Operation (4) 216
Figure 13.26 Example of Reset Synchronous PWM Mode Setting Procedure 218

Figure 13.27 Example of Reset Synchronous PWM Mode Operation (OLS0 = OLS1 = 1) 219
Figure 13.28 Example of Reset Synchronous PWM Mode Operation (OLS0 = OLS1 = 0) 220
Figure 13.29 Example of Complementar y PWM Mode Setting Procedure 222
Figure 13.30 Canceling Procedure of Complementary PWM Mode 223
Figure 13.31 Example of Complementary PWM Mode Operation (1) 224
Figure 13.32 (1) Example of Complementary PWM Mode Operation
(TPSC2 = TPSC1 = TPSC0 = 0) (2) 225
Figure 13.32 (2) Example of Complementary PWM Mode Operation
(TPSC2 = TPSC1 = TPSC0 ≠ 0) (3) 226
Figure 13.33 Timing of Overshooting 227
Figure 13.34 Timing of Undershooting 227
Figure 13.35 Compare Match Buffer Operation 230
Figure 13.36 Input Capture Buffer Operation 231
Figure 13.37 Example of Buffer Operation Setting Procedure 231
Figure 13.38 Example of Buffer Operation (1)
(Buffer Operation for Output Compare Register) 232
Figure 13.39 Example of Compare Match Timing for Buffer Operation 233
Figure 13.40 Example of Buffer Operation (2)
(Buffer Operation for Input Capture Register) 234
Figure 13.41 Input Capture Timing of Buffer Operation 235
Figure 13.42 Buffer Operation (3) (Buffer Operation in Complementary
PWM Mode CMD1 = CMD0 = 1) 236
Figure 13.43 Buffer Operation (4) (Buffer Operation in
Complementary PWM Mode CMD1 = CMD0 = 1) 237
Figure 13.44 Example of Output Disable Timing of Timer Z by Writing to TOER 238
Figure 13.45 Example of Output Disable Timing of Timer Z by External Trigger 238
Figure 13.46 Example of Output Inverse Timing of Timer Z by Writing to TFCR 239
Figure 13.47 Example of Output Inverse Timing of Timer Z by Writing to POCR 239
Figure 13.48 IMF Flag Set Timing when Compare Match Occurs 240
Figure 13.49 IMF Flag Set Timing at Input Capture 241
Figure 13.50 OVF Flag Set Timing 241
Figure 13.51 Status Flag Clearing Timing 242
Figure 13.52 Contention between TCNT Write and Clear Operations 242
Figure 13.53 Contention between TCNT Write and Increment Operations 243
Figure 13.54 Contention between GR Write and Compare Match 244
Figure 13.55 Contention between TCNT Write and Overflow 245
Figure 13.56 Contention between GR Read and Input Capture 246
Figure 13.57 Contention between Count Clearing and Increment
Operations by Input Capture 247
Figure 13.58 Contention between GR Write and Input Capture 248

Figure 13.59 When Compare Match and Bit Manipulation Instruction to TOCR Occur at the Same Timing 250

Section 14 Watchdog Timer

Figure 14.1 Block Diagram of Watchdog Timer 251
Figure 14.2 Watchdog Timer Operation Example 254

Section 15 14-Bit PWM

Figure 15.1 Block Diagram of 14-Bit PWM 255
Figure 15.2 Waveform Output by 14-Bit PWM 258

Section 16 Serial Communication Interface 3 (SCI3)

Figure 16.1 Block Diagram of SCI3 261
Figure 16.2 Data Format in Asynchronous Communication 276
Figure 16.3 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits) 276
Figure 16.4 Sample SCI3 Initialization Flowchart 277
Figure 16.5 Example of SCI3 Transmission in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit) 278
Figure 16.6 Sample Serial Transmission Data Flowchart (Asynchronous Mode) 279
Figure 16.7 Example of SCI3 Reception in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit) 280
Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode) (1) 282
Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode) (2) 283
Figure 16.9 Data Format in Clocked Synchronous Communication 284
Figure 16.10 Example of SCI3 Transmission in Clocked Synchronous Mode 286
Figure 16.11 Sample Serial Transmission Flowchart (Clocked Synchronous Mode) 287
Figure 16.12 Example of SCI3 Reception in Clocked Synchronous Mode 288
Figure 16.13 Sample Serial Reception Flowchart (Clocked Synchronous Mode) 289
Figure 16.14 Sample Flowchart of Simultaneous Serial Transmit and Receive Operations (Clocked Synchronous Mode) 291
Figure 16.15 Example of Inter-Processor Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A) 293
Figure 16.16 Sample Multiprocessor Serial Transmission Flowchart 294
Figure 16.17 Sample Multiprocessor Serial Reception Flowchart (1) 296
Figure 16.17 Sample Multiprocessor Serial Reception Flowchart (2) 297
Figure 16.18 Example of SCI3 Reception Using Multiprocessor Format (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit) 298
Figure 16.19 Receive Data Sampling Timing in Asynchronous Mode 301

Section 17 I^2C Bus Interface 2 (IIC2)

Figure 17.1 Block Diagram of I^2C Bus Interface 2 304

Figure 17.2 External Circuit Connections of I/O Pins 305
Figure 17.3 I^2C Bus Formats 318
Figure 17.4 I^2C Bus Timing 318
Figure 17.5 Master Transmit Mode Operation Timing (1) 320
Figure 17.6 Master Transmit Mode Operation Timing (2) 320
Figure 17.7 Master Receive Mode Operation Timing (1) 322
Figure 17.8 Master Receive Mode Operation Timing (2) 323
Figure 17.9 Slave Transmit Mode Operation Timing (1) 324
Figure 17.10 Slave Transmit Mode Operation Timing (2) 325
Figure 17.11 Slave Receive Mode Operation Timing (1) 326
Figure 17.12 Slave Receive Mode Operation Timing (2) 326
Figure 17.13 Clocked Synchronous Serial Transfer Format 327
Figure 17.14 Transmit Mode Operation Timing 328
Figure 17.15 Receive Mode Operation Timing 329
Figure 17.16 Block Diagram of Noise Conceler 329
Figure 17.17 Sample Flowchart for Master Transmit Mode 330
Figure 17.18 Sample Flowchart for Master Receive Mode 331
Figure 17.19 Sample Flowchart for Slave Transmit Mode 332
Figure 17.20 Sample Flowchart for Slave Receive Mode 333
Figure 17.21 The Timing of the Bit Synchronous Circuit 335

Section 18 A/D Converter

Figure 18.1 Block Diagram of A/D Converter 338
Figure 18.2 A/D Conversion Timing 344
Figure 18.3 External Trigger Input Timing 345
Figure 18.4 A/D Conversion Accuracy Definitions (1) 347
Figure 18.5 A/D Conversion Accuracy Definitions (2) 347
Figure 18.6 Analog Input Circuit Example 348

Section 19 EEPROM

Figure 19.1 Block Diagram of EEPROM 350
Figure 19.2 EEPROM Bus Format and Bus Timing 352
Figure 19.3 Byte Write Operation 355
Figure 19.4 Page Write Operation 356
Figure 19.5 Current Address Read Operation 357
Figure 19.6 Random Address Read Operation 358
Figure 19.7 Sequential Read Operation (when current address read is used) 358

Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional)

Figure 20.1 Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit 362
Figure 20.2 Operational Timing of Power-On Reset Circuit 366
Figure 20.3 Operational Timing of LVDR Circuit 367

Figure 20.4 Operational Timing of LVDI Circuit 368
Figure 20.5 Timing for Operation/Release of Low-Voltage Detection Circuit 369

Section 21 Power Supply Circuit

Figure 21.1 Power Supply Connection when Internal Step-Down Circuit is Used 371
Figure 21.2 Power Supply Connection when Internal Step-Down Circuit is Not Used 372

Section 23 Electrical Characteristics

Figure 23.1 System Clock Input Timing 428
Figure 23.2 $\overline{\text{RES}}$ Low Width Timing 429
Figure 23.3 Input Timing 429
Figure 23.4 I^2C Bus Interface Input/Output Timing 429
Figure 23.5 SCK3 Input Clock Timing 430
Figure 23.6 SCI Input/Output Timing in Clocked Synchronous Mode 430
Figure 23.7 EEPROM Bus Timing 431
Figure 23.8 Output Load Circuit 431

Appendix B I/O Port Block Diagrams

Figure B.1 Port 1 Block Diagram (P17) 463
Figure B.2 Port 1 Block Diagram (P14, P16) 464
Figure B.3 Port 1 Block Diagram (P15) 465
Figure B.4 Port 1 Block Diagram (P12) 466
Figure B.5 Port 2 Block Diagram (P11) 467
Figure B.6 Port 1 Block Diagram (P10) 468
Figure B.7 Port 2 Block Diagram (P24, P23) 469
Figure B.8 Port 2 Block Diagram (P22) 470
Figure B.9 Port 2 Block Diagram (P21) 471
Figure B.10 Port 2 Block Diagram (P20) 472
Figure B.11 Port 3 Block Diagram (P37 to P30) 473
Figure B.12 Port 5 Block Diagram (P57, P56) 474
Figure B.13 Port 5 Block Diagram (P55) 475
Figure B.14 Port 5 Block Diagram (P54 to P50) 476
Figure B.15 Port 6 Block Diagram (P67 to P60) 477
Figure B.16 Port 7 Block Diagram (P76) 478
Figure B.17 Port 7 Block Diagram (P75) 479
Figure B.18 Port 7 Block Diagram (P74) 480
Figure B.19 Port 7 Block Diagram (P72) 481
Figure B.20 Port 7 Block Diagram (P71) 482
Figure B.21 Port 7 Block Diagram (P70) 482
Figure B.22 Port 8 Block Diagram (P87 to P85) 483
Figure B.23 Port B Block Diagram (PB7 to PB0) 484

Appendix D Package Dimensions
Figure D.1 FP-64E Package Dimensions 488
Figure D.2 FP-64A Package Dimensions 489

Appendix E EEPROM Stacked-Structure Cross-Sectional View
Figure E.1 EEPROM Stacked-Structure Cross-Sectional View 490

Tables

Section 1 Overview
Table 1.1 Pin Functions 7

Section 2 CPU
Table 2.1 Operation Notation 22
Table 2.2 Data Transfer Instructions 23
Table 2.3 Arithmetic Operations Instructions (1) 24
Table 2.3 Arithmetic Operations Instructions (2) 25
Table 2.4 Logic Operations Instructions 26
Table 2.5 Shift Instructions 26
Table 2.6 Bit Manipulation Instructions (1) 27
Table 2.6 Bit Manipulation Instructions (2) 28
Table 2.7 Branch Instructions 29
Table 2.8 System Control Instructions 30
Table 2.9 Block Data Transfer Instructions 31
Table 2.10 Addressing Modes 33
Table 2.11 Absolute Address Access Ranges 35
Table 2.12 Effective Address Calculation (1) 36
Table 2.12 Effective Address Calculation (2) 37

Section 3 Exception Handling
Table 3.1 Exception Sources and Vector Address 48
Table 3.2 Interrupt Wait States 60

Section 4 Address Break
Table 4.1 Access and Data Bus Used 65

Section 5 Clock Pulse Generators
Table 5.1 Crystal Resonator Parameters 71

Section 6 Power-Down Modes
Table 6.1 Operating Frequency and Waiting Time 77
Table 6.2 Transition Mode after SLEEP Instruction Execution and Transition Mode due to Interrupt 82
Table 6.3 Internal State in Each Operating Mode 83

Section 7 ROM
Table 7.1 Setting Programming Modes 95
Table 7.2 Boot Mode Operation 97
Table 7.3 System Clock Frequencies for which Automatic Adjustment of LSI Bit Rate is Possible 98

Table 7.4 Reprogram Data Computation Table 102
Table 7.5 Additional-Program Data Computation Table 102
Table 7.6 Programming Time 102
Table 7.7 Flash Memory Operating States 106

Section 10 Realtime Clock (RTC)
Table 10.1 Pin Configuration 142
Table 10.2 Interrupt Source 152

Section 11 Timer B1
Table 11.1 Pin Configuration 154
Table 11.2 Timer B1 Operating Modes 157

Section 12 Timer V
Table 12.1 Pin Configuration 161
Table 12.2 Clock Signals to Input to TCNTV and Counting Conditions 163

Section 13 Timer Z
Table 13.1 Timer Z Functions 176
Table 13.2 Pin Configuration 180
Table 13.3 Initial Output Level of FTIOB0 Pin 212
Table 13.4 Output Pins in Reset Synchronous PWM Mode 217
Table 13.5 Register Settings in Reset Synchronous PWM Mode 217
Table 13.6 Output Pins in Complementary PWM Mode 221
Table 13.7 Register Settings in Complementary PWM Mode 221
Table 13.8 Register Combinations in Buffer Operation 230

Section 15 14-Bit PWM
Table 15.1 Pin Configuration 256

Section 16 Serial Communication Interface 3 (SCI3)
Table 16.1 Channel Configuration 260
Table 16.2 Pin Configuration 262
Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (1) 270
Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (2) 271
Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (3) 272
Table 16.4 Maximum Bit Rate for Each Frequency (Asynchronous Mode) 273
Table 16.5 Examples of BRR Settings for Various Bit Rates (Clocked Synchronous Mode) (1) 274
Table 16.5 Examples of BRR Settings for Various Bit Rates (Clocked Synchronous Mode) (2) 275
Table 16.6 SSR Status Flags and Receive Data Handling 281
Table 16.7 SCI3 Interrupt Requests 299

Section 17 I²C Bus Interface 2 (IIC2)
Table 17.1 I²C Bus Interface Pins 305
Table 17.2 Transfer Rate 307
Table 17.3 Interrupt Requests 334
Table 17.4 Time for Monitoring SCL 335

Section 18 A/D Converter
Table 18.1 Pin Configuration 339
Table 18.2 Analog Input Channels and Corresponding ADDR Registers 340
Table 18.3 A/D Conversion Time (Single Mode) 345

Section 19 EEPROM
Table 19.1 Pin Configuration 351
Table 19.2 Slave Addresses 354

Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional)
Table 20.1 LVDCR Settings and Select Functions 364

Section 23 Electrical Characteristics
Table 23.1 Absolute Maximum Ratings 391
Table 23.2 DC Characteristics (1) 394
Table 23.2 DC Characteristics (2) 398
Table 23.2 DC Characteristics (3) 399
Table 23.3 AC Characteristics 400
Table 23.4 I²C Bus Interface Timing 402
Table 23.5 Serial Communication Interface (SCI) Timing 403
Table 23.6 A/D Converter Characteristics 404
Table 23.7 Watchdog Timer Characteristics 405
Table 23.8 Flash Memory Characteristics 406
Table 23.9 EEPROM Characteristics 408
Table 23.10 Power-Supply-Voltage Detection Circuit Characteristics 409
Table 23.11 Power-On Reset Circuit Characteristics 409
Table 23.12 DC Characteristics (1) 413
Table 23.12 DC Characteristics (2) 418
Table 23.12 DC Characteristics (3) 419
Table 23.13 AC Characteristics 420
Table 23.14 I²C Bus Interface Timing 422
Table 23.15 Serial Communication Interface (SCI) Timing 423
Table 23.16 A/D Converter Characteristics 424
Table 23.17 Watchdog Timer Characteristics 425
Table 23.18 EEPROM Characteristics 426
Table 23.19 Power-Supply-Voltage Detection Circuit Characteristics 427

Table 23.20 Power-On Reset Circuit Characteristics 428

Appendix A Instruction Set

Table A.1 Instruction Set 435
Table A.2 Operation Code Map (1) 448
Table A.2 Operation Code Map (2) 449
Table A.2 Operation Code Map (3) 450
Table A.3 Number of Cycles in Each Instruction 452
Table A.4 Number of Cycles in Each Instruction 453
Table A.5 Combinations of Instructions and Addressing Modes 462

Section 1 Overview

1.1 Features

- High-speed H8/300H central processing unit with an internal 16-bit architecture
  - Upward-compatible with H8/300 CPU on an object level
  - Sixteen 16-bit general registers
  - 62 basic instructions
- Various peripheral functions
  - RTC (can be used as a free running counter)
  - Timer B1 (8-bit timer)
  - Timer V (8-bit timer)
  - Timer Z (16-bit timer)
  - 14-bit PWM
  - Watchdog timer
  - SCI (Asynchronous or clocked synchronous serial communication interface) × 2 channels
  - I^2C Bus Interface (conforms to the I^2C bus interface format that is advocated by Philips Electronics)
  - 10-bit A/D converter

- On-chip memory

Product Classification			Model: Standard Version	Model: On-Chip Power-On Reset and Low-Voltage Detecting Circuit Version	ROM	RAM	Remarks
Flash memory version (F-ZTAT™ version)		H8/3687F	HD64F3687	HD64F3687G	56 kbytes	4 kbytes	
		H8/3684F	HD64F3684	HD64F3684G	32 kbytes	4 kbytes	
Mask-ROM version		H8/3687	HD6433687	HD6433687G	56 kbytes	3 kbytes	
		H8/3686	HD6433686	HD6433686G	48 kbytes	3 kbytes	
		H8/3685	HD6433685	HD6433685G	40 kbytes	3 kbytes	
		H8/3684	HD6433684	HD6433684G	32 kbytes	3 kbytes	
		H8/3683	HD6433683	HD6433683G	24 kbytes	3 kbytes	
		H8/3682	HD6433682	HD6433682G	16 kbytes	3 kbytes	
EEPROM stacked version (512 bytes)	Flash memory version	H8/3687N	—	HD64N3687G	56 kbytes	4 kbytes	
	Mask-ROM version		—	HD6483687G	56 kbytes	3 kbytes	

- General I/O ports
  - — I/O pins: 45 I/O pins (43 I/O pins for H8/3687N), including 8 large current ports (I_{OL} = 20 mA, @V_{OL} = 1.5 V)
  - — Input-only pins: 8 input pins (also used for analog input)
- EEPROM interface (only for H8/3687N)
  - — I^2C bus interface (conforms to the I^2C bus interface format that is advocated by Philips Electronics)
- Supports various power-down states

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

- Compact package

Package	Code	Body Size	Pin Pitch
LQFP-64	FP-64E	10.0 × 10.0 mm	0.5 mm
QFP-64	FP-64A	14.0 × 14.0 mm	0.8 mm

Only LQFP-64 (FP-64E) for H8/3687N package

1.2 Internal Block Diagram

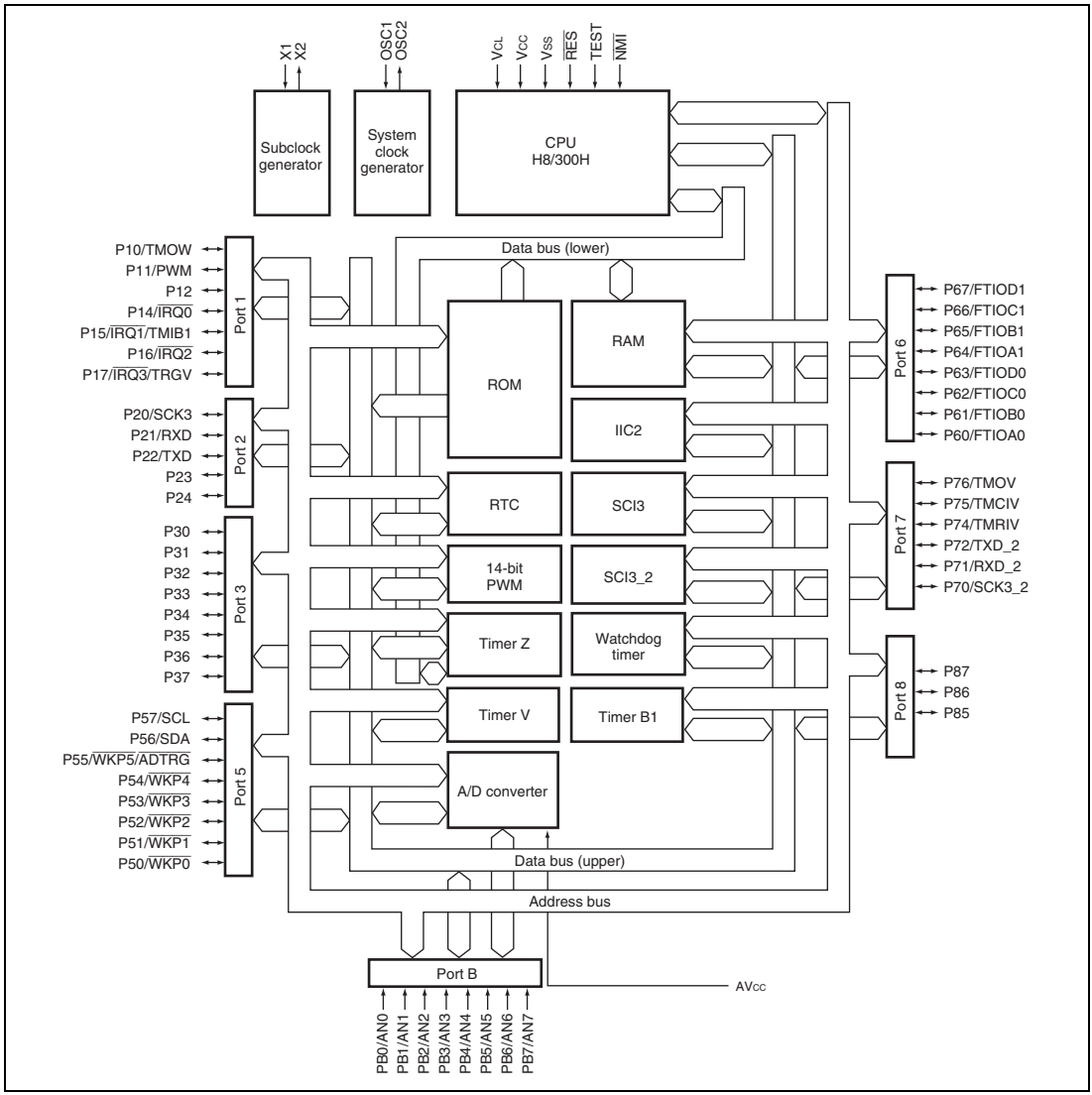
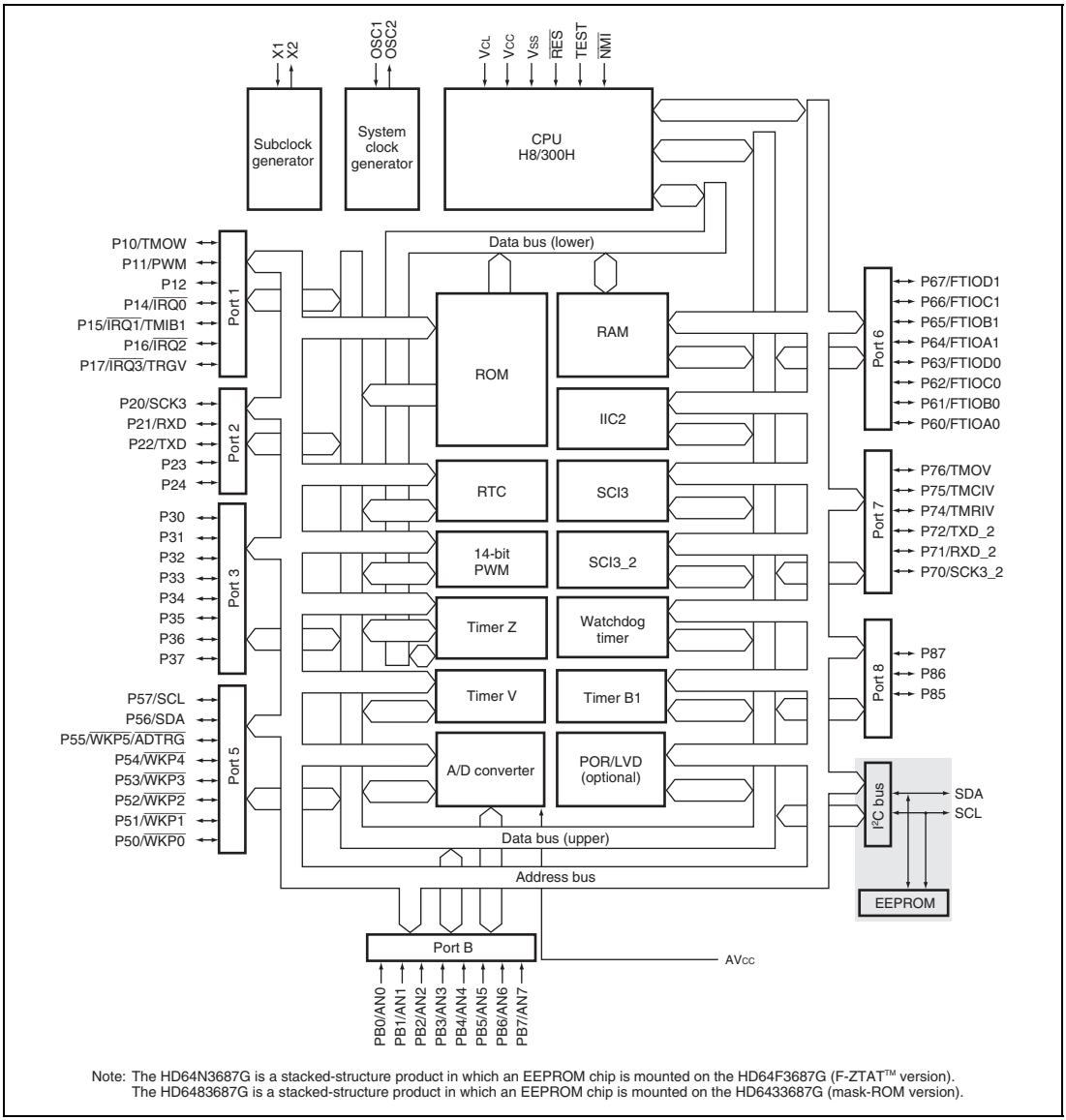


Figure 1.1 Internal Block Diagram of H8/3687 Group of F-ZTAT™ and Mask-ROM Versions



Note: The HD64N3687G is a stacked-structure product in which an EEPROM chip is mounted on the HD64F3687G (F-ZTAT™ version).
The HD6483687G is a stacked-structure product in which an EEPROM chip is mounted on the HD6433687G (mask-ROM version).

Figure 1.2 Internal Block Diagram of H8/3687N (EEPROM Stacked Version)

1.3 Pin Arrangement

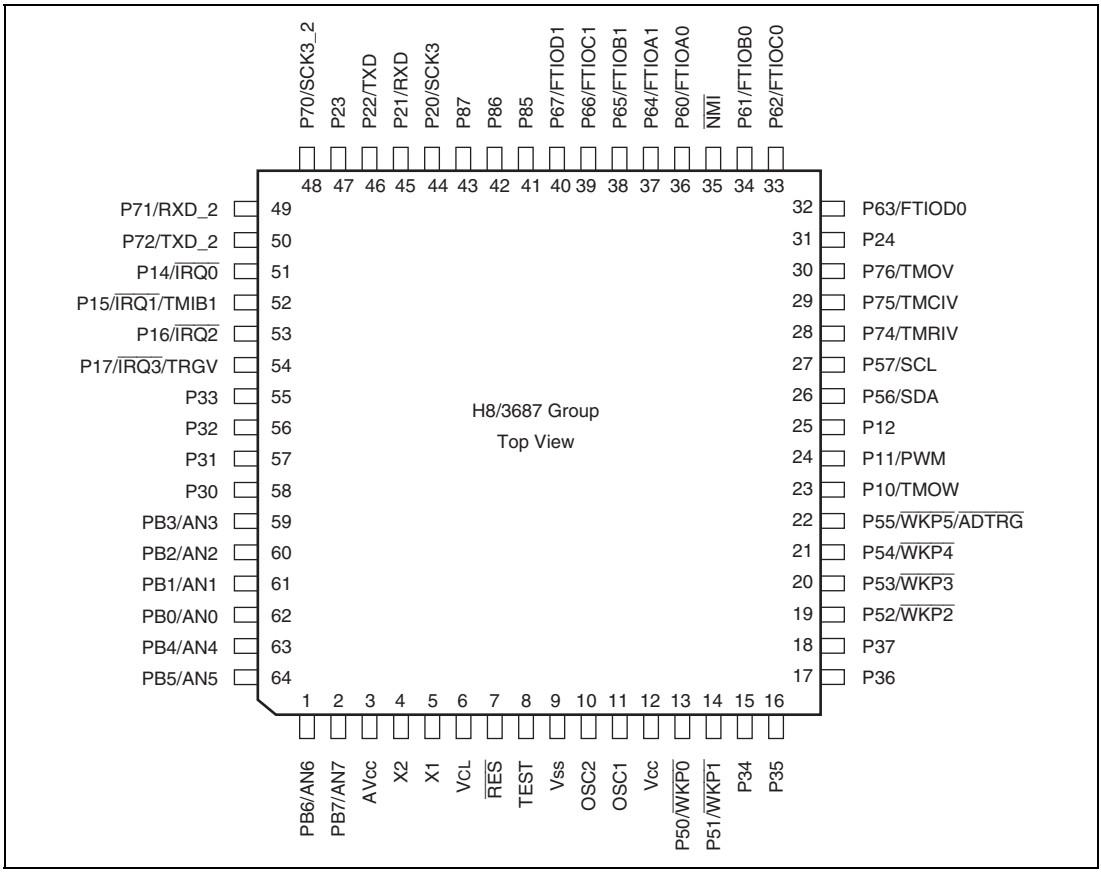


Figure 1.3 Pin Arrangement of H8/3687 Group of F-ZTAT™ and Mask-ROM Versions (FP-64E, FP-64A)

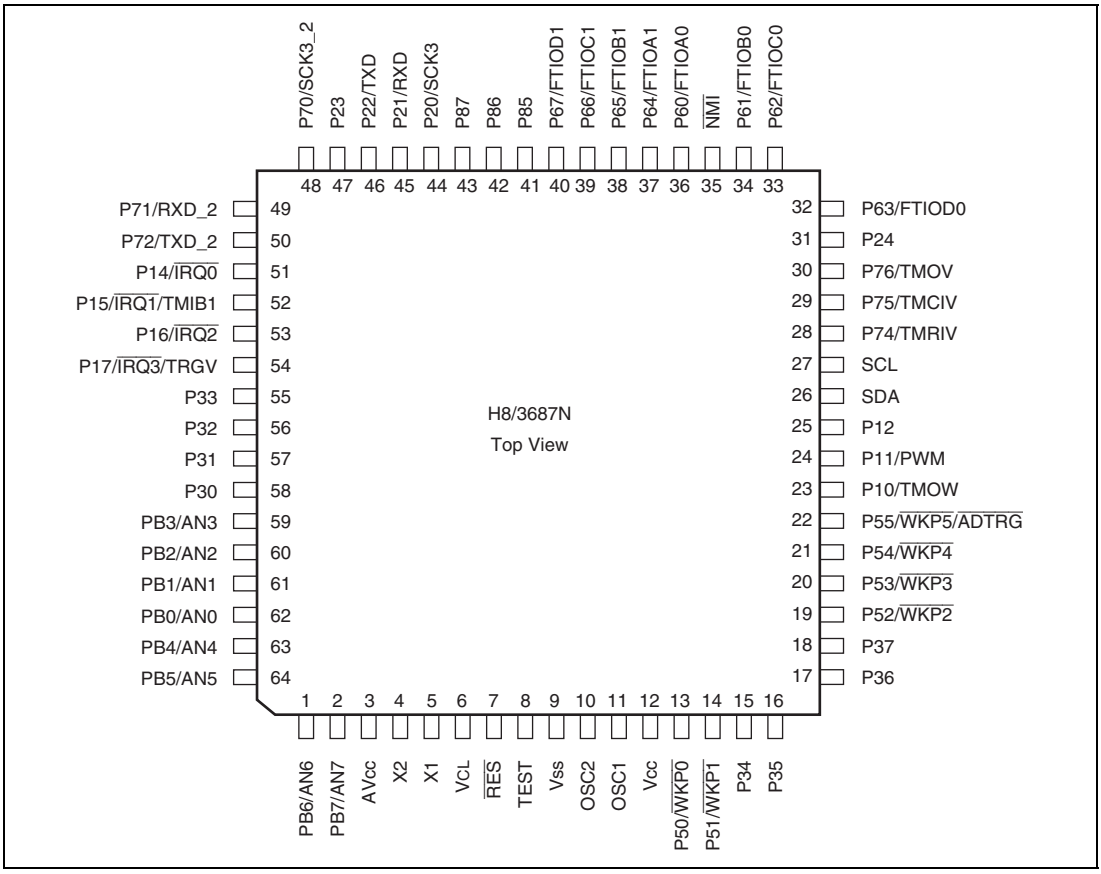


Figure 1.4 Pin Arrangement of H8/3687N (EEPROM Stacked Version) (FP-64E)

1.4 Pin Functions

Table 1.1 Pin Functions

Type	Symbol	Pin No. FP-64E FP-64A	I/O	Functions
Power source pins	V_{CC}	12	Input	Power supply pin. Connect this pin to the system power supply.
	V_{SS}	9	Input	Ground pin. Connect this pin to the system power supply (0V).
	AV_{CC}	3	Input	Analog power supply pin for the A/D converter. When the A/D converter is not used, connect this pin to the system power supply.
	V_{CL}	6	Input	Internal step-down power supply pin. Connect a capacitor of around 0.1 μF between this pin and the Vss pin for stabilization.
Clock pins	OSC1	11	Input	These pins connect with crystal or ceramic resonator for the system clock, or can be used to input an external clock. See section 5, Clock Pulse Generators, for a typical connection.
	OSC2	10	Output	
	X1	5	Input	These pins connect with a 32.768 kHz crystal resonator for the subclock. See section 5, Clock Pulse Generators, for a typical connection.
	X2	4	Output	
System control	$\overline{RES}$	7	Input	Reset pin. The pull-up resistor (typ. 150 kΩ) is incorporated. When driven low, the chip is reset.
	TEST	8	Input	Test pin. Connect this pin to Vss.
Interrupt pins	$\overline{NMI}$	35	Input	Non-maskable interrupt request input pin. Be sure to pull-up by a pull-up resistor.
	$\overline{IRQ0}$ to $\overline{IRQ3}$	51 to 54	Input	External interrupt request input pins. Can select the rising or falling edge.
	$\overline{WKP0}$ to $\overline{WKP5}$	13, 14, 19 to 22	Input	External interrupt request input pins. Can select the rising or falling edge.
RTC	TMOW	23	Output	This is an output pin for divided clocks.
Timer B1	TMIB1	52	Input	External event input pin.

Type	Symbol	Pin No. FP-64E FP-64A	I/O	Functions
Timer V	TMOV	30	Output	This is an output pin for waveforms generated by the output compare function.
	TMCIV	29	Input	External event input pin.
	TMRIV	28	Input	Counter reset input pin.
	TRGV	54	Input	Counter start trigger input pin.
Timer Z	FTIOA0	36	I/O	Output compare output/input capture input/external clock input pin
	FTIOB0	34	I/O	Output compare output/input capture input/PWM output pin
	FTIOC0	33	I/O	Output compare output/input capture input/PWM sync output pin (at a reset, complementary PWM mode)
	FTIOD0	32	I/O	Output compare output/input capture input/PWM output pin
	FTIOA1	37	I/O	Output compare output/input capture input/PWM output pin (at a reset, complementary PWM mode)
	FTIOB1 to FTIOD1	38 to 40	I/O	Output compare output/input capture input/PWM output pin
14-bit PWM	PWM	24	Output	14-bit PWM square wave output pin
I^2C bus interface (IIC)	SDA*[1]	26	I/O	IIC data I/O pin. Can directly drive a bus by NMOS open-drain output. When using this pin, external pull-up resistance is required.
	SCL*[1]	27	I/O (EEPROM: Input)	IIC clock I/O pin. Can directly drive a bus by NMOS open-drain output. When using this pin, external pull-up resistance is required.
Serial communication interface (SCI)	TXD, TXD_2	46, 50	Output	Transmit data output pin
	RXD, RXD_2	45, 49	Input	Receive data input pin
	SCK3, SCK3_2	44, 48	I/O	Clock I/O pin
A/D converter	AN7 to AN0	1, 2, 59 to 64	Input	Analog input pin
	$\overline{\text{ADTRG}}$	22	Input	A/D converter trigger input pin.

Type	Symbol	Pin No. FP-64E FP-64A	I/O	Functions
I/O ports	PB7 to PB0	1, 2, 59 to 64	Input	8-bit input port.
	P17 to P14, P12 to P10	51 to 54, 23 to 25	I/O	7-bit I/O port.
	P24 to P20	31, 44 to 47	I/O	5-bit I/O port.
	P37 to P30	15 to 18, 55 to 58	I/O	8-bit I/O port
	P57 to P50	13, 14, 19 to 22, 26*[2], 27*[2]	I/O	8-bit I/O port
	P67 to P60	32 to 34, 36, 37 to 40	I/O	8-bit I/O port
	P76 to P74, P72 to P70	28 to 30, 48 to 50	I/O	6-bit I/O port
	P87 to P85	41 to 43	I/O	3-bit I/O port.

Notes: 1. These pins are only available for the I^2C bus interface in the H8/3687N. Since the I^2C bus is disabled after canceling a reset, the ICE bit in ICCR1 must be set to 1 by using the program.

2. The P57 and P56 pins are not available in the H8/3687N.

Section 2 CPU

This LSI has an H8/300H CPU with an internal 32-bit architecture that is upward-compatible with the H8/300CPU, and supports only normal mode, which has a 64-kbyte address space.

- Upward-compatible with H8/300 CPUs
  - Can execute H8/300 CPUs object programs
  - Additional eight 16-bit extended registers
  - 32-bit transfer and arithmetic and logic instructions are added
  - Signed multiply and divide instructions are added.
- General-register architecture
  - Sixteen 16-bit general registers also usable as sixteen 8-bit registers and eight 16-bit registers, or eight 32-bit registers
- Sixty-two basic instructions
  - 8/16/32-bit data transfer and arithmetic and logic instructions
  - Multiply and divide instructions
  - Powerful bit-manipulation instructions
- Eight addressing modes
  - Register direct [Rn]
  - Register indirect [@ERn]
  - Register indirect with displacement [@(d:16,ERn) or @(d:24,ERn)]
  - Register indirect with post-increment or pre-decrement [@ERn+ or @–ERn]
  - Absolute address [@aa:8, @aa:16, @aa:24]
  - Immediate [#xx:8, #xx:16, or #xx:32]
  - Program-counter relative [@(d:8,PC) or @(d:16,PC)]
  - Memory indirect [@@aa:8]
- 64-kbyte address space
- High-speed operation
  - All frequently-used instructions execute in one or two states
  - 8/16/32-bit register-register add/subtract : 2 state
  - 8 × 8-bit register-register multiply : 14 states
  - 16 ÷ 8-bit register-register divide : 14 states
  - 16 × 16-bit register-register multiply : 22 states
  - 32 ÷ 16-bit register-register divide : 22 states
- Power-down state
  - Transition to power-down state by SLEEP instruction

2.1 Address Space and Memory Map

The address space of this LSI is 64 kbytes, which includes the program area and the data area. Figures 2.1 show the memory map.

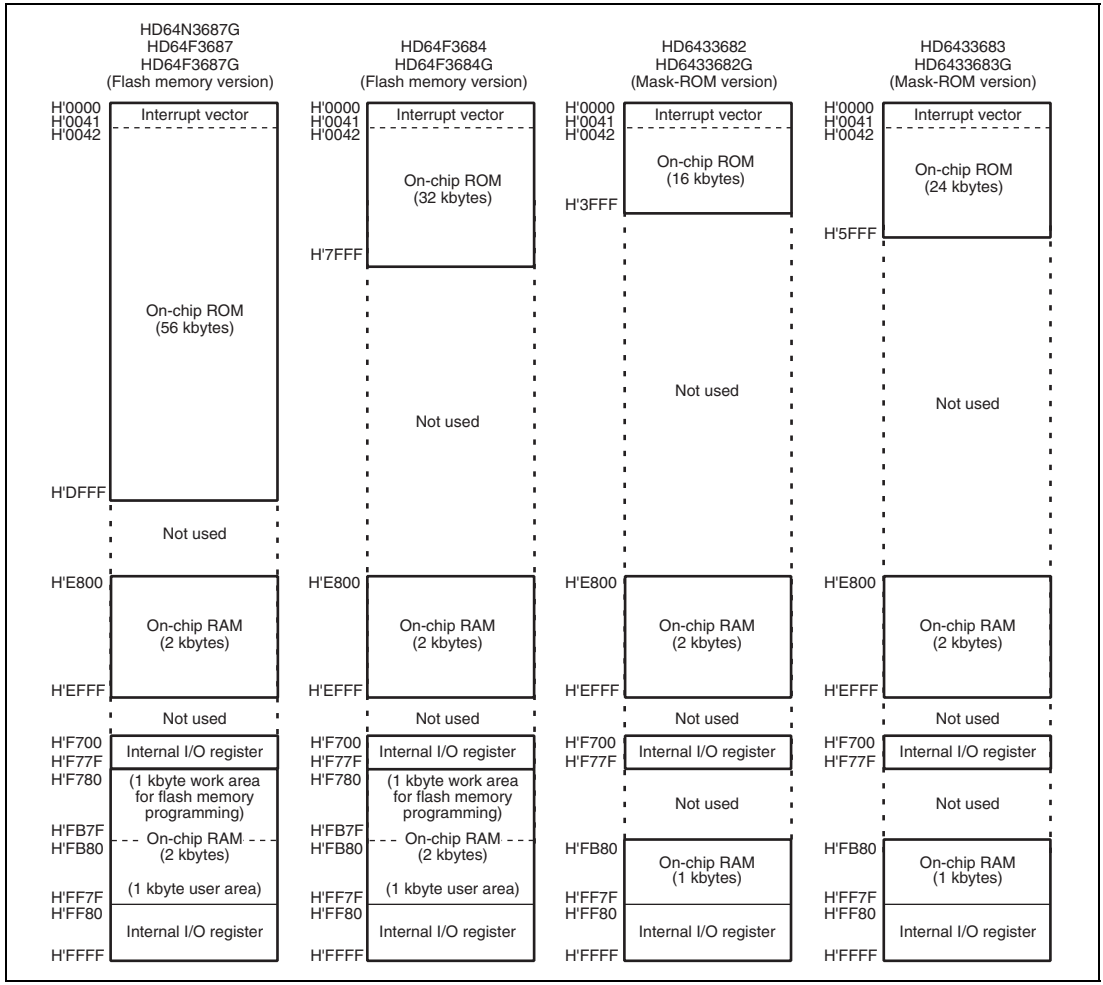


Figure 2.1 Memory Map (1)

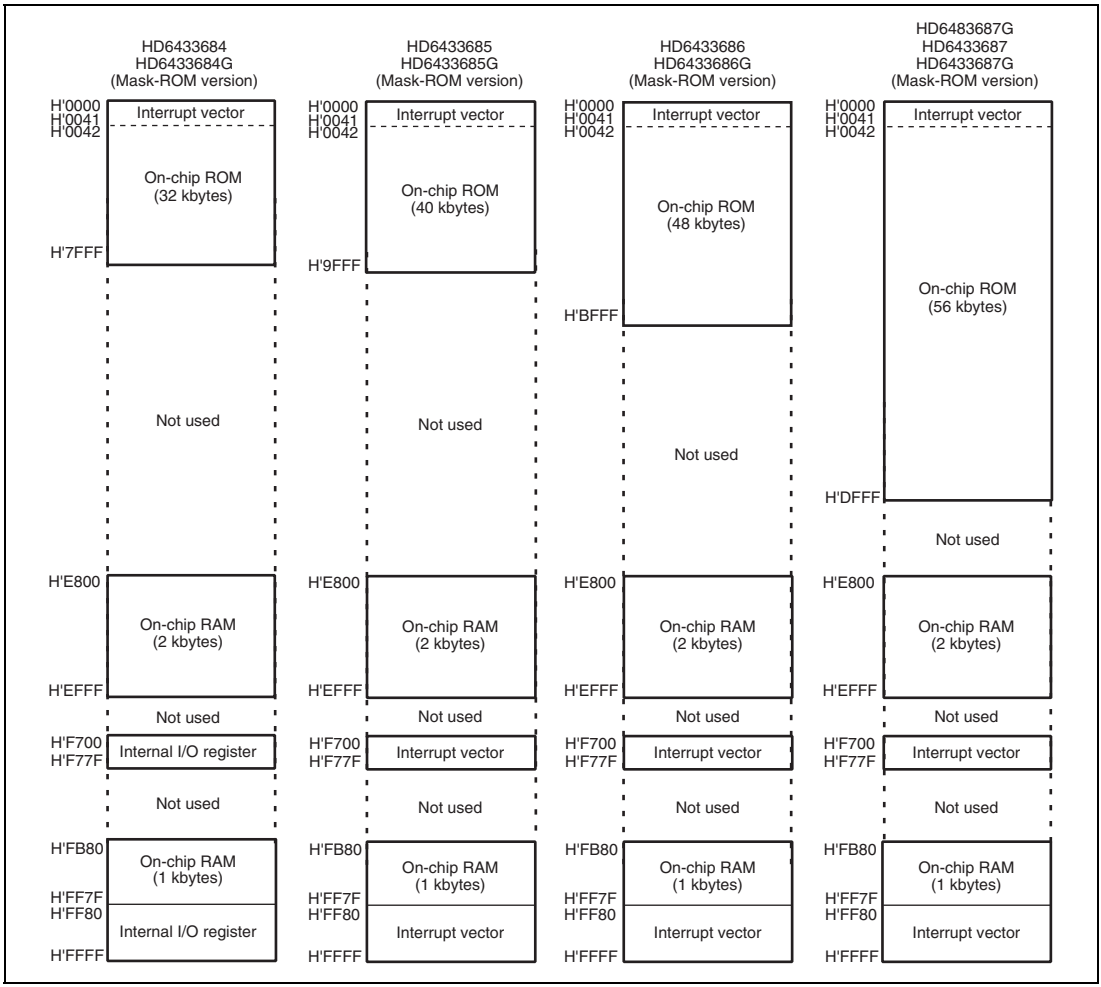


Figure 2.1 Memory Map (2)

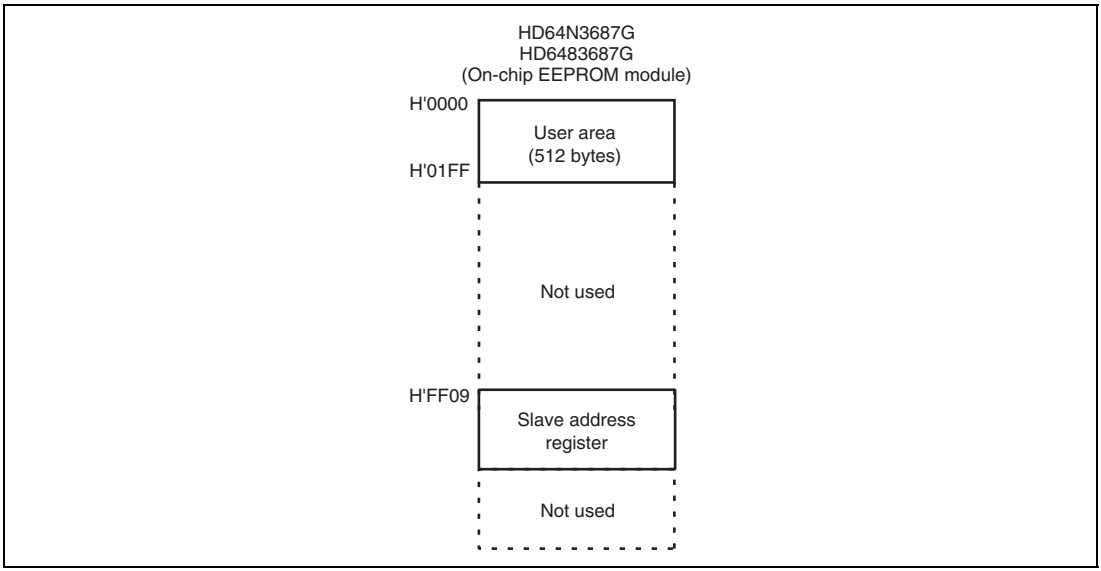


Figure 2.1 Memory Map (3)

2.2 Register Configuration

The H8/300H CPU has the internal registers shown in figure 2.2. There are two types of registers; general registers and control registers. The control registers are a 24-bit program counter (PC), and an 8-bit condition-code register (CCR).

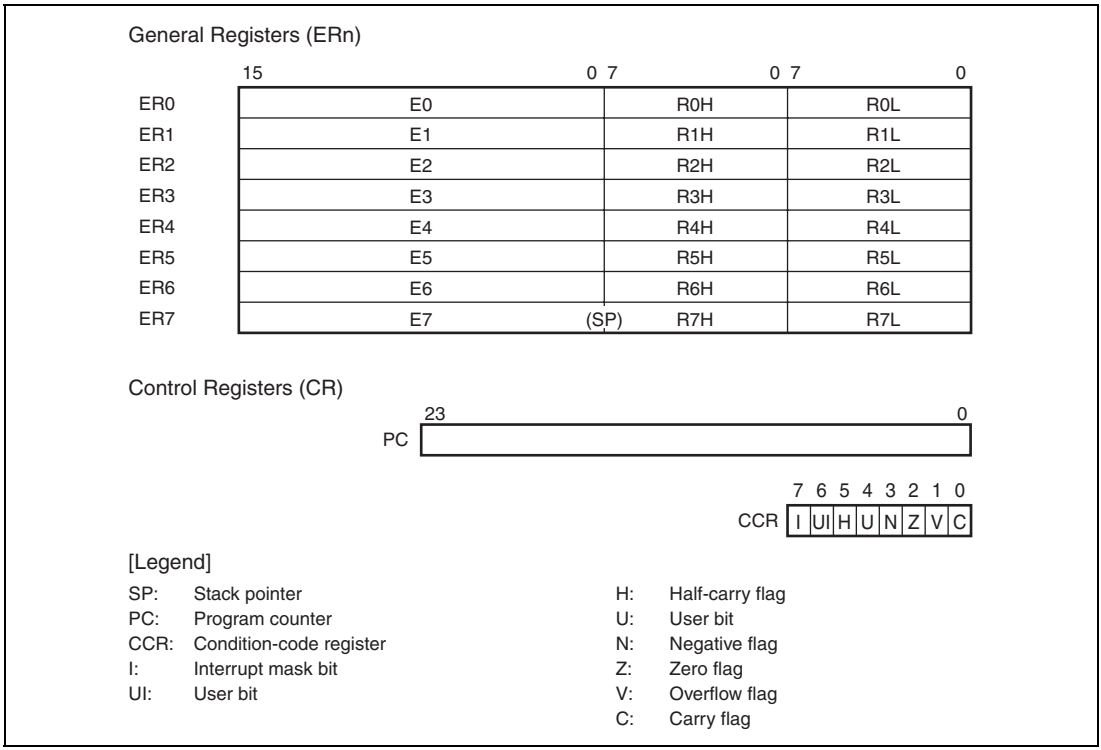


Figure 2.2 CPU Registers

2.2.1 General Registers

The H8/300H 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 2.3 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 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.

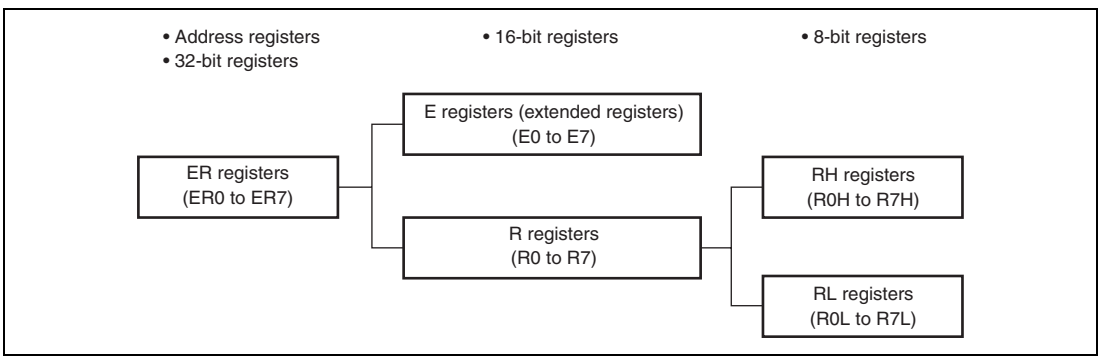


Figure 2.3 Usage of General Registers

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 2.4 shows the relationship between the stack pointer and the stack area.

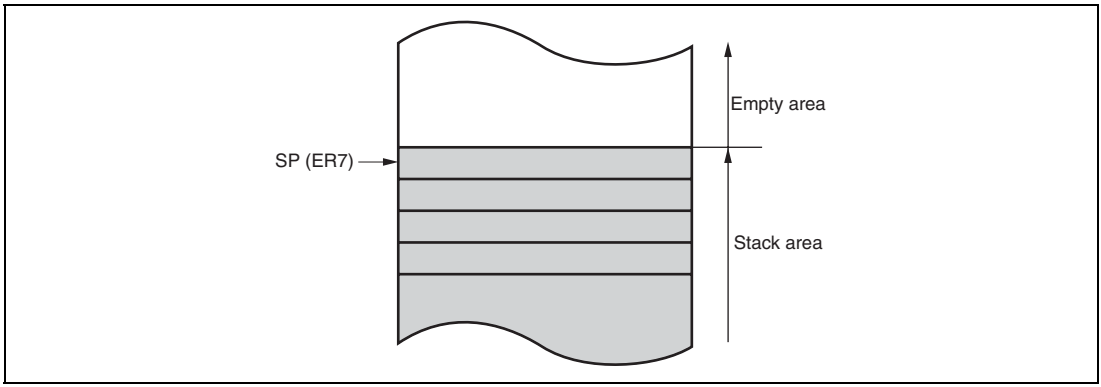


Figure 2.4 Relationship between Stack Pointer and Stack Area

2.2.2 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). The PC is initialized when the start address is loaded by the vector address generated during reset exception-handling sequence.

2.2.3 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. The I bit is initialized to 1 by reset exception-handling sequence, but other bits are not initialized.

Some instructions leave flag bits unchanged. 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.

For the action of each instruction on the flag bits, see appendix A.1, Instruction List.

Bit	Bit Name	Initial Value	R/W	Description
7	I	1	R/W	Interrupt Mask Bit Masks interrupts other than NMI when set to 1. NMI is accepted regardless of the I bit setting. The I bit is set to 1 at the start of an exception-handling sequence.
6	UI	Undefined	R/W	User Bit Can be written and read by software using the LDC, STC, ANDC, ORC, and XORC instructions.
5	H	Undefined	R/W	Half-Carry Flag When the ADD.B, ADDX.B, SUB.B, SUBX.B, CMP.B, or NEG.B instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and cleared to 0 otherwise. When the ADD.W, SUB.W, CMP.W, or NEG.W instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 11, and cleared to 0 otherwise. When the ADD.L, SUB.L, CMP.L, or NEG.L instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 27, and cleared to 0 otherwise.
4	U	Undefined	R/W	User Bit Can be written and read by software using the LDC, STC, ANDC, ORC, and XORC instructions.
3	N	Undefined	R/W	Negative Flag Stores the value of the most significant bit of data as a sign bit.
2	Z	Undefined	R/W	Zero Flag Set to 1 to indicate zero data, and cleared to 0 to indicate non-zero data.
1	V	Undefined	R/W	Overflow Flag Set to 1 when an arithmetic overflow occurs, and cleared to 0 at other times.
0	C	Undefined	R/W	Carry Flag Set to 1 when a carry occurs, and cleared to 0 otherwise. Used by: • Add instructions, to indicate a carry • Subtract instructions, to indicate a borrow • Shift and rotate instructions, to indicate a carry The carry flag is also used as a bit accumulator by bit manipulation instructions.

2.3 Data Formats

The H8/300H 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.

2.3.1 General Register Data Formats

Figure 2.5 shows the data formats in general registers.

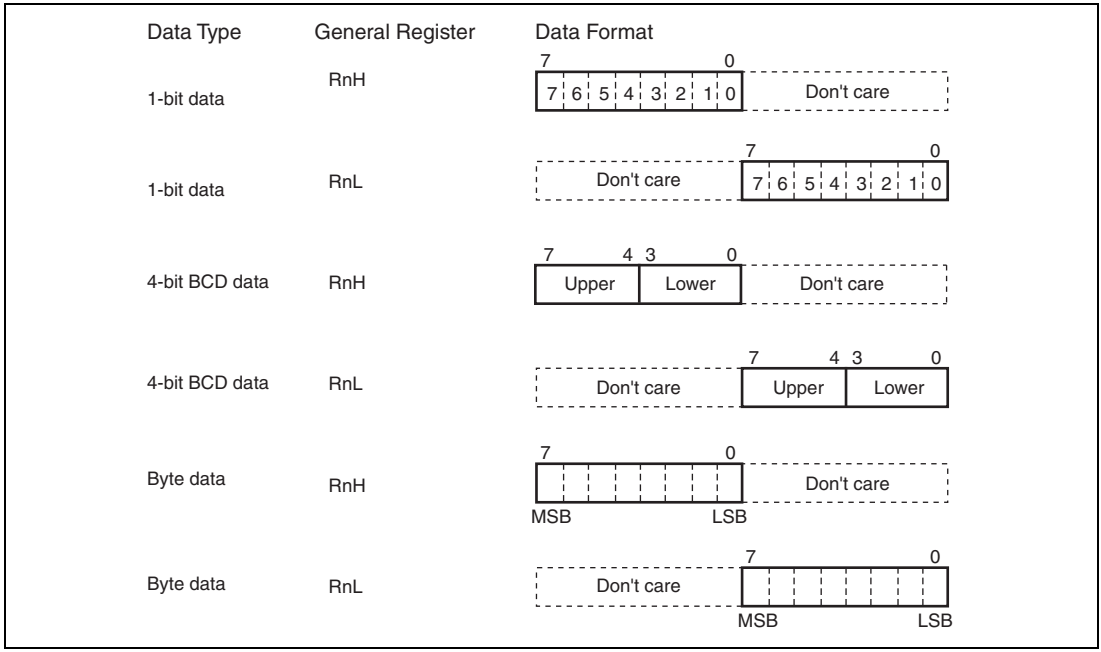


Figure 2.5 General Register Data Formats (1)

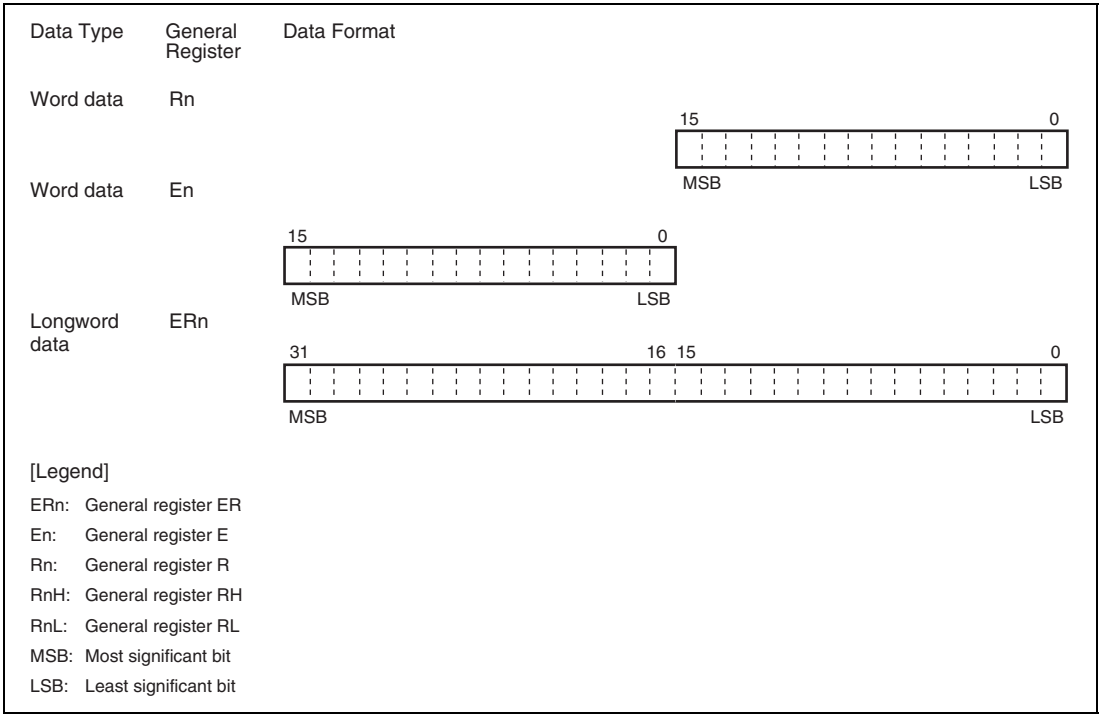


Figure 2.5 General Register Data Formats (2)

2.3.2 Memory Data Formats

Figure 2.6 shows the data formats in memory. The H8/300H 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 (SP) is used as an address register to access the stack area, the operand size should be word or longword.

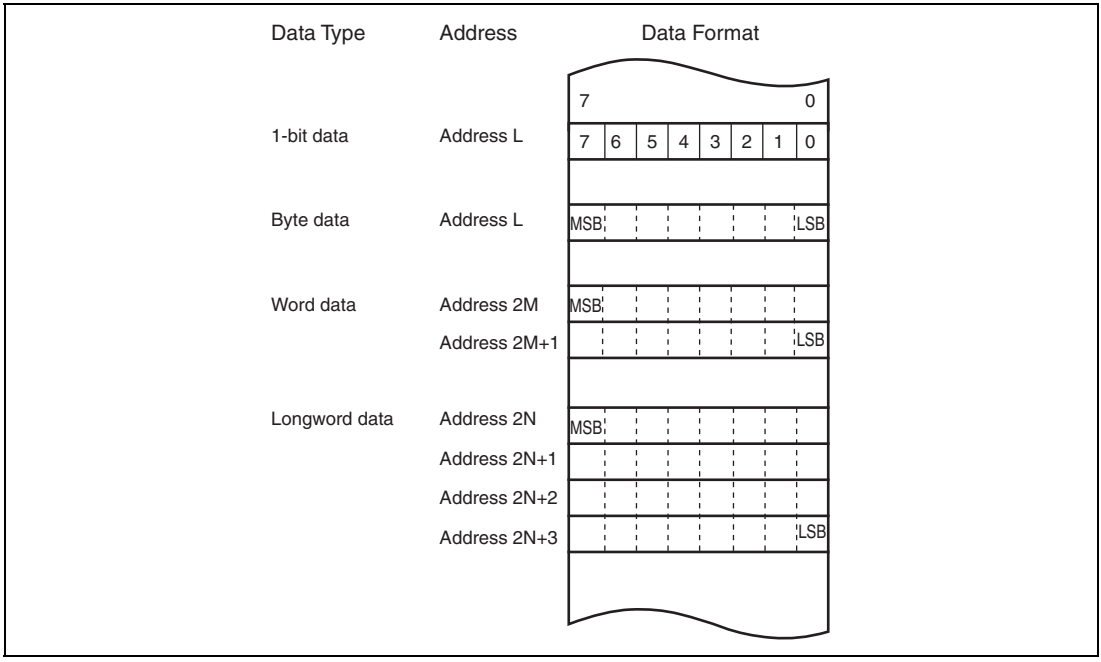


Figure 2.6 Memory Data Formats

2.4 Instruction Set

2.4.1 Table of Instructions Classified by Function

The H8/300H CPU has 62 instructions. Tables 2.2 to 2.9 summarize the instructions in each functional category. The notation used in tables 2.2 to 2.9 is defined below.

Table 2.1 Operation Notation

Symbol	Description
Rd	General register (destination)*
Rs	General register (source)*
Rn	General register*
ERn	General register (32-bit register or address register)
(EAd)	Destination operand
(EAs)	Source operand
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)
:3/:8/:16/:24	3-, 8-, 16-, or 24-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/address register (ER0 to ER7).

Table 2.2 Data Transfer Instructions

Instruction	Size*	Function
MOV	B/W/L	(EAs) → Rd, Rs → (EAd) Moves data between two general registers or between a general register and memory, or moves immediate data to a general register.
MOVFPE	B	(EAs) → Rd Cannot be used in this LSI.
MOVTPE	B	Rs → (EAs) Cannot be used in this LSI.
POP	W/L	@SP+ → Rn Pops a general register from the stack. POP.W Rn is identical to MOV.W @SP+, Rn. POP.L ERn is identical to MOV.L @SP+, ERn.
PUSH	W/L	Rn → @–SP Pushes a general register onto the stack. PUSH.W Rn is identical to MOV.W Rn, @–SP. PUSH.L ERn is identical to MOV.L ERn, @–SP.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

Table 2.3 Arithmetic Operations Instructions (1)

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

Note: * efers to the operand size.
B: Byte
W: Word
L: Longword

Table 2.3 Arithmetic Operations Instructions (2)

Instruction	Size*	Function
DIVXS	B/W	Rd ÷ Rs → Rd Performs signed division on data in two general registers: either 16 bits ÷ 8 bits → 8-bit quotient and 8-bit remainder or 32 bits ÷ 16 bits → 16-bit quotient and 16-bit remainder.
CMP	B/W/L	Rd – Rs, Rd – #IMM Compares data in a general register with data in another general register or with immediate data, and sets CCR bits according to the result.
NEG	B/W/L	0 – Rd → Rd Takes the two's complement (arithmetic complement) of data in a general register.
EXTU	W/L	Rd (zero extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the lower 16 bits of a 32-bit register to longword size, by padding with zeros on the left.
EXTS	W/L	Rd (sign extension) → Rd Extends the lower 8 bits of a 16-bit register to word size, or the lower 16 bits of a 32-bit register to longword size, by extending the sign bit.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

Table 2.4 Logic Operations Instructions

Instruction	Size*	Function
AND	B/W/L	Rd ∧ Rs → Rd, Rd ∧ #IMM → Rd Performs a logical AND operation on a general register and another general register or immediate data.
OR	B/W/L	Rd ∨ Rs → Rd, Rd ∨ #IMM → Rd Performs a logical OR operation on a general register and another general register or immediate data.
XOR	B/W/L	Rd ⊕ Rs → Rd, Rd ⊕ #IMM → Rd Performs a logical exclusive OR operation on a general register and another general register or immediate data.
NOT	B/W/L	¬ (Rd) → (Rd) Takes the one's complement (logical complement) of general register contents.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

Table 2.5 Shift Instructions

Instruction	Size*	Function
SHAL SHAR	B/W/L	Rd (shift) → Rd Performs an arithmetic shift on general register contents.
SHLL SHLR	B/W/L	Rd (shift) → Rd Performs a logical shift on general register contents.
ROTL ROTR	B/W/L	Rd (rotate) → Rd Rotates general register contents.
ROTXL ROTXR	B/W/L	Rd (rotate) → Rd Rotates general register contents through the carry flag.

Note: * Refers to the operand size.
B: Byte
W: Word
L: Longword

Table 2.6 Bit Manipulation Instructions (1)

Instruction	Size*	Function
BSET	B	1 → (<bit-No.> of <EAd>) 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 → (<bit-No.> of <EAd>) 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	¬ (<bit-No.> of <EAd>) → (<bit-No.> of <EAd>) 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	¬ (<bit-No.> of <EAd>) → Z Tests a specified bit in a general register or memory operand and sets or clears the Z flag accordingly. The bit number is specified by 3-bit immediate data or the lower three bits of a general register.
BAND	B	C ∧ (<bit-No.> of <EAd>) → C ANDs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIAND	B	C ∧ ¬ (<bit-No.> of <EAd>) → C ANDs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.
BOR	B	C ∨ (<bit-No.> of <EAd>) → C ORs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIOR	B	C ∨ ¬ (<bit-No.> of <EAd>) → C ORs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.

Note: * Refers to the operand size.
B: Byte

Table 2.6 Bit Manipulation Instructions (2)

Instruction	Size*	Function
BXOR	B	C ⊕ (<bit-No.> of <EAd>) → C XORs the carry flag with a specified bit in a general register or memory operand and stores the result in the carry flag.
BIXOR	B	C ⊕ ¬ (<bit-No.> of <EAd>) → C XORs the carry flag with the inverse of a specified bit in a general register or memory operand and stores the result in the carry flag. The bit number is specified by 3-bit immediate data.
BLD	B	(<bit-No.> of <EAd>) → C Transfers a specified bit in a general register or memory operand to the carry flag.
BILD	B	¬ (<bit-No.> of <EAd>) → C Transfers the inverse of a specified bit in a general register or memory operand to the carry flag. The bit number is specified by 3-bit immediate data.
BST	B	C → (<bit-No.> of <EAd>) Transfers the carry flag value to a specified bit in a general register or memory operand.
BIST	B	¬ C → (<bit-No.> of <EAd>) Transfers the inverse of the carry flag value to a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data.

Note: * Refers to the operand size.
B: Byte

Table 2.7 Branch Instructions

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

Note: * Bcc is the general name for conditional branch instructions.

Table 2.8 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 Moves the source operand contents to the CCR. The CCR size is one byte, but in transfer from memory, data is read by word access.
STC	B/W	CCR → (EAd) Transfers the CCR contents to a destination location. The condition code register size is one byte, but in transfer to memory, data is written by word access.
ANDC	B	CCR ∧ #IMM → CCR Logically ANDs the CCR with immediate data.
ORC	B	CCR ∨ #IMM → CCR Logically ORs the CCR with immediate data.
XORC	B	CCR ⊕ #IMM → CCR Logically XORs the CCR with immediate data.
NOP	—	PC + 2 → PC Only increments the program counter.

Note: * Refers to the operand size.
B: Byte
W: Word

Table 2.9 Block Data Transfer Instructions

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

2.4.2 Basic Instruction Formats

H8/300H CPU instructions consist of 2-byte (1-word) units. An instruction consists of an operation field (op), a register field (r), an effective address extension (EA), and a condition field (cc).

Figure 2.7 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. A24-bit address or displacement is treated as a 32-bit data in which the first 8 bits are 0 (H'00).
- Condition Field

  Specifies the branching condition of Bcc instructions.

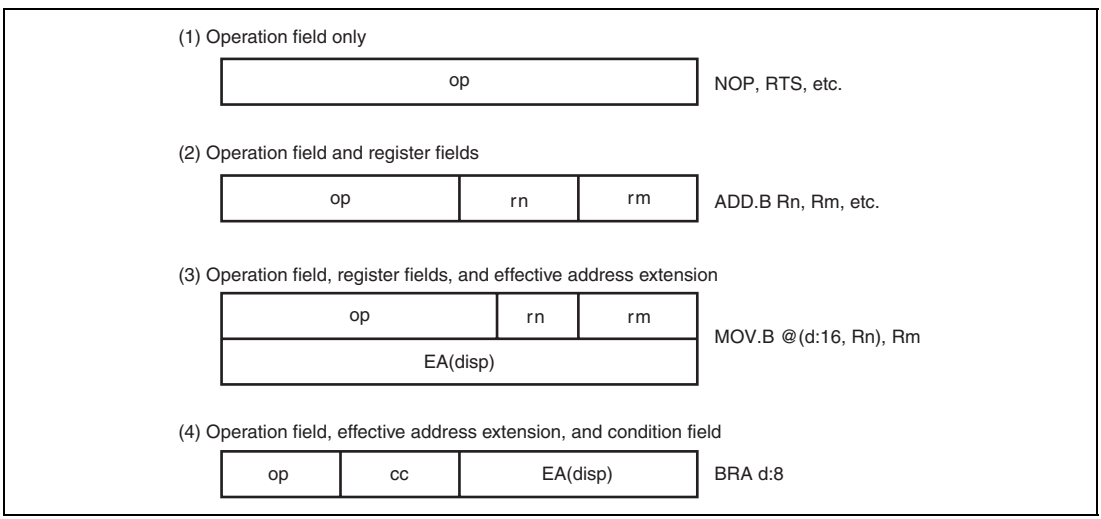


Figure 2.7 Instruction Formats

2.5 Addressing Modes and Effective Address Calculation

The following describes the H8/300H CPU. In this LSI, the upper eight bits are ignored in the generated 24-bit address, so the effective address is 16 bits.

2.5.1 Addressing Modes

The H8/300H CPU supports the eight addressing modes listed in table 2.10. Each instruction uses a subset of these addressing modes. Addressing modes that can be used differ depending on the instruction. For details, refer to appendix A.4, Combinations of Instructions and 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 (@aa:8) 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 2.10 Addressing Modes

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

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.

Register Indirect—@ERn

The register field of the instruction code specifies an address register (ERn), the lower 24 bits of which contain the address of the operand on memory.

Register Indirect with Displacement—@(d:16, ERn) or @(d:24, ERn)

A 16-bit or 24-bit displacement contained in the instruction is added to an address register (ERn) specified by the register field of the instruction, and the lower 24 bits of the sum the address of a memory operand. A 16-bit displacement is sign-extended when added.

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) the lower 24 bits of which contains the address of a memory operand. After the operand is accessed, 1, 2, or 4 is added to the address register contents (32 bits) and the sum is stored in the address register. The value added is 1 for byte access, 2 for word access, or 4 for longword access. For the word or longword access, 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 lower 24 bits of 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 access, or 4 for longword access. For the word or longword access, the register value should be even.

Absolute Address—@aa:8, @aa:16, @aa:24

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)

For an 8-bit absolute address, the upper 16 bits are all assumed to be 1 (H'FFFF). For a 16-bit absolute address the upper 8 bits are a sign extension. A 24-bit absolute address can access the entire address space.

The access ranges of absolute addresses for the group of this LSI are those shown in table 2.11, because the upper 8 bits are ignored.

Table 2.11 Absolute Address Access Ranges

Absolute Address	Access Range
8 bits (@aa:8)	H'FF00 to H'FFFF
16 bits (@aa:16)	H'0000 to H'FFFF
24 bits (@aa:24)	H'0000 to H'FFFF

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.

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

This mode is used in the BSR instruction. 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. 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.

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 memory operand is accessed by longword access. The first byte of the memory operand is ignored, generating a 24-bit branch address. Figure 2.8 shows how to specify branch address for in memory indirect mode. 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).

Note that the first part of the address range is also the exception vector area.

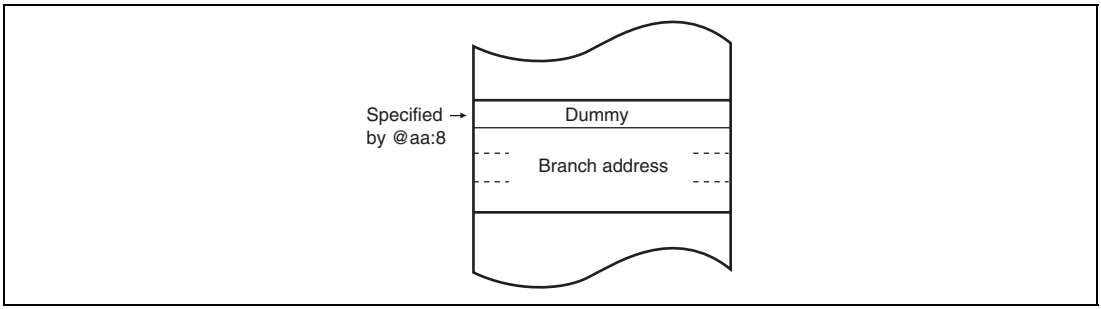


Figure 2.8 Branch Address Specification in Memory Indirect Mode

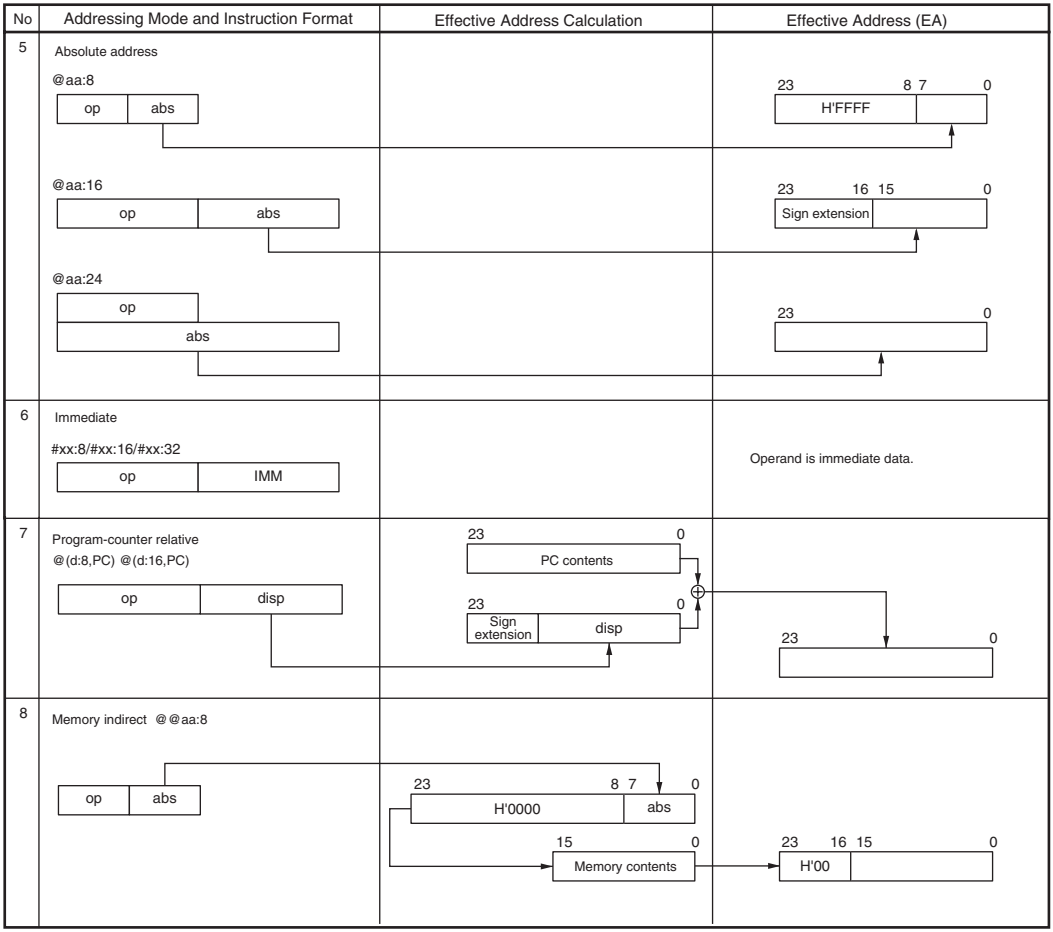
2.5.2 Effective Address Calculation

Table 2.12 indicates how effective addresses are calculated in each addressing mode. In this LSI the upper 8 bits of the effective address are ignored in order to generate a 16-bit effective address.

Table 2.12 Effective Address Calculation (1)

No	Addressing Mode and Instruction Format	Effective Address Calculation	Effective Address (EA)
1	Register direct(Rn) op rm rn		Operand is general register contents.
2	Register indirect(@ERn) op r	31 0 General register contents	23 0
3	Register indirect with displacement @(d:16,ERn) or @(d:24,ERn) op r disp	31 0 General register contents 31 0 Sign extension disp	23 0
4	Register indirect with post-increment or pre-decrement •Register indirect with post-increment @ERn+ op r •Register indirect with pre-decrement @-ERn op r	31 0 General register contents 1, 2, or 4 31 0 General register contents 1, 2, or 4 The value to be added or subtracted is 1 when the operand is byte size, 2 for word size, and 4 for longword size.	23 0 23 0

Table 2.12 Effective Address Calculation (2)



[Legend]
r, rm,rn: Register field
op: Operation field
disp: Displacement
IMM: Immediate data
abs: Absolute address

2.6 Basic Bus Cycle

CPU operation is synchronized by a system clock (ϕ) or a subclock (ϕ_{SUB}). The period from a rising edge of ϕ or ϕ_{SUB} to the next rising edge is called one state. A bus cycle consists of two states or three states. The cycle differs depending on whether access is to on-chip memory or to on-chip peripheral modules.

2.6.1 Access to On-Chip Memory (RAM, ROM)

Access to on-chip memory takes place in two states. The data bus width is 16 bits, allowing access in byte or word size. Figure 2.9 shows the on-chip memory access cycle.

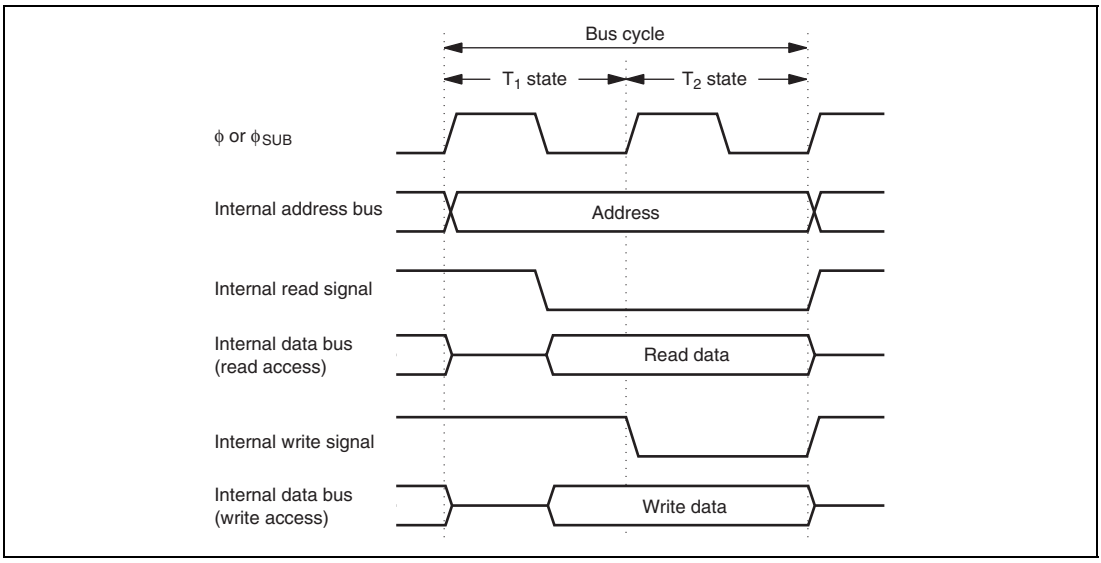


Figure 2.9 On-Chip Memory Access Cycle

2.6.2 On-Chip Peripheral Modules

On-chip peripheral modules are accessed in two states or three states. The data bus width is 8 bits or 16 bits depending on the register. For description on the data bus width and number of accessing states of each register, refer to section 22.1, Register Addresses (Address Order). Registers with 16-bit data bus width can be accessed by word size only. Registers with 8-bit data bus width can be accessed by byte or word size. When a register with 8-bit data bus width is accessed by word size, a bus cycle occurs twice. In two-state access, the operation timing is the same as that for on-chip memory.

Figure 2.10 shows the operation timing in the case of three-state access to an on-chip peripheral module.

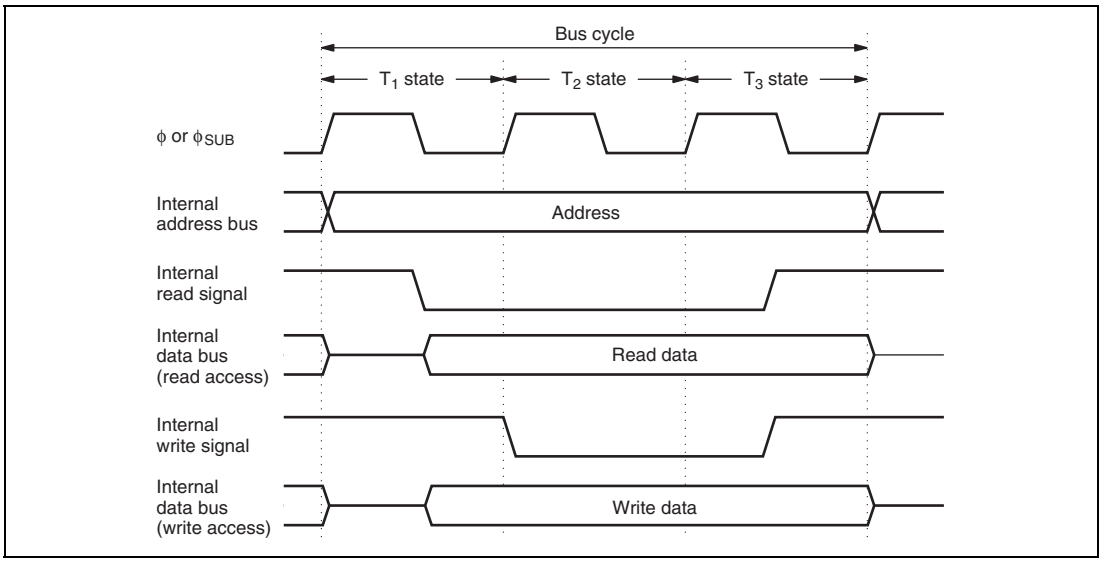


Figure 2.10 On-Chip Peripheral Module Access Cycle (3-State Access)

2.7 CPU States

There are four CPU states: the reset state, program execution state, program halt state, and exception-handling state. The program execution state includes active mode and subactive mode. For the program halt state, there are a sleep mode, standby mode, and sub-sleep mode. These states are shown in figure 2.11. Figure 2.12 shows the state transitions. For details on program execution state and program halt state, refer to section 6, Power-Down Modes. For details on exception processing, refer to section 3, Exception Handling.

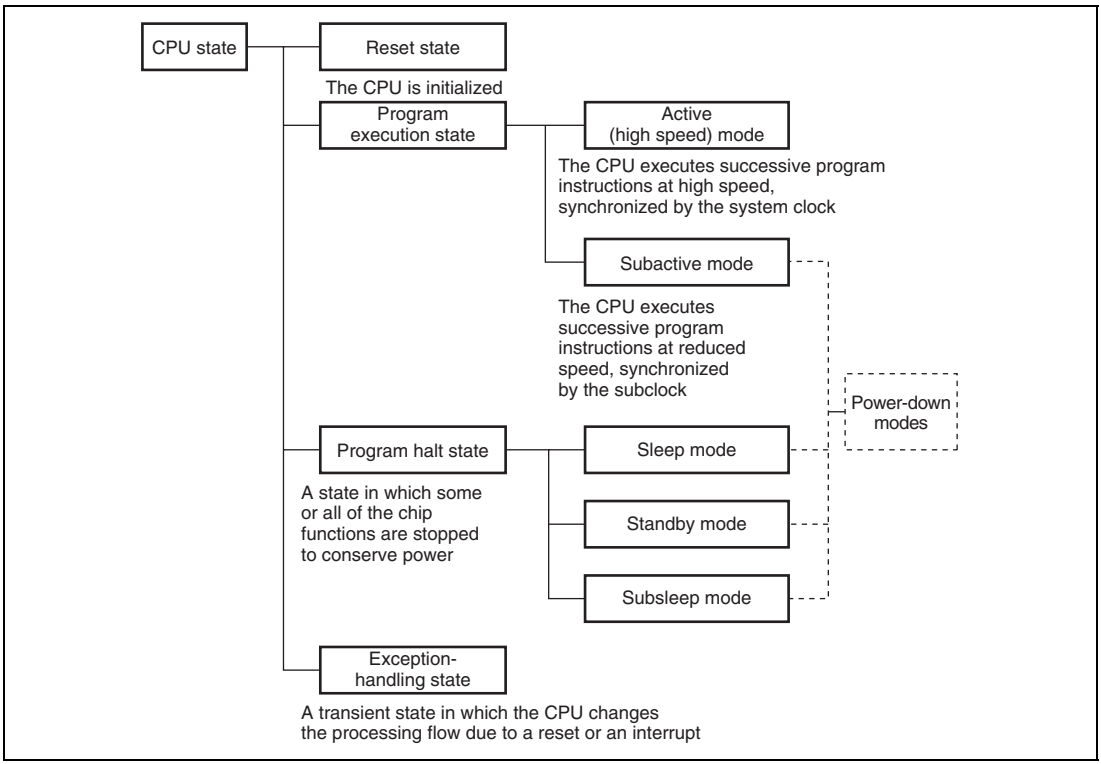


Figure 2.11 CPU Operation States

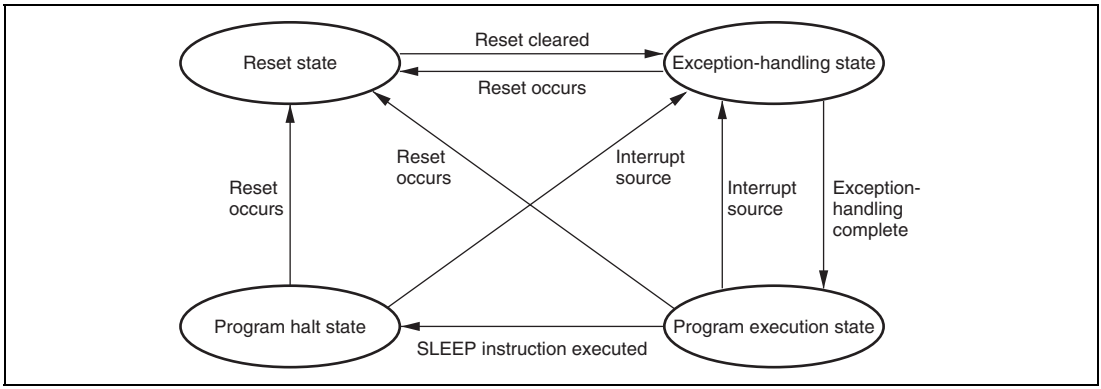


Figure 2.12 State Transitions

2.8 Usage Notes

2.8.1 Notes on Data Access to Empty Areas

The address space of this LSI includes empty areas in addition to the ROM, RAM, and on-chip I/O registers areas available to the user. When data is transferred from CPU to empty areas, the transferred data will be lost. This action may also cause the CPU to malfunction. When data is transferred from an empty area to CPU, the contents of the data cannot be guaranteed.

2.8.2 EEPMOV Instruction

EEPMOV is a block-transfer instruction and transfers the byte size of data indicated by R4L, which starts from the address indicated by R5, to the address indicated by R6. Set R4L and R6 so that the end address of the destination address (value of R6 + R4L) does not exceed H'FFFF (the value of R6 must not change from H'FFFF to H'0000 during execution).

2.8.3 Bit-Manipulation Instruction

The BSET, BCLR, BNOT, BST, and BIST instructions read data from the specified address in byte units, manipulate the data of the target bit, and write data to the same address again in byte units. Special care is required when using these instructions in cases where two registers are assigned to the same address, or when a bit is directly manipulated for a port or a register containing a write-only bit, because this may rewrite data of a bit other than the bit to be manipulated.

Bit manipulation for two registers assigned to the same address

Example 1: Bit manipulation for the timer load register and timer counter

(Applicable for timer B1 in the H8/3687 Group.)

Figure 2.13 shows an example of a timer in which two timer registers are assigned to the same address. When a bit-manipulation instruction accesses the timer load register and timer counter of a reloadable timer, since these two registers share the same address, the following operations takes place.

1. Data is read in byte units.
2. The CPU sets or resets the bit to be manipulated with the bit-manipulation instruction.
3. The written data is written again in byte units to the timer load register.

The timer is counting, so the value read is not necessarily the same as the value in the timer load register. As a result, bits other than the intended bit in the timer counter may be modified and the modified value may be written to the timer load register.

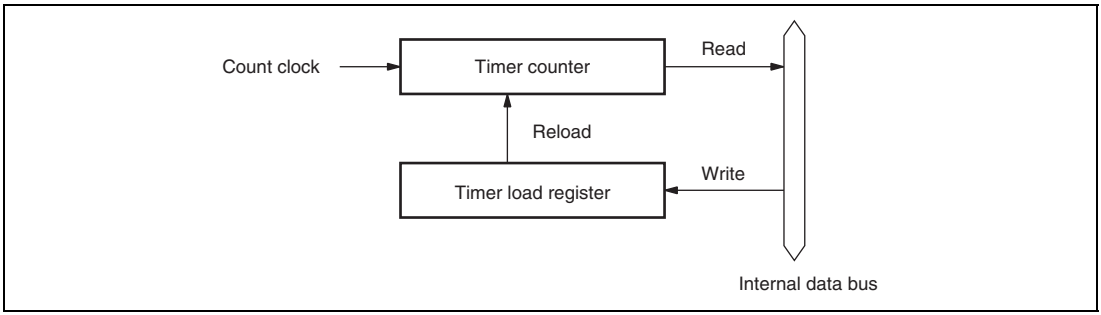


Figure 2.13 Example of Timer Configuration with Two Registers Allocated to Same Address

Example 2: The BSET instruction is executed for port 5.

P57 and P56 are input pins, with a low-level signal input at P57 and a high-level signal input at P56. P55 to P50 are output pins and output low-level signals. An example to output a high-level signal at P50 with a BSET instruction is shown below.

- Prior to executing BSET instruction

	P57	P56	P55	P54	P53	P52	P51	P50
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1	1
PDR5	1	0	0	0	0	0	0	0

- BSET instruction executed instruction

```
BSET    #0,     @PDR5
```

The BSET instruction is executed for port 5.

- After executing BSET instruction

	P57	P56	P55	P54	P53	P52	P51	P50
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR5	0	0	1	1	1	1	1	1
PDR5	0	1	0	0	0	0	0	1

- Description on operation

1. When the BSET instruction is executed, first the CPU reads port 5.
   Since P57 and P56 are input pins, the CPU reads the pin states (low-level and high-level input).
   P55 to P50 are output pins, so the CPU reads the value in PDR5. In this example PDR5 has a value of H'80, but the value read by the CPU is H'40.
2. Next, the CPU sets bit 0 of the read data to 1, changing the PDR5 data to H'41.
3. Finally, the CPU writes H'41 to PDR5, completing execution of BSET instruction.
   As a result of the BSET instruction, bit 0 in PDR5 becomes 1, and P50 outputs a high-level signal. However, bits 7 and 6 of PDR5 end up with different values. To prevent this problem, store a copy of the PDR5 data in a work area in memory. Perform the bit manipulation on the data in the work area, then write this data to PDR5.

- Prior to executing BSET instruction

```
MOV.B   #80,  R0L
MOV.B   R0L,  @RAM0
MOV.B   R0L,  @PDR5
```

The PDR5 value (H'80) is written to a work area in memory (RAM0) as well as to PDR5.

	P57	P56	P55	P54	P53	P52	P51	P50
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1	1
PDR5	1	0	0	0	0	0	0	0
RAM0	1	0	0	0	0	0	0	0

- BSET instruction executed

```
BSET    #0,    @RAM0
```

The BSET instruction is executed designating the PDR5 work area (RAM0).

- After executing BSET instruction

```
MOV.B   @RAM0, R0L
MOV.B   R0L,  @PDR5
```

The work area (RAM0) value is written to PDR5.

	P57	P56	P55	P54	P53	P52	P51	P50
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR5	0	0	1	1	1	1	1	1
PDR5	1	0	0	0	0	0	0	1
RAM0	1	0	0	0	0	0	0	1

Bit Manipulation in a Register Containing a Write-Only Bit

Example 3: BCLR instruction executed designating port 5 control register PCR5

P57 and P56 are input pins, with a low-level signal input at P57 and a high-level signal input at P56. P55 to P50 are output pins that output low-level signals. An example of setting the P50 pin as an input pin by the BCLR instruction is shown below. It is assumed that a high-level signal will be input to this input pin.

- Prior to executing BCLR instruction

	P57	P56	P55	P54	P53	P52	P51	P50
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1	1
PDR5	1	0	0	0	0	0	0	0

- BCLR instruction executed

```
BCLR    #0,     @PCR5
```

The BCLR instruction is executed for PCR5.

- After executing BCLR instruction

	P57	P56	P55	P54	P53	P52	P51	P50
Input/output	Output	Output	Output	Output	Output	Output	Output	Input
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR5	1	1	1	1	1	1	1	0
PDR5	1	0	0	0	0	0	0	0

- Description on operation

1. When the BCLR instruction is executed, first the CPU reads PCR5. Since PCR5 is a write-only register, the CPU reads a value of H'FF, even though the PCR5 value is actually H'3F.
2. Next, the CPU clears bit 0 in the read data to 0, changing the data to H'FE.
3. Finally, H'FE is written to PCR5 and BCLR instruction execution ends.

   As a result of this operation, bit 0 in PCR5 becomes 0, making P50 an input port. However, bits 7 and 6 in PCR5 change to 1, so that P57 and P56 change from input pins to output pins. To prevent this problem, store a copy of the PDR5 data in a work area in memory and manipulate data of the bit in the work area, then write this data to PDR5.

- Prior to executing BCLR instruction

```
MOV.B   #3F,  R0L
MOV.B   R0L,  @RAM0
MOV.B   R0L,  @PCR5
```

The PCR5 value (H'3F) is written to a work area in memory (RAM0) as well as to PCR5.

	P57	**P56**	**P55**	**P54**	**P53**	**P52**	**P51**	**P50**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	Low level
PCR5	0	0	1	1	1	1	1	1
PDR5	1	0	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1	1

- BCLR instruction executed

```
BCLR    #0,     @RAM0
```

The BCLR instructions executed for the PCR5 work area (RAM0).

- After executing BCLR instruction

```
MOV.B   @RAM0, R0L
MOV.B   R0L,  @PCR5
```

The work area (RAM0) value is written to PCR5.

	P57	**P56**	**P55**	**P54**	**P53**	**P52**	**P51**	**P50**
Input/output	Input	Input	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low level	High level
PCR5	0	0	1	1	1	1	1	0
PDR5	1	0	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1	0

Section 3 Exception Handling

Exception handling may be caused by a reset, a trap instruction (TRAPA), or interrupts.

- Reset

A reset has the highest exception priority. Exception handling starts as soon as the reset is cleared by the $\overline{\text{RES}}$ pin. The chip is also reset when the watchdog timer overflows, and exception handling starts. Exception handling is the same as exception handling by the $\overline{\text{RES}}$ pin.

- Trap Instruction

Exception handling starts when a trap instruction (TRAPA) is executed. The TRAPA instruction generates a vector address corresponding to a vector number from 0 to 3, as specified in the instruction code. Exception handling can be executed at all times in the program execution state, regardless of the setting of the I bit in CCR.

- Interrupts

External interrupts other than NMI and internal interrupts other than address break are masked by the I bit in CCR, and kept masked while the I bit is set to 1. Exception handling starts when the current instruction or exception handling ends, if an interrupt request has been issued.

3.1 Exception Sources and Vector Address

Table 3.1 shows the vector addresses and priority of each exception handling. When more than one interrupt is requested, handling is performed from the interrupt with the highest priority.

Table 3.1 Exception Sources and Vector Address

Relative Module	Exception Sources	Vector Number	Vector Address	Priority
$\overline{\text{RES}}$ pin Watchdog timer	Reset	0	H'0000 to H'0001	High
—	Reserved for system use	1 to 6	H'0002 to H'000D	
External interrupt pin	NMI	7	H'000E to H'000F	
CPU	Trap instruction (#0)	8	H'0010 to H'0011	
	(#1)	9	H'0012 to H'0013	
	(#2)	10	H'0014 to H'0015	
	(#3)	11	H'0016 to H'0017	
Address break	Break conditions satisfied	12	H'0018 to H'0019	
CPU	Direct transition by executing the SLEEP instruction	13	H'001A to H'001B	
External interrupt pin	IRQ0 Low-voltage detection interrupt*	14	H'001C to H'001D	
	IRQ1	15	H'001E to H'001F	
	IRQ2	16	H'0020 to H'0021	
	IRQ3	17	H'0022 to H'0023	
	WKP	18	H'0024 to H'0025	
RTC	Overflow	19	H'0026 to H'0027	
—	Reserved for system use	20	H'0028 to H'0029	
Timer V	Timer V compare match A Timer V compare match B Timer V overflow	22	H'002C to H'002D	
SCI3	SCI3 receive data full SCI3 transmit data empty SCI3 transmit end SCI3 receive error	23	H'002E to H'002F	Low

Relative Module	Exception Sources	Vector Number	Vector Address	Priority
IIC2	Transmit data empty Transmit end Receive data full Arbitration lost/Overrun error NACK detection Stop conditions detected	24	H'0030 to H'0031	High
A/D converter	A/D conversion end	25	H'0032 to H'0033	
Timer Z	Compare match/input capture A0 to D0 Timer Z overflow	26	H'0034 to H'0035	
	Compare match/input capture A1 to D1 Timer Z overflow Timer Z underflow	27	H'0036 to H'0037	
Timer B1	Timer B1 overflow	29	H'003A to H'003B	
SCI3_2	Receive data full Transmit data empty Transmit end Receive error	32	H'0040 to H'0041	Low

Note: * A low-voltage detection interrupt is enabled only in the product with an on-chip power-on reset and low-voltage detection circuit.

3.2 Register Descriptions

Interrupts are controlled by the following registers.

- Interrupt edge select register 1 (IEGR1)
- Interrupt edge select register 2 (IEGR2)
- Interrupt enable register 1 (IENR1)
- Interrupt enable register 2 (IENR2)
- Interrupt flag register 1 (IRR1)
- Interrupt flag register 2 (IRR2)
- Wakeup interrupt flag register (IWPR)

3.2.1 Interrupt Edge Select Register 1 (IEGR1)

IEGR1 selects the direction of an edge that generates interrupt requests of pins $\overline{\text{NMI}}$ and $\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$.

Bit	Bit Name	Initial Value	R/W	Description
7	NMIEG	0	R/W	NMI Edge Select 0: Falling edge of $\overline{\text{NMI}}$ pin input is detected 1: Rising edge of $\overline{\text{NMI}}$ pin input is detected
6 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	IEG3	0	R/W	IRQ3 Edge Select 0: Falling edge of $\overline{\text{IRQ3}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ3}}$ pin input is detected
2	IEG2	0	R/W	IRQ2 Edge Select 0: Falling edge of $\overline{\text{IRQ2}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ2}}$ pin input is detected
1	IEG1	0	R/W	IRQ1 Edge Select 0: Falling edge of $\overline{\text{IRQ1}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ1}}$ pin input is detected
0	IEG0	0	R/W	IRQ0 Edge Select 0: Falling edge of $\overline{\text{IRQ0}}$ pin input is detected 1: Rising edge of $\overline{\text{IRQ0}}$ pin input is detected

3.2.2 Interrupt Edge Select Register 2 (IEGR2)

IEGR2 selects the direction of an edge that generates interrupt requests of the pins $\overline{\text{ADTRG}}$ and $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1.
5	WPEG5	0	R/W	WKP5 Edge Select 0: Falling edge of $\overline{\text{WKP5}}$($\overline{\text{ADTRG}}$) pin input is detected 1: Rising edge of $\overline{\text{WKP5}}$($\overline{\text{ADTRG}}$) pin input is detected
4	WPEG4	0	R/W	WKP4 Edge Select 0: Falling edge of $\overline{\text{WKP4}}$ pin input is detected 1: Rising edge of $\overline{\text{WKP4}}$ pin input is detected
3	WPEG3	0	R/W	WKP3 Edge Select 0: Falling edge of $\overline{\text{WKP3}}$ pin input is detected 1: Rising edge of $\overline{\text{WKP3}}$ pin input is detected
2	WPEG2	0	R/W	WKP2 Edge Select 0: Falling edge of $\overline{\text{WKP2}}$ pin input is detected 1: Rising edge of $\overline{\text{WKP2}}$ pin input is detected
1	WPEG1	0	R/W	WKP1Edge Select 0: Falling edge of $\overline{\text{WKP1}}$ pin input is detected 1: Rising edge of $\overline{\text{WKP1}}$ pin input is detected
0	WPEG0	0	R/W	WKP0 Edge Select 0: Falling edge of $\overline{\text{WKP0}}$ pin input is detected 1: Rising edge of $\overline{\text{WKP0}}$ pin input is detected

3.2.3 Interrupt Enable Register 1 (IENR1)

IENR1 enables direct transition interrupts, RTC interrupts, and external pin interrupts.

Bit	Bit Name	Initial Value	R/W	Description
7	IENDT	0	R/W	Direct Transfer Interrupt Enable When this bit is set to 1, direct transition interrupt requests are enabled.
6	IENTA	0	R/W	RTC Interrupt Enable When this bit is set to 1, RTC interrupt requests are enabled.
5	IENWP	0	R/W	Wakeup Interrupt Enable This bit is an enable bit, which is common to the pins $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$. When the bit is set to 1, interrupt requests are enabled.
4	—	1	—	Reserved This bit is always read as 1.
3	IEN3	0	R/W	IRQ3 Interrupt Enable When this bit is set to 1, interrupt requests of the $\overline{\text{IRQ3}}$ pin are enabled.
2	IEN2	0	R/W	IRQ2 Interrupt Enable When this bit is set to 1, interrupt requests of the $\overline{\text{IRQ2}}$ pin are enabled.
1	IEN1	0	R/W	IRQ1 Interrupt Enable When this bit is set to 1, interrupt requests of the $\overline{\text{IRQ1}}$ pin are enabled.
0	IEN0	0	R/W	IRQ0 Interrupt Enable When this bit is set to 1, interrupt requests of the $\overline{\text{IRQ0}}$ pin are enabled.

When disabling interrupts by clearing bits in an interrupt enable register, or when clearing bits in an interrupt flag register, always do so while interrupts are masked (I = 1). If the above clear operations are performed while I = 0, and as a result a conflict arises between the clear instruction and an interrupt request, exception handling for the interrupt will be executed after the clear instruction has been executed.

3.2.4 Interrupt Enable Register 2 (IENR2)

IENR2 enables, timer B1 overflow interrupts.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	—	Reserved These bits are always read as 0.
5	IENTB1	0	R/W	Timer B1 Interrupt Enable When this bit is set to 1, timer B1 overflow interrupt requests are enabled.
4 to 0	—	All 1	—	Reserved These bits are always read as 1.

When disabling interrupts by clearing bits in an interrupt enable register, or when clearing bits in an interrupt flag register, always do so while interrupts are masked (I = 1). If the above clear operations are performed while I = 0, and as a result a conflict arises between the clear instruction and an interrupt request, exception handling for the interrupt will be executed after the clear instruction has been executed.

3.2.5 Interrupt Flag Register 1 (IRR1)

IRR1 is a status flag register for direct transition interrupts, RTC interrupts, and $\overline{IRQ3}$ to $\overline{IRQ0}$ interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
7	IRRDT	0	R/W	Direct Transfer Interrupt Request Flag [Setting condition] When a direct transfer is made by executing a SLEEP instruction while DTON in SYSCR2 is set to 1. [Clearing condition] When IRRDT is cleared by writing 0
6	IRRTA	0	R/W	RTC Interrupt Request Flag [Setting condition] When the RTC counter value overflows [Clearing condition] When IRRTA is cleared by writing 0

Bit	Bit Name	Initial Value	R/W	Description
5, 4	—	All 1	—	Reserved These bits are always read as 1.
3	IRRI3	0	R/W	IRQ3 Interrupt Request Flag [Setting condition] When $\overline{\text{IRQ3}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IRRI3 is cleared by writing 0
2	IRRI2	0	R/W	IRQ2 Interrupt Request Flag [Setting condition] When $\overline{\text{IRQ2}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IRRI2 is cleared by writing 0
1	IRRI1	0	R/W	IRQ1 Interrupt Request Flag [Setting condition] When $\overline{\text{IRQ1}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IRRI1 is cleared by writing 0
0	IRRI0	0	R/W	IRQ0 Interrupt Request Flag [Setting condition] When $\overline{\text{IRQ0}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IRRI0 is cleared by writing 0

3.2.6 Interrupt Flag Register 2 (IRR2)

IRR2 is a status flag register for timer B1 overflow interrupts.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	—	Reserved These bits are always read as 0.
5	IRRTB1	0	R/W	Timer B1 Interrupt Request flag [Setting condition] When the timer B1 counter value overflows [Clearing condition] When IRRTB1 is cleared by writing 0
4 to 0	—	All 1	—	Reserved These bits are always read as 1.

3.2.7 Wakeup Interrupt Flag Register (IWPR)

IWPR is a status flag register for $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$ interrupt requests.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1.
5	IWPF5	0	R/W	WKP5 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP5}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IWPF5 is cleared by writing 0.
4	IWPF4	0	R/W	WKP4 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP4}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IWPF4 is cleared by writing 0.

Bit	Bit Name	Initial Value	R/W	Description
3	IWPF3	0	R/W	WKP3 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP3}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IWPF3 is cleared by writing 0.
2	IWPF2	0	R/W	WKP2 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP2}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IWPF2 is cleared by writing 0.
1	IWPF1	0	R/W	WKP1 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP1}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IWPF1 is cleared by writing 0.
0	IWPF0	0	R/W	WKP0 Interrupt Request Flag [Setting condition] When $\overline{\text{WKP0}}$ pin is designated for interrupt input and the designated signal edge is detected. [Clearing condition] When IWPF0 is cleared by writing 0.

3.3 Reset Exception Handling

When the $\overline{\text{RES}}$ pin goes low, all processing halts and this LSI enters the reset. The internal state of the CPU and the registers of the on-chip peripheral modules are initialized by the reset. To ensure that this LSI is reset at power-up, hold the $\overline{\text{RES}}$ pin low until the clock pulse generator output stabilizes. To reset the chip during operation, hold the $\overline{\text{RES}}$ pin low for at least 10 system clock cycles. When the $\overline{\text{RES}}$ pin goes high after being held low for the necessary time, this LSI starts reset exception handling. The reset exception handling sequence is shown in figure 3.1. However, for the reset exception handling sequence of the product with on-chip power-on reset circuit, refer to section 20, Power-On Reset and Low-Voltage Detection Circuits (Optional).

The reset exception handling sequence is as follows:

1. Set the I bit in the condition code register (CCR) to 1.
2. The CPU generates a reset exception handling vector address (from H'0000 to H'0001), the data in that address is sent to the program counter (PC) as the start address, and program execution starts from that address.

3.4 Interrupt Exception Handling

3.4.1 External Interrupts

As the external interrupts, there are NMI, IRQ3 to IRQ0, and WKP5 to WKP0 interrupts.

NMI Interrupt

NMI interrupt is requested by input signal edge to pin $\overline{\text{NMI}}$. This interrupt is detected by either rising edge sensing or falling edge sensing, depending on the setting of bit NMIEG in IEGR1.

NMI is the highest-priority interrupt, and can always be accepted without depending on the I bit value in CCR.

IRQ3 to IRQ0 Interrupts

IRQ3 to IRQ0 interrupts are requested by input signals to pins $\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$. These four interrupts are given different vector addresses, and are detected individually by either rising edge sensing or falling edge sensing, depending on the settings of bits IEG3 to IEG0 in IEGR1.

When pins $\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$ are designated for interrupt input in PMR1 and the designated signal edge is input, the corresponding bit in IRR1 is set to 1, requesting the CPU of an interrupt. These interrupts can be masked by setting bits IEN3 to IEN0 in IENR1.

WKP5 to WKP0 Interrupts

WKP5 to WKP0 interrupts are requested by input signals to pins $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$. These six interrupts have the same vector addresses, and are detected individually by either rising edge sensing or falling edge sensing, depending on the settings of bits WPEG5 to WPEG0 in IEGR2.

When pins $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$ are designated for interrupt input in PMR5 and the designated signal edge is input, the corresponding bit in IWPR is set to 1, requesting the CPU of an interrupt. These interrupts can be masked by setting bit IENWP in IENR1.

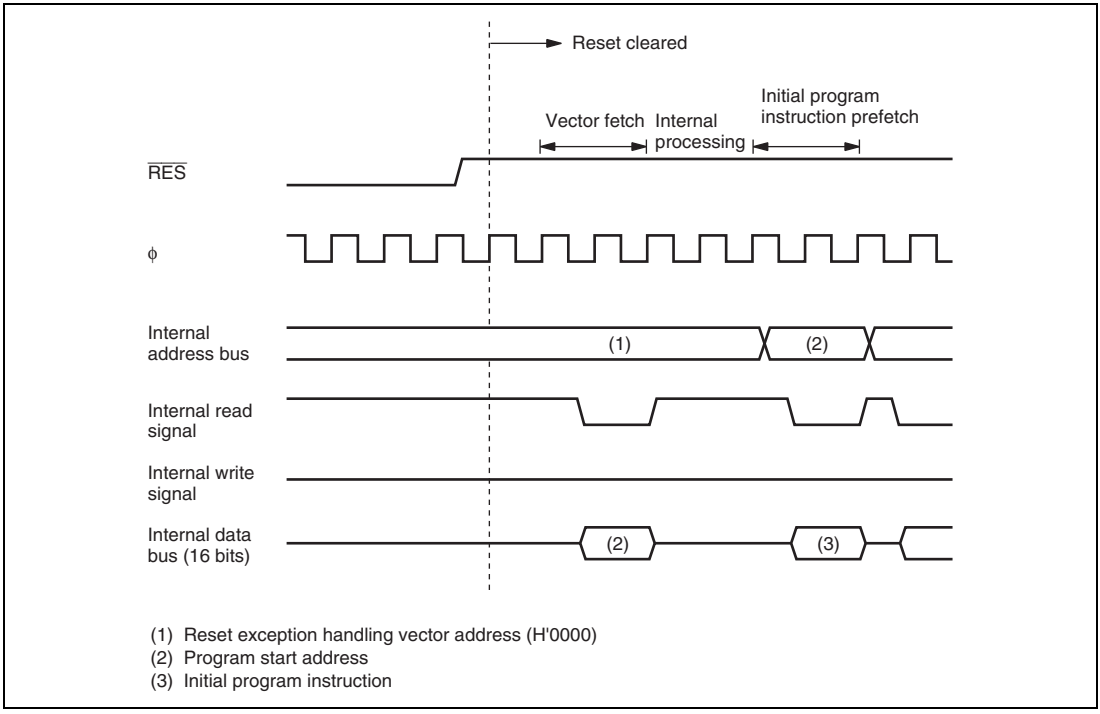


Figure 3.1 Reset Sequence

3.4.2 Internal Interrupts

Each on-chip peripheral module has a flag to show the interrupt request status and the enable bit to enable or disable the interrupt. For RTC interrupt requests and direct transfer interrupt requests generated by execution of a SLEEP instruction, this function is included in IRR1, IRR2, IENR1, and IENR2.

When an on-chip peripheral module requests an interrupt, the corresponding interrupt request status flag is set to 1, requesting the CPU of an interrupt. These interrupts can be masked by writing 0 to clear the corresponding enable bit.

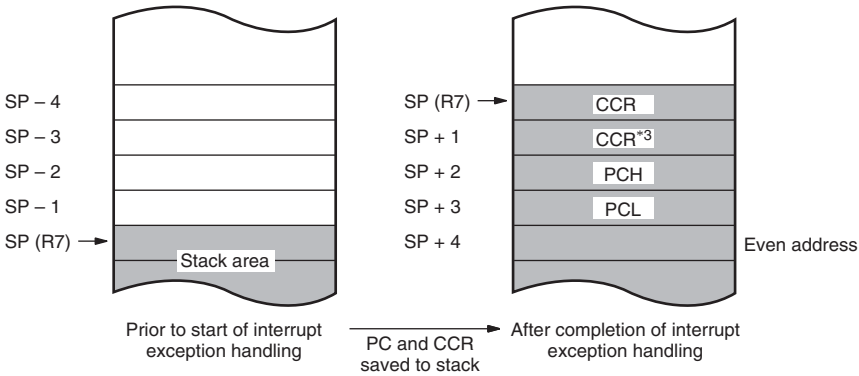
3.4.3 Interrupt Handling Sequence

Interrupts are controlled by an interrupt controller.

Interrupt operation is described as follows.

1. If an interrupt occurs while the NMI or interrupt enable bit is set to 1, an interrupt request signal is sent to the interrupt controller.
2. When multiple interrupt requests are generated, the interrupt controller requests to the CPU for the interrupt handling with the highest priority at that time according to table 3.1. Other interrupt requests are held pending.
3. The CPU accepts the NMI and address break without depending on the I bit value. Other interrupt requests are accepted, if the I bit is cleared to 0 in CCR; if the I bit is set to 1, the interrupt request is held pending.
4. If the CPU accepts the interrupt after processing of the current instruction is completed, interrupt exception handling will begin. First, both PC and CCR are pushed onto the stack. The state of the stack at this time is shown in figure 3.2. The PC value pushed onto the stack is the address of the first instruction to be executed upon return from interrupt handling.
5. Then, the I bit of CCR is set to 1, masking further interrupts excluding the NMI and address break. Upon return from interrupt handling, the values of I bit and other bits in CCR will be restored and returned to the values prior to the start of interrupt exception handling.
6. Next, the CPU generates the vector address corresponding to the accepted interrupt, and transfers the address to PC as a start address of the interrupt handling-routine. Then a program starts executing from the address indicated in PC.

Figure 3.3 shows a typical interrupt sequence where the program area is in the on-chip ROM and the stack area is in the on-chip RAM.



[Legend]
PCH: Upper 8 bits of program counter (PC)
PCL: Lower 8 bits of program counter (PC)
CCR: Condition code register
SP: Stack pointer

Notes: 1. PC shows the address of the first instruction to be executed upon return from the interrupt handling routine.
2. Register contents must always be saved and restored by word length, starting from an even-numbered address.
3. Ignored when returning from the interrupt handling routine.

Figure 3.2 Stack Status after Exception Handling

3.4.4 Interrupt Response Time

Table 3.2 shows the number of wait states after an interrupt request flag is set until the first instruction of the interrupt handling-routine is executed.

Table 3.2 Interrupt Wait States

Item	States	Total
Waiting time for completion of executing instruction*	1 to 23	15 to 37
Saving of PC and CCR to stack	4	
Vector fetch	2	
Instruction fetch	4	
Internal processing	4	

Note: * Not including EEPMOV instruction.

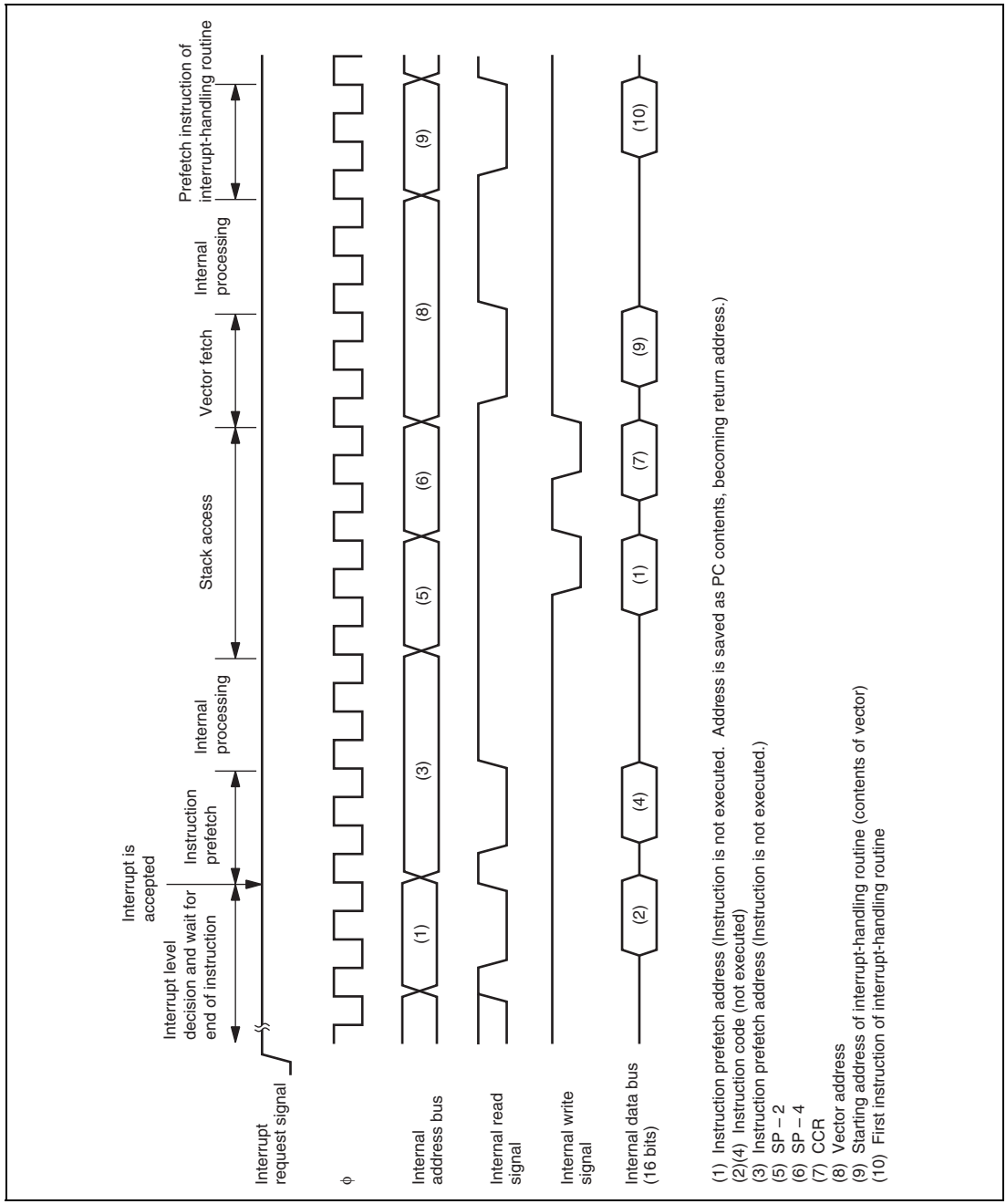


Figure 3.3 Interrupt Sequence

3.5 Usage Notes

3.5.1 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.W #xx: 16, SP).

3.5.2 Notes on Stack Area Use

When word data is accessed, the least significant bit of the address is regarded as 0. Access to the stack always takes place in word size, so the stack pointer (SP: R7) should never indicate an odd address. Use PUSH Rn (MOV.W Rn, @–SP) or POP Rn (MOV.W @SP+, Rn) to save or restore register values.

3.5.3 Notes on Rewriting Port Mode Registers

When a port mode register is rewritten to switch the functions of external interrupt pins, $\overline{\text{IRQ3}}$ to $\overline{\text{IRQ0}}$, and $\overline{\text{WKP5}}$ to $\overline{\text{WKP0}}$, the interrupt request flag may be set to 1.

When switching a pin function, mask the interrupt before setting the bit in the port mode register. After accessing the port mode register, execute at least one instruction (e.g., NOP), then clear the interrupt request flag from 1 to 0.

Figure 3.4 shows a port mode register setting and interrupt request flag clearing procedure.

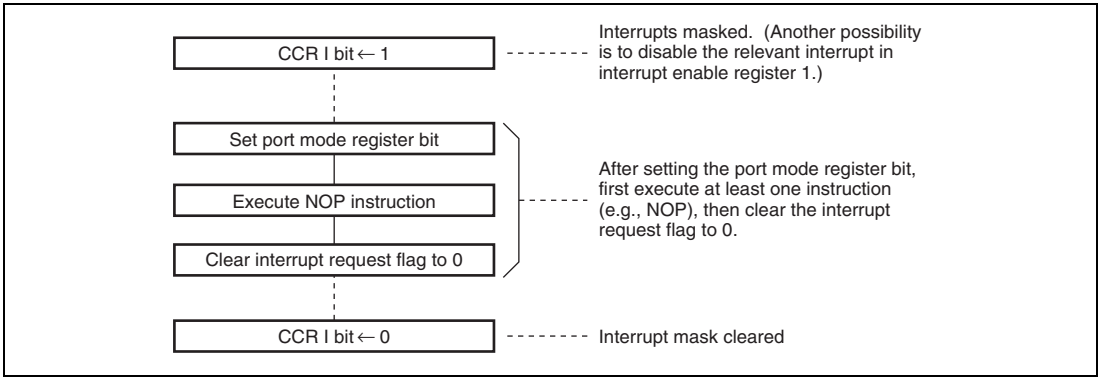


Figure 3.4 Port Mode Register Setting and Interrupt Request Flag Clearing Procedure

Section 4 Address Break

The address break simplifies on-board program debugging. It requests an address break interrupt when the set break condition is satisfied. The interrupt request is not affected by the I bit of CCR. Break conditions that can be set include instruction execution at a specific address and a combination of access and data at a specific address. With the address break function, the execution start point of a program containing a bug is detected and execution is branched to the correcting program. Figure 4.1 shows a block diagram of the address break.

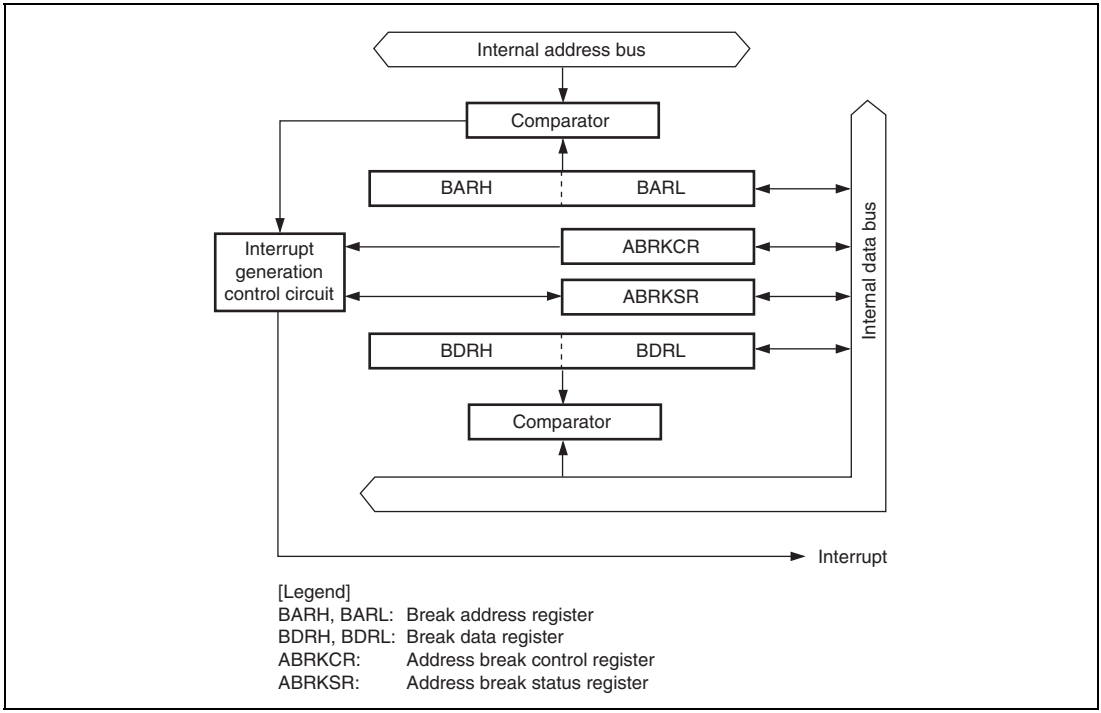


Figure 4.1 Block Diagram of Address Break

4.1 Register Descriptions

Address break has the following registers.

- Address break control register (ABRKCR)
- Address break status register (ABRKSR)
- Break address register (BARH, BARL)

- Break address register (BARH, BARL)
- Break data register (BDRH, BDRL)

4.1.1 Address Break Control Register (ABRKCR)

ABRKCR sets address break conditions.

Bit	Bit Name	Initial Value	R/W	Description
7	RTINTE	1	R/W	RTE Interrupt Enable When this bit is 0, the interrupt immediately after executing RTE is masked and then one instruction must be executed. When this bit is 1, the interrupt is not masked.
6	CSEL1	0	R/W	Condition Select 1 and 0
5	CSEL0	0	R/W	These bits set address break conditions. 00: Instruction execution cycle 01: CPU data read cycle 10: CPU data write cycle 11: CPU data read/write cycle
4	ACMP2	0	R/W	Address Compare Condition Select 2 to 0
3	ACMP1	0	R/W	These bits set the comparison condition between the address set in BAR and the internal address bus.
2	ACMP0	0	R/W	000: Compares 16-bit addresses 001: Compares upper 12-bit addresses 010: Compares upper 8-bit addresses 011: Compares upper 4-bit addresses 1XX: Reserved (setting prohibited)
1	DCMP1	0	R/W	Data Compare Condition Select 1 and 0
0	DCMP0	0	R/W	These bits set the comparison condition between the data set in BDR and the internal data bus. 00: No data comparison 01: Compares lower 8-bit data between BDRL and data bus 10: Compares upper 8-bit data between BDRH and data bus 11: Compares 16-bit data between BDR and data bus

Legend: X: Don't care.

When an address break is set in the data read cycle or data write cycle, the data bus used will depend on the combination of the byte/word access and address. Table 4.1 shows the access and data bus used. When an I/O register space with an 8-bit data bus width is accessed in word size, a byte access is generated twice. For details on data widths of each register, see section 22.1, Register Addresses (Address Order).

Table 4.1 Access and Data Bus Used

	Word Access		Byte Access	
	Even Address	**Odd Address**	**Even Address**	**Odd Address**
ROM space	Upper 8 bits	Lower 8 bits	Upper 8 bits	Upper 8 bits
RAM space	Upper 8 bits	Lower 8 bits	Upper 8 bits	Upper 8 bits
I/O register with 8-bit data bus width	Upper 8 bits	Upper 8 bits	Upper 8 bits	Upper 8 bits
I/O register with 16-bit data bus width	Upper 8 bits	Lower 8 bits	—	—

4.1.2 Address Break Status Register (ABRKSR)

ABRKSR consists of the address break interrupt flag and the address break interrupt enable bit.

Bit	Bit Name	Initial Value	R/W	Description
7	ABIF	0	R/W	Address Break Interrupt Flag [Setting condition] When the condition set in ABRKCR is satisfied [Clearing condition] When 0 is written after ABIF=1 is read
6	ABIE	0	R/W	Address Break Interrupt Enable When this bit is 1, an address break interrupt request is enabled.
5 to 0	—	All 1	—	Reserved These bits are always read as 1.

4.1.3 Break Address Registers (BARH, BARL)

BARH and BARL are 16-bit read/write registers that set the address for generating an address break interrupt. When setting the address break condition to the instruction execution cycle, set the first byte address of the instruction. The initial value of this register is H'FFFF.

4.1.4 Break Data Registers (BDRH, BDRL)

BDRH and BDRL are 16-bit read/write registers that set the data for generating an address break interrupt. BDRH is compared with the upper 8-bit data bus. BDRL is compared with the lower 8-bit data bus. When memory or registers are accessed by byte, the upper 8-bit data bus is used for even and odd addresses in the data transmission. Therefore, comparison data must be set in BDRH for byte access. For word access, the data bus used depends on the address. See section 4.1.1, Address Break Control Register (ABRKCR), for details. The initial value of this register is undefined.

4.2 Operation

When the ABIF and ABIE bits in ABRKSR are set to 1, the address break function generates an interrupt request to the CPU. The ABIF bit in ABRKSR is set to 1 by the combination of the address set in BAR, the data set in BDR, and the conditions set in ABRKCR. When the interrupt request is accepted, interrupt exception handling starts after the instruction being executed ends. The address break interrupt is not masked by the I bit in CCR of the CPU.

Figures 4.2 show the operation examples of the address break interrupt setting.

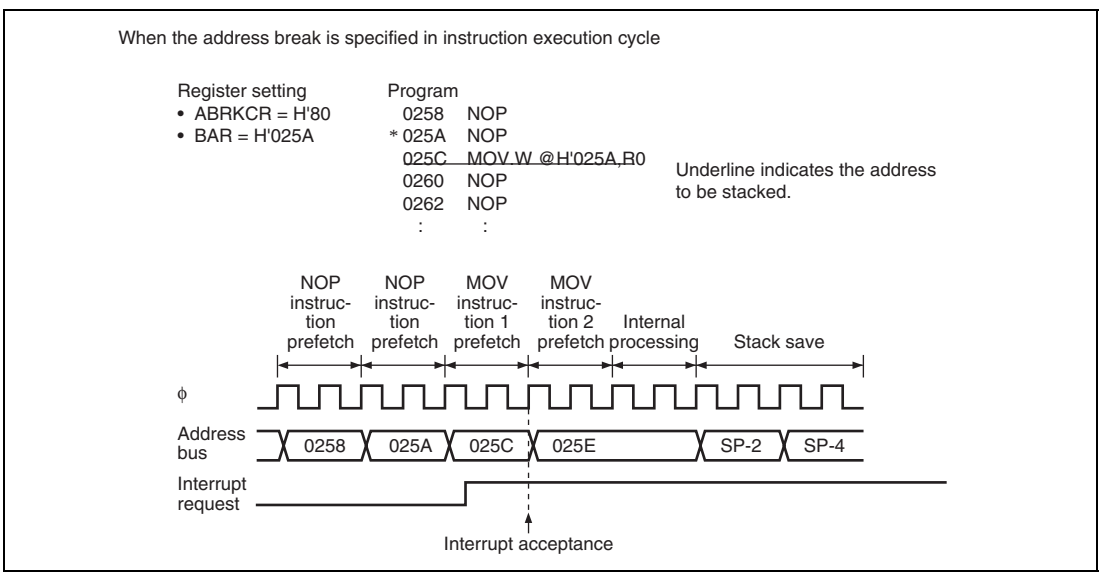


Figure 4.2 Address Break Interrupt Operation Example (1)

When the address break is specified in the data read cycle

Register setting
- ABRKCR = H'A0
- BAR = H'025A

Program

```
  0258   NOP
  025A   NOP
* 025C   MOV.W @H'025A,R0
  0260   NOP
  0262   NOP
   :      :
```

Underline indicates the address to be stacked.

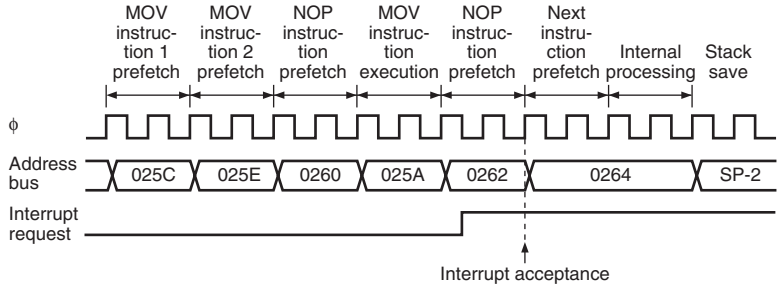


Figure 4.2 Address Break Interrupt Operation Example (2)

Section 5 Clock Pulse Generators

Clock oscillator circuitry (CPG: clock pulse generator) is provided on-chip, including both a system clock pulse generator and a subclock pulse generator. The system clock pulse generator consists of a system clock oscillator, a duty correction circuit, and system clock dividers. The subclock pulse generator consists of a subclock oscillator circuit and a subclock divider.

Figure 5.1 shows a block diagram of the clock pulse generators.

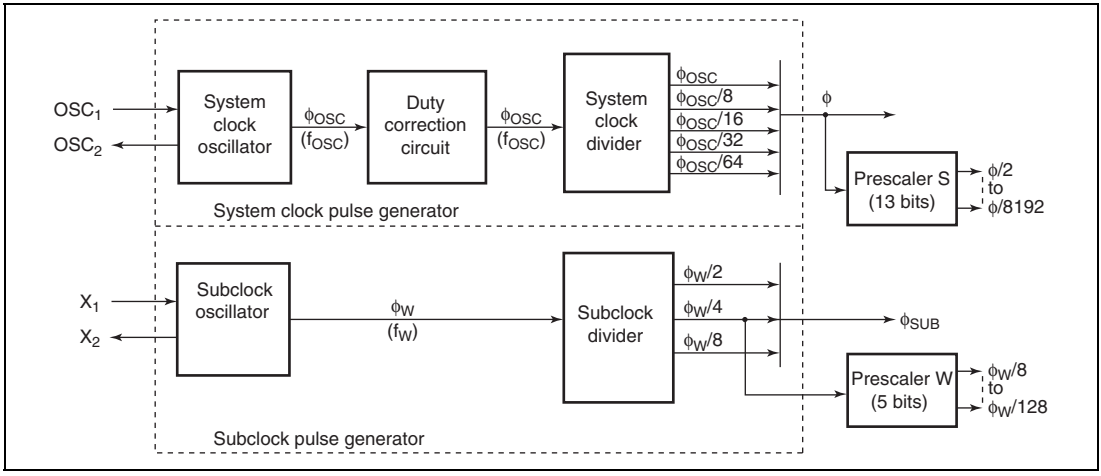
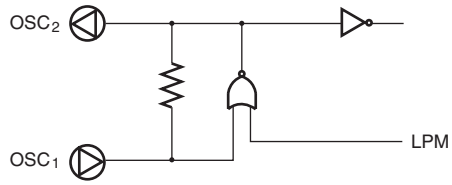


Figure 5.1 Block Diagram of Clock Pulse Generators

The basic clock signals that drive the CPU and on-chip peripheral modules are ϕ and ϕ_{SUB}. The system clock is divided by prescaler S to become a clock signal from $\phi/8192$ to $\phi/2$, and the subclock is divided by prescaler W to become a clock signal from $\phi w/128$ to $\phi w/8$. Both the system clock and subclock signals are provided to the on-chip peripheral modules.

5.1 System Clock Generator

Clock pulses can be supplied to the system clock divider either by connecting a crystal or ceramic resonator, or by providing external clock input. Figure 5.2 shows a block diagram of the system clock generator.



LPM: Low-power mode (standby mode, subactive mode, subsleep mode)

Figure 5.2 Block Diagram of System Clock Generator

5.1.1 Connecting Crystal Resonator

Figure 5.3 shows a typical method of connecting a crystal resonator. An AT-cut parallel-resonance crystal resonator should be used. Figure 5.4 shows the equivalent circuit of a crystal resonator. A resonator having the characteristics given in table 5.1 should be used.

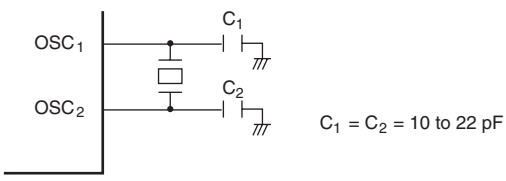


Figure 5.3 Typical Connection to Crystal Resonator

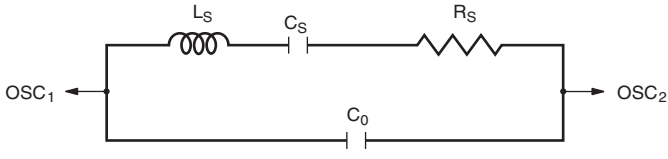


Figure 5.4 Equivalent Circuit of Crystal Resonator

Table 5.1 Crystal Resonator Parameters

Frequency (MHz)	2	4	8	10	16	20
R_s (max)	500 Ω	120 Ω	80 Ω	60 Ω	50 Ω	40 Ω
C_0 (max)	7 pF	7 pF	7 pF	7 pF	7 pF	7 pF

5.1.2 Connecting Ceramic Resonator

Figure 5.5 shows a typical method of connecting a ceramic resonator.

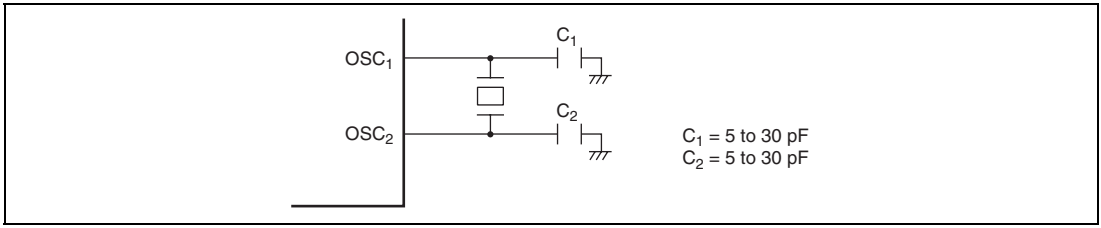


Figure 5.5 Typical Connection to Ceramic Resonator

5.1.3 External Clock Input Method

Connect an external clock signal to pin OSC_1, and leave pin OSC_2 open. Figure 5.6 shows a typical connection. The duty cycle of the external clock signal must be 45 to 55%.

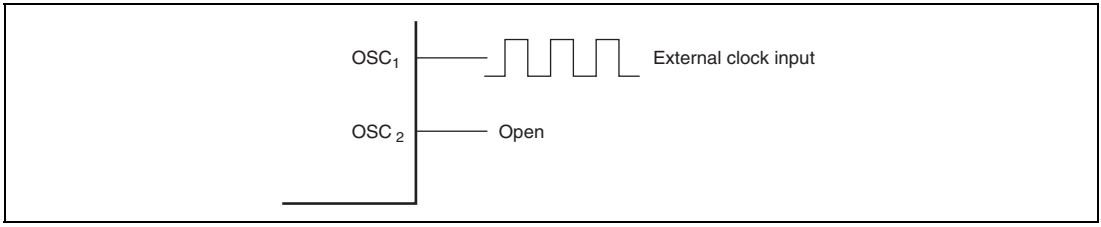
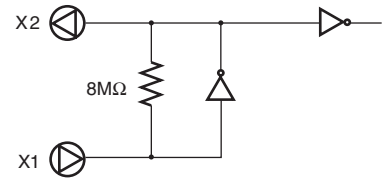


Figure 5.6 Example of External Clock Input

5.2 Subclock Generator

Figure 5.7 shows a block diagram of the subclock generator.



Note : Registance is a reference value.

Figure 5.7 Block Diagram of Subclock Generator

5.2.1 Connecting 32.768-kHz Crystal Resonator

Clock pulses can be supplied to the subclock divider by connecting a 32.768-kHz crystal resonator, as shown in figure 5.8. Figure 5.9 shows the equivalent circuit of the 32.768-kHz crystal resonator.

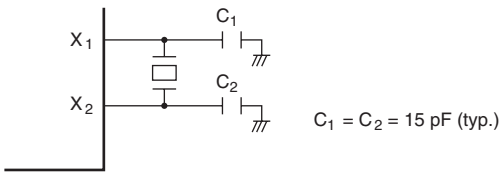
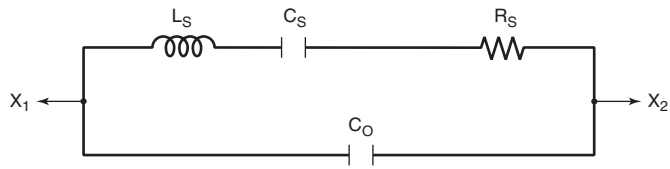


Figure 5.8 Typical Connection to 32.768-kHz Crystal Resonator



C_O = 1.5 pF (typ.)
R_S = 14 kΩ (typ.)
f_W = 32.768 kHz

Note: Constants are reference values.

Figure 5.9 Equivalent Circuit of 32.768-kHz Crystal Resonator

5.2.2 Pin Connection when Not Using Subclock

When the subclock is not used, connect pin X_1 to V_{CL} or V_{SS} and leave pin X_2 open, as shown in figure 5.10.

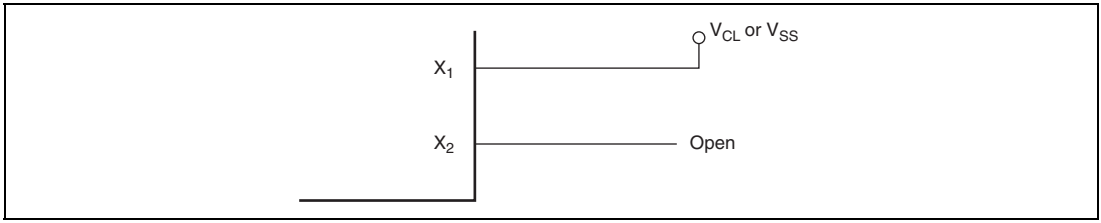


Figure 5.10 Pin Connection when not Using Subclock

5.3 Prescalers

5.3.1 Prescaler S

Prescaler S is a 13-bit counter using the system clock (ø) as its input clock. It is incremented once per clock period. Prescaler S is initialized to H'0000 by a reset, and starts counting on exit from the reset state. In standby mode, subactive mode, and subsleep mode, the system clock pulse generator stops. Prescaler S also stops and is initialized to H'0000. The CPU cannot read or write prescaler S. The output from prescaler S is shared by the on-chip peripheral modules. The divider ratio can be set separately for each on-chip peripheral function. In active mode and sleep mode, the clock input to prescaler S is determined by the division factor designated by MA2 to MA0 in SYSCR2.

5.3.2 Prescaler W

Prescaler W is a 5-bit counter using a 32.768 kHz signal divided by 4 ($ø_w/4$) as its input clock. The divided output is used for clock time base operation of timer A. Prescaler W is initialized to H'00 by a reset, and starts counting on exit from the reset state. Even in standby mode, subactive mode, or subsleep mode, prescaler W continues functioning so long as clock signals are supplied to pins X_1 and X_2.

5.4 Usage Notes

5.4.1 Note on Resonators

Resonator characteristics are closely related to board design and should be carefully evaluated by the user, referring to the examples shown in this section. Resonator circuit constants will differ depending on the resonator element, stray capacitance in its interconnecting circuit, and other factors. Suitable constants should be determined in consultation with the resonator element manufacturer. Design the circuit so that the resonator element never receives voltages exceeding its maximum rating.

5.4.2 Notes on Board Design

When using a crystal resonator (ceramic resonator), place the resonator and its load capacitors as close as possible to the OSC_1 and OSC_2 pins. Other signal lines should be routed away from the resonator circuit to prevent induction from interfering with correct oscillation (see figure 5.11).

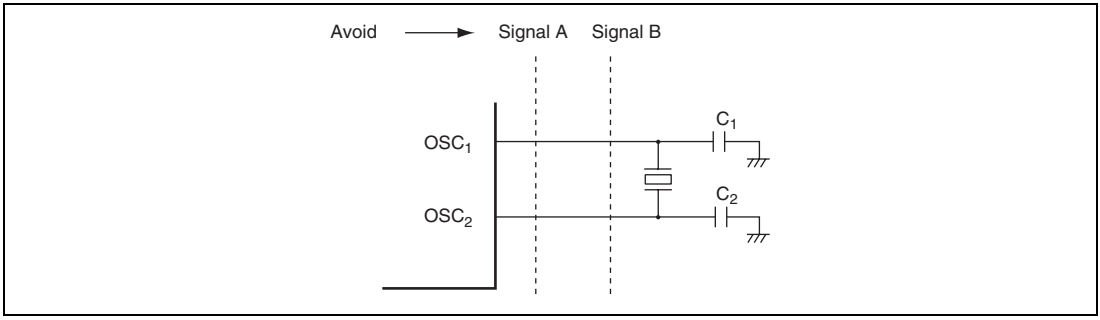


Figure 5.11 Example of Incorrect Board Design

Section 6 Power-Down Modes

This LSI has six modes of operation after a reset. These include a normal active mode and four power-down modes, in which power consumption is significantly reduced. Module standby mode reduces power consumption by selectively halting on-chip module functions.

- Active mode

  The CPU and all on-chip peripheral modules are operable on the system clock. The system clock frequency can be selected from ϕosc, ϕosc/8, ϕosc/16, ϕosc/32, and ϕosc/64.
- Subactive mode

  The CPU and all on-chip peripheral modules are operable on the subclock. The subclock frequency can be selected from ϕw/2, ϕw/4, and ϕw/8.
- Sleep mode

  The CPU halts. On-chip peripheral modules are operable on the system clock.
- Subsleep mode

  The CPU halts. On-chip peripheral modules are operable on the subclock.
- Standby mode

  The CPU and all on-chip peripheral modules halt. When the clock time-base function is selected, the RTC is operable.
- Module standby mode

  Independent of the above modes, power consumption can be reduced by halting on-chip peripheral modules that are not used in module units.

6.1 Register Descriptions

The registers related to power-down modes are listed below.

- System control register 1 (SYSCR1)
- System control register 2 (SYSCR2)
- Module standby control register 1 (MSTCR1)
- Module standby control register 2 (MSTCR2)

6.1.1 System Control Register 1 (SYSCR1)

SYSCR1 controls the power-down modes, as well as SYSCR2.

Bit	Bit Name	Initial Value	R/W	Description
7	SSBY	0	R/W	Software Standby This bit selects the mode to transit after the execution of the SLEEP instruction. 0: Enters sleep mode or subsleep mode. 1: Enters standby mode. For details, see table 6.2.
6	STS2	0	R/W	Standby Timer Select 2 to 0 These bits designate the time the CPU and peripheral modules wait for stable clock operation after exiting from standby mode, subactive mode, or subsleep mode to active mode or sleep mode due to an interrupt. The designation should be made according to the clock frequency so that the waiting time is at least 6.5 ms. The relationship between the specified value and the number of wait states is shown in table 6.1. When an external clock is to be used, the minimum value (STS2 = STS1 = STS0 =1) is recommended.
5	STS1	0	R/W	
4	STS0	0	R/W	
3	NESEL	0	R/W	Noise Elimination Sampling Frequency Select The subclock pulse generator generates the watch clock signal (ϕ_W) and the system clock pulse generator generates the oscillator clock (ϕ_{OSC}). This bit selects the sampling frequency of the oscillator clock when the watch clock signal (ϕ_W) is sampled. When ϕ_{OSC} = 4 to 20 MHz, clear NESEL to 0. 0: Sampling rate is ϕ_{OSC}/16 1: Sampling rate is ϕ_{OSC}/4
2 to 0	—	All 0	—	Reserved These bits are always read as 0.

Table 6.1 Operating Frequency and Waiting Time

Bit Name				Operating Frequency							
STS2	STS1	STS0	Waiting Time	20 MHz	16 MHz	10 MHz	8 MHz	4 MHz	2 MHz	1 MHz	0.5 MHz
0	0	0	8,192 states	0.4	0.5	0.8	1.0	2.0	4.1	8.1	16.4
		1	16,384 states	0.8	1.0	1.6	2.0	4.1	8.2	16.4	32.8
	1	0	32,768 states	1.6	2.0	3.3	4.1	8.2	16.4	32.8	65.5
		1	65,536 states	3.3	4.1	6.6	8.2	16.4	32.8	65.5	131.1
1	0	0	131,072 states	6.6	8.2	13.1	16.4	32.8	65.5	131.1	262.1
		1	1,024 states	0.05	0.06	0.10	0.13	0.26	0.51	1.02	2.05
	1	0	128 states	0.00	0.00	0.01	0.02	0.03	0.06	0.13	0.26
		1	16 states	0.00	0.00	0.00	0.00	0.00	0.01	0.02	0.03

Note: Time unit is ms.

6.1.2 System Control Register 2 (SYSCR2)

SYSCR2 controls the power-down modes, as well as SYSCR1.

Bit	Bit Name	Initial Value	R/W	Description
7	SMSEL	0	R/W	Sleep Mode Selection
6	LSON	0	R/W	Low Speed on Flag
5	DTON	0	R/W	Direct Transfer on Flag These bits select the mode to enter after the execution of a SLEEP instruction, as well as bit SSBY of SYSCR1. For details, see table 6.2.
4	MA2	0	R/W	Active Mode Clock Select 2 to 0
3	MA1	0	R/W	These bits select the operating clock frequency in active and sleep modes. The operating clock frequency changes to the set frequency after the SLEEP instruction is executed. 0XX: ϕ_{osc} 100: $\phi_{osc}/8$ 101: $\phi_{osc}/16$ 110: $\phi_{osc}/32$ 111: $\phi_{osc}/64$
2	MA0	0	R/W	
1	SA1	0	R/W	Subactive Mode Clock Select 1 and 0
0	SA0	0	R/W	These bits select the operating clock frequency in subactive and subsleep modes. The operating clock frequency changes to the set frequency after the SLEEP instruction is executed. 00: $\phi_w/8$ 01: $\phi_w/4$ 1X: $\phi_w/2$

Legend: Don't care.

6.1.3 Module Standby Control Register 1 (MSTCR1)

MSTCR1 allows the on-chip peripheral modules to enter a standby state in module units.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved This bit is always read as 0.
6	MSTIIC	0	R/W	IIC2 Module Standby IIC2 enters standby mode when this bit is set to 1
5	MSTS3	0	R/W	SCI3 Module Standby SCI3 enters standby mode when this bit is set to 1
4	MSTAD	0	R/W	A/D Converter Module Standby A/D converter enters standby mode when this bit is set to 1
3	MSTWD	0	R/W	Watchdog Timer Module Standby Watchdog timer enters standby mode when this bit is set to 1.When the internal oscillator is selected for the watchdog timer clock, the watchdog timer operates regardless of the setting of this bit
2	—	0	—	Reserved This bit is always read as 0.
1	MSTTV	0	R/W	Timer V Module Standby Timer V enters standby mode when this bit is set to 1
0	MSTTA	0	R/W	RTC Module Standby RTC enters standby mode when this bit is set to 1

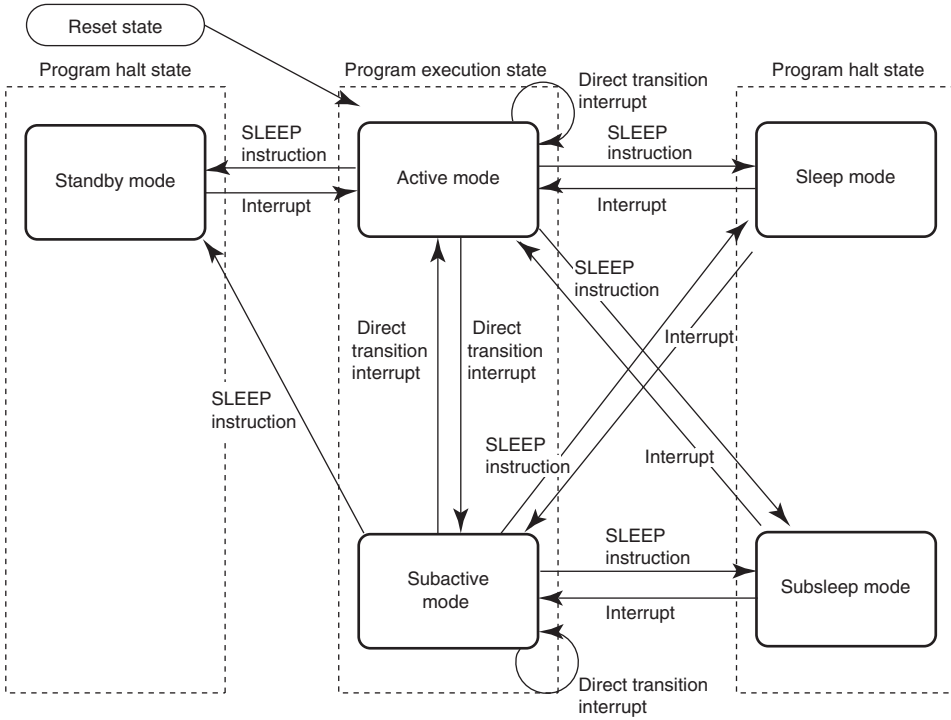
6.1.4 Module Standby Control Register 2 (MSTCR2)

MSTCR2 allows the on-chip peripheral modules to enter a standby state in module units.

Bit	Bit Name	Initial Value	R/W	Description
7	MSTS3_2	0	R/W	SCI3_2 Module Standby SCI3_2 enters standby mode when this bit is set to1
6, 5	—	All 0	—	Reserved These bits are always read as 0.
4	MSTTB1	0	R/W	Timer B1 Module Standby Timer B1 enters standby mode when this bit is set to1
3, 2	—	All 0	—	Reserved These bits are always read as 0.
1	MSTTZ	0	R/W	Timer Z Module Standby Timer Z enters standby mode when this bit is set to1
0	MSTPWM	0	R/W	PWM Module Standby PWM enters standby mode when this bit is set to1

6.2 Mode Transitions and States of LSI

Figure 6.1 shows the possible transitions among these operating modes. A transition is made from the program execution state to the program halt state by executing a SLEEP instruction. Interrupts allow for returning from the program halt state to the program execution state. A direct transition between active mode and subactive mode, which are both program execution states, can be made without halting the program. The operating frequency can also be changed in the same modes by making a transition directly from active mode to active mode, and from subactive mode to subactive mode. $\overline{RES}$ input enables transitions from a mode to the reset state. Table 6.2 shows the transition conditions of each mode after the SLEEP instruction is executed and a mode to return by an interrupt. Table 6.3 shows the internal states of the LSI in each mode.



Notes: 1. To make a transition to another mode by an interrupt, make sure interrupt handling is after the interrupt is accepted.
2. Details on the mode transition conditions are given in table 6.2.

Figure 6.1 Mode Transition Diagram

Table 6.2 Transition Mode after SLEEP Instruction Execution and Transition Mode due to Interrupt

DTON	SSBY	SMSEL	LSON	Transition Mode after SLEEP Instruction Execution	Transition Mode due to Interrupt
0	0	0	0	Sleep mode	Active mode
			1		Subactive mode
		1	0	Subsleep mode	Active mode
			1		Subactive mode
	1	X	X	Standby mode	Active mode
1	X	0*	0	Active mode (direct transition)	—
	X	X	1	Subactive mode (direct transition)	—

Legend: X: Don't care.

* When a state transition is performed while SMSEL is 1, timer V, SCI3, SCI3_2 and the A/D converter are reset, and all registers are set to their initial values. To use these functions after entering active mode, reset the registers.

Table 6.3 Internal State in Each Operating Mode

Function		Active Mode	Sleep Mode	Subactive Mode	Subsleep Mode	Standby Mode
System clock oscillator		Functioning	Functioning	Halted	Halted	Halted
Subclock oscillator		Functioning	Functioning	Functioning	Functioning	Functioning
CPU operations	Instructions	Functioning	Halted	Functioning	Halted	Halted
	Registers	Functioning	Retained	Functioning	Retained	Retained
RAM		Functioning	Retained	Functioning	Retained	Retained
IO ports		Functioning	Retained	Functioning	Retained	Register contents are retained, but output is the high-impedance state.
External interrupts	IRQ3 to IRQ0	Functioning	Functioning	Functioning	Functioning	Functioning
	WKP5 to WKP0	Functioning	Functioning	Functioning	Functioning	Functioning
Peripheral functions	RTC	Functioning	Functioning	Functioning if the timekeeping time-base function is selected, and retained if not selected		
	Timer V	Functioning	Functioning	Reset	Reset	Reset
	Watchdog timer	Functioning	Functioning	Retained (functioning if the internal oscillator is selected as a count clock*)		
	SCI3, SCI3_2	Functioning	Functioning	Reset	Reset	Reset
	IIC2	Functioning	Functioning	Retained*	Retained	Retained
	Timer B1	Functioning	Functioning	Retained*	Retained	Retained
	Timer Z	Functioning	Functioning	Retained (the counter increments according to subclocks if the internal clock (ϕ) is selected as a count clock*)		
	A/D converter	Functioning	Functioning	Reset	Reset	Reset

Note: * Registers can be read or written in subactive mode.

6.2.1 Sleep Mode

In sleep mode, CPU operation is halted but the on-chip peripheral modules function at the clock frequency set by the MA2, MA1, and MA0 bits in SYSCR2. CPU register contents are retained. When an interrupt is requested, sleep mode is cleared and interrupt exception handling starts. Sleep mode is not cleared if the I bit of the condition code register (CCR) is set to 1 or the requested interrupt is disabled in the interrupt enable register. After sleep mode is cleared, a transition is made to active mode when the LSON bit in SYSCR2 is 0, and a transition is made to subactive mode when the bit is 1.

When the $\overline{\text{RES}}$ pin goes low, the CPU goes into the reset state and sleep mode is cleared.

6.2.2 Standby Mode

In standby mode, the clock pulse generator stops, so the CPU and on-chip peripheral modules stop functioning. However, as long as the rated voltage is supplied, the contents of CPU registers, on-chip RAM, and some on-chip peripheral module registers are retained. On-chip RAM contents will be retained as long as the voltage set by the RAM data retention voltage is provided. The I/O ports go to the high-impedance state.

Standby mode is cleared by an interrupt. When an interrupt is requested, the system clock pulse generator starts. After the time set in bits STS2 to STS0 in SYSCR1 has elapsed, and interrupt exception handling starts. Standby mode is not cleared if the I bit of CCR is set to 1 or the requested interrupt is disabled in the interrupt enable register.

When the $\overline{\text{RES}}$ pin goes low, the system clock pulse generator starts. Since system clock signals are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output stabilizes. After the pulse generator output has stabilized, the CPU starts reset exception handling if the $\overline{\text{RES}}$ pin is driven high.

6.2.3 Subsleep Mode

In subsleep mode, operation of the CPU and on-chip peripheral modules other than RTC is halted. As long as a required voltage is applied, the contents of CPU registers, the on-chip RAM, and some registers of the on-chip peripheral modules are retained. I/O ports keep the same states as before the transition.

Subsleep mode is cleared by an interrupt. When an interrupt is requested, subsleep mode is cleared and interrupt exception handling starts. Subsleep mode is not cleared if the I bit of CCR is set to 1 or the requested interrupt is disabled in the interrupt enable register. After subsleep mode is

cleared, a transition is made to active mode when the LSON bit in SYSCR2 is 0, and a transition is made to subactive mode when the bit is 1. After the time set in bits STS2 to STS0 in SYSCR1 has elapsed, a transition is made to active mode.

When the $\overline{\text{RES}}$ pin goes low, the system clock pulse generator starts. Since system clock signals are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output stabilizes. After the pulse generator output has stabilized, the CPU starts reset exception handling if the $\overline{\text{RES}}$ pin is driven high.

6.2.4 Subactive Mode

The operating frequency of subactive mode is selected from $\phi_w/2$, $\phi_w/4$, and $\phi_w/8$ by the SA1 and SA0 bits in SYSCR2. After the SLEEP instruction is executed, the operating frequency changes to the frequency which is set before the execution. When the SLEEP instruction is executed in subactive mode, a transition to sleep mode, subsleep mode, standby mode, active mode, or subactive mode is made, depending on the combination of SYSCR1 and SYSCR2. When the $\overline{\text{RES}}$ pin goes low, the system clock pulse generator starts. Since system clock signals are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{\text{RES}}$ pin must be kept low until the pulse generator output stabilizes. After the pulse generator output has stabilized, the CPU starts reset exception handling if the $\overline{\text{RES}}$ pin is driven high.

6.3 Operating Frequency in Active Mode

Operation in active mode is clocked at the frequency designated by the MA2, MA1, and MA0 bits in SYSCR2. The operating frequency changes to the set frequency after SLEEP instruction execution.

6.4 Direct Transition

The CPU can execute programs in two modes: active and subactive modes. A direct transition is a transition between these two modes without stopping program execution. A direct transition can be made by executing a SLEEP instruction while the DTON bit in SYSCR2 is set to 1. The direct transition also enables operating frequency modification in active or subactive mode. After the mode transition, direct transition interrupt exception handling starts.

If the direct transition interrupt is disabled in interrupt enable register 1, a transition is made instead to sleep or subsleep mode. Note that if a direct transition is attempted while the I bit in CCR is set to 1, sleep or subsleep mode will be entered, and the resulting mode cannot be cleared by means of an interrupt.

6.4.1 Direct Transition from Active Mode to Subactive Mode

The time from the start of SLEEP instruction execution to the end of interrupt exception handling (the direct transition time) is calculated by equation (1).

Direct transition time = {(number of SLEEP instruction execution states) + (number of internal processing states)}× (tcyc before transition) + (number of interrupt exception handling states) × (tsubcyc after transition) (1)

Example
Direct transition time = (2 + 1) × tosc + 14 × 8tw = 3tosc + 112tw
(when the CPU operating clock of $\phi_{osc} \rightarrow \phi_w/8$ is selected)

Legend
tosc: OSC clock cycle time
tw: Watch clock cycle time
tcyc: System clock (ϕ) cycle time
tsubcyc: Subclock (ϕ_{SUB}) cycle time

6.4.2 Direct Transition from Subactive Mode to Active Mode

The time from the start of SLEEP instruction execution to the end of interrupt exception handling (the direct transition time) is calculated by equation (2).

Direct transition time = {(number of SLEEP instruction execution states) + (number of internal processing states)} × (tsubcyc before transition) + {(waiting time set in bits STS2 to STS0) + (number of interrupt exception handling states)} × (tcyc after transition) (2)

Example
Direct transition time = (2 + 1) × 8tw + (8192 + 14) × tosc = 24tw + 8206tosc
(when the CPU operating clock of $\phi_w/8 \rightarrow \phi_{osc}$ and a waiting time of 8192 states are selected)

Legend
tosc: OSC clock cycle time
tw: Watch clock cycle time
tcyc: System clock (ϕ) cycle time
tsubcyc: Subclock (ϕ_{SUB}) cycle time

6.5 Module Standby Function

The module-standby function can be set to any peripheral module. In module standby mode, the clock supply to modules stops to enter the power-down mode. Module standby mode enables each on-chip peripheral module to enter the standby state by setting a bit that corresponds to each module to 1 and cancels the mode by clearing the bit to 0.

Section 7 ROM

The features of the 56-kbyte or 32-kbyte flash memories built into the flash memory (F-ZTAT) version are summarized below.

- 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: 1 kbyte × 4 blocks, 28 kbytes × 1 block, 16 kbytes × 1 block, and 8 kbytes × 1 block for H8/3687F and 1 kbyte × 4 blocks and 28 kbytes × 1 block for H8/3684F. To erase the entire flash memory, each block must be erased in turn.
- Reprogramming capability
  - The flash memory can be reprogrammed up to 1,000 times.
- On-board programming
  - 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.
- Power-down mode
  - Operation of the power supply circuit can be partly halted in subactive mode. As a result, flash memory can be read with low power consumption.

7.1 Block Configuration

Figure 7.1 shows the block configuration of flash memory. The thick lines indicate erasing units, the narrow lines indicate programming units, and the values are addresses. The 56-kbyte flash memory is divided into 1 kbyte × 4 blocks, 28 kbytes × 1 block, 16 kbytes × 1 block, and 8 kbytes × 1 block. The 32-kbyte flash memory is divided into 1 kbyte × 4 blocks and 28 kbytes × 1 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.

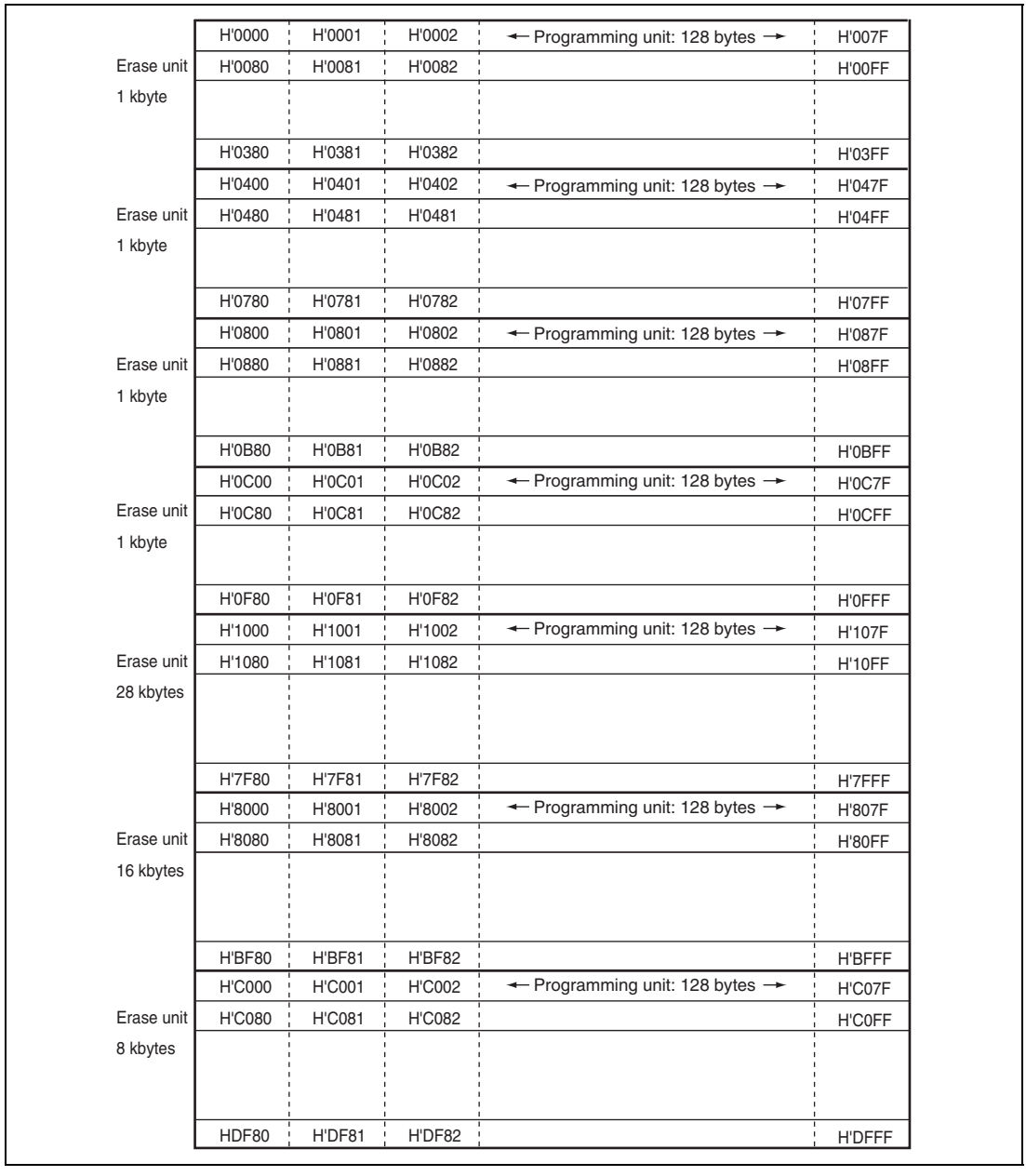


Figure 7.1 Flash Memory Block Configuration

7.2 Register Descriptions

The flash memory has the following registers.

- Flash memory control register 1 (FLMCR1)
- Flash memory control register 2 (FLMCR2)
- Erase block register 1 (EBR1)
- Flash memory power control register (FLPWCR)
- Flash memory enable register (FENR)

7.2.1 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 7.4, Flash Memory Programming/Erasing.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved This bit is always read as 0.
6	SWE	0	R/W	Software Write Enable 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 bits cannot be set.
5	ESU	0	R/W	Erase Setup When this bit is set to 1, the flash memory changes to the erase setup state. When it is cleared to 0, the erase setup state is cancelled. Set this bit to 1 before setting the E bit to 1 in FLMCR1.
4	PSU	0	R/W	Program Setup 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 P bit in FLMCR1.
3	EV	0	R/W	Erase-Verify When this bit is set to 1, the flash memory changes to erase-verify mode. When it is cleared to 0, erase-verify mode is cancelled.

Bit	Bit Name	Initial Value	R/W	Description
2	PV	0	R/W	Program-Verify 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	E	0	R/W	Erase When this bit is set to 1 while SWE=1 and ESU=1, the flash memory changes to erase mode. When it is cleared to 0, erase mode is cancelled.
0	P	0	R/W	Program When this bit is set to 1 while SWE=1 and PSU=1, the flash memory changes to program mode. When it is cleared to 0, program mode is cancelled.

7.2.2 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	Flash Memory Error 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. See section 7.5.3, Error Protection, for details.
6 to 0	—	All 0	—	Reserved These bits are always read as 0.

7.2.3 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	—	0	—	Reserved This bit is always read as 0.
6	EB6	0	R/W	When this bit is set to 1, 8 bytes of H'C000 to H'DFFF will be erased.
5	EB5	0	R/W	When this bit is set to 1, 16 bytes of H'8000 to H'BFFF will be erased.
4	EB4	0	R/W	When this bit is set to 1, 28 kbytes of H'1000 to H'7FFF will be erased.
3	EB3	0	R/W	When this bit is set to 1, 1 kbyte of H'0C00 to H'0FFF will be erased.
2	EB2	0	R/W	When this bit is set to 1, 1 kbyte of H'0800 to H'0BFF will be erased.
1	EB1	0	R/W	When this bit is set to 1, 1 kbyte of H'0400 to H'07FF will be erased.
0	EB0	0	R/W	When this bit is set to 1, 1 kbyte of H'0000 to H'03FF will be erased.

7.2.4 Flash Memory Power Control Register (FLPWCR)

FLPWCR enables or disables a transition to the flash memory power-down mode when the LSI switches to subactive mode. There are two modes: mode in which operation of the power supply circuit of flash memory is partly halted in power-down mode and flash memory can be read, and mode in which even if a transition is made to subactive mode, operation of the power supply circuit of flash memory is retained and flash memory can be read.

Bit	Bit Name	Initial Value	R/W	Description
7	PDWND	0	R/W	Power-Down Disable When this bit is 0 and a transition is made to subactive mode, the flash memory enters the power-down mode. When this bit is 1, the flash memory remains in the normal mode even after a transition is made to subactive mode.
6 to 0	—	All 0	—	Reserved These bits are always read as 0.

7.2.5 Flash Memory Enable Register (FENR)

Bit 7 (FLSHE) in FENR enables or disables the CPU access to the flash memory control registers, FLMCR1, FLMCR2, EBR1, and FLPWCR.

Bit	Bit Name	Initial Value	R/W	Description
7	FLSHE	0	R/W	Flash Memory Control Register Enable Flash memory control registers can be accessed when this bit is set to 1. Flash memory control registers cannot be accessed when this bit is set to 0.
6 to 0	—	All 0	—	Reserved These bits are always read as 0.

7.3 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 TEST pin settings, $\overline{\text{NMI}}$ pin settings, and input level of each port, as shown in table 7.1. 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 SCI3. 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 7.1 Setting Programming Modes

TEST	$\overline{\text{NMI}}$	P85	PB0	PB1	PB2	LSI State after Reset End
0	1	X	X	X	X	User Mode
0	0	1	X	X	X	Boot Mode
1	X	X	0	0	0	Programmer Mode

Legend: X : Don't care.

7.3.1 Boot Mode

Table 7.2 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 7.4, Flash Memory Programming/Erasing.
2. SCI3 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 SCI3 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 7.3.
5. In boot mode, a part of the on-chip RAM area is used by the boot program. The area H'F780 to H'FEEF 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 SCI3 (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 program data or verify data with the host. The TxD pin is high (PCR22 = 1, P22 = 1). 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 $\overline{\text{NMI}}$ pin. Boot mode is also cleared when a WDT overflow occurs.
8. Do not change the TEST pin and NMI pin input levels in boot mode.

Table 7.2 Boot Mode Operation

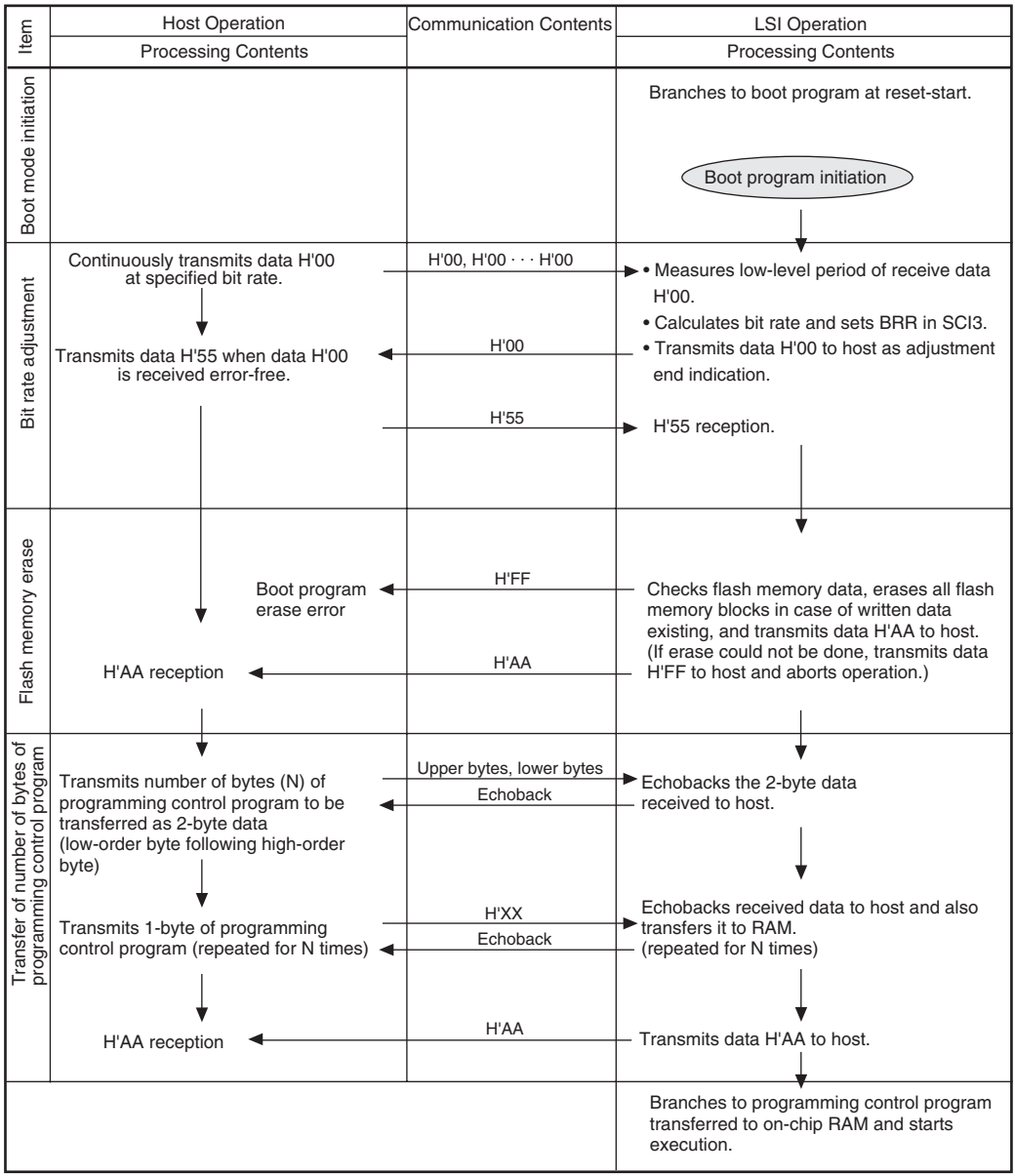


Table 7.3 System Clock Frequencies for which Automatic Adjustment of LSI Bit Rate is Possible

Host Bit Rate	System Clock Frequency Range of LSI
19,200 bps	16 to 20 MHz
9,600 bps	8 to 16 MHz
4,800 bps	4 to 16 MHz
2,400 bps	2 to 16 MHz

7.3.2 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 7.2 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 7.4, Flash Memory Programming/Erasing.

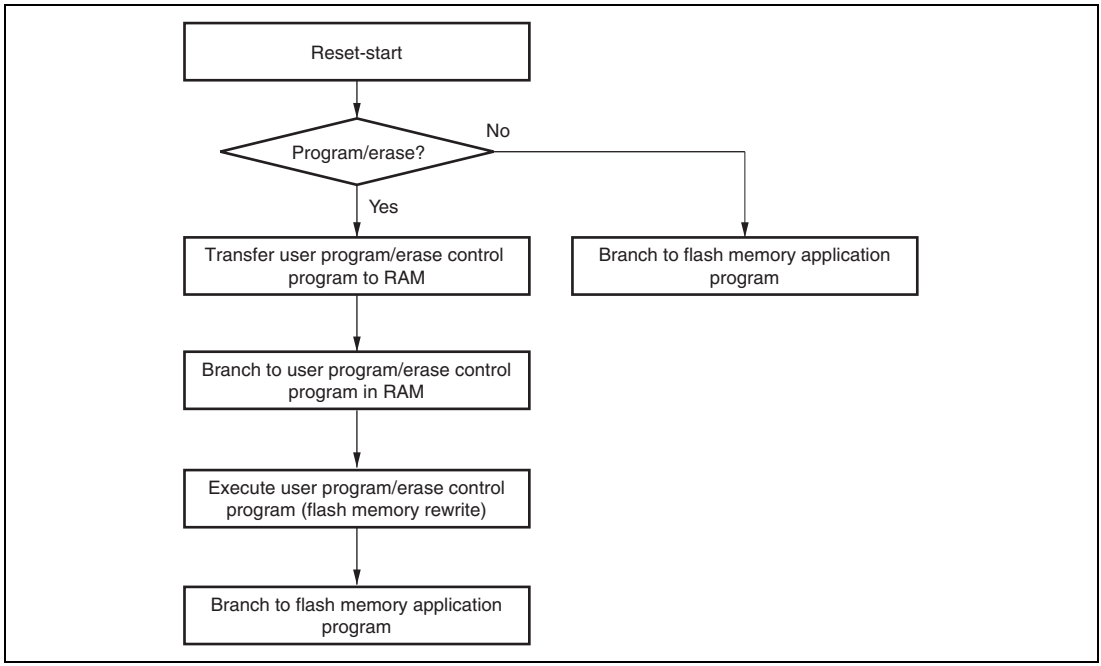


Figure 7.2 Programming/Erasing Flowchart Example in User Program Mode

7.4 Flash Memory Programming/Erasing

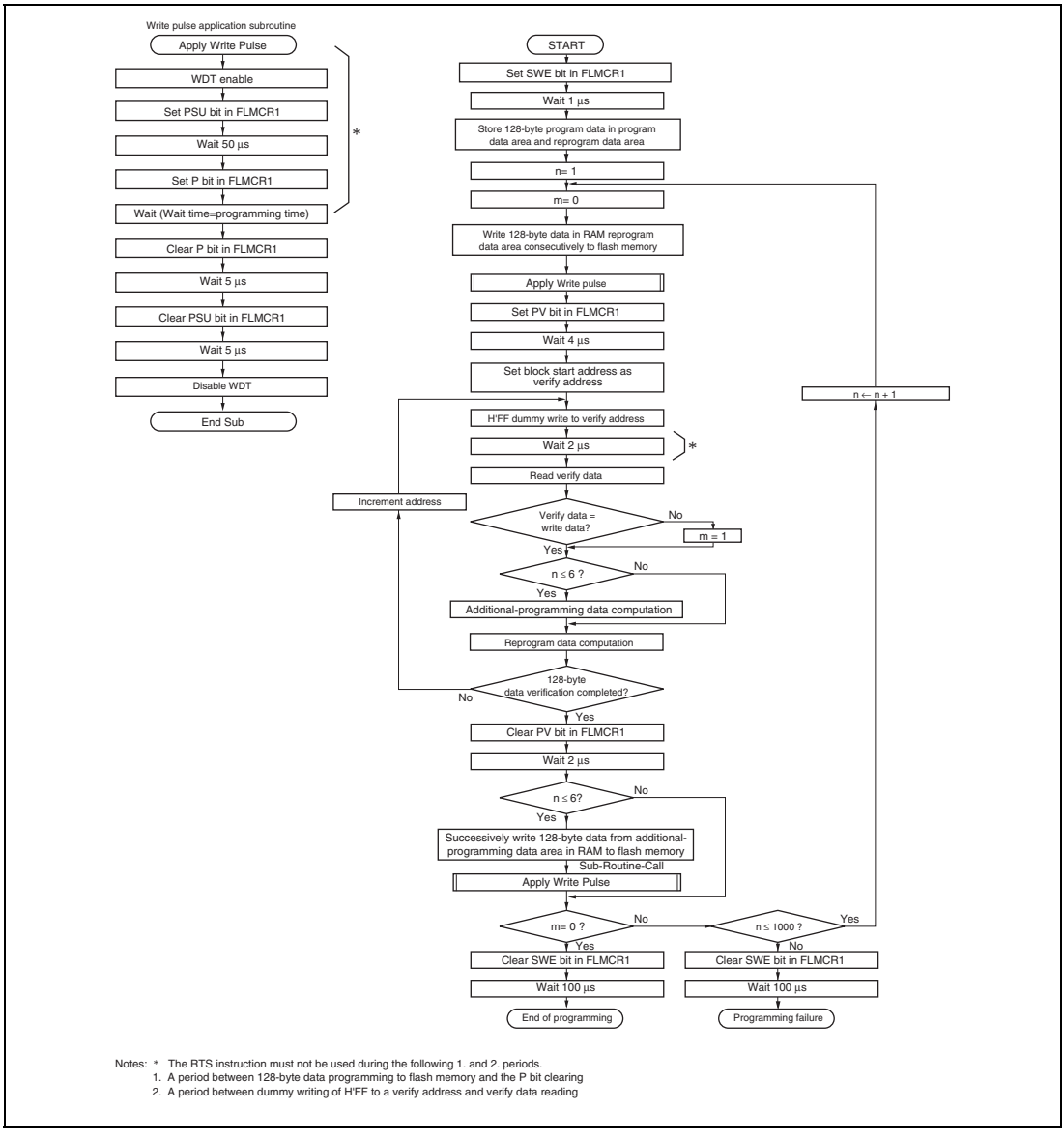
A software method using the CPU is employed to program and erase flash memory in the on-board programming modes. Depending on the FLMCR1 setting, the flash memory operates in one of the following four modes: Program mode, program-verify mode, erase mode, and erase-verify mode. The programming control 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 7.4.1, Program/Program-Verify and section 7.4.2, Erase/Erase-Verify, respectively.

7.4.1 Program/Program-Verify

When writing data or programs to the flash memory, the program/program-verify flowchart shown in figure 7.3 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 according to table 7.4, and additional programming data computation according to table 7.5.
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. Table 7.6 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 words or 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: * The RTS instruction must not be used during the following 1. and 2. periods.
1. A period between 128-byte data programming to flash memory and the P bit clearing
2. A period between dummy writing of H'FF to a verify address and verify data reading

Figure 7.3 Program/Program-Verify Flowchart

Table 7.4 Reprogram Data Computation Table

Program Data	Verify Data	Reprogram Data	Comments
0	0	1	Programming completed
0	1	0	Reprogram bit
1	0	1	—
1	1	1	Remains in erased state

Table 7.5 Additional-Program Data Computation Table

Reprogram Data	Verify Data	Additional-Program Data	Comments
0	0	0	Additional-program bit
0	1	1	No additional programming
1	0	1	No additional programming
1	1	1	No additional programming

Table 7.6 Programming Time

n (Number of Writes)	Programming Time	In Additional Programming	Comments
1 to 6	30	10	
7 to 1,000	200	—	

Note: Time shown in µs.

7.4.2 Erase/Erase-Verify

When erasing flash memory, the erase/erase-verify flowchart shown in figure 7.4 should be followed.

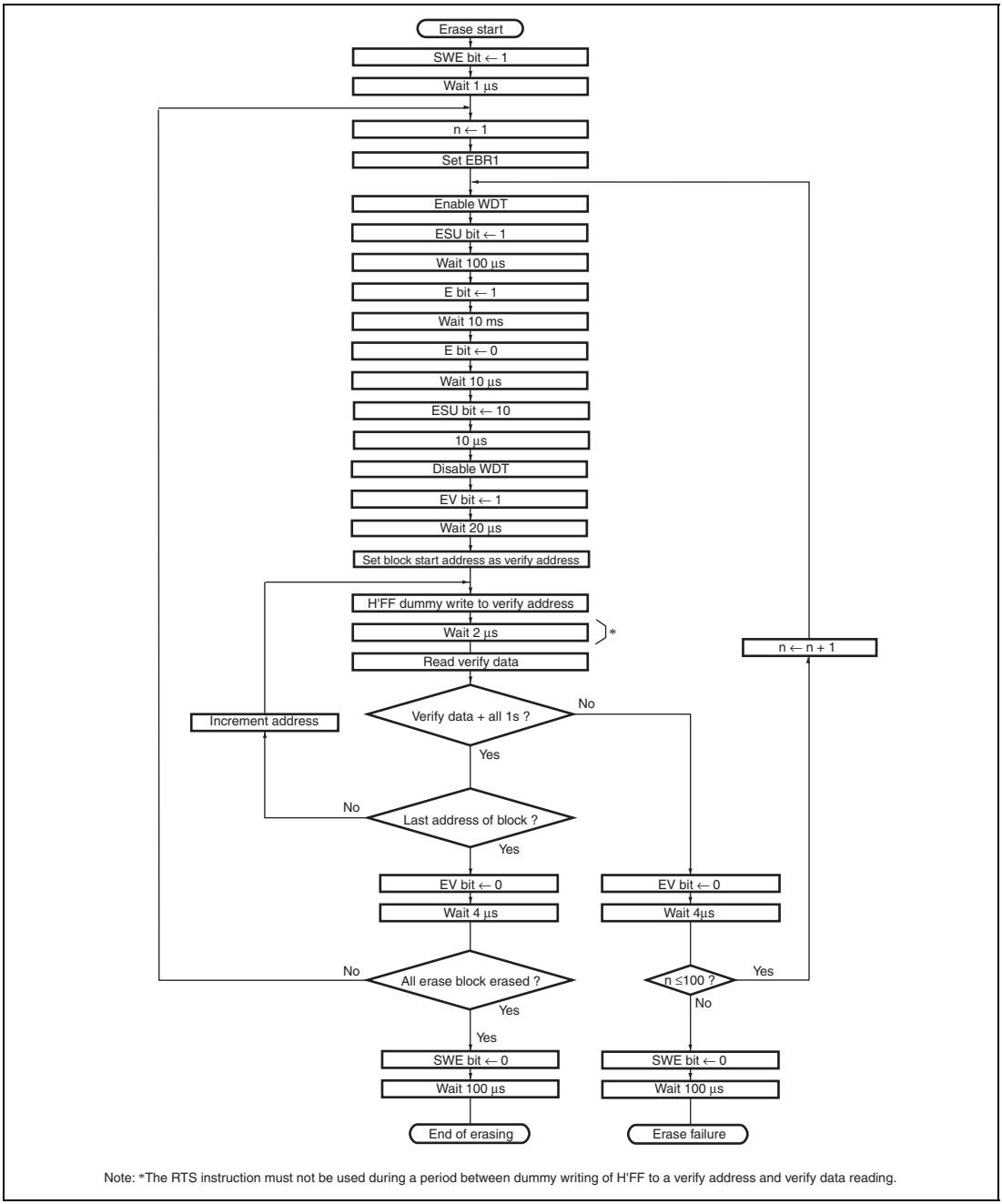
1. Prewriting (setting erase block data to all 0s) is not necessary.
2. Erasing is performed in block units. Make only a single-bit specification in the erase block register (EBR1). To erase multiple blocks, each block must be erased in turn.
3. The time during which the E bit is set to 1 is the flash 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.

7.4.3 Interrupt Handling when Programming/Erasing Flash Memory

All interrupts, including the 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.



Note: *The RTS instruction must not be used during a period between dummy writing of H'FF to a verify address and verify data reading.

Figure 7.4 Erase/Erase-Verify Flowchart

7.5 Program/Erase Protection

There are three kinds of flash memory program/erase protection; hardware protection, software protection, and error protection.

7.5.1 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, subactive mode, subsleep mode, 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.

7.5.2 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 P or E 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.

7.5.3 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 forcibly 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 P or E bit. However, PV and EV bit settings are retained, and a transition can be made to verify mode. Error protection can be cleared only by a reset.

7.6 Programmer Mode

In programmer mode, a PROM programmer can be used to perform programming/erasing via a socket adapter, just as a discrete flash memory. Use a PROM programmer that supports the MCU device type with the on-chip 64-kbyte flash memory (FZTAT64V5).

7.7 Power-Down States for Flash Memory

In user mode, the flash memory will operate in either of the following states:

- Normal operating mode
  The flash memory can be read and written to at high speed.
- Power-down operating mode
  The power supply circuit of flash memory can be partly halted. As a result, flash memory can be read with low power consumption.
- Standby mode
  All flash memory circuits are halted.

Table 7.7 shows the correspondence between the operating modes of this LSI and the flash memory. In subactive mode, the flash memory can be set to operate in power-down mode with the PDWND bit in FLPWCR. When the flash memory returns to its normal operating state from power-down mode or standby mode, a period to stabilize operation of the power supply circuits that were stopped is needed. When the flash memory returns to its normal operating state, bits STS2 to STS0 in SYSCR1 must be set to provide a wait time of at least 20 µs, even when the external clock is being used.

Table 7.7 Flash Memory Operating States

	Flash Memory Operating State	
LSI Operating State	**PDWND = 0 (Initial Value)**	**PDWND = 1**
Active mode	Normal operating mode	Normal operating mode
Subactive mode	Power-down mode	Normal operating mode
Sleep mode	Normal operating mode	Normal operating mode
Subsleep mode	Standby mode	Standby mode
Standby mode	Standby mode	Standby mode

Section 8 RAM

This LSI has an on-chip high-speed static RAM. The RAM is connected to the CPU by a 16-bit data bus, enabling two-state access by the CPU to both byte data and word data.

Product Classification			RAM Size	RAM Address
Flash memory version		H8/3687F	4 kbytes	H'E800 to H'EFFF, H'F780 to H'FF7F*
(F-ZTAT™ version)		H8/3684F	4 kbytes	H'E800 to H'EFFF, H'F780 to H'FF7F*
Mask-ROM version		H8/3687	3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F
		H8/3686	3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F
		H8/3685	3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F
		H8/3684	3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F
		H8/3683	3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F
		H8/3682	3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F
EEPROM stacked version	Flash memory version	H8/3687N	4 kbytes	H'E800 to H'EFFF, H'F780 to H'FF7F*
	Mask-ROM version		3 kbytes	H'E800 to H'EFFF, H'FB80 to H'FF7F

Note: * When the E7 or E8 is used, area H'F780 to H'FB7F must not be accessed.

Section 9 I/O Ports

The group of this LSI has forty-five general I/O ports (forty-three general I/O ports in the H8/3687N) and eight general input-only ports. Port 6 is a large current port, which can drive 20 mA ($@V_{OL}$ = 1.5 V) when a low level signal is output. Any of these ports can become an input port immediately after a reset. They can also be used as I/O pins of the on-chip peripheral modules or external interrupt input pins, and these functions can be switched depending on the register settings. The registers for selecting these functions can be divided into two types: those included in I/O ports and those included in each on-chip peripheral module. General I/O ports are comprised of the port control register for controlling inputs/outputs and the port data register for storing output data and can select inputs/outputs in bit units.

For functions in each port, see appendix B.1, I/O Port Block Diagrams. For the execution of bit-manipulation instructions to the port control register and port data register, see section 2.8.3, Bit Manipulation Instruction.

9.1 Port 1

Port 1 is a general I/O port also functioning as IRQ interrupt input pins, an RTC output pin, a 14-bit PWM output pin, a timer B1 input pin, and a timer V input pin. Figure 9.1 shows its pin configuration.

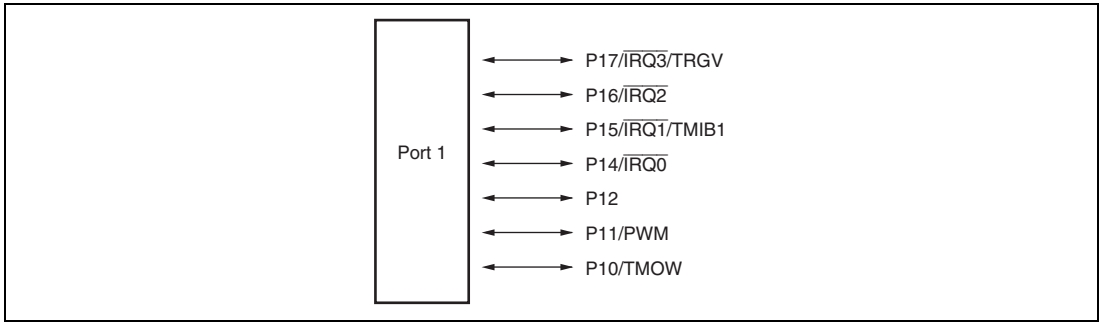


Figure 9.1 Port 1 Pin Configuration

Port 1 has the following registers.

- Port mode register 1 (PMR1)
- Port control register 1 (PCR1)
- Port data register 1 (PDR1)
- Port pull-up control register 1 (PUCR1)

9.1.1 Port Mode Register 1 (PMR1)

PMR1 switches the functions of pins in port 1 and port 2.

Bit	Bit Name	Initial Value	R/W	Description
7	IRQ3	0	R/W	This bit selects the function of pin P17/$\overline{\text{IRQ3}}$/TRGV. 0: General I/O port 1: $\overline{\text{IRQ3}}$/TRGV input pin
6	IRQ2	0	R/W	This bit selects the function of pin P16/$\overline{\text{IRQ2}}$. 0: General I/O port 1: $\overline{\text{IRQ2}}$ input pin
5	IRQ1	0	R/W	This bit selects the function of pin P15/$\overline{\text{IRQ1}}$/TMIB1. 0: General I/O port 1: $\overline{\text{IRQ1}}$/TMIB1 input pin
4	IRQ0	0	R/W	This bit selects the function of pin P14/$\overline{\text{IRQ0}}$. 0: General I/O port 1: $\overline{\text{IRQ0}}$ input pin
3	TXD2	0	R/W	This bit selects the function of pin P72/TXD_2. 0: General I/O port 1: TXD_2 output pin
2	PWM	0	R/W	This bit selects the function of pin P11/PWM. 0: General I/O port 1: PWM output pin
1	TXD	0	R/W	This bit selects the function of pin P22/TXD. 0: General I/O port 1: TXD output pin
0	TMOW	0	R/W	This bit selects the function of pin P10/TMOW. 0: General I/O port 1: TMOW output pin

9.1.2 Port Control Register 1 (PCR1)

PCR1 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 1.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR17	0	W	When the corresponding pin is designated in PMR1 as a general I/O pin, setting a PCR1 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port.
6	PCR16	0	W	
5	PCR15	0	W	
4	PCR14	0	W	Bit 3 is a reserved bit.
3	—	—	—	
2	PCR12	0	W	
1	PCR11	0	W	
0	PCR10	0	W	

9.1.3 Port Data Register 1 (PDR1)

PDR1 is a general I/O port data register of port 1.

Bit	Bit Name	Initial Value	R/W	Description
7	P17	0	R/W	PDR1 stores output data for port 1 pins.
6	P16	0	R/W	If PDR1 is read while PCR1 bits are set to 1, the value stored in PDR1 are read. If PDR1 is read while PCR1 bits are cleared to 0, the pin states are read regardless of the value stored in PDR1.
5	P15	0	R/W	
4	P14	0	R/W	
3	—	1	—	Bit 3 is a reserved bit. This bit is always read as 1.
2	P12	0	R/W	
1	P11	0	R/W	
0	P10	0	R/W	

9.1.4 Port Pull-Up Control Register 1 (PUCR1)

PUCR1 controls the pull-up MOS in bit units of the pins set as the input ports.

Bit	Bit Name	Initial Value	R/W	Description
7	PUCR17	0	R/W	Only bits for which PCR1 is cleared are valid. The pull-up MOS of P17 to P14 and P12 to P10 pins enter the on-state when these bits are set to 1, while they enter the off-state when these bits are cleared to 0. Bit 3 is a reserved bit. This bit is always read as 1.
6	PUCR16	0	R/W	
5	PUCR15	0	R/W	
4	PUCR14	0	R/W	
3	—	1	—	
2	PUCR12	0	R/W	
1	PUCR11	0	R/W	
0	PUCR10	0	R/W	

9.1.5 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P17/$\overline{\text{IRQ3}}$/TRGV pin

Register	PMR1	PCR1	
Bit Name	**IRQ3**	**PCR17**	**Pin Function**
Setting value	0	0	P17 input pin
		1	P17 output pin
	1	X	$\overline{\text{IRQ3}}$ input/TRGV input pin

Legend: X: Don't care.

P16/$\overline{\text{IRQ2}}$ pin

Register	PMR1	PCR1	
Bit Name	**IRQ2**	**PCR16**	**Pin Function**
Setting value	0	0	P16 input pin
		1	P16 output pin
	1	X	$\overline{\text{IRQ2}}$ input pin

Legend: X: Don't care.

P15/$\overline{\text{IRQ1}}$/TMIB1 pin

Register	PMR1	PCR1	
Bit Name	**IRQ1**	**PCR15**	**Pin Function**
Setting value	0	0	P15 input pin
		1	P15 output pin
	1	X	$\overline{\text{IRQ1}}$ input/TMIB1 input pin

Legend: X: Don't care.

P14/$\overline{\text{IRQ0}}$ pin

Register	PMR1	PCR1	
Bit Name	**IRQ0**	**PCR14**	**Pin Function**
Setting value	0	0	P14 input pin
		1	P14 output pin
	1	X	$\overline{\text{IRQ0}}$ input pin

Legend: X: Don't care.

P12 pin

Register	PCR1	
Bit Name	**PCR12**	**Pin Function**
Setting value	0	P12 input pin
	1	P12 output pin

P11/PWM pin

Register	PMR1	PCR1	
Bit Name	**PWM**	**PCR11**	**Pin Function**
Setting value	0	0	P11 input pin
		1	P11 output pin
	1	X	PWM output pin

Legend: X: Don't care.

P10/TMOW pin

Register	PMR1	PCR1	
Bit Name	**TMOW**	**PCR10**	**Pin Function**
Setting value	0	0	P10 input pin
		1	P10 output pin
	1	X	TMOW output pin

Legend: X: Don't care.

9.2 Port 2

Port 2 is a general I/O port also functioning as SCI3 I/O pins. Each pin of the port 2 is shown in figure 9.2. The register settings of PMR1and SCI3 have priority for functions of the pins for both uses.

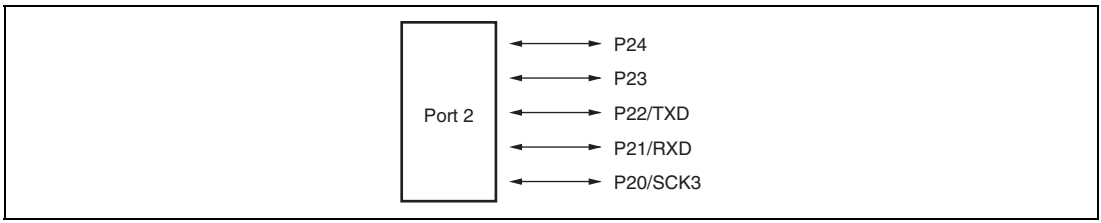


Figure 9.2 Port 2 Pin Configuration

Port 2 has the following registers.

- Port control register 2 (PCR2)
- Port data register 2 (PDR2)
- Port mode register 3 (PMR3)

9.2.1 Port Control Register 2 (PCR2)

PCR2 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 2.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	—	—	Reserved
4	PCR24	0	W	When each of the port 2 pins P24 to P20 functions as a general I/O port, setting a PCR2 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port.
3	PCR23	0	W	
2	PCR22	0	W	
1	PCR21	0	W	
0	PCR20	0	W	

9.2.2 Port Data Register 2 (PDR2)

PDR2 is a general I/O port data register of port 2.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	—	Reserved These bits are always read as 1.
4	P24	0	R/W	PDR2 stores output data for port 2 pins.
3	P23	0	R/W	If PDR2 is read while PCR2 bits are set to 1, the value stored in PDR2 is read. If PDR2 is read while PCR2 bits are cleared to 0, the pin states are read regardless of the value stored in PDR2.
2	P22	0	R/W	
1	P21	0	R/W	
0	P20	0	R/W	

9.2.3 Port Mode Register 3 (PMR3)

PMR3 selects the CMOS output or NMOS open-drain output for port 2.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	—	Reserved These bits are always read as 0.
4	POF24	0	R/W	When the bit is set to 1, the corresponding pin is cut off by PMOS and it functions as the NMOS open-drain output. When cleared to 0, the pin functions as the CMOS output.
3	POF23	0	R/W	
2 to 0	—	All 1	—	Reserved These bits are always read as 1.

9.2.4 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P24 pin

Register	PCR2	
Bit Name	**PCR24**	**Pin Function**
Setting Value	0	P24 input pin
	1	P24 output pin

P23 pin

Register	PCR2	
Bit Name	**PCR23**	**Pin Function**
Setting Value	0	P23 input pin
	1	P23 output pin

P22/TXD pin

Register	PMR1	PCR2	
Bit Name	**TXD**	**PCR22**	**Pin Function**
Setting Value	0	0	P22 input pin
		1	P22 output pin
	1	X	TXD output pin

Legend: X: Don't care.

P21/RXD pin

Register	SCR3	PCR2	
Bit Name	**RE**	**PCR21**	**Pin Function**
Setting Value	0	0	P21 input pin
		1	P21 output pin
	1	X	RXD input pin

Legend: X: Don't care.

P20/SCK3 pin

Register	SCR3		SMR	PCR2	
Bit Name	**CKE1**	**CKE0**	**COM**	**PCR20**	**Pin Function**
Setting Value	0	0	0	0	P20 input pin
				1	P20 output pin
	0	0	1	X	SCK3 output pin
	0	1	X	X	SCK3 output pin
	1	X	X	X	SCK3 input pin

Legend: X: Don't care.

9.3 Port 3

Port 3 is a general I/O port. Each pin of the port 3 is shown in figure 9.3.

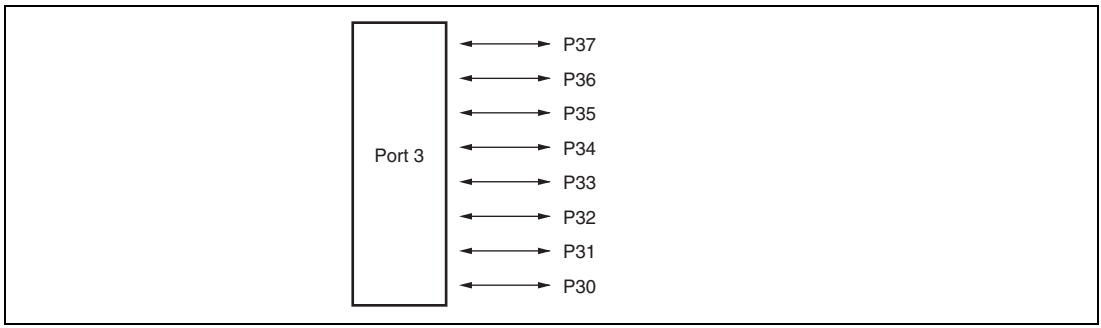


Figure 9.3 Port 3 Pin Configuration

Port 3 has the following registers.

- Port control register 3 (PCR3)
- Port data register 3 (PDR3)

9.3.1 Port Control Register 3 (PCR3)

PCR3 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 3.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR37	0	W	Setting a PCR3 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port.
6	PCR36	0	W	
5	PCR35	0	W	
4	PCR34	0	W	
3	PCR33	0	W	
2	PCR32	0	W	
1	PCR31	0	W	
0	PCR30	0	W	

9.3.2 Port Data Register 3 (PDR3)

PDR3 is a general I/O port data register of port 3.

Bit	Bit Name	Initial Value	R/W	Description
7	P37	0	R/W	PDR3 stores output data for port 3 pins.
6	P36	0	R/W	If PDR3 is read while PCR3 bits are set to 1, the value stored in PDR3 is read. If PDR3 is read while PCR3 bits are cleared to 0, the pin states are read regardless of the value stored in PDR3.
5	P35	0	R/W	
4	P34	0	R/W	
3	P33	0	R/W	
2	P32	0	R/W	
1	P31	0	R/W	
0	P30	0	R/W	

9.3.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P37 pin

Register	PCR3	
Bit Name	PCR37	Pin Function
Setting Value	0	P37 input pin
	1	P37 output pin

P36 pin

Register	PCR3	
Bit Name	PCR36	Pin Function
Setting Value	0	P36 input pin
	1	P36 output pin

P35 pin

Register	PCR3	
Bit Name	**PCR35**	**Pin Function**
Setting Value	0	P35 input pin
	1	P35 output pin

P34 pin

Register	PCR3	
Bit Name	**PCR34**	**Pin Function**
Setting Value	0	P34 input pin
	1	P34 output pin

P33 pin

Register	PCR3	
Bit Name	**PCR33**	**Pin Function**
Setting Value	0	P33 input pin
	1	P33 output pin

P32 pin

Register	PCR3	
Bit Name	**PCR32**	**Pin Function**
Setting Value	0	P32 input pin
	1	P32 output pin

P31 pin

Register	PCR3	
Bit Name	**PCR31**	**Pin Function**
Setting Value	0	P31 input pin
	1	P31 output pin

P30 pin

Register	PCR3	
Bit Name	**PCR30**	**Pin Function**
Setting Value	0	P30 input pin
	1	P30 output pin

9.4 Port 5

Port 5 is a general I/O port also functioning as an I^2C bus interface I/O pin, an A/D trigger input pin, and wakeup interrupt input pin. Each pin of the port 5 is shown in figure 9.4. The register setting of the I^2C bus interface register has priority for functions of the pins P57/SCL and P56/SDA. Since the output buffer for pins P56 and P57 has the NMOS push-pull structure, it differs from an output buffer with the CMOS structure in the high-level output characteristics (see section 23, Electrical Characteristics).

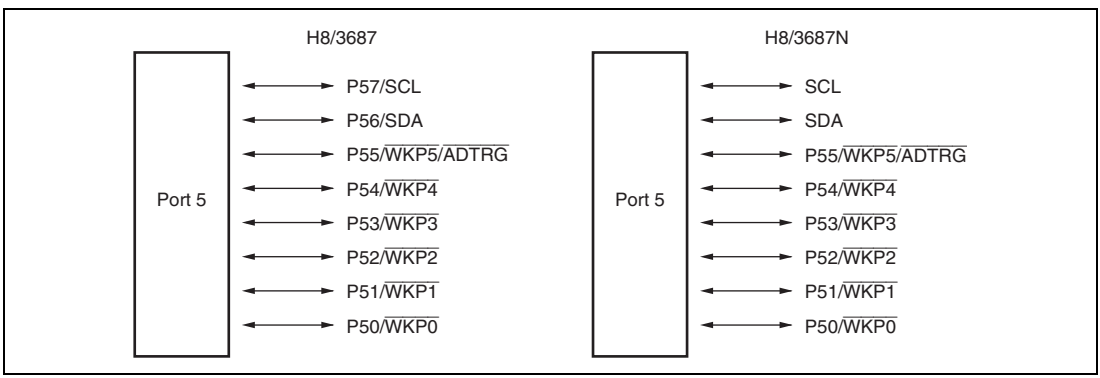


Figure 9.4 Port 5 Pin Configuration

Port 5 has the following registers.

- Port mode register 5 (PMR5)
- Port control register 5 (PCR5)
- Port data register 5 (PDR5)
- Port pull-up control register 5 (PUCR5)

9.4.1 Port Mode Register 5 (PMR5)

PMR5 switches the functions of pins in port 5.

Bit	Bit Name	Initial Value	R/W	Description
7	POF57	0	R/W	When the bit is set to 1, the corresponding pin is cut off by PMOS and it functions as the NMOS open-drain output. When cleared to 0, the pin functions as the CMOS output.
6	POF56	0	R/W	
5	WKP5	0	R/W	This bit selects the function of pin P55/$\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$. 0: General I/O port 1: $\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ input pin
4	WKP4	0	R/W	This bit selects the function of pin P54/$\overline{\text{WKP4}}$. 0: General I/O port 1: $\overline{\text{WKP4}}$ input pin
3	WKP3	0	R/W	This bit selects the function of pin P53/$\overline{\text{WKP3}}$. 0: General I/O port 1: $\overline{\text{WKP3}}$ input pin
2	WKP2	0	R/W	This bit selects the function of pin P52/$\overline{\text{WKP2}}$. 0: General I/O port 1: $\overline{\text{WKP2}}$ input pin
1	WKP1	0	R/W	This bit selects the function of pin P51/$\overline{\text{WKP1}}$. 0: General I/O port 1: $\overline{\text{WKP1}}$ input pin
0	WKP0	0	R/W	This bit selects the function of pin P50/$\overline{\text{WKP0}}$. 0: General I/O port 1: $\overline{\text{WKP0}}$ input pin

9.4.2 Port Control Register 5 (PCR5)

PCR5 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 5.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR57	0	W	When each of the port 5 pins P57 to P50 functions as a general I/O port, setting a PCR5 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port. Note: The PCR57 and PCR56 bits should not be set to 1 in the H8/3687N.
6	PCR56	0	W	
5	PCR55	0	W	
4	PCR54	0	W	
3	PCR53	0	W	
2	PCR52	0	W	
1	PCR51	0	W	
0	PCR50	0	W	

9.4.3 Port Data Register 5 (PDR5)

PDR5 is a general I/O port data register of port 5.

Bit	Bit Name	Initial Value	R/W	Description
7	P57	0	R/W	Stores output data for port 5 pins.
6	P56	0	R/W	If PDR5 is read while PCR5 bits are set to 1, the value stored in PDR5 are read. If PDR5 is read while PCR5 bits are cleared to 0, the pin states are read regardless of the value stored in PDR5. Note: The P57 and P56 bits should not be set to 1 in the H8/3687N.
5	P55	0	R/W	
4	P54	0	R/W	
3	P53	0	R/W	
2	P52	0	R/W	
1	P51	0	R/W	
0	P50	0	R/W	

9.4.4 Port Pull-Up Control Register 5 (PUCR5)

PUCR5 controls the pull-up MOS in bit units of the pins set as the input ports.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	—	Reserved These bits are always read as 0.
5	PUCR55	0	R/W	Only bits for which PCR5 is cleared are valid. The pull-up MOS of the corresponding pins enter the on-state when these bits are set to 1, while they enter the off-state when these bits are cleared to 0.
4	PUCR54	0	R/W	
3	PUCR53	0	R/W	
2	PUCR52	0	R/W	
1	PUCR51	0	R/W	
0	PUCR50	0	R/W	

9.4.5 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P57/SCL pin

Register	ICCR1	PCR5	
Bit Name	**ICE**	**PCR57**	**Pin Function**
Setting Value	0	0	P57 input pin
		1	P57 output pin
	1	X	SCL I/O pin

Legend: X: Don't care.

SCL performs the NMOS open-drain output, that enables a direct bus drive.

P56/SDA pin

Register	ICCR1	PCR5	
Bit Name	**ICE**	**PCR56**	**Pin Function**
Setting Value	0	0	P56 input pin
		1	P56 output pin
	1	X	SDA I/O pin

Legend: X: Don't care.

SDA performs the NMOS open-drain output, that enables a direct bus drive.

P55/$\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ pin

Register	PMR5	PCR5	
Bit Name	**WKP5**	**PCR55**	**Pin Function**
Setting Value	0	0	P55 input pin
		1	P55 output pin
	1	X	$\overline{\text{WKP5}}$/$\overline{\text{ADTRG}}$ input pin

Legend: X: Don't care.

P54/$\overline{\text{WKP4}}$ pin

Register	PMR5	PCR5	
Bit Name	**WKP4**	**PCR54**	**Pin Function**
Setting Value	0	0	P54 input pin
		1	P54 output pin
	1	X	$\overline{\text{WKP4}}$ input pin

Legend: X: Don't care.

P53/$\overline{\text{WKP3}}$ pin

Register	PMR5	PCR5	
Bit Name	**WKP3**	**PCR53**	**Pin Function**
Setting Value	0	0	P53 input pin
		1	P53 output pin
	1	X	$\overline{\text{WKP3}}$ input pin

Legend: X: Don't care.

P52/$\overline{\text{WKP2}}$ pin

Register	PMR5	PCR5	
Bit Name	**WKP2**	**PCR52**	**Pin Function**
Setting Value	0	0	P52 input pin
		1	P52 output pin
	1	X	$\overline{\text{WKP2}}$ input pin

Legend: X: Don't care.

P51/$\overline{\text{WKP1}}$ pin

Register	PMR5	PCR5	
Bit Name	**WKP1**	**PCR51**	**Pin Function**
Setting Value	0	0	P51 input pin
		1	P51 output pin
	1	X	$\overline{\text{WKP1}}$ input pin

Legend: X: Don't care.

P50/$\overline{\text{WKP0}}$ pin

Register	PMR5	PCR5	
Bit Name	**WKP0**	**PCR50**	**Pin Function**
Setting Value	0	0	P50 input pin
		1	P50 output pin
	1	X	$\overline{\text{WKP0}}$ input pin

Legend: X: Don't care.

9.5 Port 6

Port 6 is a general I/O port also functioning as a timer Z I/O pin. Each pin of the port 6 is shown in figure 9.5. The register setting of the timer Z has priority for functions of the pins for both uses.

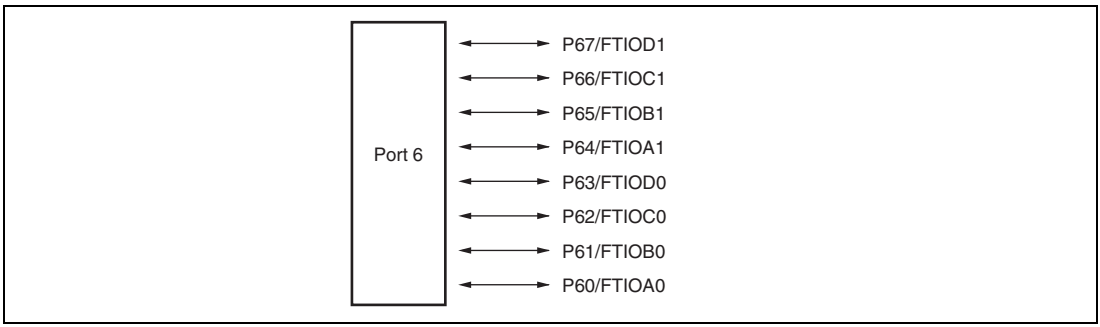


Figure 9.5 Port 6 Pin Configuration

Port 6 has the following registers.

- Port control register 6 (PCR6)
- Port data register 6 (PDR6)

9.5.1 Port Control Register 6 (PCR6)

PCR6 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 6.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR67	0	W	When each of the port 6 pins P67 to P60 functions as a general I/O port, setting a PCR6 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port.
6	PCR66	0	W	
5	PCR65	0	W	
4	PCR64	0	W	
3	PCR63	0	W	
2	PCR62	0	W	
1	PCR61	0	W	
0	PCR60	0	W	

9.5.2 Port Data Register 6 (PDR6)

PDR6 is a general I/O port data register of port 6.

Bit	Bit Name	Initial Value	R/W	Description
7	P67	0	R/W	Stores output data for port 6 pins.
6	P66	0	R/W	If PDR6 is read while PCR6 bits are set to 1, the value stored in PDR6 are read. If PDR6 is read while PCR6 bits are cleared to 0, the pin states are read regardless of the value stored in PDR6.
5	P65	0	R/W	
4	P64	0	R/W	
3	P63	0	R/W	
2	P62	0	R/W	
1	P61	0	R/W	
0	P60	0	R/W	

9.5.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P67/FTIOD1 pin

Register	TOER	TFCR	TPMR	TIORC1	PCR6	
Bit Name	ED1	CMD1 and CMD0	PWMD1	IOD2 to IOD0	PCR67	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P67 input/FTIOD1 input pin
					1	P67 output pin
	0	00	0	001 or 01X	X	FTIOD1 output pin
			1	XXX		
		Other than 00	X	XXX		

Legend: X: Don't care.

P66/FTIOC1 pin

Register	TOER	TFCR	TPMR	TIORC1	PCR6	
Bit Name	EC1	CMD1 and CMD0	PWMC1	IOC2 to IOC0	PCR66	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P66 input/FTIOC1 input pin
					1	P66 output pin
	0	00	0	001 or 01X	X	FTIOC1 output pin
			1	XXX		
		Other than 00	X	XXX		

Legend: X: Don't care.

P65/FTIOB1 pin

Register	TOER	TFCR	TPMR	TIORA1	PCR6	
Bit Name	EB1	CMD1 to CMD0	PWMB1	IOB2 to IOB0	PCR65	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P65 input/FTIOB1 input pin
					1	P65 output pin
	0	00	0	001 or 01X	X	FTIOB1 output pin
			1	XXX		
		Other than 00	X	XXX		

Legend: X: Don't care.

P64/FTIOA1 pin

Register	TOER	TFCR	TIORA1	PCR6	
Bit Name	EB1	CMD1 to CMD0	IOA2 to IOA0	PCR64	Pin Function
Setting Value	1	XX	000 or	0	P64 input/FTIOA1 input pin
			1XX	1	P64 output pin
	0	00	001 or 01X	X	FTIOA1 output pin

Legend: X: Don't care.

P63/FTIOD0 pin

Register	TOER	TFCR	TPMR	TIORC0	PCR6	
Bit Name	ED0	CMD1 to CMD0	PWMD0	IOD2 to IOD0	PCR63	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P63 input/FTIOD0 input pin
					1	P63 output pin
	0	00	0	001 or 01X	X	FTIOD0 output pin
			1	XXX		
		Other than 00	X	XXX		

Legend: X: Don't care.

P62/FTIOC0 pin

Register	TOER	TFCR	TPMR	TIORC0	PCR6	
Bit Name	EC0	CMD1 to CMD0	PWMC0	IOC2 to IOC0	PCR62	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P62 input/FTIOC0 input pin
					1	P62 output pin
	0	00	0	001 or 01X	X	FTIOC0 output pin
			1	XXX		
		Other than 00	X	XXX		

Legend: X: Don't care.

P61/FTIOB0 pin

Register	TOER	TFCR	TPMR	TIORA0	PCR6	
Bit Name	EB0	CMD1 to CMD0	PWMB0	IOB2 to IOB0	PCR61	Pin Function
Setting Value	1	00	0	000 or 1XX	0	P61 input/FTIOB0 input pin
					1	P61 output pin
	0	00	0	001 or 01X	X	FTIOB0 output pin
			1	XXX		
		Other than 00	X	XXX		

Legend: X: Don't care.

P60/FTIOA0 pin

Register	TOER	TFCR	TFCR	TIORA0	PCR6	
Bit Name	EA0	CMD1 to CMD0	STCLK	IOA2 to IOA0	PCR60	Pin Function
Setting Value	1	XX	X	000 or	0	P60 input/FTIOA0 input pin
				1XX	1	P60 output pin
	0	00	0	001 or 01X	X	FTIOA0 output pin

Legend: X: Don't care.

9.6 Port 7

Port 7 is a general I/O port also functioning as a timer V I/O pin and SCI3_2 I/O pin. Each pin of the port 7 is shown in figure 9.6. The register settings of the timer V and SCI3_2 have priority for functions of the pins for both uses.

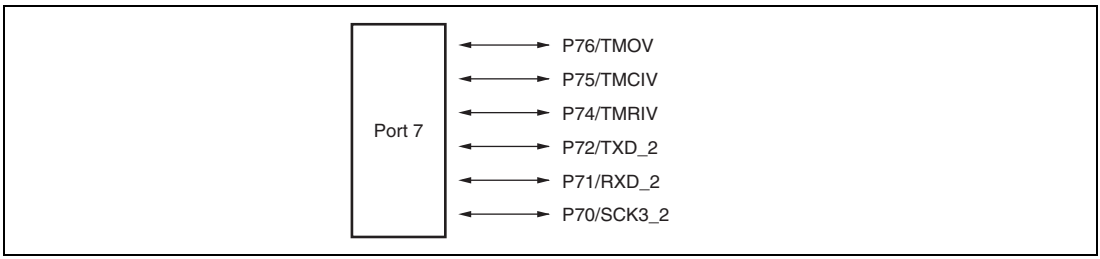


Figure 9.6 Port 7 Pin Configuration

Port 7 has the following registers.

- Port control register 7 (PCR7)
- Port data register 7 (PDR7)

9.6.1 Port Control Register 7 (PCR7)

PCR7 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 7.

Bit	Bit Name	Initial Value	R/W	Description
7	—	—	—	When each of the port 7 pins P76 to P74 and P72 to P70 functions as a general I/O port, setting a PCR7 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port. Bits 7 and 3 are reserved bits.
6	PCR76	0	W	
5	PCR75	0	W	
4	PCR74	0	W	
3	—	—	—	
2	PCR72	0	W	
1	PCR71	0	W	
0	PCR70	0	W	

9.6.2 Port Data Register 7 (PDR7)

PDR7 is a general I/O port data register of port 7.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Stores output data for port 7 pins.
6	P76	0	R/W	If PDR7 is read while PCR7 bits are set to 1, the value stored in PDR7 are read. If PDR7 is read while PCR7 bits are cleared to 0, the pin states are read regardless of the value stored in PDR7.
5	P75	0	R/W	
4	P74	0	R/W	
3	—	1	—	Bits 7 and 3 are reserved bits. These bits are always read as 1.
2	P72	0	R/W	
1	P71	0	R/W	
0	P70	0	R/W	

9.6.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P76/TMOV pin

Register	TCSRV	PCR7	
Bit Name	**OS3 to OS0**	**PCR76**	**Pin Function**
Setting Value	0000	0	P76 input pin
		1	P76 output pin
	Other than the above values	X	TMOV output pin

Legend: X: Don't care.

P75/TMCIV pin

Register	PCR7	
Bit Name	**PCR75**	**Pin Function**
Setting Value	0	P75 input/TMCIV input pin
	1	P75 output/TMCIV input pin

P74/TMRIV pin

Register	PCR7	
Bit Name	**PCR74**	**Pin Function**
Setting Value	0	P74 input/TMRIV input pin
	1	P74 output/TMRIV input pin

P72/TXD_2 pin

Register	PMR1	PCR7	
Bit Name	**TXD2**	**PCR72**	**Pin Function**
Setting Value	0	0	P72 input pin
		1	P72 output pin
	1	X	TXD_2 output pin

Legend: X: Don't care.

P71/RXD_2 pin

Register	SCR3_2	PCR7	
Bit Name	**RE**	**PCR71**	**Pin Function**
Setting Value	0	0	P71 input pin
		1	P71 output pin
	1	X	RXD_2 input pin

Legend: X: Don't care.

P70/SCK3_2 pin

Register	SCR3_2		SMR2	PCR7	
Bit Name	**CKE1**	**CKE0**	**COM**	**PCR70**	**Pin Function**
Setting Value	0	0	0	0	P70 input pin
				1	P70 output pin
	0	0	1	X	SCK3_2 output pin
	0	1	X	X	SCK3_2 output pin
	1	X	X	X	SCK3_2 input pin

Legend: X: Don't care.

9.7 Port 8

Port 8 is a general I/O port. Each pin of the port 8 is shown in figure 9.7.

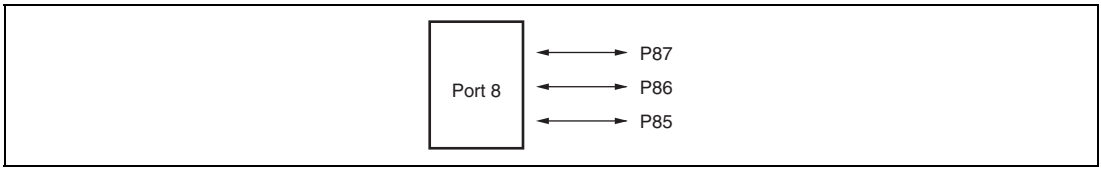


Figure 9.7 Port 8 Pin Configuration

Port 8 has the following registers.

- Port control register 8 (PCR8)
- Port data register 8 (PDR8)

9.7.1 Port Control Register 8 (PCR8)

PCR8 selects inputs/outputs in bit units for pins to be used as general I/O ports of port 8.

Bit	Bit Name	Initial Value	R/W	Description
7	PCR87	0	W	When each of the port 8 pins P87 to P85 functions as a general I/O port, setting a PCR8 bit to 1 makes the corresponding pin an output port, while clearing the bit to 0 makes the pin an input port.
6	PCR86	0	W	
5	PCR85	0	W	
4 to 0	—	—	—	Reserved

9.7.2 Port Data Register 8 (PDR8)

PDR8 is a general I/O port data register of port 8.

Bit	Bit Name	Initial Value	R/W	Description
7	P87	0	R/W	PDR8 stores output data for port 8 pins.
6	P86	0	R/W	If PDR8 is read while PCR8 bits are set to 1, the value stored in PDR8 is read. If PDR8 is read while PCR8 bits are cleared to 0, the pin states are read regardless of the value stored in PDR8.
5	P85	0	R/W	
4 to 0	—	All 1	—	Reserved These bits are always read as 1.

9.7.3 Pin Functions

The correspondence between the register specification and the port functions is shown below.

P87 pin

Register	PCR8	
Bit Name	**PCR87**	**Pin Function**
Setting Value	0	P87 input pin
	1	P87 output pin

P86 pin

Register	PCR8	
Bit Name	**PCR86**	**Pin Function**
Setting Value	0	P86 input pin
	1	P86 output pin

P85 pin

Register	PCR8	
Bit Name	**PCR85**	**Pin Function**
Setting Value	0	P85 input pin
	1	P85 output pin

9.8 Port B

Port B is an input port also functioning as an A/D converter analog input pin. Each pin of the port B is shown in figure 9.8.

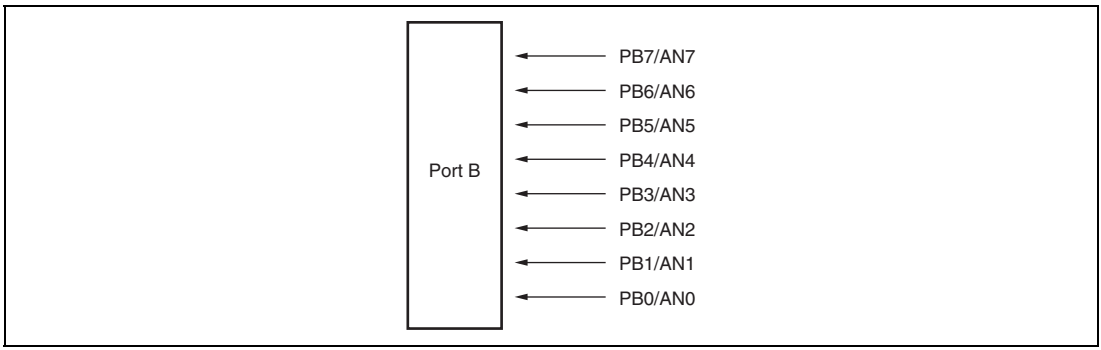


Figure 9.8 Port B Pin Configuration

Port B has the following register.

- Port data register B (PDRB)

9.8.1 Port Data Register B (PDRB)

PDRB is a general input-only port data register of port B.

Bit	Bit Name	Initial Value	R/W	Description
7	PB7	—	R	The input value of each pin is read by reading this register. However, if a port B pin is designated as an analog input channel by ADCSR in A/D converter, 0 is read.
6	PB6	—	R	
5	PB5	—	R	
4	PB4	—	R	
3	PB3	—	R	
2	PB2	—	R	
1	PB1	—	R	
0	PB0	—	R	

Section 10 Realtime Clock (RTC)

The realtime clock (RTC) is a timer used to count time ranging from a second to a week. Figure 10.1 shows the block diagram of the RTC.

10.1 Features

- Counts seconds, minutes, hours, and day-of-week
- Start/stop function
- Reset function
- Readable/writable counter of seconds, minutes, hours, and day-of-week with BCD codes
- Periodic (seconds, minutes, hours, days, and weeks) interrupts
- 8-bit free running counter
- Selection of clock source

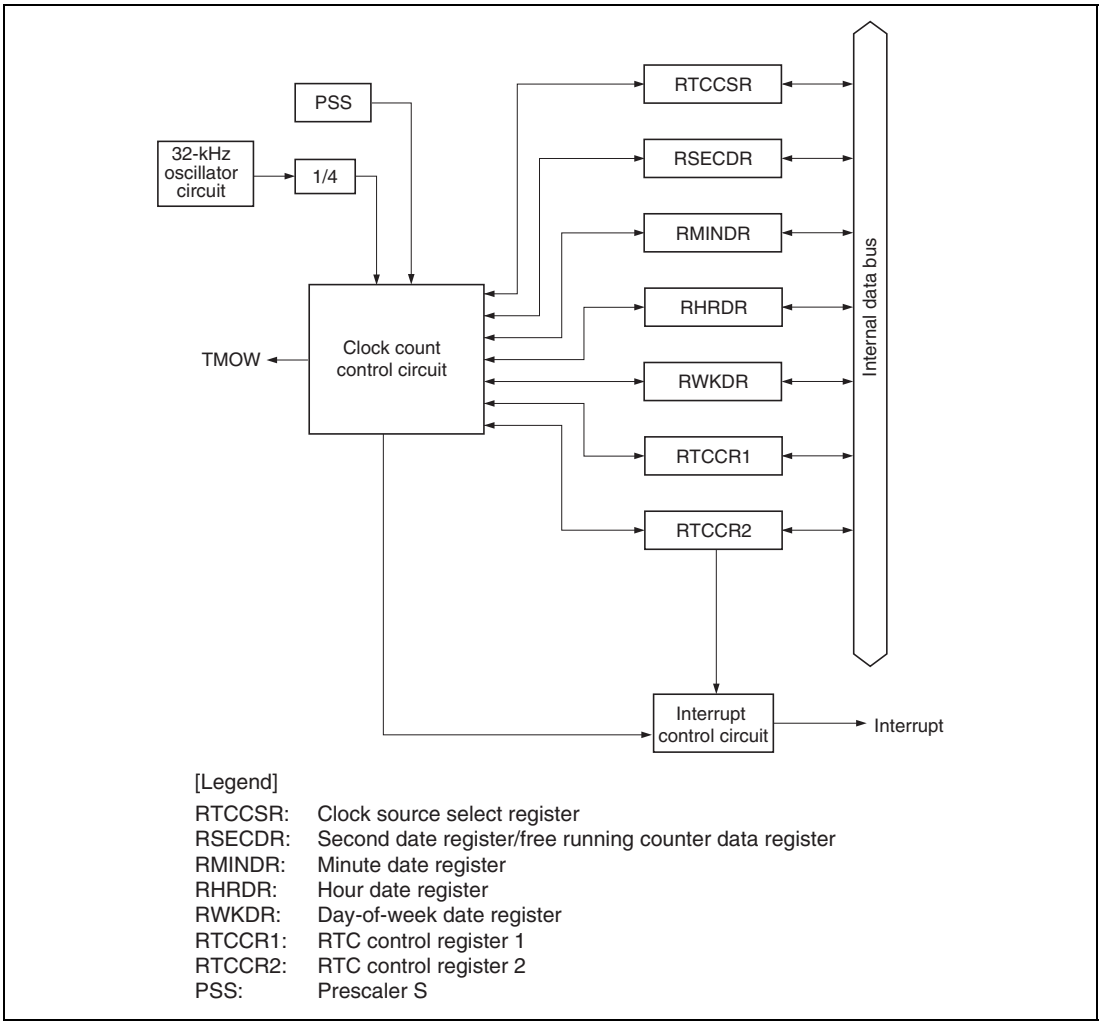


Figure 10.1 Block Diagram of RTC

10.2 Input/Output Pin

Table 10.1 shows the RTC input/output pin.

Table 10.1 Pin Configuration

Name	Abbreviation	I/O	Function
Clock output	TMOW	Output	RTC divided clock output

10.3 Register Descriptions

The RTC has the following registers.

- Second data register/free running counter data register (RSECDR)
- Minute data register (RMINDR)
- Hour data register (RHRDR)
- Day-of-week data register (RWKDR)
- RTC control register 1 (RTCCR1)
- RTC control register 2 (RTCCR2)
- Clock source select register (RTCCSR)

10.3.1 Second Data Register/Free Running Counter Data Register (RSECDR)

RSECDR counts the BCD-coded second value. The setting range is decimal 00 to 59. It is an 8-bit read register used as a counter, when it operates as a free running counter. For more information on reading seconds, minutes, hours, and day-of-week, see section 10.4.3, Data Reading Procedure.

Bit	Bit Name	Initial Value	R/W	Description
7	BSY	—	R	RTC busy This bit is set to 1 when the RTC is updating (operating) the values of second, minute, hour, and day-of-week data registers. When this bit is 0, the values of second, minute, hour, and day-of-week data registers must be adopted.
6	SC12	—	R/W	Counting ten's position of seconds
5	SC11	—	R/W	Counts on 0 to 5 for 60-second counting.
4	SC10	—	R/W	
3	SC03	—	R/W	Counting one's position of seconds
2	SC02	—	R/W	Counts on 0 to 9 once per second. When a carry is generated, 1 is added to the ten's position.
1	SC01	—	R/W	
0	SC00	—	R/W	

10.3.2 Minute Data Register (RMINDR)

RMINDR counts the BCD-coded minute value on the carry generated once per minute by the RSECDR counting. The setting range is decimal 00 to 59.

Bit	Bit Name	Initial Value	R/W	Description
7	BSY	—	R	RTC busy This bit is set to 1 when the RTC is updating (operating) the values of second, minute, hour, and day-of-week data registers. When this bit is 0, the values of second, minute, hour, and day-of-week data registers must be adopted.
6	MN12	—	R/W	Counting ten's position of minutes
5	MN11	—	R/W	Counts on 0 to 5 for 60-minute counting.
4	MN10	—	R/W	
3	MN03	—	R/W	Counting one's position of minutes
2	MN02	—	R/W	Counts on 0 to 9 once per minute. When a carry is generated, 1 is added to the ten's position.
1	MN01	—	R/W	
0	MN00	—	R/W	

10.3.3 Hour Data Register (RHRDR)

RHRDR counts the BCD-coded hour value on the carry generated once per hour by RMINDR. The setting range is either decimal 00 to 11 or 00 to 23 by the selection of the 12/24 bit in RTCCR1.

Bit	Bit Name	Initial Value	R/W	Description
7	BSY	—	R	RTC busy This bit is set to 1 when the RTC is updating (operating) the values of second, minute, hour, and day-of-week data registers. When this bit is 0, the values of second, minute, hour, and day-of-week data registers must be adopted.
6	—	0	—	Reserved This bit is always read as 0.
5	HR11	—	R/W	Counting ten's position of hours
4	HR10	—	R/W	Counts on 0 to 2 for ten's position of hours.
3	HR03	—	R/W	Counting one's position of hours
2	HR02	—	R/W	Counts on 0 to 9 once per hour. When a carry is generated, 1 is added to the ten's position.
1	HR01	—	R/W	
0	HR00	—	R/W	

10.3.4 Day-of-Week Data Register (RWKDR)

RWKDR counts the BCD-coded day-of-week value on the carry generated once per day by RHRDR. The setting range is decimal 0 to 6 using bits WK2 to WK0.

Bit	Bit Name	Initial Value	R/W	Description
7	BSY	—	R	RTC busy This bit is set to 1 when the RTC is updating (operating) the values of second, minute, hour, and day-of-week data registers. When this bit is 0, the values of second, minute, hour, and day-of-week data registers must be adopted.
6 to 3	—	All 0	—	Reserved These bits are always read as 0.
2	WK2	—	R/W	Day-of-week counting
1	WK1	—	R/W	Day-of-week is indicated with a binary code
0	WK0	—	R/W	000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

10.3.5 RTC Control Register 1 (RTCCR1)

RTCCR1 controls start/stop and reset of the clock timer. For the definition of time expression, see figure 10.2.

Bit	Bit Name	Initial Value	R/W	Description
7	RUN	—	R/W	RTC operation start 0: Stops RTC operation 1: Starts RTC operation
6	12/24	—	R/W	Operating mode 0: RTC operates in 12-hour mode. RHRDR counts on 0 to 11. 1: RTC operates in 24-hour mode. RHRDR counts on 0 to 23.
5	PM	—	R/W	A.m./p.m. 0: Indicates a.m. when RTC is in the 12-hour mode. 1: Indicates p.m. when RTC is in the 12-hour mode.
4	RST	0	R/W	Reset 0: Normal operation 1: Resets registers and control circuits except RTCCSR and this bit. Clear this bit to 0 after having been set to 1.
3 to 0	—	All 0	—	Reserved These bits are always read as 0.

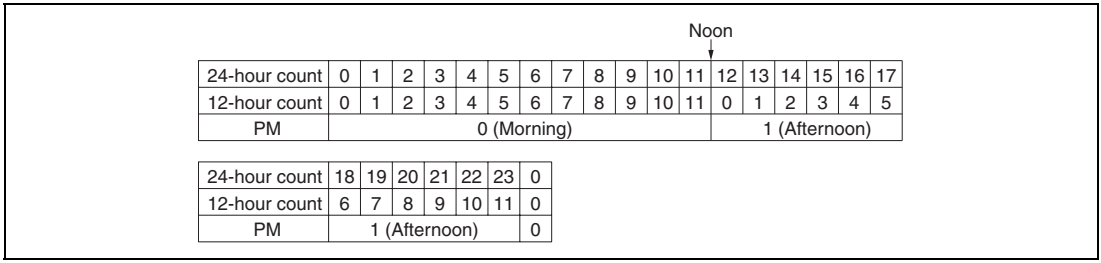


Figure 10.2 Definition of Time Expression

10.3.6 RTC Control Register 2 (RTCCR2)

RTCCR2 controls RTC periodic interrupts of weeks, days, hours, minutes, and seconds. Enabling interrupts of weeks, days, hours, minutes, and seconds sets the IRRTA flag to 1 in the interrupt flag register 1 (IRR1) when an interrupt occurs. It also controls an overflow interrupt of a free running counter when RTC operates as a free running counter.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	—	Reserved These bits are always read as 0.
5	FOIE	—	R/W	Free Running Counter Overflow Interrupt Enable 0: Disables an overflow interrupt 1: Enables an overflow interrupt
4	WKIE	—	R/W	Week Periodic Interrupt Enable 0: Disables a week periodic interrupt 1: Enables a week periodic interrupt
3	DYIE	—	R/W	Day Periodic Interrupt Enable 0: Disables a day periodic interrupt 1: Enables a day periodic interrupt
2	HRIE	—	R/W	Hour Periodic Interrupt Enable 0: Disables an hour periodic interrupt 1: Enables an hour periodic interrupt
1	MNIE	—	R/W	Minute Periodic Interrupt Enable 0: Disables a minute periodic interrupt 1: Enables a minute periodic interrupt
0	SEIE	—	R/W	Second Periodic Interrupt Enable 0: Disables a second periodic interrupt 1: Enables a second periodic interrupt

10.3.7 Clock Source Select Register (RTCCSR)

RTCCSR selects clock source. A free running counter controls start/stop of counter operation by the RUN bit in RTCCR1. When a clock other than 32.768 kHz is selected, the RTC is disabled and operates as an 8-bit free running counter. When the RTC operates as an 8-bit free running counter, RSECDR enables counter values to be read. An interrupt can be generated by setting 1 to the FOIE bit in RTCCR2 and enabling an overflow interrupt of the free running counter. A clock in which the system clock is divided by 32, 16, 8, or 4 is output in active or sleep mode.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	—	Reserved This bit is always read as 0.
6	RCS6	0	R/W	Clock output selection
5	RCS5	0	R/W	Selects a clock output from the TMOW pin when setting TMOW in PMR1 to 1. 00: φ/4 01: φ/8 10: φ/16 11: φ/32
4	—	0	—	Reserved This bit is always read as 0.
3	RCS3	1	R/W	Clock source selection
2	RCS2	0	R/W	0000: φ/8····· Free running counter operation
1	RCS1	0	R/W	0001: φ/32····· Free running counter operation
0	RCS0	0	R/W	0010: φ/128····· Free running counter operation 0011: φ/256····· Free running counter operation 0100: φ/512····· Free running counter operation 0101: φ/2048····· Free running counter operation 0110: φ/4096····· Free running counter operation 0111: φ/8192····· Free running counter operation 1XXX: 32.768 kHz·····RTC operation

Legend: X: Don't care.

10.4 Operation

10.4.1 Initial Settings of Registers after Power-On

The RTC registers that store second, minute, hour, and day-of week data are not reset by a $\overline{\text{RES}}$ input. Therefore, all registers must be set to their initial values after power-on. Once the register setting are made, the RTC provides an accurate time as long as power is supplied regardless of a $\overline{\text{RES}}$ input.

10.4.2 Initial Setting Procedure

Figure 10.3 shows the procedure for the initial setting of the RTC. To set the RTC again, also follow this procedure.

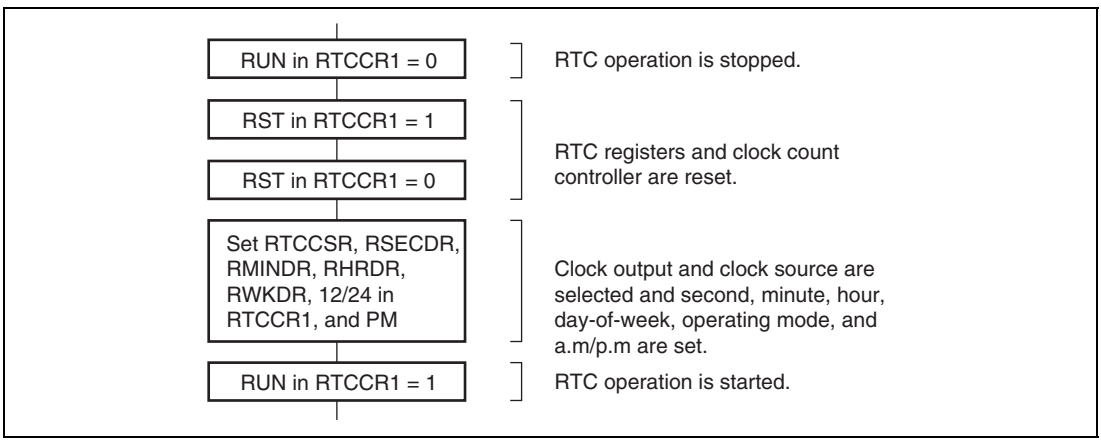


Figure 10.3 Initial Setting Procedure

10.4.3 Data Reading Procedure

When the seconds, minutes, hours, or day-of-week datum is updated while time data is being read, the data obtained may not be correct, and so the time data must be read again. Figure 10.4 shows an example in which correct data is not obtained. In this example, since only RSECDR is read after data update, about 1-minute inconsistency occurs.

To avoid reading in this timing, the following processing must be performed.

1. Check the setting of the BSY bit, and when the BSY bit changes from 1 to 0, read from the second, minute, hour, and day-of-week registers. When about 62.5 ms is passed after the BSY bit is set to 1, the registers are updated, and the BSY bit is cleared to 0.
2. Making use of interrupts, read from the second, minute, hour, and day-of week registers after the IRRTA flag in IRR1 is set to 1 and the BSY bit is confirmed to be 0.
3. Read from the second, minute, hour, and day-of week registers twice in a row, and if there is no change in the read data, the read data is used.

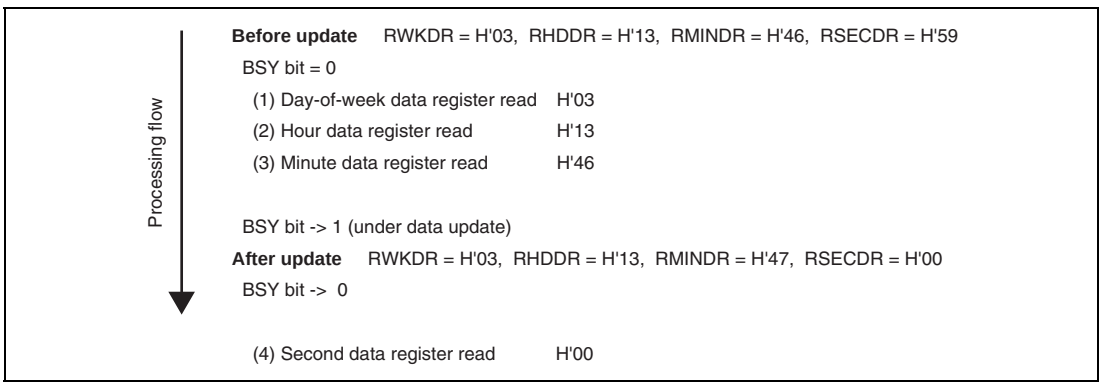


Figure 10.4 Example: Reading of Inaccurate Time Data

10.5 Interrupt Source

There are five kinds of RTC interrupts: week interrupts, day interrupts, hour interrupts, minute interrupts, and second interrupts.

When using an interrupt, initiate the RTC last after other registers are set. Do not set multiple interrupt enable bits in RTCCR2 simultaneously to 1.

When an interrupt request of the RTC occurs, the IRRTA flag in IRR1 is set to 1. When clearing the flag, write 0.

Table 10.2 Interrupt Source

Interrupt Name	Interrupt Source	Interrupt Enable Bit
Overflow interrupt	Occurs when the free running counter is overflown.	FOIE
Week periodic interrupt	Occurs every week when the day-of-week date register value becomes 0.	WKIE
Day periodic interrupt	Occurs every day when the day-of-week date register is counted.	DYIE
Hour periodic interrupt	Occurs every hour when the hour date register is counted.	HRIE
Minute periodic interrupt	Occurs every minute when the minute date register is counted.	MNIE
Second periodic interrupt	Occurs every second when the second date register is counted.	SCIE

Section 11 Timer B1

Timer B1 is an 8-bit timer that increments each time a clock pulse is input. This timer has two operating modes, interval and auto reload. Figure 11.1 shows a block diagram of timer B1.

11.1 Features

- Selection of seven internal clock sources (ϕ/8192, ϕ/2048, ϕ/512, ϕ/256, ϕ/64, ϕ/16, and ϕ/4) or an external clock (can be used to count external events).
- An interrupt is generated when the counter overflows.

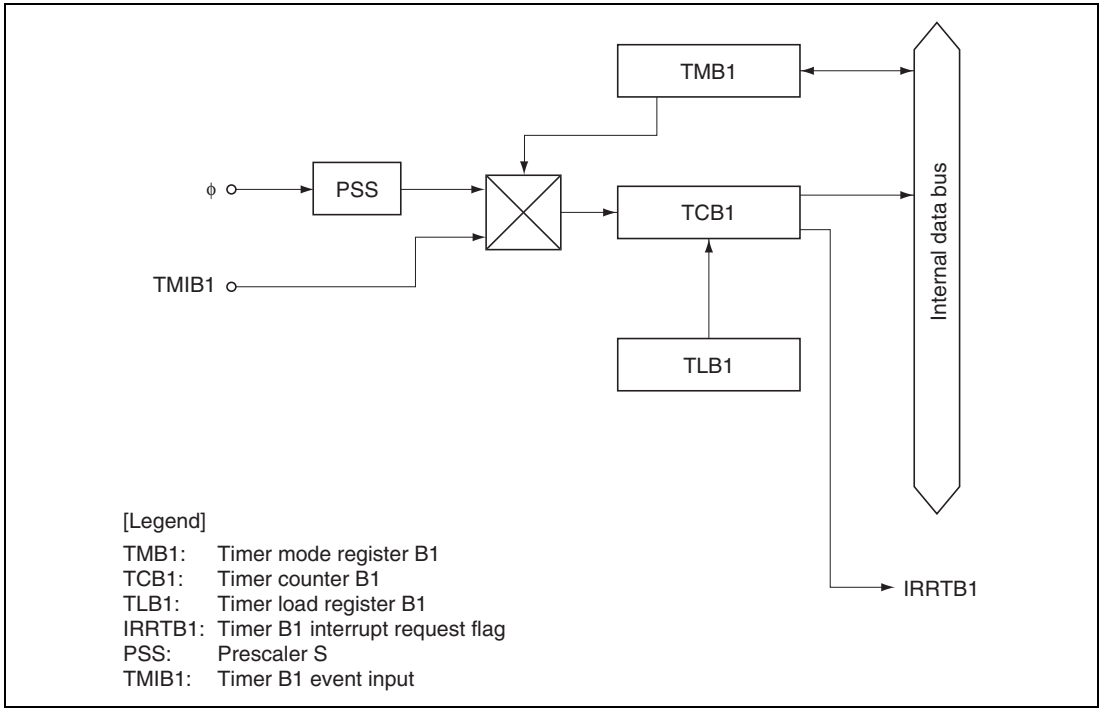


Figure 11.1 Block Diagram of Timer B1

11.2 Input/Output Pin

Table 11.1 shows the timer B1 pin configuration.

Table 11.1 Pin Configuration

Name	Abbreviation	I/O	Function
Timer B1 event input	TMIB1	Input	Event input to TCB1

11.3 Register Descriptions

The timer B1 has the following registers.

- Timer mode register B1 (TMB1)
- Timer counter B1 (TCB1)
- Timer load register B1 (TLB1)

11.3.1 Timer Mode Register B1 (TMB1)

TMB1 selects the auto-reload function and input clock.

Bit	Bit Name	Initial Value	R/W	Description
7	TMB17	0	R/W	Auto-reload function select 0: Interval timer function selected 1: Auto-reload function selected
6 to 3	—	All 1	—	Reserved These bits are always read as 1.
2	TMB12	0	R/W	Clock select
1	TMB11	0	R/W	000: Internal clock: ϕ/8192
0	TMB10	0	R/W	001: Internal clock: ϕ/2048 010: Internal clock: ϕ/512 011: Internal clock: ϕ/256 100: Internal clock: ϕ/64 101: Internal clock: ϕ/16 110: Internal clock: ϕ/4 111: External event (TMIB1): rising or falling edge* Note: * The edge of the external event signal is selected by bit IEG1 in the interrupt edge select register 1 (IEGR1). See section 3.2.1, Interrupt Edge Select Register 1 (IEGR1), for details. Before setting TMB12 to TMB10 to 1, IRQ1 in the port mode register 1 (PMR1) should be set to 1.

11.3.2 Timer Counter B1 (TCB1)

TCB1 is an 8-bit read-only up-counter, which is incremented by internal clock input. The clock source for input to this counter is selected by bits TMB12 to TMB10 in TMB1. TCB1 values can be read by the CPU at any time. When TCB1 overflows from H'FF to H'00 or to the value set in TLB1, the IRRTB1 flag in IRR2 is set to 1. TCB1 is allocated to the same address as TLB1. TCB1 is initialized to H'00.

11.3.3 Timer Load Register B1 (TLB1)

TLB1 is an 8-bit write-only register for setting the reload value of TCB1. When a reload value is set in TLB1, the same value is loaded into TCB1 as well, and TCB1 starts counting up from that value. When TCB1 overflows during operation in auto-reload mode, the TLB1 value is loaded into TCB1. Accordingly, overflow periods can be set within the range of 1 to 256 input clocks. TLB1 is allocated to the same address as TCB1. TLB1 is initialized to H'00.

11.4 Operation

11.4.1 Interval Timer Operation

When bit TMB17 in TMB1 is cleared to 0, timer B1 functions as an 8-bit interval timer. Upon reset, TCB1 is cleared to H'00 and bit TMB17 is cleared to 0, so up-counting and interval timing resume immediately. The operating clock of timer B1 is selected from seven internal clock signals output by prescaler S, or an external clock input at pin TMB1. The selection is made by bits TMB12 to TMB10 in TMB1.

After the count value in TMB1 reaches H'FF, the next clock signal input causes timer B1 to overflow, setting flag IRRTB1 in IRR2 to 1. If IENTB1 in IENR2 is 1, an interrupt is requested to the CPU.

At overflow, TCB1 returns to H'00 and starts counting up again. During interval timer operation (TMB17 = 0), when a value is set in TLB1, the same value is set in TCB1.

11.4.2 Auto-Reload Timer Operation

Setting bit TMB17 in TMB1 to 1 causes timer B1 to function as an 8-bit auto-reload timer. When a reload value is set in TLB1, the same value is loaded into TCB1, becoming the value from which TCB1 starts its count. After the count value in TCB1 reaches H'FF, the next clock signal input causes timer B1 to overflow. The TLB1 value is then loaded into TCB1, and the count continues from that value. The overflow period can be set within a range from 1 to 256 input clocks, depending on the TLB1 value.

The clock sources and interrupts in auto-reload mode are the same as in interval mode. In auto-reload mode (TMB17 = 1), when a new value is set in TLB1, the TLB1 value is also loaded into TCB1.

11.4.3 Event Counter Operation

Timer B1 can operate as an event counter in which TMIB1 is set to an event input pin. External event counting is selected by setting bits TMB12 to TMB10 in TMB1 to 1. TCB1 counts up at rising or falling edge of an external event signal input at pin TMB1.

When timer B1 is used to count external event input, bit IRQ1 in PMR1 should be set to 1 and IEN1 in IENR1 should be cleared to 0 to disable IRQ1 interrupt requests.

11.5 Timer B1 Operating Modes

Table 11.2 shows the timer B1 operating modes.

Table 11.2 Timer B1 Operating Modes

Operating Mode		Reset	Active	Sleep	Subactive	Subsleep	Standby
TCB1	Interval	Reset	Functions	Functions	Halted	Halted	Halted
	Auto-reload	Reset	Functions	Functions	Halted	Halted	Halted
TMB1		Reset	Functions	Retained	Retained	Retained	Retained

Section 12 Timer V

Timer V is an 8-bit timer based on an 8-bit counter. Timer V counts external events. Compare-match signals with two registers can also be used to reset the counter, request an interrupt, or output a pulse signal with an arbitrary duty cycle. Counting can be initiated by a trigger input at the TRGV pin, enabling pulse output control to be synchronized to the trigger, with an arbitrary delay from the trigger input. Figure 12.1 shows a block diagram of timer V.

12.1 Features

- Choice of seven clock signals is available.
  Choice of six internal clock sources (ϕ/128, ϕ/64, ϕ/32, ϕ/16, ϕ/8, ϕ/4) or an external clock.
- Counter can be cleared by compare match A or B, or by an external reset signal. If the count stop function is selected, the counter can be halted when cleared.
- Timer output is controlled by two independent compare match signals, enabling pulse output with an arbitrary duty cycle, PWM output, and other applications.
- Three interrupt sources: compare match A, compare match B, timer overflow
- Counting can be initiated by trigger input at the TRGV pin. The rising edge, falling edge, or both edges of the TRGV input can be selected.

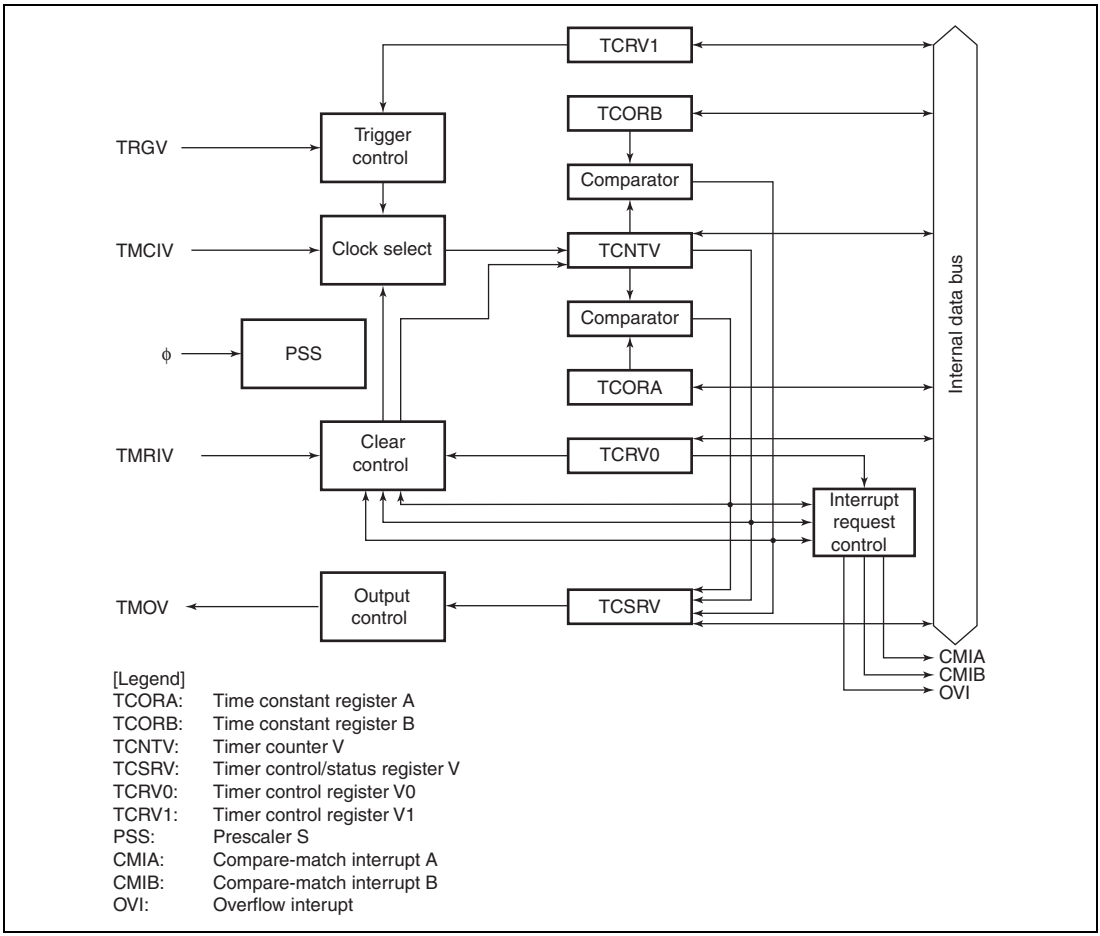


Figure 12.1 Block Diagram of Timer V

12.2 Input/Output Pins

Table 12.1 shows the timer V pin configuration.

Table 12.1 Pin Configuration

Name	Abbreviation	I/O	Function
Timer V output	TMOV	Output	Timer V waveform output
Timer V clock input	TMCIV	Input	Clock input to TCNTV
Timer V reset input	TMRIV	Input	External input to reset TCNTV
Trigger input	TRGV	Input	Trigger input to initiate counting

12.3 Register Descriptions

Time V has the following registers.

- Timer counter V (TCNTV)
- Timer constant register A (TCORA)
- Timer constant register B (TCORB)
- Timer control register V0 (TCRV0)
- Timer control/status register V (TCSRV)
- Timer control register V1 (TCRV1)

12.3.1 Timer Counter V (TCNTV)

TCNTV is an 8-bit up-counter. The clock source is selected by bits CKS2 to CKS0 in timer control register V0 (TCRV0). The TCNTV value can be read and written by the CPU at any time. TCNTV can be cleared by an external reset input signal, or by compare match A or B. The clearing signal is selected by bits CCLR1 and CCLR0 in TCRV0.

When TCNTV overflows, OVF is set to 1 in timer control/status register V (TCSRV).

TCNTV is initialized to H'00.

12.3.2 Time Constant Registers A and B (TCORA, TCORB)

TCORA and TCORB have the same function.

TCORA and TCORB are 8-bit read/write registers.

TCORA and TCNTV are compared at all times. When the TCORA and TCNTV contents match, CMFA is set to 1 in TCSRV. If CMIEA is also set to 1 in TCRV0, a CPU interrupt is requested. Note that they must not be compared during the T3 state of a TCORA write cycle.

Timer output from the TMOV pin can be controlled by the identifying signal (compare match A) and the settings of bits OS3 to OS0 in TCSRV.

TCORA and TCORB are initialized to H'FF.

12.3.3 Timer Control Register V0 (TCRV0)

TCRV0 selects the input clock signals of TCNTV, specifies the clearing conditions of TCNTV, and controls each interrupt request.

Bit	Bit Name	Initial Value	R/W	Description
7	CMIEB	0	R/W	Compare Match Interrupt Enable B When this bit is set to 1, interrupt request from the CMFB bit in TCSRV is enabled.
6	CMIEA	0	R/W	Compare Match Interrupt Enable A When this bit is set to 1, interrupt request from the CMFA bit in TCSRV is enabled.
5	OVIE	0	R/W	Timer Overflow Interrupt Enable When this bit is set to 1, interrupt request from the OVF bit in TCSRV is enabled.
4	CCLR1	0	R/W	Counter Clear 1 and 0
3	CCLR0	0	R/W	These bits specify the clearing conditions of TCNTV. 00: Clearing is disabled 01: Cleared by compare match A 10: Cleared by compare match B 11: Cleared on the rising edge of the TMRIV pin. The operation of TCNTV after clearing depends on TRGE in TCRV1.
2	CKS2	0	R/W	Clock Select 2 to 0
1	CKS1	0	R/W	These bits select clock signals to input to TCNTV and the counting condition in combination with ICKS0 in TCRV1. Refer to table 12.2.
0	CKS0	0	R/W	

Table 12.2 Clock Signals to Input to TCNTV and Counting Conditions

TCRV0			TCRV1	
Bit 2	Bit 1	Bit 0	Bit 0	
CKS2	CKS1	CKS0	ICKS0	Description
0	0	0	—	Clock input prohibited
		1	0	Internal clock: counts on ϕ/4, falling edge
			1	Internal clock: counts on ϕ/8, falling edge
	1	0	0	Internal clock: counts on ϕ/16, falling edge
			1	Internal clock: counts on ϕ/32, falling edge
		1	0	Internal clock: counts on ϕ/64, falling edge
			1	Internal clock: counts on ϕ/128, falling edge
1	0	0	—	Clock input prohibited
		1	—	External clock: counts on rising edge
	1	0	—	External clock: counts on falling edge
		1	—	External clock: counts on rising and falling edge

12.3.4 Timer Control/Status Register V (TCSRV)

TCSRV indicates the status flag and controls outputs by using a compare match.

Bit	Bit Name	Initial Value	R/W	Description
7	CMFB	0	R/W	Compare Match Flag B Setting condition: When the TCNTV value matches the TCORB value Clearing condition: After reading CMFB = 1, cleared by writing 0 to CMFB
6	CMFA	0	R/W	Compare Match Flag A Setting condition: When the TCNTV value matches the TCORA value Clearing condition: After reading CMFA = 1, cleared by writing 0 to CMFA
5	OVF	0	R/W	Timer Overflow Flag Setting condition: When TCNTV overflows from H'FF to H'00 Clearing condition: After reading OVF = 1, cleared by writing 0 to OVF
4	—	1	—	Reserved This bit is always read as 1.
3	OS3	0	R/W	Output Select 3 and 2
2	OS2	0	R/W	These bits select an output method for the TMOV pin by the compare match of TCORB and TCNTV. 00: No change 01: 0 output 10: 1 output 11: Output toggles
1	OS1	0	R/W	Output Select 1 and 0
0	OS0	0	R/W	These bits select an output method for the TMOV pin by the compare match of TCORA and TCNTV. 00: No change 01: 0 output 10: 1 output 11: Output toggles

OS3 and OS2 select the output level for compare match B. OS1 and OS0 select the output level for compare match A. The two output levels can be controlled independently. After a reset, the timer output is 0 until the first compare match.

12.3.5 Timer Control Register V1 (TCRV1)

TCRV1 selects the edge at the TRGV pin, enables TRGV input, and selects the clock input to TCNTV.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	—	Reserved These bits are always read as 1.
4	TVEG1	0	R/W	TRGV Input Edge Select
3	TVEG0	0	R/W	These bits select the TRGV input edge. 00: TRGV trigger input is prohibited 01: Rising edge is selected 10: Falling edge is selected 11: Rising and falling edges are both selected
2	TRGE	0	R/W	TCNT starts counting up by the input of the edge which is selected by TVEG1 and TVEG0. 0: Disables starting counting-up TCNTV by the input of the TRGV pin and halting counting-up TCNTV when TCNTV is cleared by a compare match. 1: Enables starting counting-up TCNTV by the input of the TRGV pin and halting counting-up TCNTV when TCNTV is cleared by a compare match.
1	—	1	—	Reserved This bit is always read as 1.
0	ICKS0	0	R/W	Internal Clock Select 0 This bit selects clock signals to input to TCNTV in combination with CKS2 to CKS0 in TCRV0. Refer to table 12.2.

12.4 Operation

12.4.1 Timer V Operation

1. According to table 12.2, six internal/external clock signals output by prescaler S can be selected as the timer V operating clock signals. When the operating clock signal is selected, TCNTV starts counting-up. Figure 12.2 shows the count timing with an internal clock signal selected, and figure 12.3 shows the count timing with both edges of an external clock signal selected.
2. When TCNTV overflows (changes from H'FF to H'00), the overflow flag (OVF) in TCRV0 will be set. The timing at this time is shown in figure 12.4. An interrupt request is sent to the CPU when OVIE in TCRV0 is 1.
3. TCNTV is constantly compared with TCORA and TCORB. Compare match flag A or B (CMFA or CMFB) is set to 1 when TCNTV matches TCORA or TCORB, respectively. The compare-match signal is generated in the last state in which the values match. Figure 12.5 shows the timing. An interrupt request is generated for the CPU when CMIEA or CMIEB in TCRV0 is 1.
4. When a compare match A or B is generated, the TMOV responds with the output value selected by bits OS3 to OS0 in TCSRV. Figure 12.6 shows the timing when the output is toggled by compare match A.
5. When CCLR1 or CCLR0 in TCRV0 is 01 or 10, TCNTV can be cleared by the corresponding compare match. Figure 12.7 shows the timing.
6. When CCLR1 or CCLR0 in TCRV0 is 11, TCNTV can be cleared by the rising edge of the input of TMRIV pin. A TMRIV input pulse-width of at least 1.5 system clocks is necessary. Figure 12.8 shows the timing.
7. When a counter-clearing source is generated with TRGE in TCRV1 set to 1, the counting-up is halted as soon as TCNTV is cleared. TCNTV resumes counting-up when the edge selected by TVEG1 or TVEG0 in TCRV1 is input from the TGRV pin.

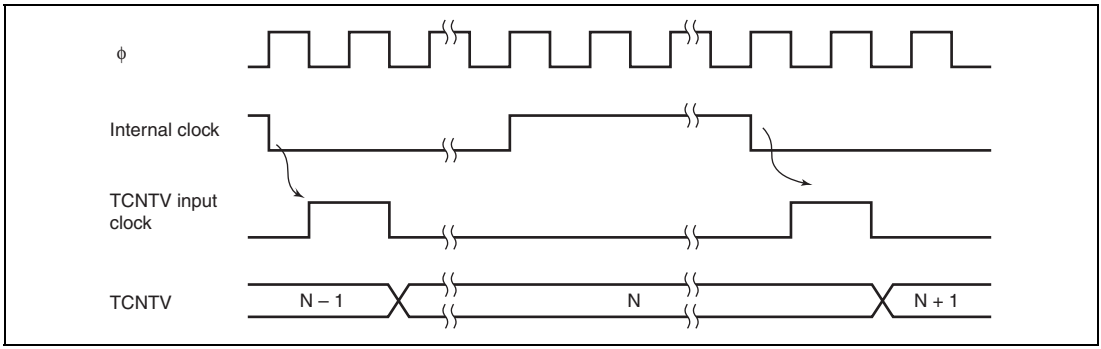


Figure 12.2 Increment Timing with Internal Clock

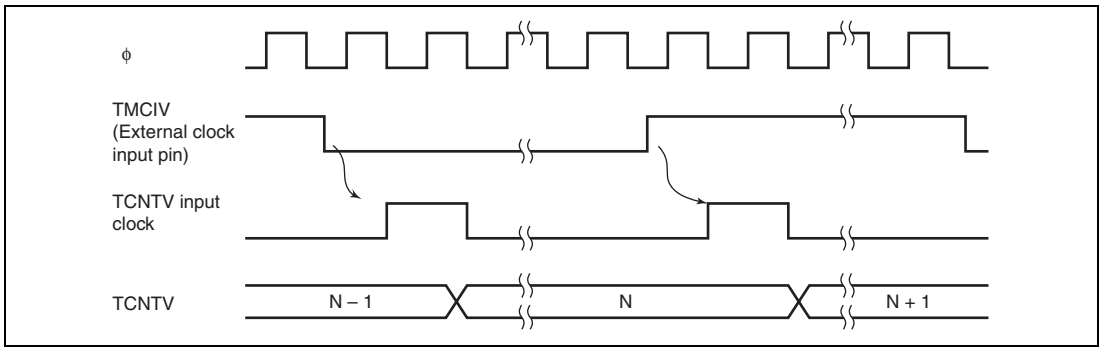


Figure 12.3 Increment Timing with External Clock

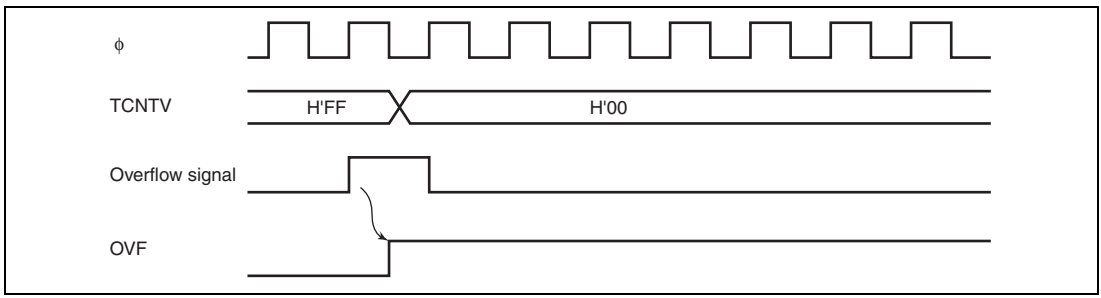


Figure 12.4 OVF Set Timing

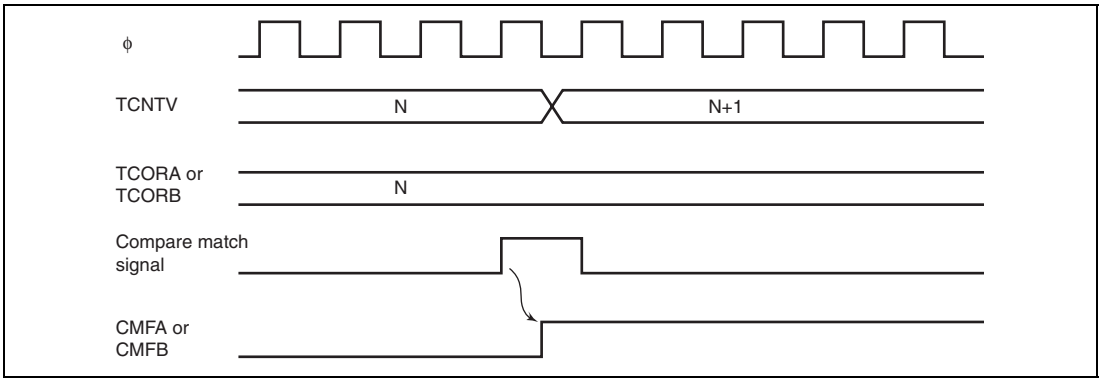


Figure 12.5 CMFA and CMFB Set Timing

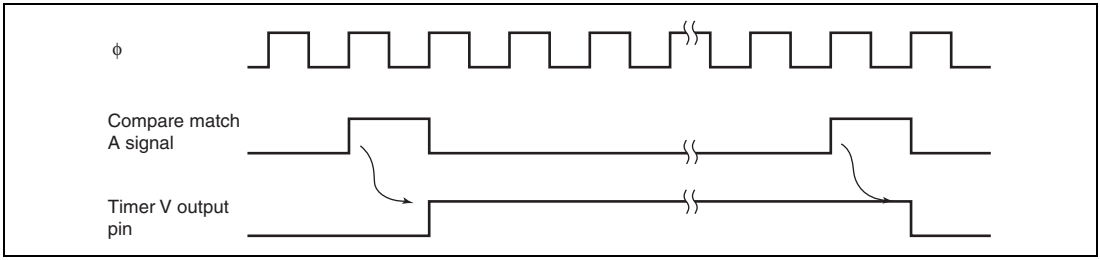


Figure 12.6 TMOV Output Timing

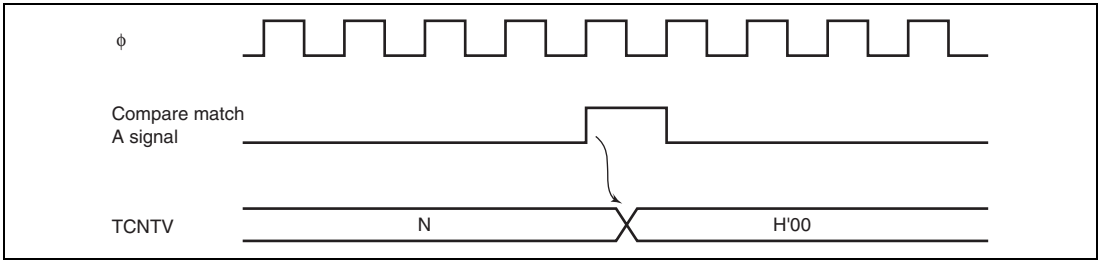


Figure 12.7 Clear Timing by Compare Match

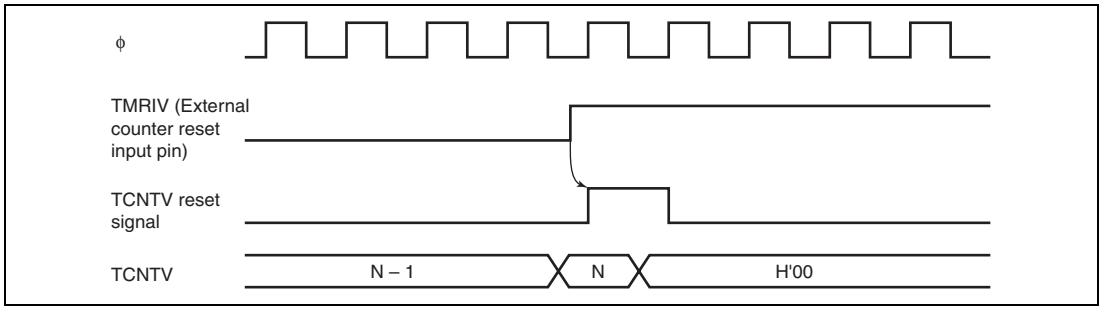


Figure 12.8 Clear Timing by TMRIV Input

12.5 Timer V Application Examples

12.5.1 Pulse Output with Arbitrary Duty Cycle

Figure 12.9 shows an example of output of pulses with an arbitrary duty cycle.

1. Set bits CCLR1 and CCLR0 in TCRV0 so that TCNTV will be cleared by compare match with TCORA.
2. Set bits OS3 to OS0 in TCSRV so that the output will go to 1 at compare match with TCORA and to 0 at compare match with TCORB.
3. Set bits CKS2 to CKS0 in TCRV0 and bit ICKS0 in TCRV1 to select the desired clock source.
4. With these settings, a waveform is output without further software intervention, with a period determined by TCORA and a pulse width determined by TCORB.

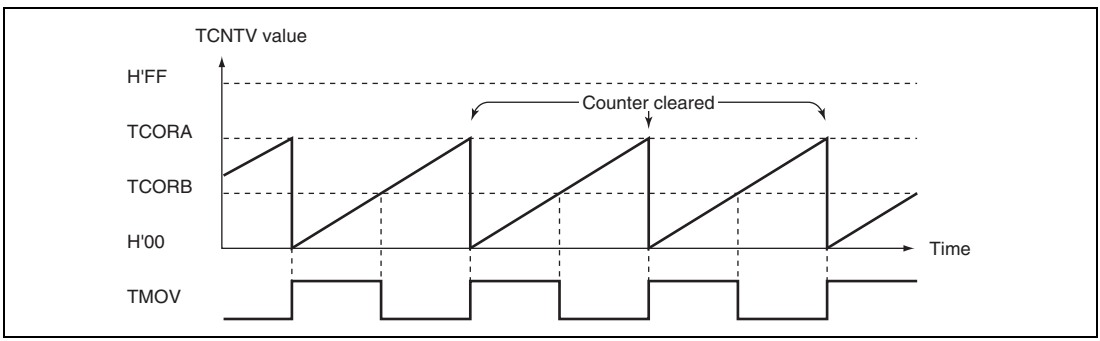


Figure 12.9 Pulse Output Example

12.5.2 Pulse Output with Arbitrary Pulse Width and Delay from TRGV Input

The trigger function can be used to output a pulse with an arbitrary pulse width at an arbitrary delay from the TRGV input, as shown in figure 12.10. To set up this output:

1. Set bits CCLR1 and CCLR0 in TCRV0 so that TCNTV will be cleared by compare match with TCORB.
2. Set bits OS3 to OS0 in TCSRV so that the output will go to 1 at compare match with TCORA and to 0 at compare match with TCORB.
3. Set bits TVEG1 and TVEG0 in TCRV1 and set TRGE to select the falling edge of the TRGV input.
4. Set bits CKS2 to CKS0 in TCRV0 and bit ICKS0 in TCRV1 to select the desired clock source.
5. After these settings, a pulse waveform will be output without further software intervention, with a delay determined by TCORA from the TRGV input, and a pulse width determined by (TCORB – TCORA).

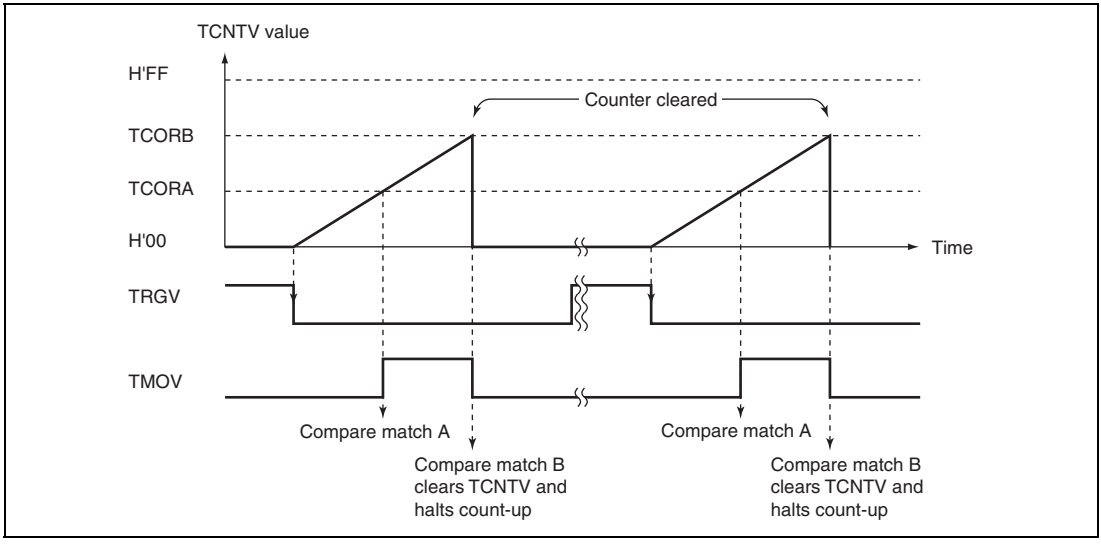


Figure 12.10 Example of Pulse Output Synchronized to TRGV Input

12.6 Usage Notes

The following types of contention or operation can occur in timer V operation.

1. Writing to registers is performed in the T3 state of a TCNTV write cycle. If a TCNTV clear signal is generated in the T3 state of a TCNTV write cycle, as shown in figure 12.11, clearing takes precedence and the write to the counter is not carried out. If counting-up is generated in the T3 state of a TCNTV write cycle, writing takes precedence.
2. If a compare match is generated in the T3 state of a TCORA or TCORB write cycle, the write to TCORA or TCORB takes precedence and the compare match signal is inhibited. Figure 12.12 shows the timing.
3. If compare matches A and B occur simultaneously, any conflict between the output selections for compare match A and compare match B is resolved by the following priority: toggle output > output 1 > output 0.
4. Depending on the timing, TCNTV may be incremented by a switch between different internal clock sources. When TCNTV is internally clocked, an increment pulse is generated from the falling edge of an internal clock signal, that is divided system clock (ϕ). Therefore, as shown in figure 12.3 the switch is from a high clock signal to a low clock signal, the switchover is seen as a falling edge, causing TCNTV to increment. TCNTV can also be incremented by a switch between internal and external clocks.

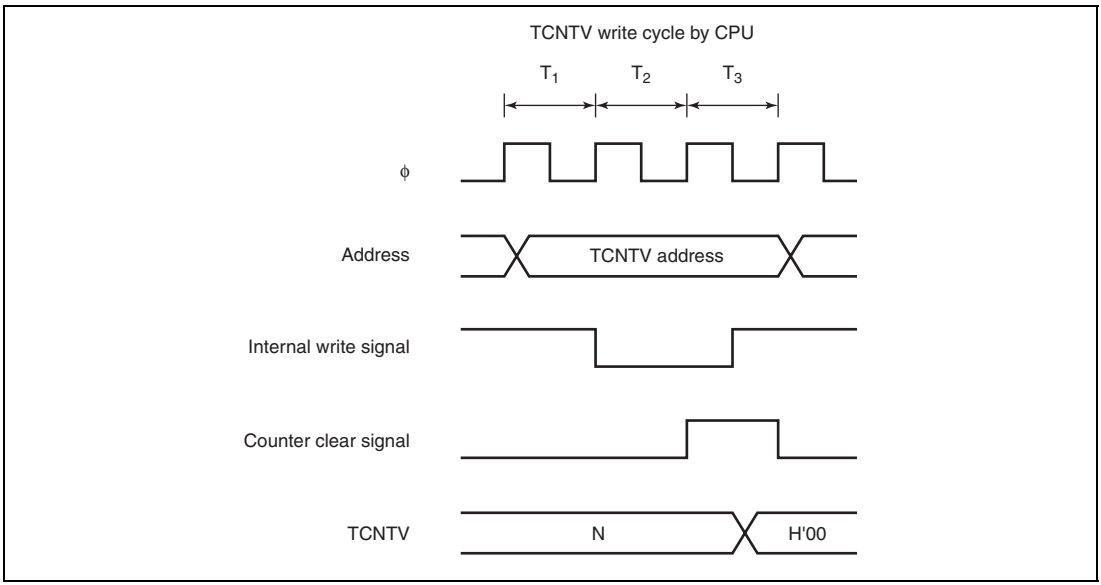


Figure 12.11 Contention between TCNTV Write and Clear

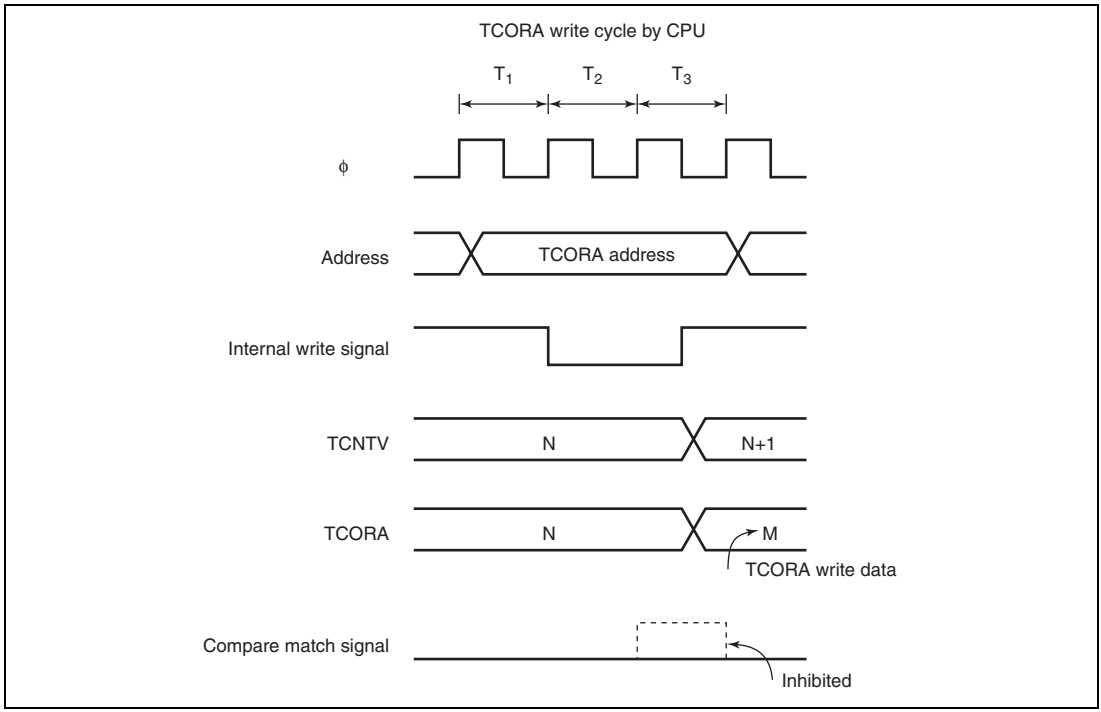


Figure 12.12 Contention between TCORA Write and Compare Match

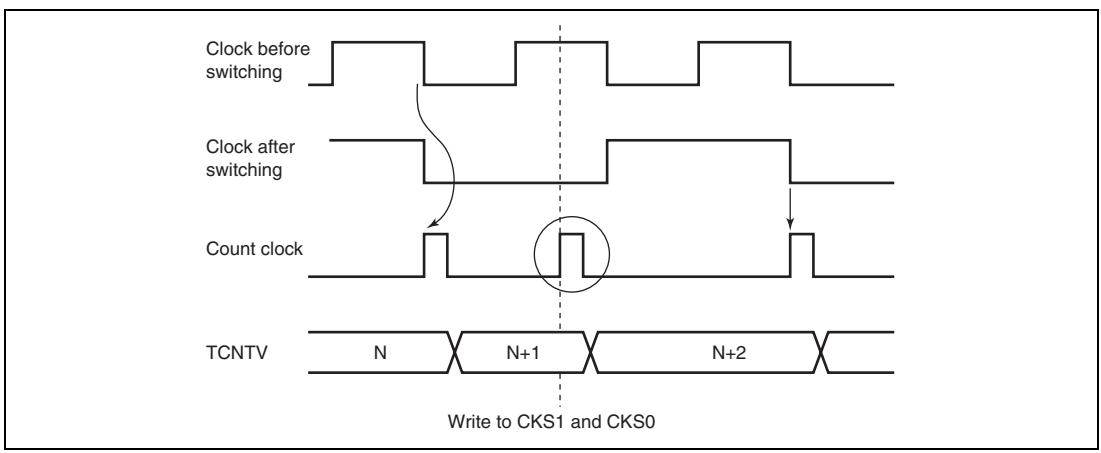


Figure 12.13 Internal Clock Switching and TCNTV Operation

Section 13 Timer Z

The timer Z has a 16-bit timer with two channels. Figures 13.1, 13.2, and 13.3 show the block diagrams of entire timer Z, its channel 0, and its channel 1, respectively. For details on the timer Z functions, refer to table 13.1.

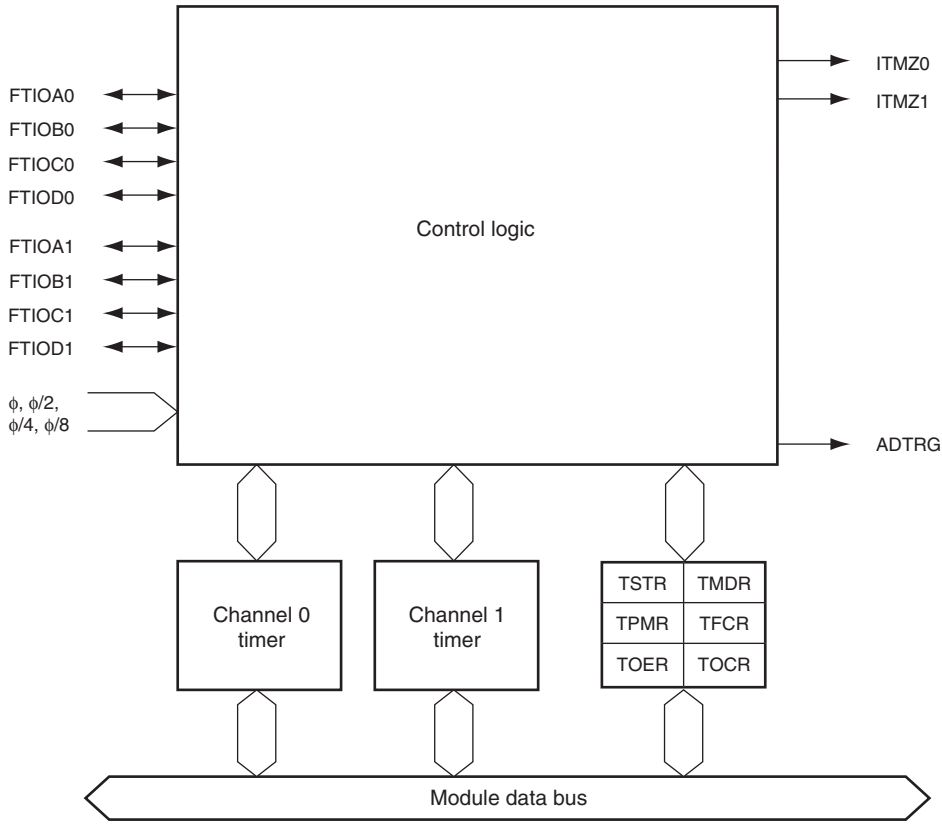
13.1 Features

- Capability to process up to eight inputs/outputs
- Eight general registers (GR): four registers for each channel
  - — Independently assignable output compare or input capture functions
- Selection of five counter clock sources: four internal clocks (ϕ, ϕ/2, ϕ/4, and ϕ/8) and an external clock
- Seven selectable operating modes
  - — Output compare function
    Selection of 0 output, 1 output, or toggle output
  - — Input capture function
    Rising edge, falling edge, or both edges
  - — Synchronous operation
    Timer counters_0 and _1 (TCNT_0 and TCNT_1) can be written simultaneously.
    Simultaneous clearing by compare match or input capture is possible.
  - — PWM mode
    Up to six-phase PWM output can be provided with desired duty ratio.
  - — Reset synchronous PWM mode
    Three-phase PWM output for normal and counter phases
  - — Complementary PWM mode
    Three-phase PWM output for non-overlapped normal and counter phases
    The A/D conversion start trigger can be set for PWM cycles.
  - — Buffer operation
    The input capture register can be consisted of double buffers.
    The output compare register can automatically be modified.
- High-speed access by the internal 16-bit bus
  - — 16-bit TCNT and GR registers can be accessed in high speed by a 16-bit bus interface
- Any initial timer output value can be set
- Output of the timer is disabled by external trigger

- Eleven interrupt sources
  - Four compare match/input capture interrupts and an overflow interrupt are available for each channel. An underflow interrupt can be set for channel 1.

Table 13.1 Timer Z Functions

Item		Channel 0	Channel 1
Count clock		Internal clocks: φ, φ/2, φ/4, φ/8 External clock: FTIOA0 (TCLK)	
General registers (output compare/input capture registers)		GRA_0, GRB_0, GRC_0, GRD_0	GRA_1, GRB_1, GRC_1, GRD_1
Buffer register		GRC_0, GRD_0	GRC_1, GRD_1
I/O pins		FTIOA0, FTIOB0, FTIOC0, FTIOD0	FTIOA1, FTIOB1, FTIOC1, FTIOD1
Counter clearing function		Compare match/input capture of GRA_0, GRB_0, GRC_0, or GRD_0	Compare match/input capture of GRA_1, GRB_1, GRC_1, or GRD_1
Compare match output	0 output	Yes	Yes
	1 output	Yes	Yes
	output	Yes	Yes
Input capture function		Yes	Yes
Synchronous operation		Yes	Yes
PWM mode		Yes	Yes
Reset synchronous PWM mode		Yes	Yes
Complementary PWM mode		Yes	Yes
Buffer function		Yes	Yes
Interrupt sources		Compare match/input capture A0 to D0 Overflow	Compare match/input capture A1 to D1 Overflow Underflow



[Legend]

TSTR : Timer start register (8 bits)

TMDR : Timer mode register (8 bits)

TPMR : Timer PWM mode register (8 bits)

TFCR : Timer function control register (8 bits)

TOER : Timer output master enable register (8 bits)

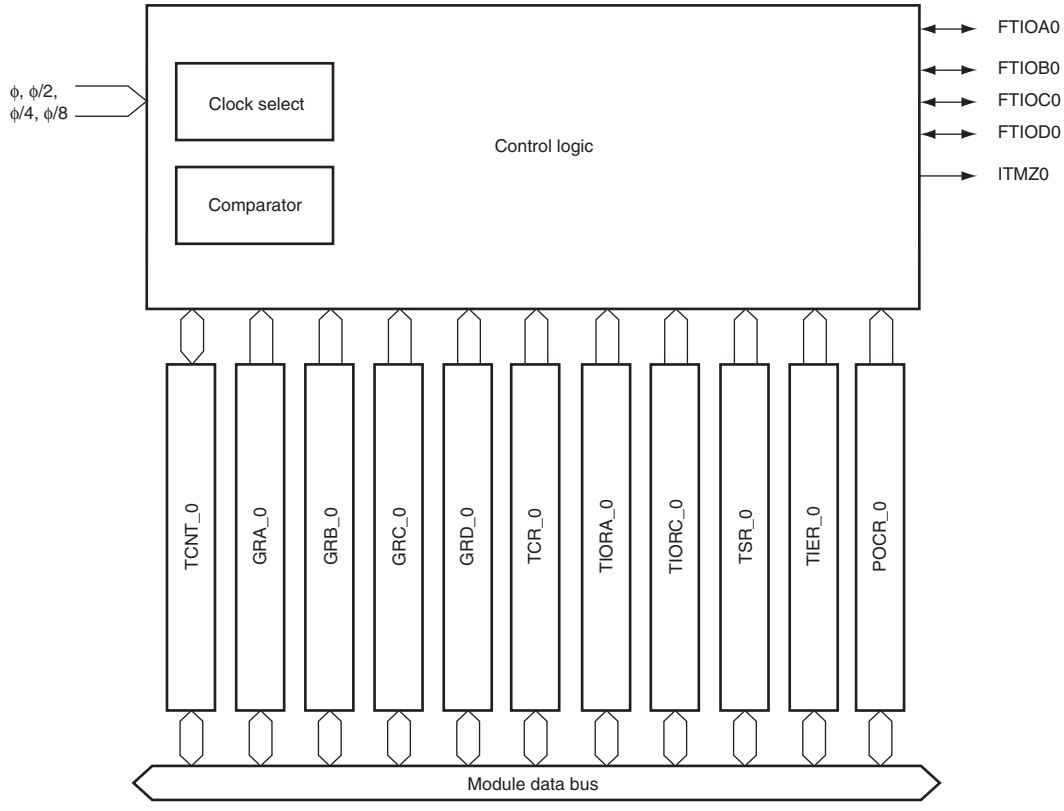
TOCR : Timer output control register (8 bits)

$\overline{\text{ADTRG}}$: A/D conversion start trigger output signal

ITMZ0 : Channel 0 interrupt

ITMZ1 : Channel 1 interrupt

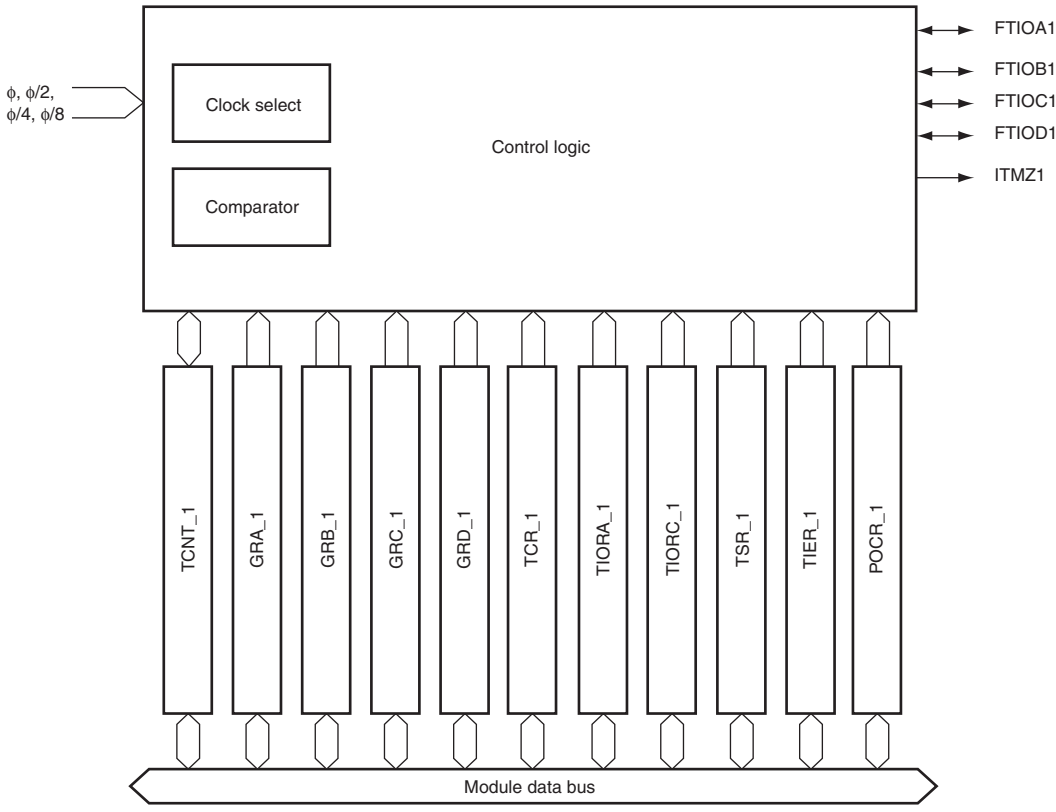
Figure 13.1 Timer Z Block Diagram



[Legend]

TCNT_0 :	Timer counter_0 (16 bits)
GRA_0, GRB_0: GRC_0, GRD_0 :	General registers A_0, B_0, C_0, and D_0 (input capture/output compare registers: 16 bits × 4)
TCR_0 :	Timer control register_0 (8 bits)
TIORA_0 :	Timer I/O control register A_0 (8 bits)
TIORC_0 :	Timer I/O control register C_0 (8 bits)
TSR_0 :	Timer status register_0 (8 bits)
TIER_0 :	Timer interrupt enable register_0 (8 bits)
POCR_0 :	PWM mode output level control register_0 (8 bits)
ITMZ0 :	Channel 0 interrupt

Figure 13.2 Timer Z (Channel 0) Block Diagram



[Legend]

TCNT_1 :	Timer counter_1 (16 bits)
GRA_1, GRB_1: GRC_1, GRD_1 :	General registers A_1, B_1, C_1, and D_1 (input capture/output compare registers: 16 bits × 4)
TCR_1 :	Timer control register_1 (8 bits)
TIORA_1 :	Timer I/O control register A_1 (8 bits)
TIORC_1 :	Timer I/O control register C_1 (8 bits)
TSR_1 :	Timer status register_1 (8 bits)
TIER_1 :	Timer interrupt enable register_1 (8 bits)
POCR_1 :	PWM mode output level control register_1 (8 bits)
ITMZ1 :	Channel 1 interrupt

Figure 13.3 Timer Z (Channel 1) Block Diagram

13.2 Input/Output Pins

Table 13.2 summarizes the timer Z pins.

Table 13.2 Pin Configuration

Name	Abbreviation	Input/Output	Function
Input capture/output compare A0	FTIOA0	Input/output	GRA_0 output compare output, GRA_0 input capture input, or external clock input (TCLK)
Input capture/output compare B0	FTIOB0	Input/output	GRB_0 output compare output, GRB_0 input capture input, or PWM output
Input capture/output compare C0	FTIOC0	Input/output	GRC_0 output compare output, GRC_0 input capture input, or PWM synchronous output (in reset synchronous PWM and complementary PWM modes)
Input capture/output compare D0	FTIOD0	Input/output	GRD_0 output compare output, GRD_0 input capture input, or PWM output
Input capture/output compare A1	FTIOA1	Input/output	GRA_1 output compare output, GRA_1 input capture input, or PWM output (in reset synchronous PWM and complementary PWM modes)
Input capture/output compare B1	FTIOB1	Input/output	GRB_1 output compare output, GRB_1 input capture input, or PWM output
Input capture/output compare C1	FTIOC1	Input/output	GRC_1 output compare output, GRC_1 input capture input, or PWM output
Input capture/output compare D1	FTIOD1	Input/output	GRD_1 output compare output, GRD_1 input capture input, or PWM output

13.3 Register Descriptions

The timer Z has the following registers.

Common

- Timer start register (TSTR)
- Timer mode register (TMDR)
- Timer PWM mode register (TPMR)
- Timer function control register (TFCR)
- Timer output master enable register (TOER)
- Timer output control register (TOCR)

Channel 0

- Timer control register_0 (TCR_0)
- Timer I/O control register A_0 (TIORA_0)
- Timer I/O control register C_0 (TIORC_0)
- Timer status register_0 (TSR_0)
- Timer interrupt enable register_0 (TIER_0)
- PWM mode output level control register_0 (POCR_0)
- Timer counter_0 (TCNT_0)
- General register A_0 (GRA_0)
- General register B_0 (GRB_0)
- General register C_0 (GRC_0)
- General register D_0 (GRD_0)

Channel 1

- Timer control register_1 (TCR_1)
- Timer I/O control register A_1 (TIORA_1)
- Timer I/O control register C_1 (TIORC_1)
- Timer status register_1 (TSR_1)
- Timer interrupt enable register_1 (TIER_1)
- PWM mode output level control register_1 (POCR_1)
- Timer counter_1 (TCNT_1)
- General register A_1 (GRA_1)
- General register B_1 (GRB_1)

- General register C_1 (GRC_1)
- General register D_1 (GRD_1)

13.3.1 Timer Start Register (TSTR)

TSTR selects the operation/stop for the TCNT counter.

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
1	STR1	0	R/W	Channel 1 Counter Start 0: TCNT_1 halts counting 1: TCNT_1 starts counting
0	STR0	0	R/W	Channel 0 Counter Start 0: TCNT_0 halts counting 1: TCNT_0 starts counting

13.3.2 Timer Mode Register (TMDR)

TMDR selects buffer operation settings and synchronized operation.

Bit	Bit Name	Initial Value	R/W	Description
7	BFD1	0	R/W	Buffer Operation D1 0: GRD_1 operates normally 1: GRB_1 and GRD_1 are used together for buffer operation
6	BFC1	0	R/W	Buffer Operation C1 0: GRC_1 operates normally 1: GRA_1 and GRD_1 are used together for buffer operation
5	BFD0	0	R/W	Buffer Operation D0 0: GRD_0 operates normally 1: GRB_0 and GRD_0 are used together for buffer operation
4	BFC0	0	R/W	Buffer Operation C0 0: GRC_0 operates normally 1: GRA_0 and GRC_0 are used together for buffer operation
3 to 1	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
0	SYNC	0	R/W	Timer Synchronization 0: TCNT_1 and TCNT_0 operate as a different timer 1: TCNT_1 and TCNT_0 are synchronized TCNT_1 and TCNT_0 can be pre-set or cleared synchronously

13.3.3 Timer PWM Mode Register (TPMR)

TPMR sets the pin to enter PWM mode.

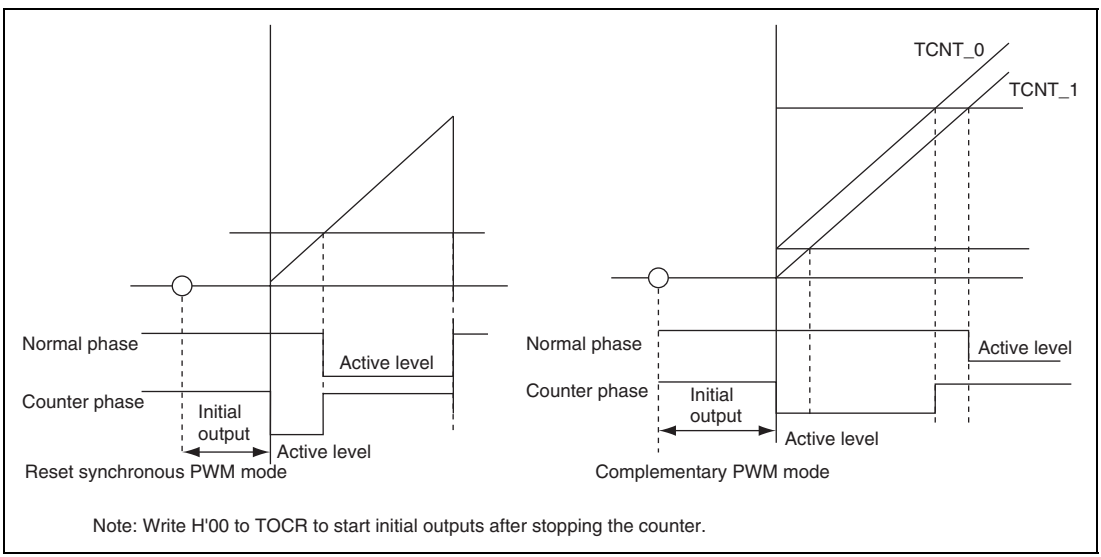
Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1, and cannot be modified.
6	PWMD1	0	R/W	PWM Mode D1 0: FTIOD1 operates normally 1: FTIOD1 operates in PWM mode
5	PWMC1	0	R/W	PWM Mode C1 0: FTIOC1 operates normally 1: FTIOC1 operates in PWM mode
4	PWMB1	0	R/W	PWM Mode B1 0: FTIOB1 operates normally 1: FTIOB1 operates in PWM mode
3	—	1	—	Reserved This bit is always read as 1, and cannot be modified.
2	PWMD0	0	R/W	PWM Mode D0 0: FTIOD0 operates normally 1: FTIOD0 operates in PWM mode
1	PWMC0	0	R/W	PWM Mode C0 0: FTIOC0 operates normally 1: FTIOC0 operates in PWM mode
0	PWMB0	0	R/W	PWM Mode B0 0: FTIOB0 operates normally 1: FTIOB0 operates in PWM mode

13.3.4 Timer Function Control Register (TFCR)

TFCR selects the settings and output levels for each operating mode.

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	STCLK	0	R/W	External Clock Input Select 0: External clock input is disabled 1: External clock input is enabled
5	ADEG	0	R/W	A/D Trigger Edge Select A/D module should be set to start an A/D conversion by the external trigger 0: A/D trigger at the crest in complementary PWM mode 1: A/D trigger at the trough in complementary PWM mode
4	ADTRG	0	R/W	External Trigger Disable 0: A/D trigger for PWM cycles is disabled in complementary PWM mode 1: A/D trigger for PWM cycles is enabled in complementary PWM mode
3	OLS1	0	R/W	Output Level Select 1 Selects the counter-phase output levels in reset synchronous PWM mode or complementary PWM mode. 0: Initial output is high and the active level is low. 1: Initial output is low and the active level is high.
2	OLS0	0	R/W	Output Level Select 0 Selects the normal-phase output levels in reset synchronous PWM mode or complementary PWM mode. 0: Initial output is high and the active level is low. 1: Initial output is low and the active level is high. Figure 13.4 shows an example of outputs in reset synchronous PWM mode and complementary PWM mode when OLS1 = 0 and OLS0 = 0.

Bit	Bit Name	Initial Value	R/W	Description
1	CMD1	0	R/W	Combination Mode 1 and 0
0	CMD0	0	R/W	00: Channel 0 and channel 1 operate normally 01: Channel 0 and channel 1 are used together to operate in reset synchronous PWM mode 10: Channel 0 and channel 1 are used together to operate in complementary PWM mode (transferred at the trough) 11: Channel 0 and channel 1 are used together to operate in complementary PWM mode (transferred at the crest) Note: When reset synchronous PWM mode or complementary PWM mode is selected by these bits, this setting has the priority to the settings for PWM mode by each bit in TPMR. Stop TCNT_0 and TCNT_1 before making settings for reset synchronous PWM mode or complementary PWM mode.



Note: Write H'00 to TOCR to start initial outputs after stopping the counter.

Figure 13.4 Example of Outputs in Reset Synchronous PWM Mode and Complementary PWM Mode

13.3.5 Timer Output Master Enable Register (TOER)

TOER enables/disables the outputs for channel 0 and channel 1. When $\overline{\text{WKP4}}$ is selected for inputs, if a low level signal is input to $\overline{\text{WKP4}}$, the bits in TOER are set to 1 to disable the output for timer Z.

Bit	Bit Name	Initial Value	R/W	Description
7	ED1	1	R/W	Master Enable D1 0: FTIOD1 pin output is enabled according to the TPMR, TFCR, and TIORC_1 settings 1: FTIOD1 pin output is disabled regardless of the TPMR, TFCR, and TIORC_1 settings (FTIOD1 pin is operated as an I/O port).
6	EC1	1	R/W	Master Enable C1 0: FTIOC1 pin output is enabled according to the TPMR, TFCR, and TIORC_1 settings 1: FTIOC1 pin output is disabled regardless of the TPMR, TFCR, and TIORC_1 settings (FTIOC1 pin is operated as an I/O port).
5	EB1	1	R/W	Master Enable B1 0: FTIOB1 pin output is enabled according to the TPMR, TFCR, and TIORA_1 settings 1: FTIOB1 pin output is disabled regardless of the TPMR, TFCR, and TIORA_1 settings (FTIOB1 pin is operated as an I/O port).
4	EA1	1	R/W	Master Enable A1 0: FTIOA1 pin output is enabled according to the TPMR, TFCR, and TIORA_1 settings 1: FTIOA1 pin output is disabled regardless of the TPMR, TFCR, and TIORA_1 settings (FTIOA1 pin is operated as an I/O port).
3	ED0	1	R/W	Master Enable D0 0: FTIOD0 pin output is enabled according to the TPMR, TFCR, and TIORC_0 settings 1: FTIOD0 pin output is disabled regardless of the TPMR, TFCR, and TIORC_0 settings (FTIOD0 pin is operated as an I/O port).

Bit	Bit Name	Initial Value	R/W	Description
2	EC0	1	R/W	Master Enable C0 0: FTIOC0 pin output is enabled according to the TPMR, TFCR, and TIORC_0 settings 1: FTIOC0 pin output is disabled regardless of the TPMR, TFCR, and TIORC_0 settings (FTIOC0 pin is operated as an I/O port).
1	EB0	1	R/W	Master Enable B0 0: FTIOB0 pin output is enabled according to the TPMR, TFCR, and TIORA_0 settings 1: FTIOB0 pin output is disabled regardless of the TPMR, TFCR, and TIORA_0 settings (FTIOB0 pin is operated as an I/O port).
0	EA0	1	R/W	Master Enable A0 0: FTIOA0 pin output is enabled according to the TPMR, TFCR, and TIORA_0 settings 1: FTIOA0 pin output is disabled regardless of the TPMR, TFCR, and TIORA_0 settings (FTIOA0 pin is operated as an I/O port).

13.3.6 Timer Output Control Register (TOCR)

TOCR selects the initial outputs before the first occurrence of a compare match. Note that bits OLS1 and OLS0 in TFCR set these initial outputs in reset synchronous PWM mode and complementary PWM mode.

Bit	Bit Name	Initial Value	R/W	Description
7	TOD1	0	R/W	Output Level Select D1 0: 0 output at the FTIOD1 pin* 1: 1 output at the FTIOD1 pin*
6	TOC1	0	R/W	Output Level Select C1 0: 0 output at the FTIOC1 pin* 1: 1 output at the FTIOC1 pin*
5	TOB1	0	R/W	Output Level Select B1 0: 0 output at the FTIOB1 pin* 1: 1 output at the FTIOB1 pin*

Bit	Bit Name	Initial Value	R/W	Description
4	TOA1	0	R/W	Output Level Select A1 0: 0 output at the FTIOA1 pin* 1: 1 output at the FTIOA1 pin*
3	TOD0	0	R/W	Output Level Select D0 0: 0 output at the FTIOD0 pin* 1: 1 output at the FTIOD0 pin*
2	TOC0	0	R/W	Output Level Select C0 0: 0 output at the FTIOC0 pin* 1: 1 output at the FTIOC0 pin*
1	TOB0	0	R/W	Output Level Select B0 0: 0 output at the FTIOB0 pin* 1: 1 output at the FTIOB0 pin*
0	TOA0	0	R/W	Output Level Select A0 0: 0 output at the FTIOA0 pin* 1: 1 output at the FTIOA0 pin*

Note: * The change of the setting is immediately reflected in the output value.

13.3.7 Timer Counter (TCNT)

The timer Z has two TCNT counters (TCNT_0 and TCNT_1), one for each channel. The TCNT counters are 16-bit readable/writable registers that increment/decrement according to input clocks. Input clocks can be selected by bits TPSC2 to TPSC0 in TCR. TCNT0 and TCNT 1 increment/decrement in complementary PWM mode, while they only increment in other modes.

The TCNT counters are initialized to H'0000 by compare matches with corresponding GRA, GRB, GRC, or GRD, or input captures to GRA, GRB, GRC, or GRD (counter clearing function). When the TCNT counters overflow, an OVF flag in TSR for the corresponding channel is set to 1. When TCNT_1 underflows, an UDF flag in TSR is set to 1. The TCNT counters cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit.

13.3.8 General Registers A, B, C, and D (GRA, GRB, GRC, and GRD)

GR are 16-bit registers. Timer Z has eight general registers (GR), four for each channel. The GR registers are dual function 16-bit readable/writable registers, functioning as either output compare or input capture registers. Functions can be switched by TIORA and TIORC.

The values in GR and TCNT are constantly compared with each other when the GR registers are used as output compare registers. When the both values match, the IMFA to IMFD flags in TSR are set to 1. Compare match outputs can be selected by TIORA and TIORC.

When the GR registers are used as input capture registers, the TCNT value is stored after detecting external signals. At this point, IMFA to IMFD flags in the corresponding TSR are set to 1. Detection edges for input capture signals can be selected by TIORA and TIORC.

When PWM mode, complementary PWM mode, or reset synchronous PWM mode is selected, the values in TIORA and TIORC are ignored. Upon reset, the GR registers are set as output compare registers (no output) and initialized to H'FFFF. The GR registers cannot be accessed in 8-bit units; they must always be accessed as a 16-bit unit.

13.3.9 Timer Control Register (TCR)

The TCR registers select a TCNT counter clock, an edge when an external clock is selected, and counter clearing sources. Timer Z has a total of two TCR registers, one for each channel.

Bit	Bit Name	Initial value	R/W	Description
7	CCLR2	0	R/W	Counter Clear 2 to 0
6	CCLR1	0	R/W	000: Disables TCNT clearing
5	CCLR0	0	R/W	001: Clears TCNT by GRA compare match/input capture*[1]
				010: Clears TCNT by GRB compare match/input capture*[1]
				011: Synchronization clear; Clears TCNT in synchronous with counter clearing of the other channel's timer*[2]
				100: Disables TCNT clearing
				101: Clears TCNT by GRC compare match/input capture*[1]
				110: Clears TCNT by GRD compare match/input capture*[1]
				111: Synchronization clear; Clears TCNT in synchronous with counter clearing of the other channel's timer*[2]
4	CKEG1	0	R/W	Clock Edge 1 and 0
3	CKEG0	0	R/W	00: Count at rising edge
				01: Count at falling edge
				1X: Count at both edges
2	TPSC2	0	R/W	Time Prescaler 2 to 0
1	TPSC1	0	R/W	000: Internal clock: count by ϕ
0	TPSC0	0	R/W	001: Internal clock: count by $\phi/2$
				010: Internal clock: count by $\phi/4$
				011: Internal clock: count by $\phi/8$
				1XX: External clock: count by FTIOA0 (TCLK) pin input

Notes: 1. When GR functions as an output compare register, TCNT is cleared by compare match. When GR functions as input capture, TCNT is cleared by input capture.
2. Synchronous operation is set by TMDR.
3. X: Don't care

13.3.10 Timer I/O Control Register (TIORA and TIORC)

The TIOR registers control the general registers (GR). Timer Z has four TIOR registers (TIORA_0, TIORA_1, TIORC_0, and TIORC_1), two for each channel. In PWM mode including complementary PWM mode and reset synchronous PWM mode, the settings of TIOR are invalid.

TIORA: TIORA selects whether GRA or GRB is used as an output compare register or an input capture register. When an output compare register is selected, the output setting is selected. When an input capture register is selected, an input edge of an input capture signal is selected. TIORA also selects the function of FTIOA or FTIOB pin.

Bit	Bit Name	Initial value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	IOB2	0	R/W	I/O Control B2 to B0
5	IOB1	0	R/W	GRB is an output compare register:
4	IOB0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRB compare match 010: 1 output by GRB compare match 011: Toggle output by GRB compare match GRB is an input capture register: 100: Input capture to GRB at the rising edge 101: Input capture to GRB at the falling edge 11X: Input capture to GRB at both rising and falling edges
3	—	1	—	Reserved This bit is always read as 1.
2	IOA2	0	R/W	I/O Control A2 to A0
1	IOA1	0	R/W	GRA is an output compare register:
0	IOA0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRA compare match 010: 1 output by GRA compare match 011: Toggle output by GRA compare match GRA is an input capture register: 100: Input capture to GRA at the rising edge 101: Input capture to GRA at the falling edge 11X: Input capture to GRA at both rising and falling edges

Legend: X: Don't care

TIORC: TIORC selects whether GRC or GRD is used as an output compare register or an input capture register. When an output compare register is selected, the output setting is selected. When an input capture register is selected, an input edge of an input capture signal is selected. TIORC also selects the function of FTIOC or FTIOD pin.

Bit	Bit Name	Initial value	R/W	Description
7	—	1	—	Reserved This bit is always read as 1.
6	IOD2	0	R/W	I/O Control D2 to D0
5	IOD1	0	R/W	GRD is an output compare register:
4	IOD0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRD compare match 010: 1 output by GRD compare match 011: Toggle output by GRD compare match GRD is an input capture register: 100: Input capture to GRD at the rising edge 101: Input capture to GRD at the falling edge 11X: Input capture to GRD at both rising and falling edges
3	—	1	—	Reserved This bit is always read as 1.
2	IOC2	0	R/W	I/O Control C2 to C0
1	IOC1	0	R/W	GRC is an output compare register:
0	IOC0	0	R/W	000: Disables pin output by compare match 001: 0 output by GRC compare match 010: 1 output by GRC compare match 011: Toggle Output by GRC compare match GRC is an input capture register: 100: Input capture to GRC at the rising edge 101: Input capture to GRC at the falling edge 11X: Input capture to GRC at both rising and falling edges

Legend: X: Don't care

13.3.11 Timer Status Register (TSR)

TSR indicates generation of an overflow/underflow of TCNT and a compare match/input capture of GRA, GRB, GRC, and GRD. These flags are interrupt sources. If an interrupt is enabled by a corresponding bit in TIER, TSR requests an interrupt for the CPU. Timer Z has two TSR registers, one for each channel.

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1.
5	UDF*	0	R/W	Underflow Flag [Setting condition] • When TCNT_1 underflows [Clearing condition] • When 0 is written to UDF after reading UDF = 1
4	OVF	0	R/W	Overflow Flag [Setting condition] • When the TCNT value underflows [Clearing condition] • When 0 is written to OVF after reading OVF = 1
3	IMFD	0	R/W	Input Capture/Compare Match Flag D [Setting conditions] • When TCNT = GRD and GRD is functioning as output compare register • When TCNT value is transferred to GRD by input capture signal and GRD is functioning as input capture register [Clearing condition] • When 0 is written to IMFD after reading IMFD = 1

Bit	Bit Name	Initial value	R/W	Description
2	IMFC	0	R/W	Input Capture/Compare Match Flag C [Setting conditions] • When TCNT = GRC and GRC is functioning as output compare register • When TCNT value is transferred to GRC by input capture signal and GRC is functioning as input capture register [Clearing condition] • When 0 is written to IMFC after reading IMFC = 1
1	IMFB	0	R/W	Input Capture/Compare Match Flag B [Setting conditions] • When TCNT = GRB and GRB is functioning as output compare register • When TCNT value is transferred to GRB by input capture signal and GRB is functioning as input capture register [Clearing condition] • When 0 is written to IMFB after reading IMFB = 1
0	IMFA	0	R/W	Input Capture/Compare Match Flag A [Setting conditions] • When TCNT = GRA and GRA is functioning as output compare register • When TCNT value is transferred to GRA by input capture signal and GRA is functioning as input capture register [Clearing condition] • When 0 is written to IMFA after reading IMFA = 1

Note: Bit 5 is not the UDF flag in TSR_0. It is a reserved bit. It is always read as 1.

13.3.12 Timer Interrupt Enable Register (TIER)

TIER enables or disables interrupt requests for overflow or GR compare match/input capture. Timer Z has two TIER registers, one for each channel.

Bit	Bit Name	Initial value	R/W	Description
7 to 5	—	All 1	—	Reserved These bits are always read as 1.
4	OVIE	0	R/W	Overflow Interrupt Enable 0: Interrupt requests (OVI) by OVF or UDF flag are disabled 1: Interrupt requests (OVI) by OVF or UDF flag are enabled
3	IMIED	0	R/W	Input Capture/Compare Match Interrupt Enable D 0: Interrupt requests (IMID) by IMFD flag are disabled 1: Interrupt requests (IMID) by IMFD flag are enabled
2	IMIEC	0	R/W	Input Capture/Compare Match Interrupt Enable C 0: Interrupt requests (IMIC) by IMFC flag are disabled 1: Interrupt requests (IMIC) by IMFC flag are enabled
1	IMIEB	0	R/W	Input Capture/Compare Match Interrupt Enable B 0: Interrupt requests (IMIB) by IMFB flag are disabled 1: Interrupt requests (IMIB) by IMFB flag are enabled
0	IMIEA	0	R/W	Input Capture/Compare Match Interrupt Enable A 0: Interrupt requests (IMIA) by IMFA flag are disabled 1: Interrupt requests (IMIA) by IMFA flag are enabled

13.3.13 PWM Mode Output Level Control Register (POCR)

POCR control the active level in PWM mode. Timer Z has two POCR registers, one for each channel.

Bit	Bit Name	Initial value	R/W	Description
7 to 3	—	All 1	—	Reserved These bits are always read as 1.
2	POLD	0	R/W	PWM Mode Output Level Control D 0: The output level of FTIOD is low-active 1: The output level of FTIOD is high-active
1	POLC	0	R/W	PWM Mode Output Level Control C 0: The output level of FTIOC is low-active 1: The output level of FTIOC is high-active
0	POLB	0	R/W	PWM Mode Output Level Control B 0: The output level of FTIOB is low-active 1: The output level of FTIOB is high-active

13.3.14 Interface with CPU

1. 16-bit register

   TCNT and GR are 16-bit registers. Reading/writing in a 16-bit unit is enabled but disabled in an 8-bit unit since the data bus with the CPU is 16-bit width. These registers must always be accessed in a 16-bit unit. Figure 13.5 shows an example of accessing the 16-bit registers.

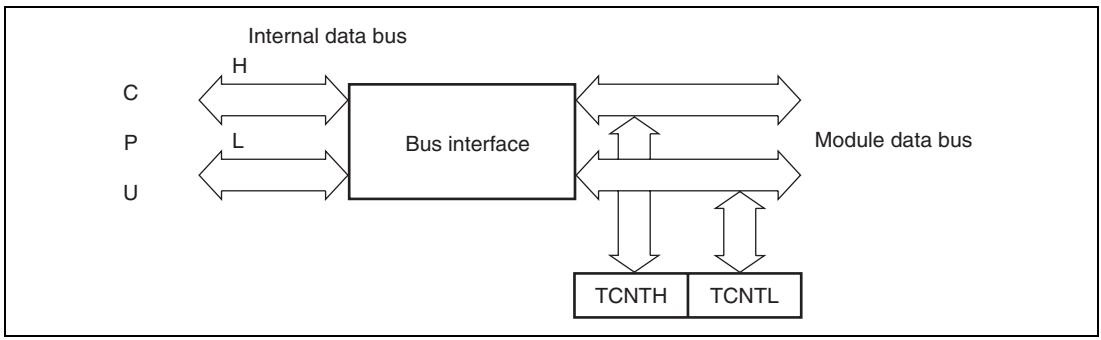


Figure 13.5 Accessing Operation of 16-Bit Register (between CPU and TCNT (16 bits))

2. 8-bit register

Registers other than TCNT and GR are 8-bit registers that are connected internally with the CPU in an 8-bit width. Figure 13.6 shows an example of accessing the 8-bit registers.

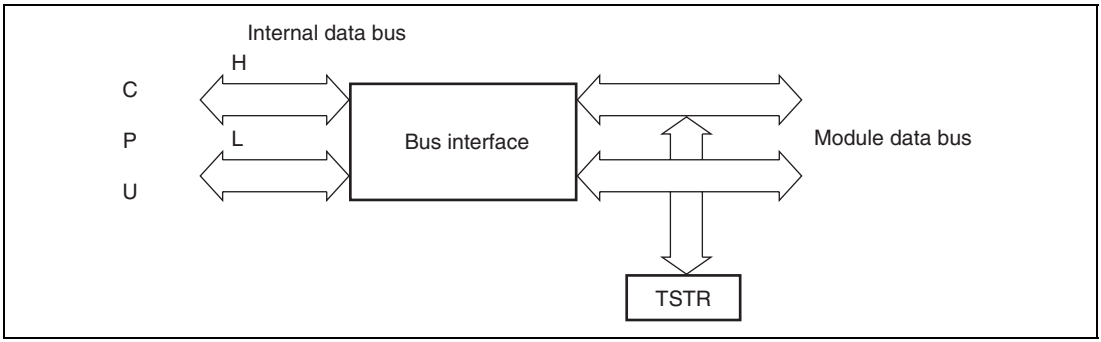


Figure 13.6 Accessing Operation of 8-Bit Register (between CPU and TSTR (8 bits))

13.4 Operation

13.4.1 Counter Operation

When one of bits STR0 and STR1 in TSTR is set to 1, the TCNT counter for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example. Figure 13.7 shows an example of the counter operation setting procedure.

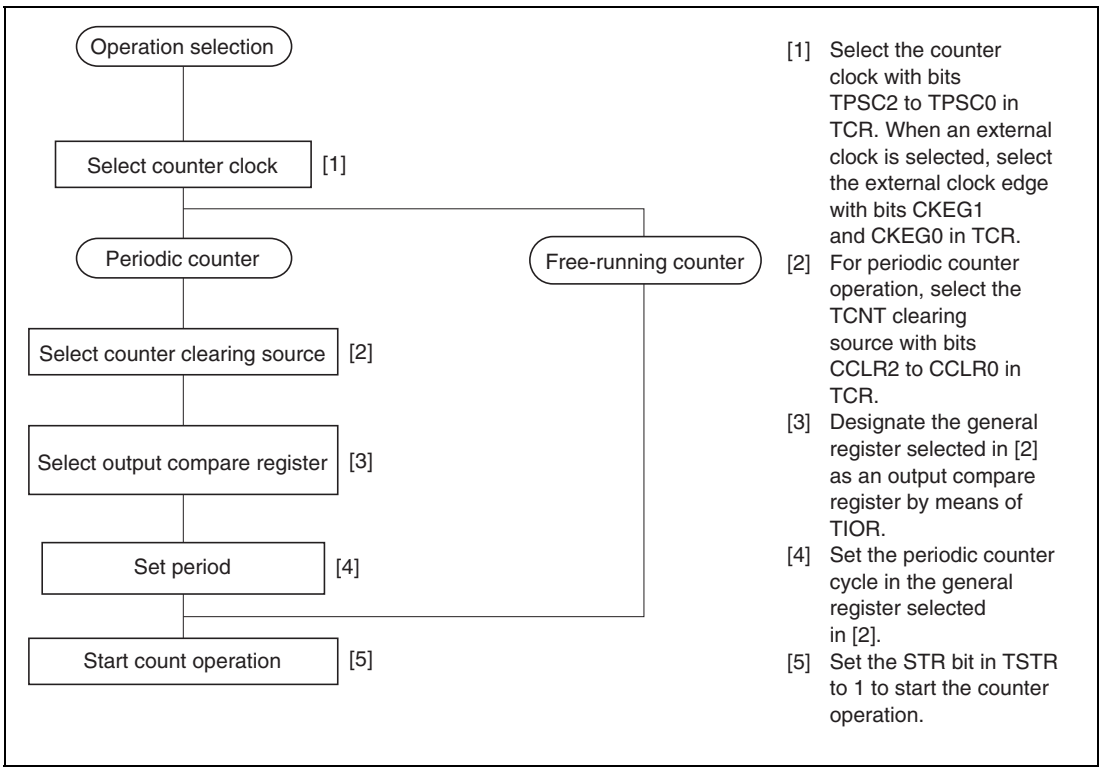


Figure 13.7 Example of Counter Operation Setting Procedure

1. Free-running count operation and periodic count operation

   Immediately after a reset, the TCNT counters for channels 0 and 1 are all designated as free-running counters. When the relevant bit in TSTR is set to 1, the corresponding TCNT counter starts an increment operation as a free-running counter. When TCNT overflows, the OVF flag in TSR is set to 1. If the value of the OVIE bit in the corresponding TIER is 1 at this point, timer Z requests an interrupt. After overflow, TCNT starts an increment operation again from H'0000.

Figure 13.8 illustrates free-running counter operation.

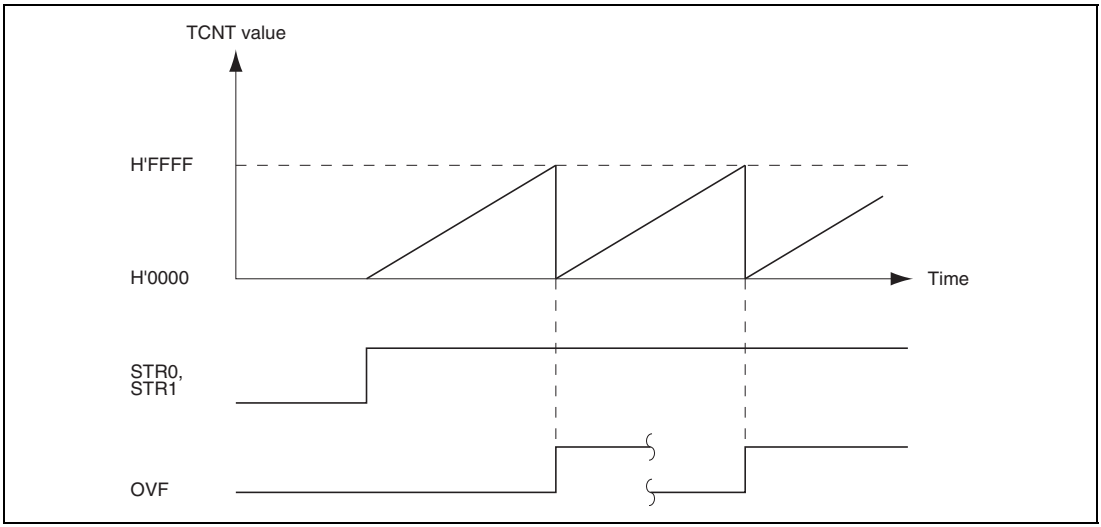


Figure 13.8 Free-Running Counter Operation

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The GR registers for setting the period are designated as output compare registers, and counter clearing by compare match is selected by means of bits CCLR1 and CCLR0 in TCR. After the settings have been made, TCNT starts an increment operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the count value matches the value in GR, the IMFA, IMFB, IMFC, or IMFD flag in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding IMIEA, IMIEB, IMIEC, or IMIED bit in TIER is 1 at this point, the timer Z requests an interrupt. After a compare match, TCNT starts an increment operation again from H'0000.

Figure 13.9 illustrates periodic counter operation.

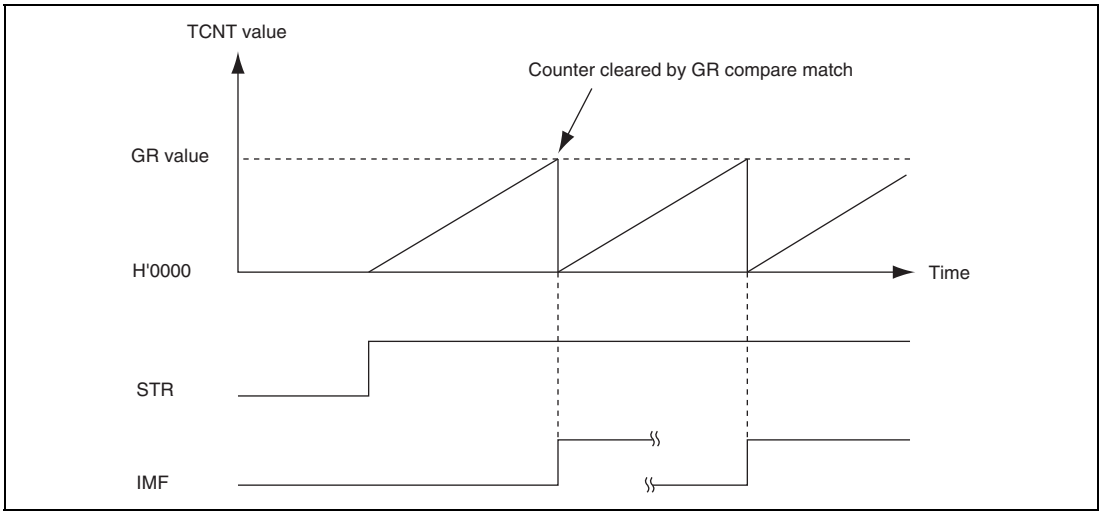


Figure 13.9 Periodic Counter Operation

2. TCNT count timing

A. Internal clock operation

A system clock (ϕ) or three types of clocks (ϕ/2, ϕ/4, or ϕ/8) that divides the system clock can be selected by bits TPSC2 to TPSC0 in TCR.

Figure 13.10 illustrates this timing.

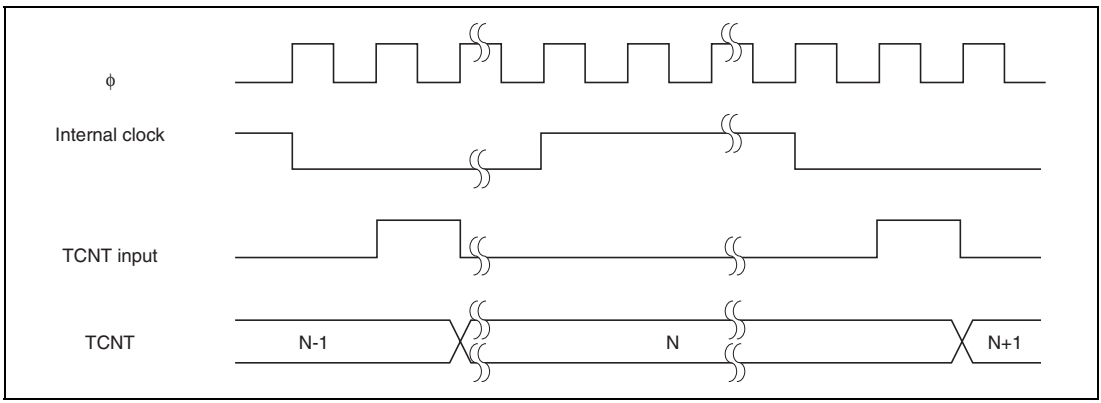


Figure 13.10 Count Timing at Internal Clock Operation

B. External clock operation

An external clock input pin (TCLK) can be selected by bits TPSC2 to TPSC0 in TCR, and a detection edge can be selected by bits CKEG1 and CKEG0. To detect an external clock, the rising edge, falling edge, or both edges can be selected. The pulse width of the external clock needs two or more system clocks. Note that an external clock does not operate correctly with the lower pulse width.

Figure 13.11 illustrates the detection timing of the rising and falling edges.

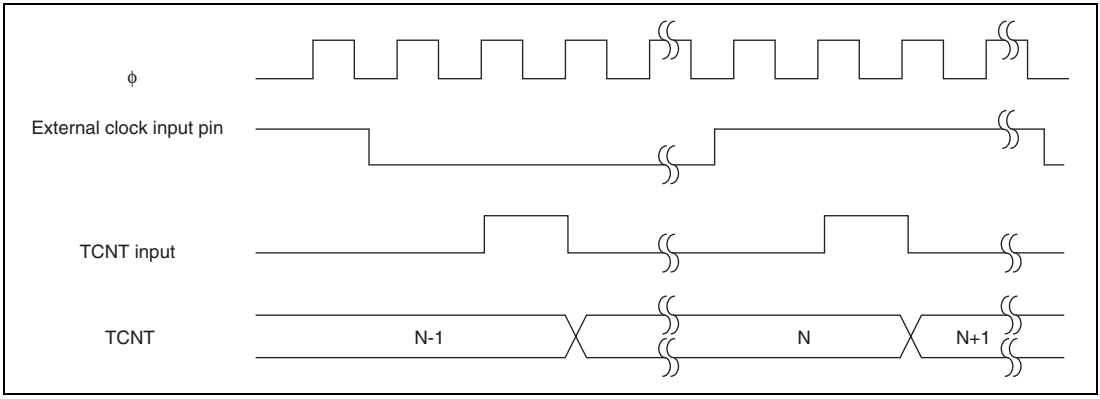


Figure 13.11 Count Timing at External Clock Operation (Both Edges Detected)

13.4.2 Waveform Output by Compare Match

Timer Z can perform 0, 1, or toggle output from the corresponding FTIOA, FTIOB, FTIOC, or FTIOD output pin using compare match A, B, C, or D.

Figure 13.12 shows an example of the setting procedure for waveform output by compare match.

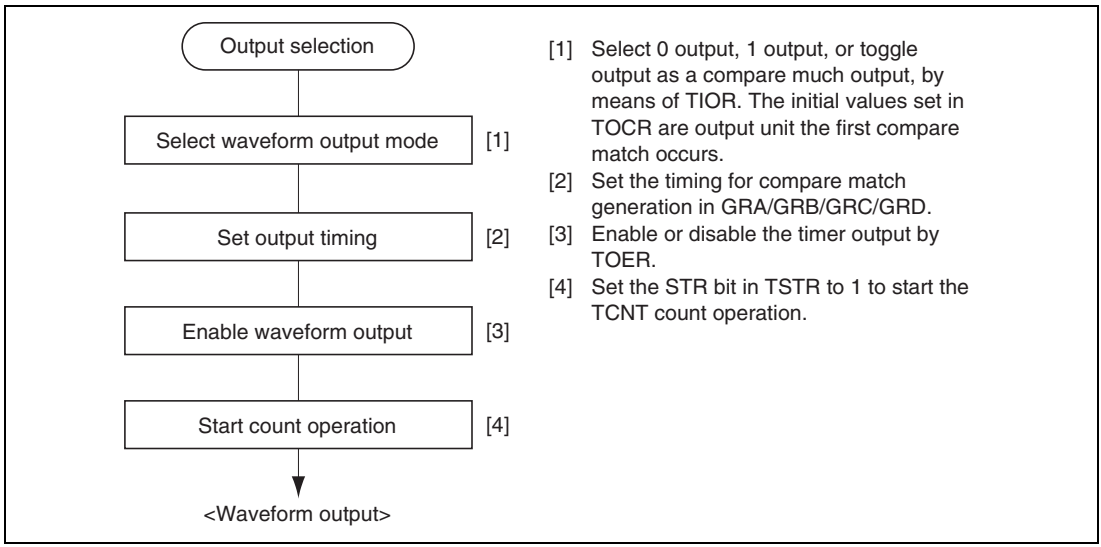


Figure 13.12 Example of Setting Procedure for Waveform Output by Compare Match

1. Examples of waveform output operation

   Figure 13.13 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 0 is output by compare match A, and 1 is output by compare match B. When the set level and the pin level coincide, the pin level does not change.

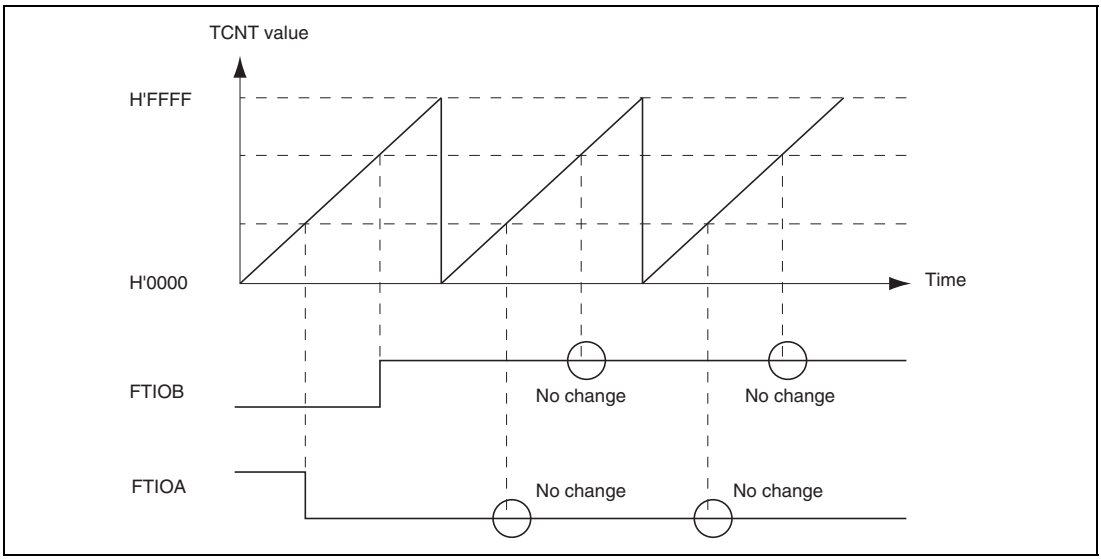


Figure 13.13 Example of 0 Output/1 Output Operation

Figure 13.14 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.

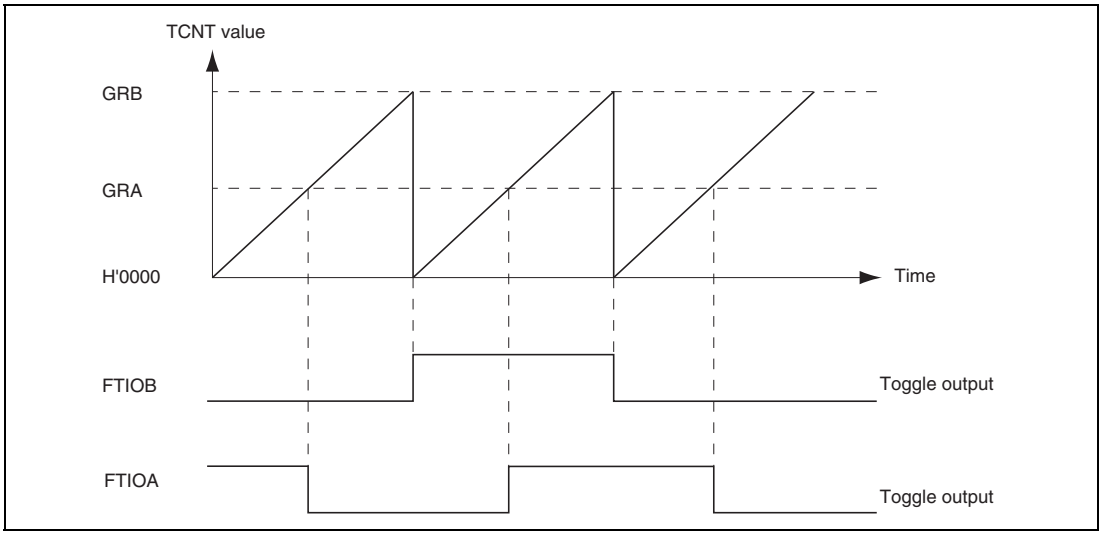


Figure 13.14 Example of Toggle Output Operation

2. Output compare timing

   The compare match signal is generated in the last state in which TCNT and GR match (when TCNT changes from the matching value to the next value). When the compare match signal is generated, the output value selected in TIOR is output at the compare match output pin (FTIOA, FTIOB, FTIOC, or FTIOD). When TCNT matches GR, the compare match signal is generated only after the next TCNT input clock pulse is input.

   Figure 13.15 shows an example of the output compare timing.

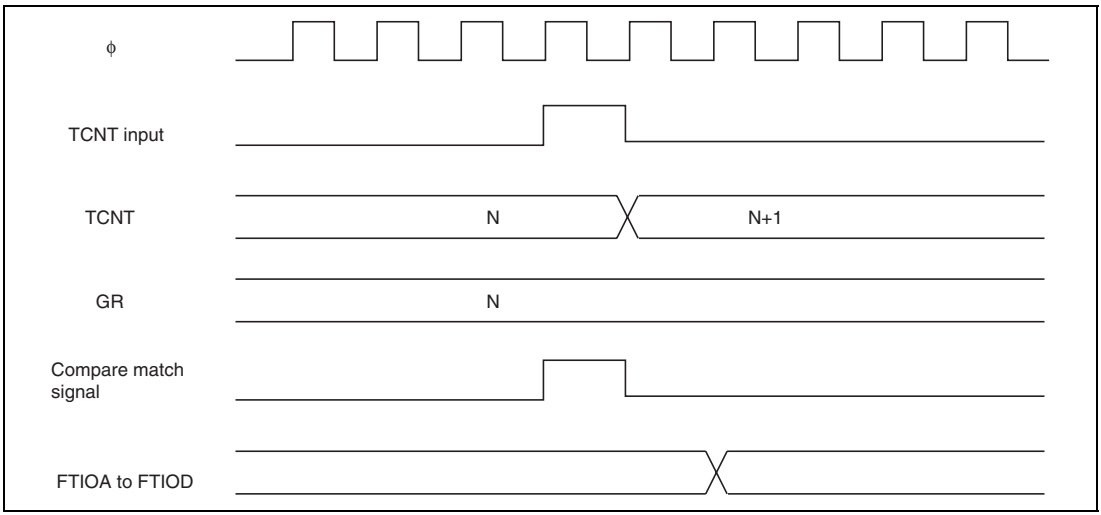


Figure 13.15 Output Compare Timing

13.4.3 Input Capture Function

The TCNT value can be transferred to GR on detection of the input edge of the input capture/output compare pin (FTIOA, FTIOB, FTIOC, or FTIOD). Rising edge, falling edge, or both edges can be selected as the detected edge. When the input capture function is used, the pulse width or period can be measured.

Figure 13.16 shows an example of the input capture operation setting procedure.

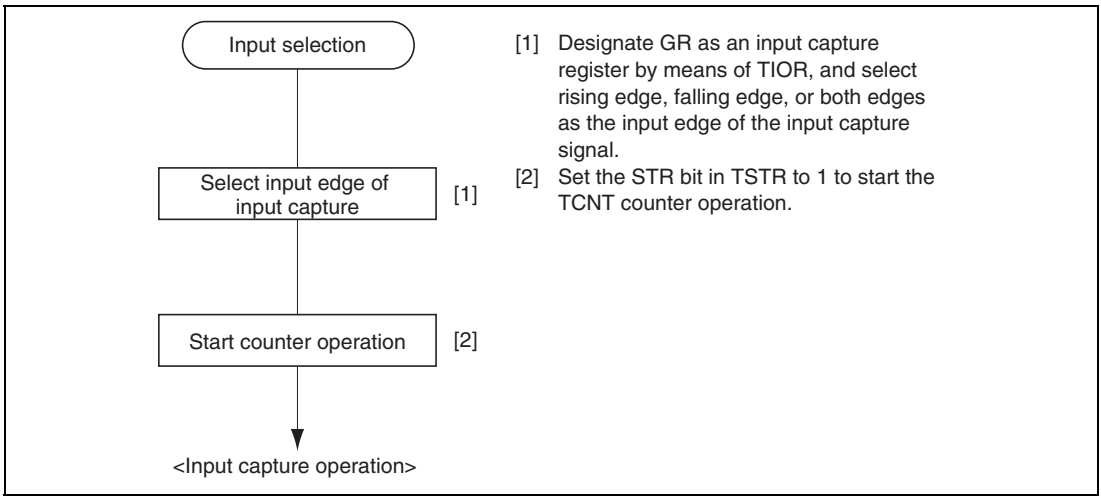


Figure 13.16 Example of Input Capture Operation Setting Procedure

1. Example of input capture operation

   Figure 13.17 shows an example of input capture operation.

   In this example, both rising and falling edges have been selected as the FTIOA pin input capture input edge, the falling edge has been selected as the FTIOB pin input capture input edge, and counter clearing by GRB input capture has been designated for TCNT.

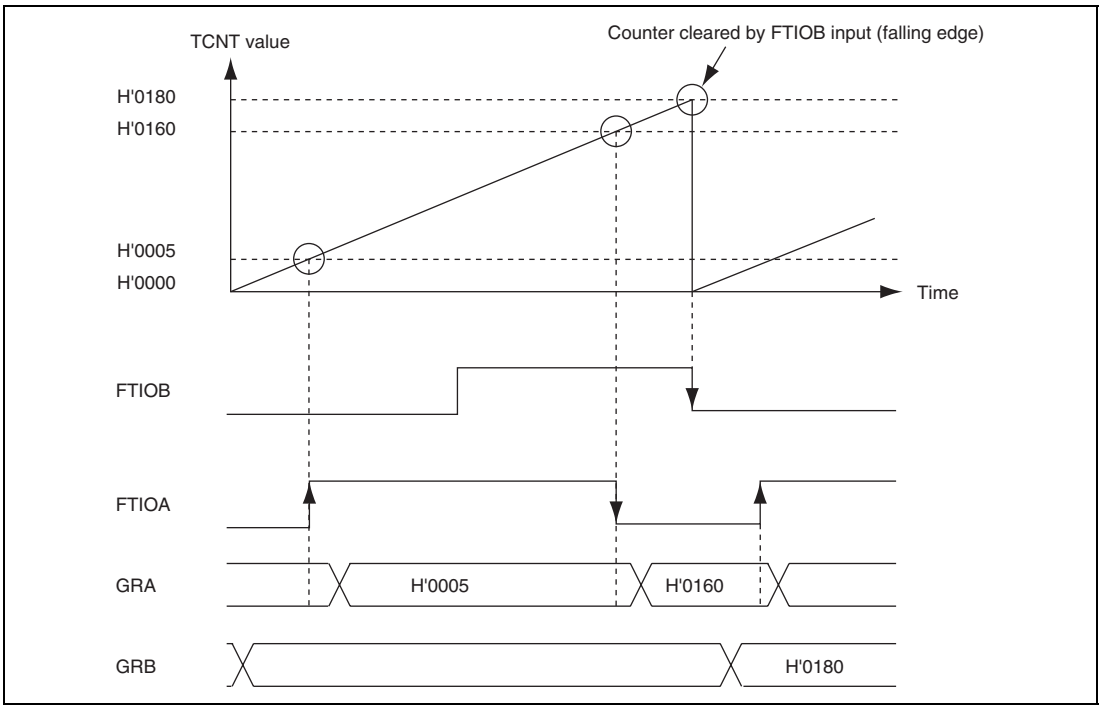


Figure 13.17 Example of Input Capture Operation

2. Input capture signal timing

   Input capture on the rising edge, falling edge, or both edges can be selected through settings in TIOR. Figure 13.18 shows the timing when the rising edge is selected. The pulse width of the input capture signal must be at least two system clock (ϕ) cycles.

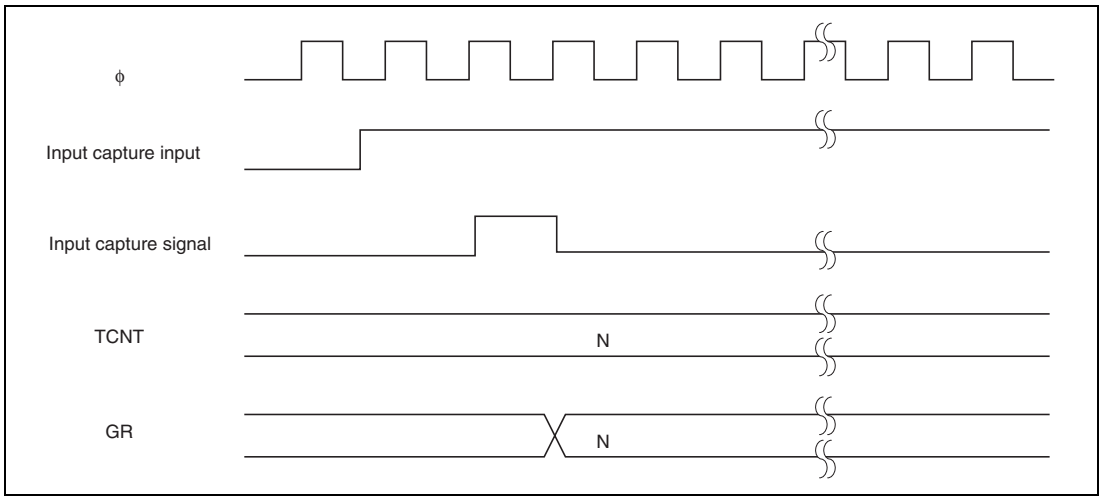


Figure 13.18 Input Capture Signal Timing

13.4.4 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 GR to be increased with respect to a single time base.

Figure 13.19 shows an example of the synchronous operation setting procedure.

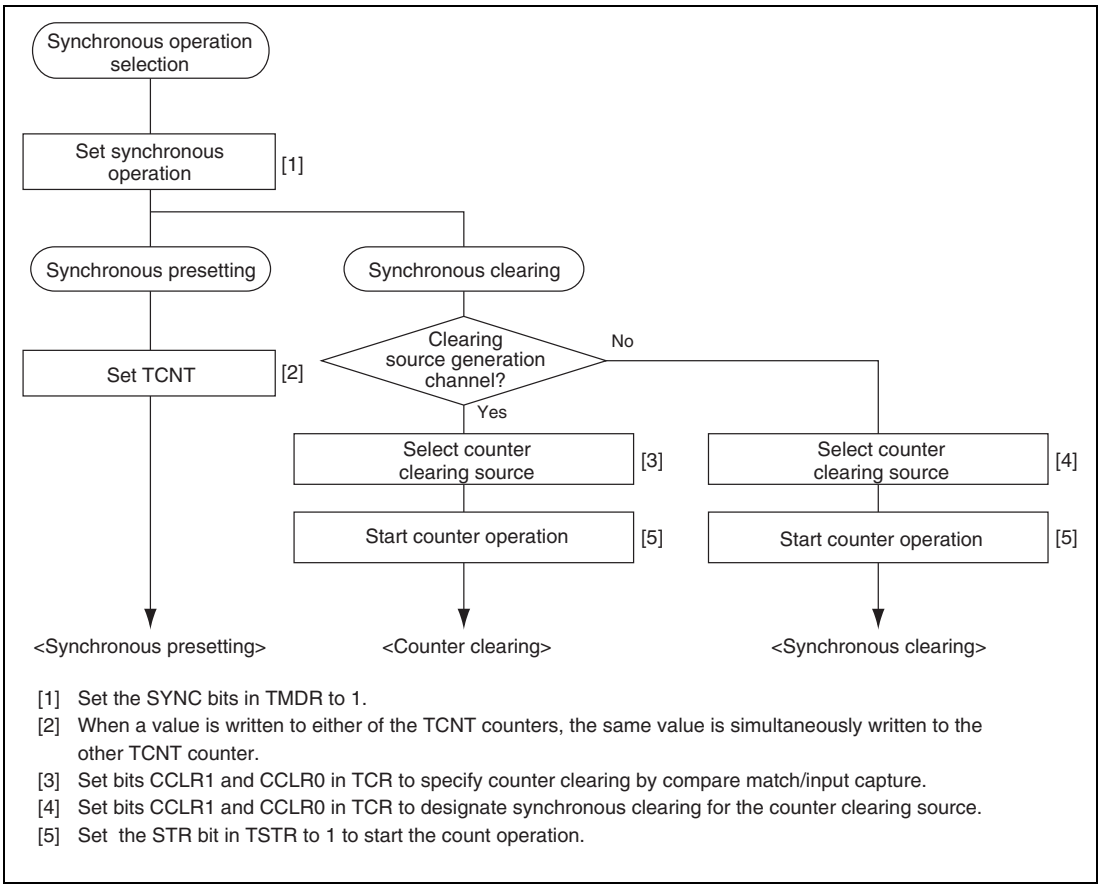


Figure 13.19 Example of Synchronous Operation Setting Procedure

Figure 13.20 shows an example of synchronous operation. In this example, synchronous operation has been selected, FTIOB0 and FTIOB1 have been designated for PWM mode, GRA_0 compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 counter clearing source. In addition, the same input clock has been set as the counter input clock for channel 0 and channel 1. Two-phase PWM waveforms are output from pins FTIOB0 and FTIOB1. At this time, synchronous presetting and synchronous operation by GRA_0 compare match are performed by TCNT counters.

For details on PWM mode, see section 13.4.5, PWM Mode.

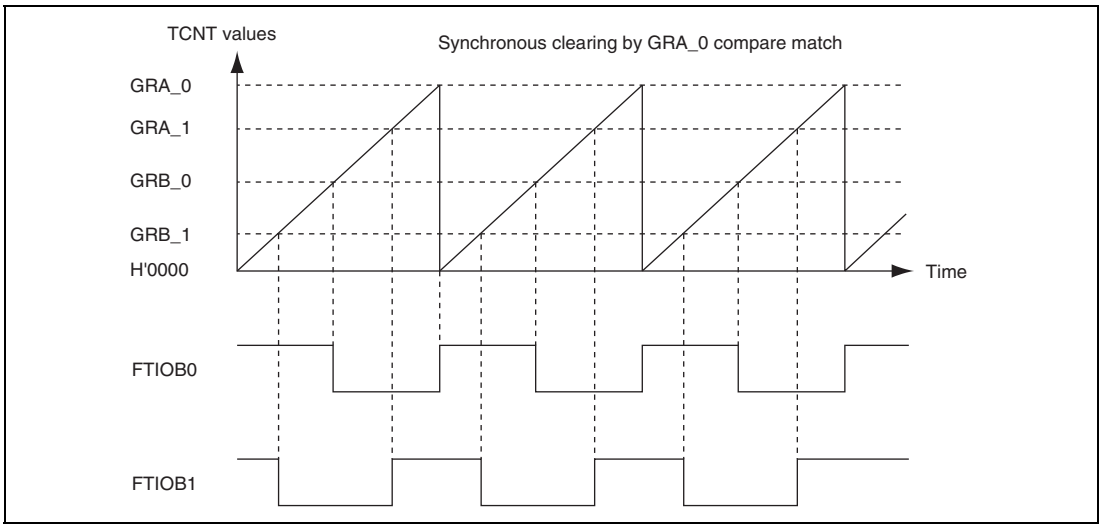


Figure 13.20 Example of Synchronous Operation

13.4.5 PWM Mode

In PWM mode, PWM waveforms are output from the FTIOB, FTIOC, and FTIOD output pins with GRA as a cycle register and GRB, GRC, and GRD as duty registers. The initial output level of the corresponding pin depends on the setting values of TOCR and POCR. Table 13.3 shows an example of the initial output level of the FTIOB0 pin.

The output level is determined by the POLB to POLD bits corresponding to POCR. When POLB is 0, the FTIOB output pin is set to 0 by compare match B and set to 1 by compare match A. When POLB is 1, the FTIOB output pin is set to 1 by compare match B and cleared to 0 by compare match A. In PWM mode, maximum 6-phase PWM outputs are possible.

Figure 13.21 shows an example of the PWM mode setting procedure.

Table 13.3 Initial Output Level of FTIOB0 Pin

TOB0	POLB	Initial Output Level
0	0	1
0	1	0
1	0	0
1	1	1

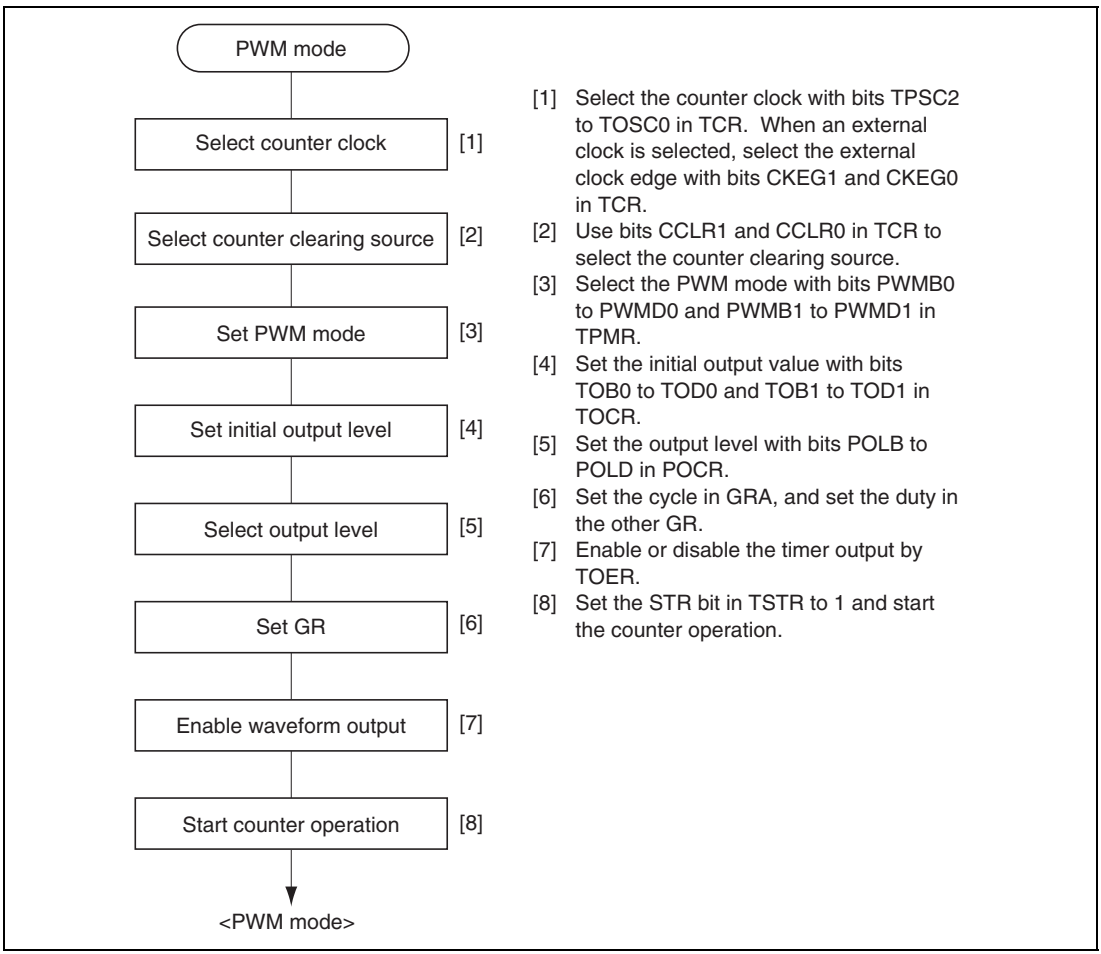


Figure 13.21 Example of PWM Mode Setting Procedure

Figure 13.22 shows an example of operation in PWM mode. The output signals go to 1 and TCNT is reset at compare match A, and the output signals go to 0 at compare match B, C, and D (TOB, TOC, and TOD = 1, POLB, POLC, and POLD = 0).

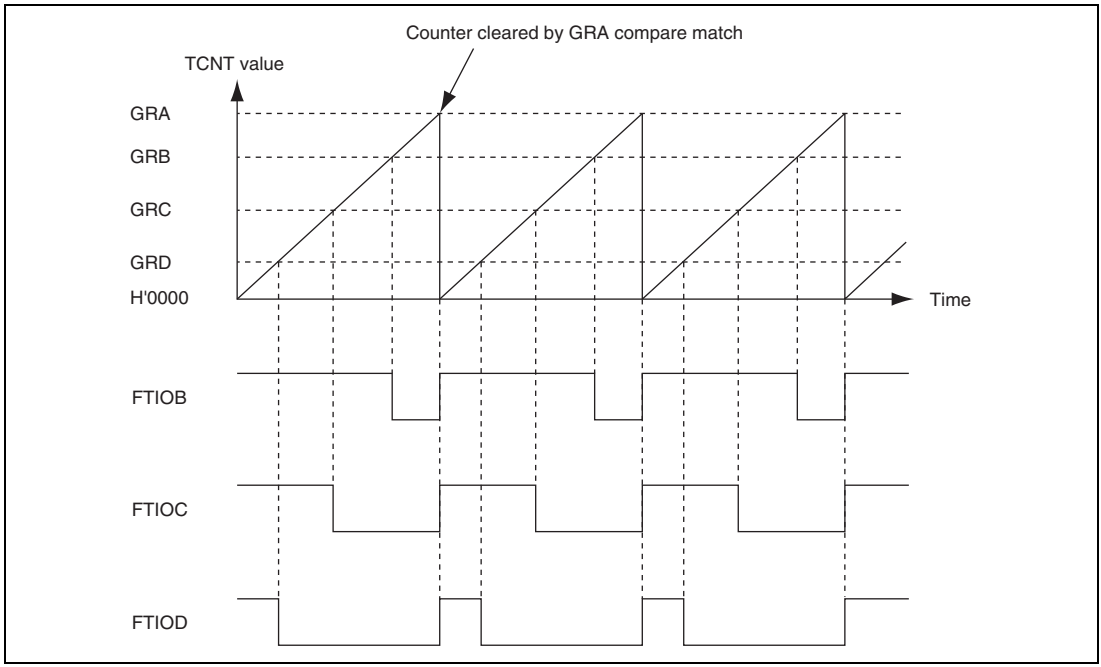


Figure 13.22 Example of PWM Mode Operation (1)

Figure 13.23 shows another example of operation in PWM mode. The output signals go to 0 and TCNT is reset at compare match A, and the output signals go to 1 at compare match B, C, and D (TOB, TOC, and TOD = 0, POLB, POLC, and POLD = 1).

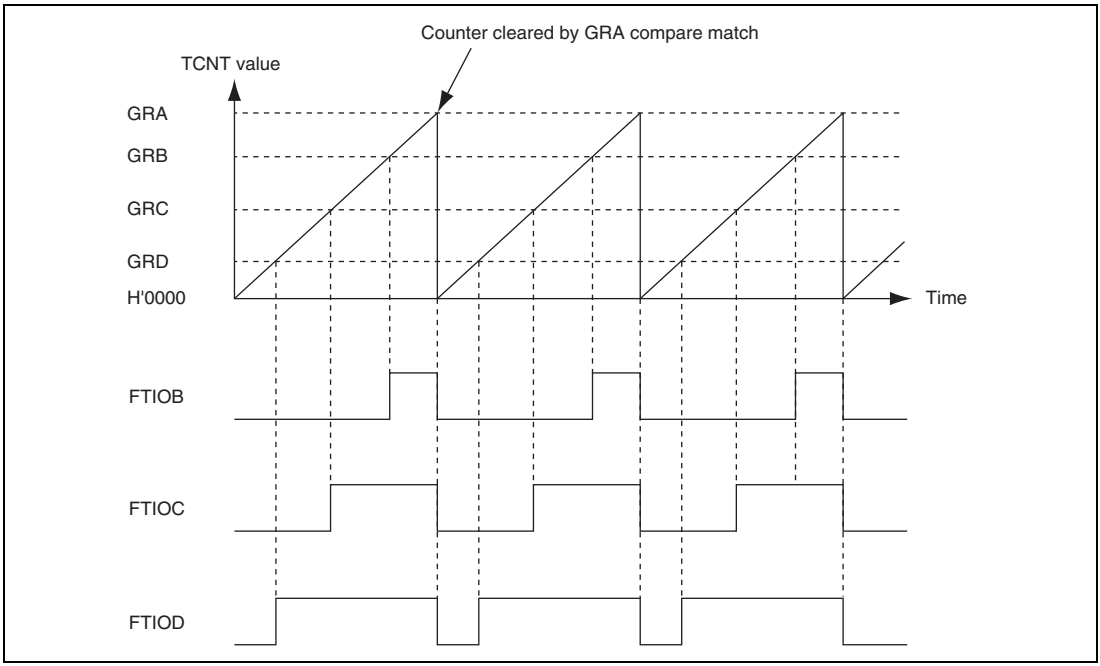


Figure 13.23 Example of PWM Mode Operation (2)

Figures 13.24 (when TOB, TOC, and TOD = 1, POLB, POLC, and POLD = 0) and 13.25 (when TOB, TOC, and TOD = 0, POLB, POLC, and POLD = 1) show examples of the output of PWM waveforms with duty cycles of 0% and 100% in PWM mode.

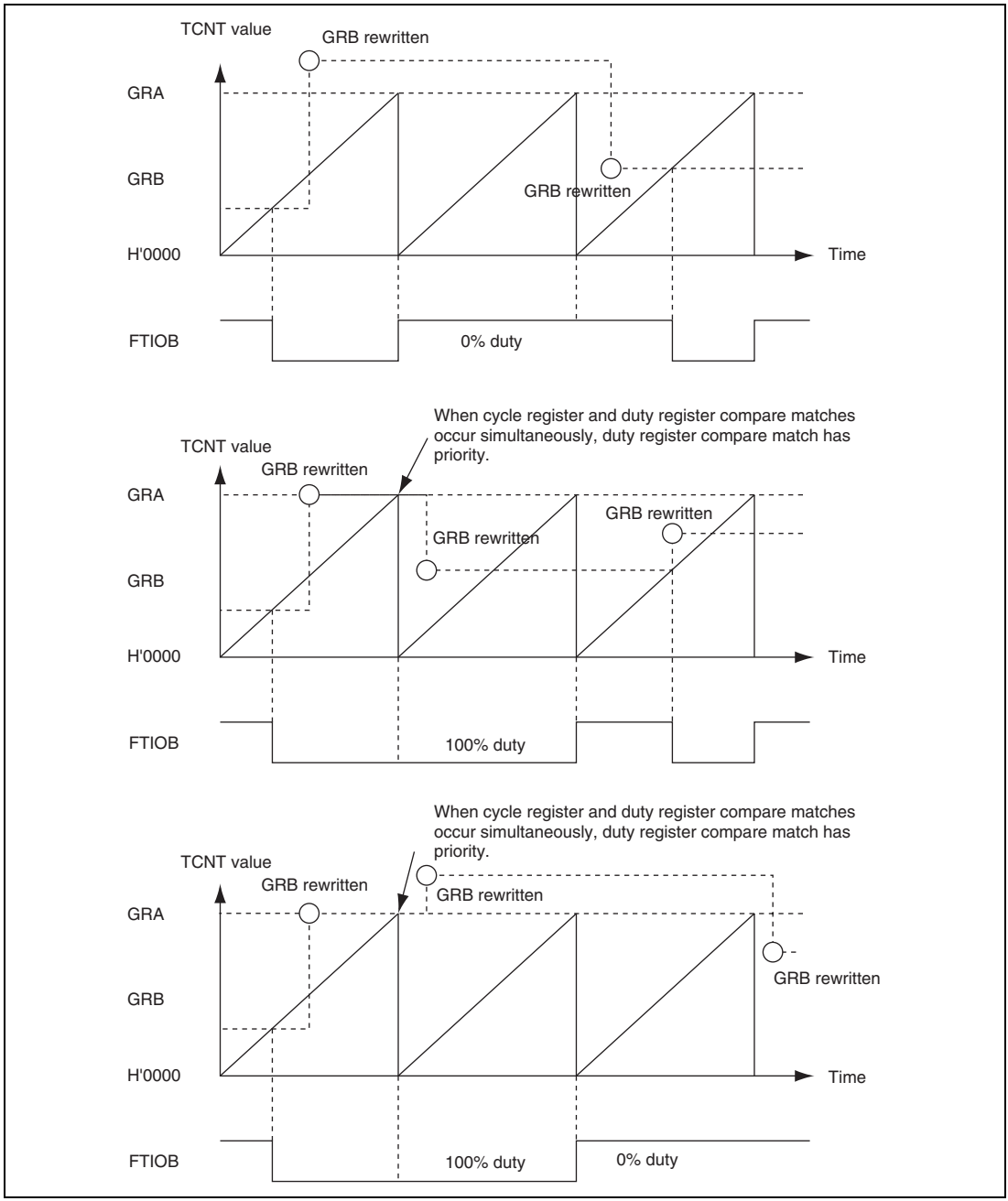


Figure 13.24 Example of PWM Mode Operation (3)

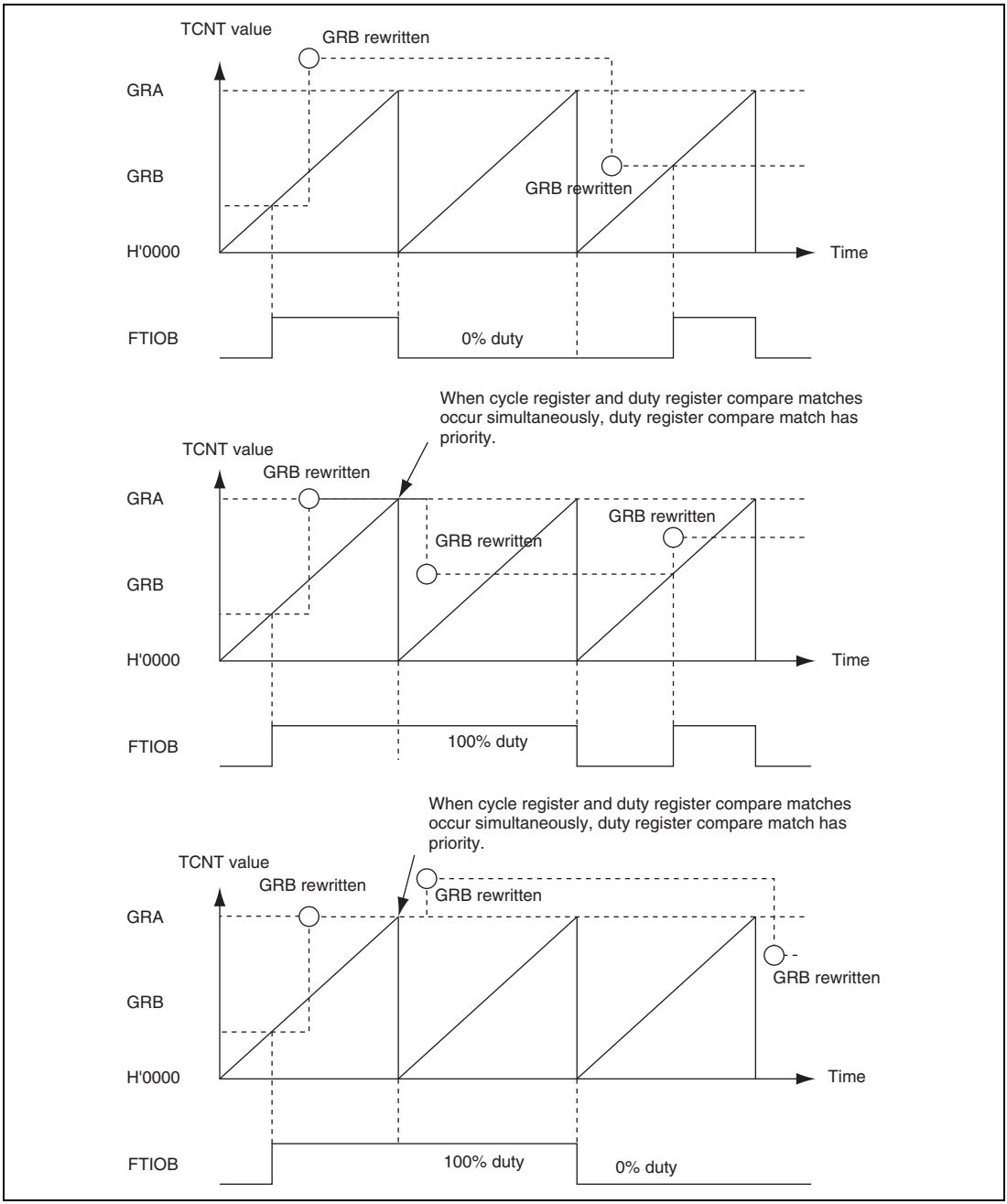


Figure 13.25 Example of PWM Mode Operation (4)

13.4.6 Reset Synchronous PWM Mode

Three normal- and counter-phase PWM waveforms are output by combining channels 0 and 1 that one of changing points of waveforms will be common.

In reset synchronous PWM mode, the FTIOB0 to FTIOD0 and FTIOA1 to FTIOD1 pins become PWM-output pins automatically. TCNT_0 performs an increment operation. Tables 13.4 and 13.5 show the PWM-output pins used and the register settings, respectively.

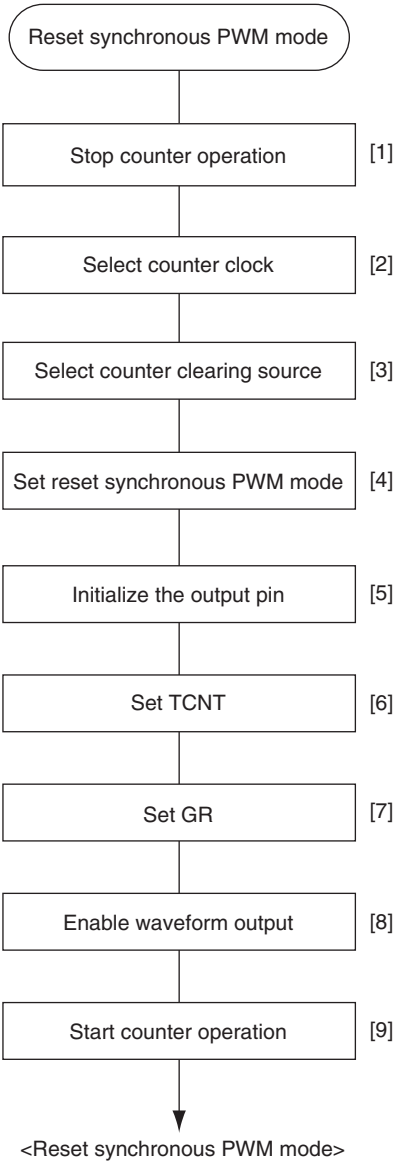
Figure 13.26 shows the example of reset synchronous PWM mode setting procedure.

Table 13.4 Output Pins in Reset Synchronous PWM Mode

Channel	Pin Name	Input/Output	Pin Function
0	FTIOC0	Output	Toggle output in synchronous with PWM cycle
0	FTIOB0	Output	PWM output 1
0	FTIOD0	Output	PWM output 1 (counter-phase waveform of PWM output 1)
1	FTIOA1	Output	PWM output 2
1	FTIOC1	Output	PWM output 2 (counter-phase waveform of PWM output 2)
1	FTIOB1	Output	PWM output 3
1	FTIOD1	Output	PWM output 3 (counter-phase waveform of PWM output 3)

Table 13.5 Register Settings in Reset Synchronous PWM Mode

Register	Description
TCNT_0	Initial setting of H'0000
TCNT_1	Not used (independently operates)
GRA_0	Sets counter cycle of TCNT_0
GRB_0	Set a changing point of the PWM waveform output from pins FTIOB0 and FTIOD0.
GRA_1	Set a changing point of the PWM waveform output from pins FTIOA1 and FTIOC1.
GRB_1	Set a changing point of the PWM waveform output from pins FTIOB1 and FTIOD1.



[1] Clear bit STR0 in TSTR to 0 and stop the counter operation of TCNT_0. Set reset synchronous PWM mode after TCNT_0 stops.
[2] Select the counter clock with bits TPSC2 to TOSC0 in TCR. When an external clock is selected, select the external clock edge with bits CKEG1 and CKEG0 in TCR.
[3] Use bits CCLR1 and CCLR0 in TCR to select counter clearing source GRA_0.
[4] Select the reset synchronous PWM mode with bits CMD1 and CMD0 in TFCR. FTIOB0 to FTIOD0 and FTIOA1 to FTIOD1 become PWM output pins automatically.
[5] Set H'00 to TOCR.
[6] Set TCNT_0 as H'0000. TCNT1 does not need to be set.
[7] GRA_0 is a cycle register. Set a cycle for GRA_0. Set the changing point timing of the PWM output waveform for GRB_0, GRA_1, and GRB_1.
[8] Enable or disable the timer output by TOER.
[9] Set the STR bit in TSTR to 1 and start the counter operation.

Figure 13.26 Example of Reset Synchronous PWM Mode Setting Procedure

Figures 13.27 and 13.28 show examples of operation in reset synchronous PWM mode.

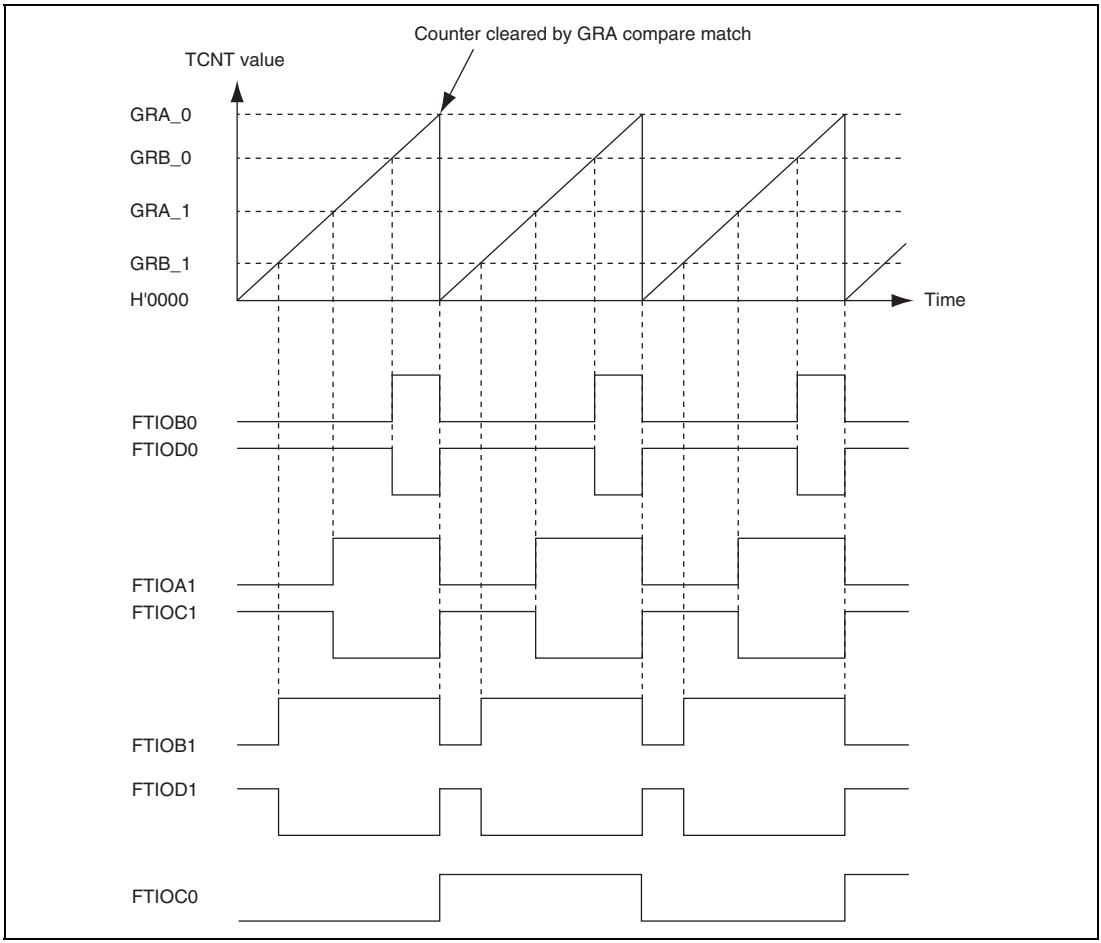


Figure 13.27 Example of Reset Synchronous PWM Mode Operation (OLS0 = OLS1 = 1)

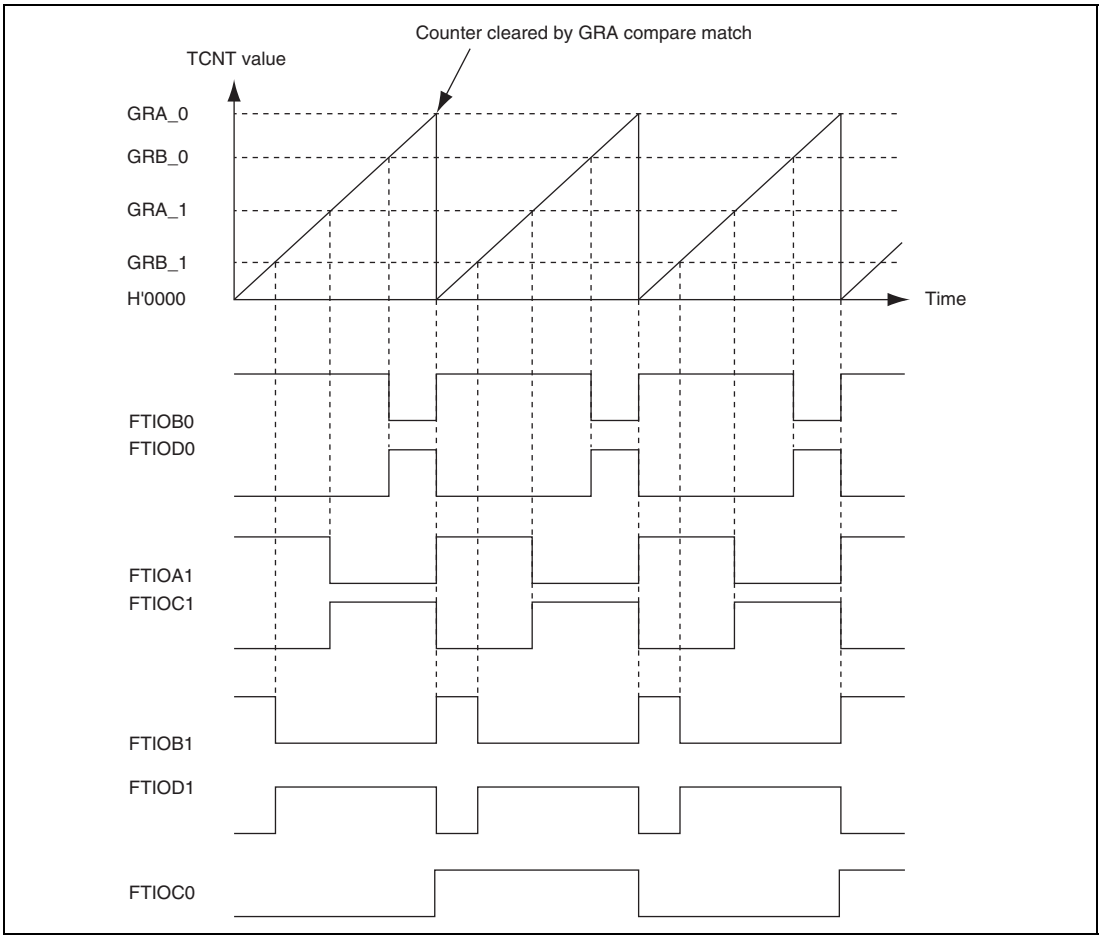


Figure 13.28 Example of Reset Synchronous PWM Mode Operation (OLS0 = OLS1 = 0)

In reset synchronous PWM mode, TCNT_0 and TCNT_1 perform increment and independent operations, respectively. However, GRA_1 and GRB_1 are separated from TCNT_1. When a compare match occurs between TCNT_0 and GRA_0, a counter is cleared and an increment operation is restarted from H'0000.

The PWM pin outputs 0 or 1 whenever a compare match between GRB_0, GRA_1, GRB_1 and TCNT_0 or counter clearing occur.

For details on operations when reset synchronous PWM mode and buffer operation are simultaneously set, refer to section 13.4.8, Buffer Operation.

13.4.7 Complementary PWM Mode

Three PWM waveforms for non-overlapped normal and counter phases are output by combining channels 0 and 1.

In complementary PWM mode, the FTIOB0 to FTIOD0 and FTIOA1 to FTIOD1 pins become PWM-output pins automatically. TCNT_0 and TCNT_1 perform an increment or decrement operation. Tables 13.6 and 13.7 show the output pins and register settings in complementary PWM mode, respectively.

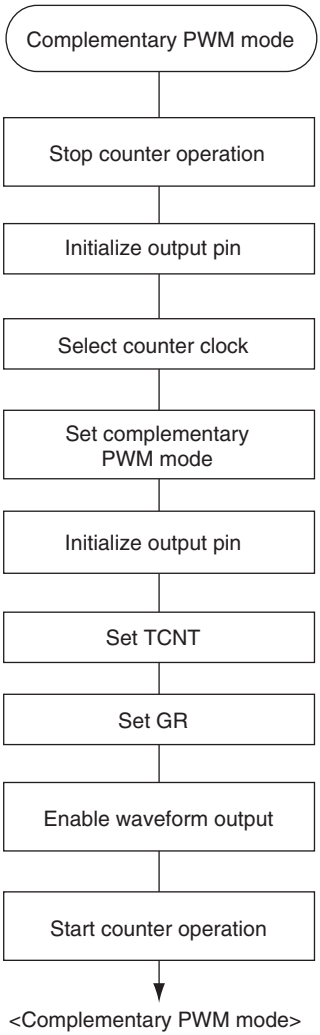
Figure 13.29 shows the example of complementary PWM mode setting procedure.

Table 13.6 Output Pins in Complementary PWM Mode

Channel	Pin Name	Input/Output	Pin Function
0	FTIOC0	Output	Toggle output in synchronous with PWM cycle
0	FTIOB0	Output	PWM output 1
0	FTIOD0	Output	PWM output 1 (counter-phase waveform non-overlapped with PWM output 1)
1	FTIOA1	Output	PWM output 2
1	FTIOC1	Output	PWM output 2 (counter-phase waveform non-overlapped with PWM output 2)
1	FTIOB1	Output	PWM output 3
1	FTIOD1	Output	PWM output 3 (counter-phase waveform non-overlapped with PWM output 3)

Table 13.7 Register Settings in Complementary PWM Mode

Register	Description
TCNT_0	Initial setting of non-overlapped periods (non-overlapped periods are differences with TCNT_1)
TCNT_1	Initial setting of H'0000
GRA_0	Sets (upper limit value – 1) of TCNT_0
GRB_0	Set a changing point of the PWM waveform output from pins FTIOB0 and FTIOD0.
GRA_1	Set a changing point of the PWM waveform output from pins FTIOA1 and FTIOC1.
GRB_1	Set a changing point of the PWM waveform output from pins FTIOB1 and FTIOD1.



[1] Clear bits STR0 and STR1 in TSTR to 0, and stop the counter operation of TCNT_0. Stop TCNT_0 and TCNT_1 and set complementary PWM mode.
[2] Write H'00 to TOCR.
[3] Use bits TPSC2 to TPSC0 in TCR to select the same counter clock for channels 0 and 1. When an external clock is selected, select the edge of the external clock by bits CKEG1 and CKEG0 in TCR. Do not use bits CCLR1 and CCLR0 in TCR to clear the counter.
[4] Use bits CMD1 and CMD0 in TFCR to set complementary PWM mode. FTIOB0 to FTIOD0 and FTIOA1 to FTIOD1 automatically become PWM output pins.
[5] Set H'00 to TOCR.
[6] TCNT_1 must be H'0000. Set a non-overlapped period to TCNT_0.
[7] GRA_0 is a cycle register. Set the cycle to GRA_0. Set the timing to change the PWM output waveform to GRB_0, GRA_1, and GRB_1. Note that the timing must be set within the range of compare match carried out for TCNT_0 and TCNT_1. For GR settings, see 3. Setting GR Value in Complementary PWM Mode in section 13.4.7.
[8] Use TOER to enable or disable the timer output.
[9] Set the STR0 and STR1 bits in TSTR to 1 to start the count operation.

Note: To re-enter complementary PWM mode, first, enter a mode other than the complementary PWM mode. After that, repeat the setting procedures from step [1].
For settings of waveform outputs with a duty cycle of 0% and 100%, see the settings shown in 2. Examples of Complementary PWM Mode Operation and 3. Setting GR Value in Complementary PWM Mode in section 13.4.7.

Figure 13.29 Example of Complementar y PWM Mode Setting Procedure

1. Canceling Procedure of Complementary PWM Mode: Figure 13.30 shows the complementary PWM mode canceling procedure.

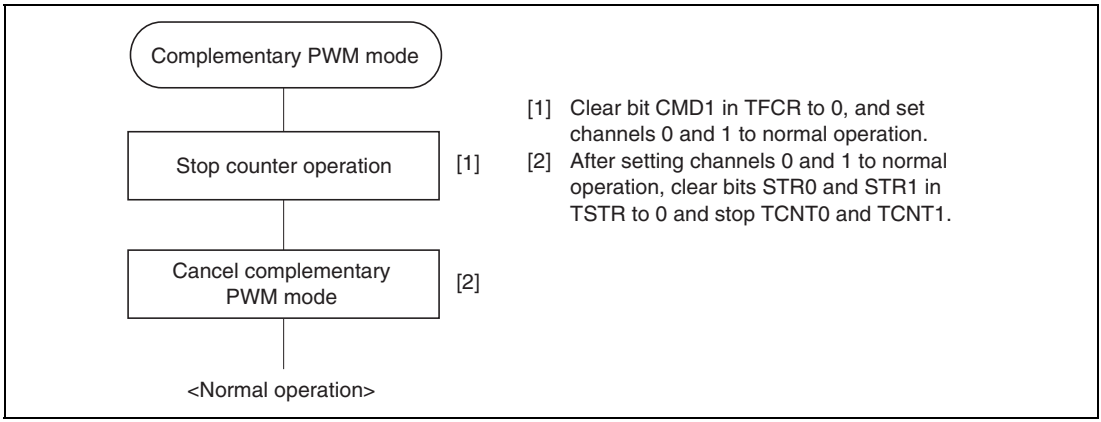


Figure 13.30 Canceling Procedure of Complementary PWM Mode

2. Examples of Complementary PWM Mode Operation: Figure 13.31 shows an example of complementary PWM mode operation. In complementary PWM mode, TCNT_0 and TCNT_1 perform an increment or decrement operation. When TCNT_0 and GRA_0 are compared and their contents match, the counter is decremented, and when TCNT_1 underflows, the counter is incremented. In GRA_0, GRA_1, and GRB_1, compare match is carried out in the order of TCNT_0 → TCNT_1 → TCNT_1 → TCNT_0 and PWM waveform is output, during one cycle of a up/down counter. In this mode, the initial setting will be TCNT_0 > TCNT_1.

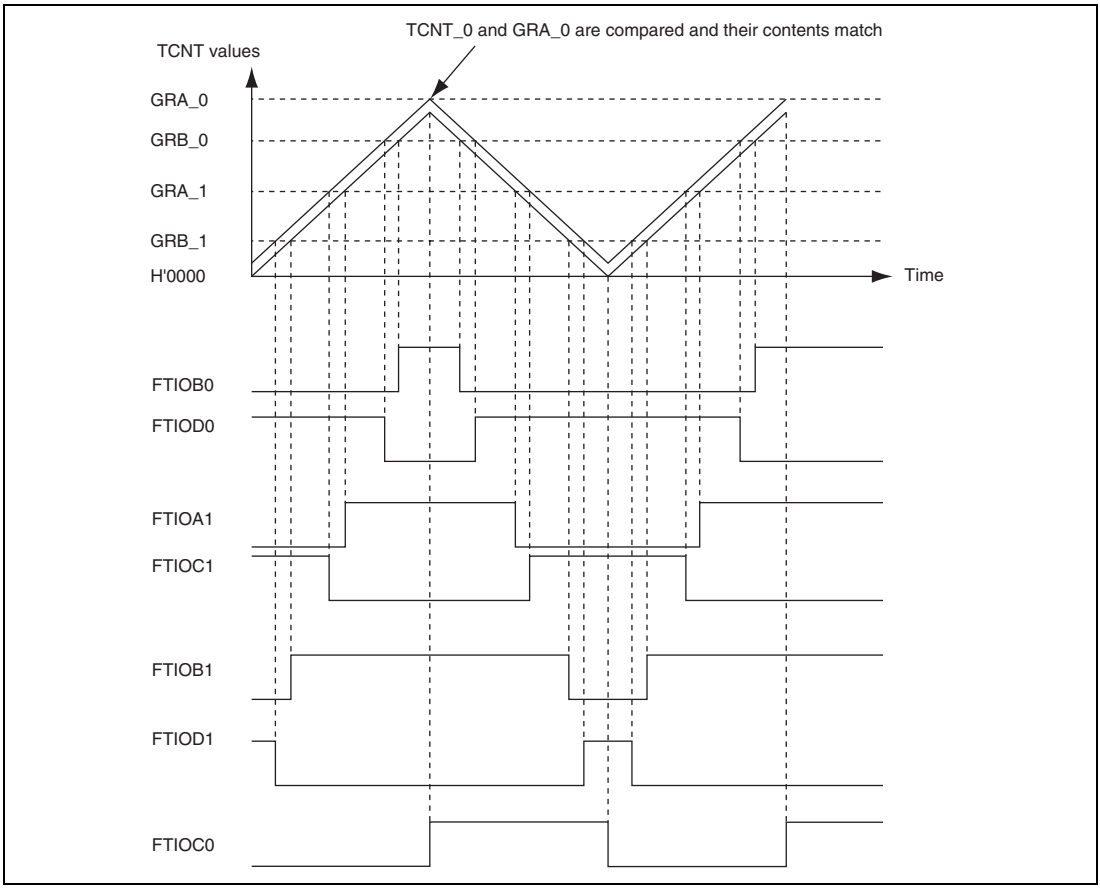


Figure 13.31 Example of Complementary PWM Mode Operation (1)

Figure 13.32 (1) and (2) show examples of PWM waveform output with 0% duty and 100% duty in complementary PWM mode (for one phase).

- TPSC2 = TPSC1 = TPSC0 = 0

  Set GRB_0 to H'0000 or a value equal to or more than GRA_0. The waveform with a duty cycle of 0% and 100% can be output. When buffer operation is used together, the duty cycles can easily be changed, including the above settings, during operation. For details on buffer operation, refer to section 13.4.8, Buffer Operation.

- Other than TPSC2 = TPSC1 = TPSC0 = 0

  Set GRB_0 to satisfy the following expression: GRA_0 + 1 < GRB_0 < H'FFFF. The waveform with a duty cycle of 0% and 100% can be output. For details on 0%- and 100%-duty cycle waveform output, see 3. C., Outputting a waveform with a duty cycle of 0% and 100% in section 13.4.7.

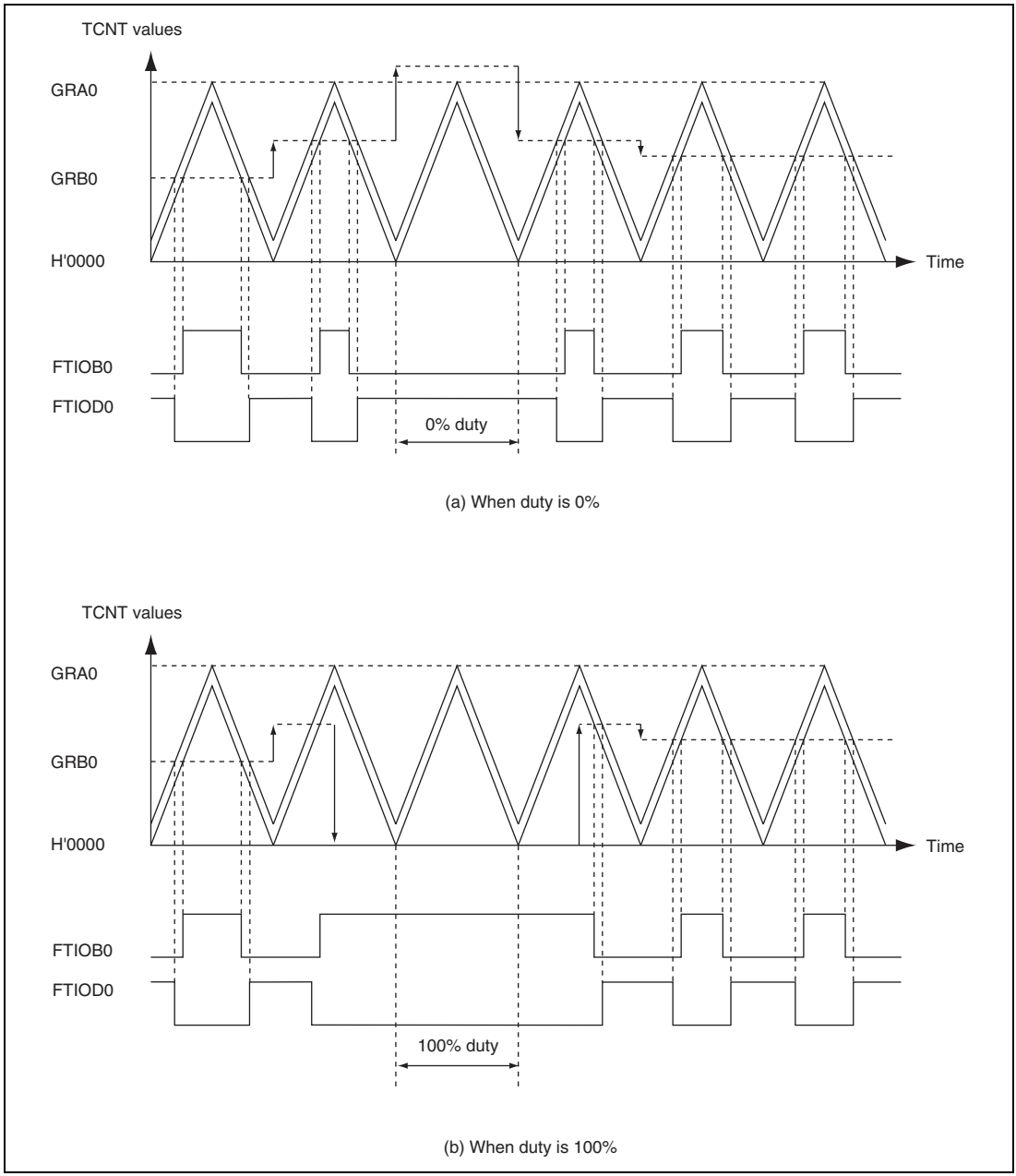


Figure 13.32 (1) Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 = 0) (2)

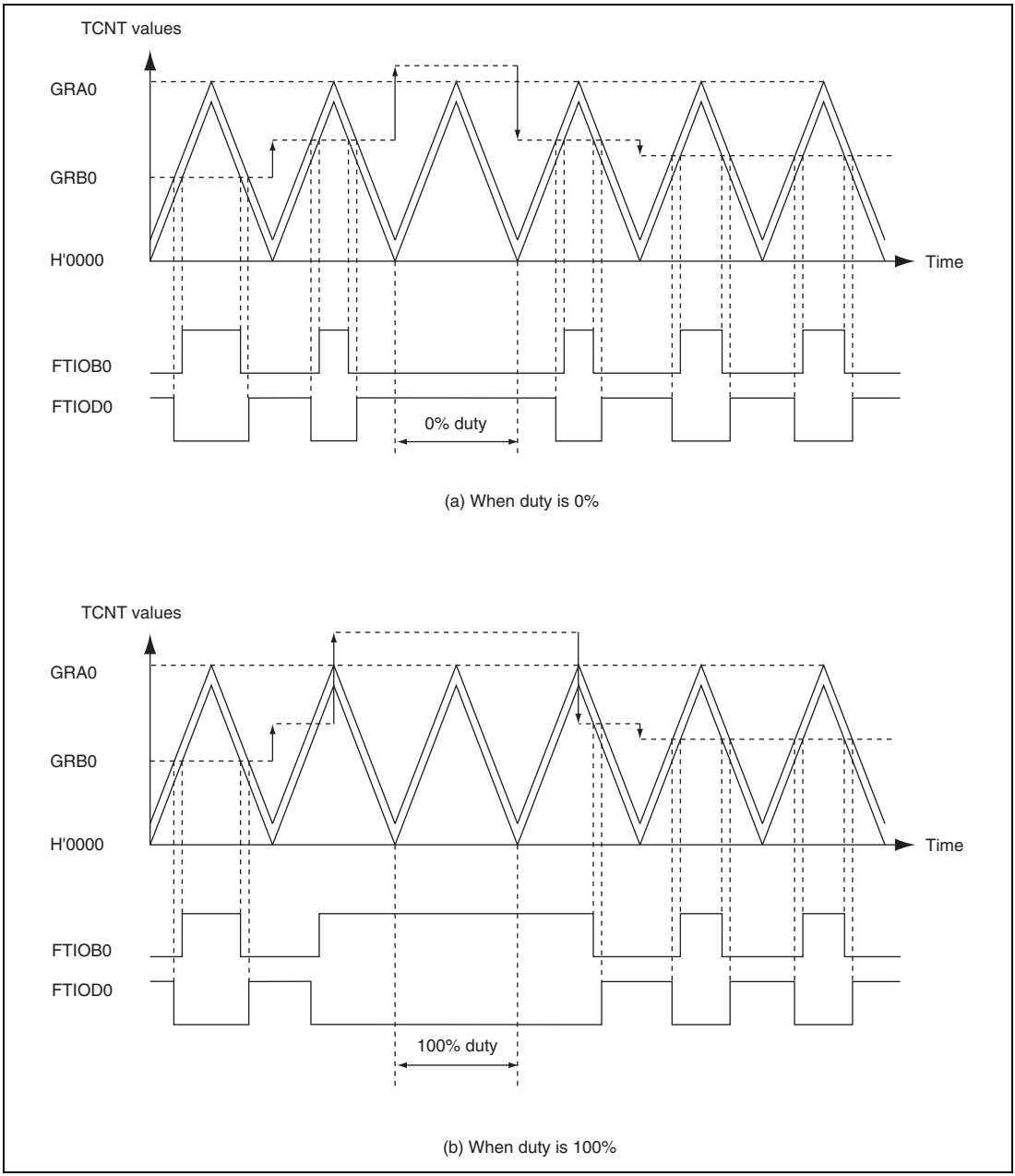


Figure 13.32 (2) Example of Complementary PWM Mode Operation (TPSC2 = TPSC1 = TPSC0 ≠ 0) (3)

In complementary PWM mode, when the counter switches from up-counter to down-counter or vice versa, TCNT_0 and TCNT_1 overshoots or undershoots, respectively. In this case, the conditions to set the IMFA flag in channel 0 and the UDF flag in channel 1 differ from usual settings. Also, the transfer conditions in buffer operation differ from usual settings. Such timings are shown in figures 13.33 and 13.34.

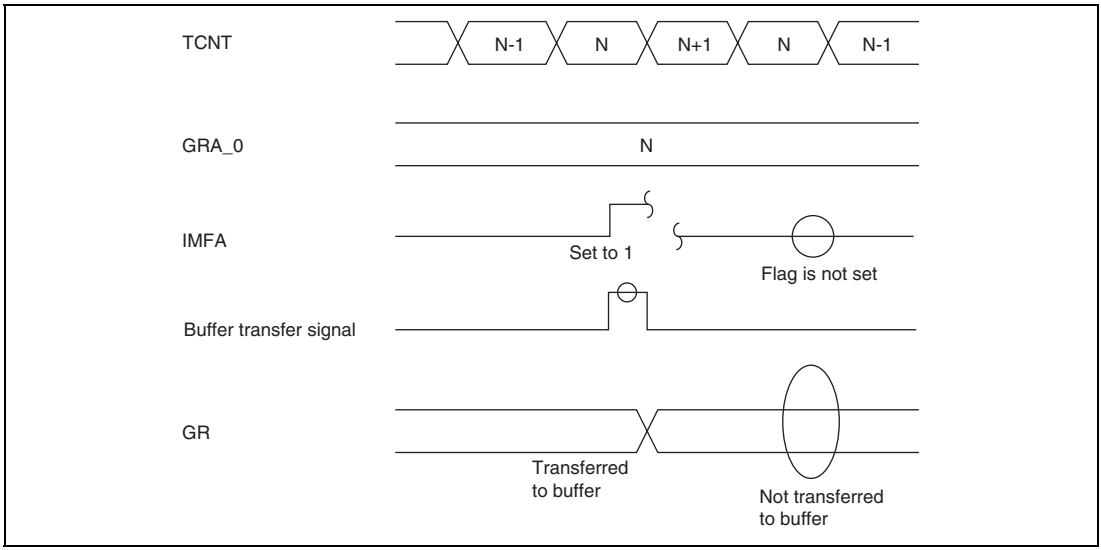


Figure 13.33 Timing of Overshooting

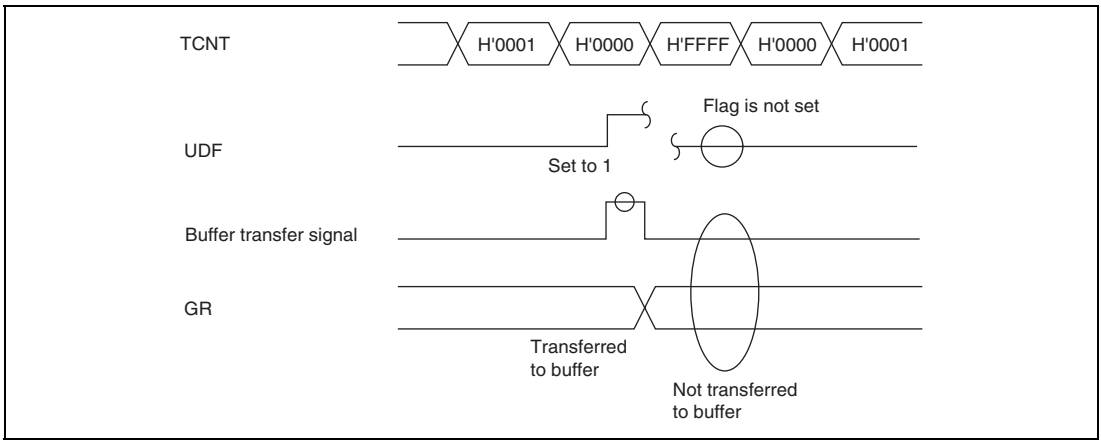


Figure 13.34 Timing of Undershooting

When the counter is incremented or decremented, the IMFA flag of channel 0 is set to 1, and when the register is underflowed, the UDF flag of channel 0 is set to 1. After buffer operation has been designated for BR, BR is transferred to GR when the counter is incremented by compare match A0 or when TCNT_1 is underflowed. If the ϕ or ϕ/2 clock is selected by TPSC2 to TPSC0 bits, the OVF flag is not set to 1 at the timing that the counter value changes from H'FFFF to H'0000. If the ϕ/4 or ϕ/8 clock is selected by TPSC2 to TPSC0 bits, the OVF flag is set to 1.

3. Setting GR Value in Complementary PWM Mode: To set the general register (GR) or modify GR during operation in complementary PWM mode, refer to the following notes.
   A. Initial value
      a. When other than TPSC2 = TPSC1 = TPSC0 = 0, the GRA_0 value must be equal to H'FFFC or less. When TPSC2 = TPSC1 = TPSC0 = 0, the GRA_0 value can be set to H'FFFF or less.
      b. H'0000 to T – 1 (T: Initial value of TCNT0) must not be set for the initial value.
      c. GRA_0 – (T – 1) or more must not be set for the initial value.
      d. When using buffer operation, the same values must be set in the buffer registers and corresponding general registers.
   B. Modifying the setting value
      a. Writing to GR directly must be performed while the TCNT_1 and TCNT_0 values should satisfy the following expression: H'0000 ≤ TCNT_1 < previous GR value, and previous GR value < TCNT_0 ≤ GRA_0. Otherwise, a waveform is not output correctly. For details on outputting a waveform with a duty cycle of 0% and 100%, see C., Outputting a waveform with a duty cycle of 0% and 100%.
      b. Do not write the following values to GR directly. When writing the values, a waveform is not output correctly.

         H'0000 ≤ GR ≤ T – 1 and GRA_0 – (T – 1) ≤ GR < GRA_0 when TPSC2 = TPSC1 = TPSC0 = 0

         H'0000 < GR ≤ T – 1 and GRA_0 – (T – 1) ≤ GR < GRA_0 + 1 when TPSC2 = TPSC1 = TPSC0 = 0
      c. Do not change settings of GRA_0 during operation.
   C. Outputting a waveform with a duty cycle of 0% and 100%
      a. Buffer operation is not used and TPSC2 = TPSC1 = TPSC0 = 0

         Write H'0000 or a value equal to or more than the GRA_0 value to GR directly at the timing shown below.

      - To output a 0%-duty cycle waveform, write a value equal to or more than the GRA_0 value while H'0000 ≤ TCNT_1 < previous GR value
      - To output a 100%-duty cycle waveform, write H'0000 while previous GR value< TCNT_0 ≤ GRA_0

To change duty cycles while a waveform with a duty cycle of 0% or 100% is being output, make sure the following procedure.

- To change duty cycles while a 0%-duty cycle waveform is being output, write to GR while H'0000 ≤ TCNT_1 < previous GR value
- To change duty cycles while a 100%-duty cycle waveform is being output, write to GR while previous GR value< TCNT_0 ≤ GRA_0

  Note that changing from a 0%-duty cycle waveform to a 100%-duty cycle waveform and vice versa is not possible.

b. Buffer operation is used and TPSC2 = TPSC1 = TPSC0 = 0

   Write H'0000 or a value equal to or more than the GRA_0 value to the buffer register.

- To output a 0%-duty cycle waveform, write a value equal to or more than the GRA_0 value to the buffer register
- To output a 100%-duty cycle waveform, write H'0000 to the buffer register

  For details on buffer operation, see section 13.4.8, Buffer Operation.

c. Buffer operation is not used and other than TPSC2 = TPSC1 = TPSC0 = 0

   Write a value which satisfies GRA_0 + 1 < GR < H'FFFF to GR directly at the timing shown below.

- To output a 0%-duty cycle waveform, write the value while H'0000 ≤ TCNT_1 < previous GR value
- To output a 100%-duty cycle waveform, write the value while previous GR value< TCNT_0 ≤ GRA_0

To change duty cycles while a waveform with a duty cycle of 0% and 100% is being output, the following procedure must be followed.

- To change duty cycles while a 0%-duty cycle waveform is being output, write to GR while H'0000 ≤ TCNT_1 < previous GR value
- To change duty cycles while a 100%-duty cycle waveform is being output, write to GR while previous GR value< TCNT_0 ≤ GRA_0

  Note that changing from a 0%-duty cycle waveform to a 100%-duty cycle waveform and vice versa is not possible.

d. Buffer operation is used and other than TPSC2 = TPSC1 = TPSC0 = 0

   Write a value which satisfies GRA_0 + 1 < GR < H'FFFF to the buffer register. A waveform with a duty cycle of 0% can be output. However, a waveform with a duty cycle of 100% cannot be output using the buffer operation. Also, the buffer operation cannot be used to change duty cycles while a waveform with a duty cycle of 100% is being output. For details on buffer operation, see section 13.4.8, Buffer Operation.

13.4.8 Buffer Operation

Buffer operation differs depending on whether GR has been designated for an input capture register or an output compare register, or in reset synchronous PWM mode or complementary PWM mode.

Table 13.8 shows the register combinations used in buffer operation.

Table 13.8 Register Combinations in Buffer Operation

General Register	Buffer Register
GRA	GRC
GRB	GRD

1. When GR is an output compare register

   When a compare match occurs, the value in the buffer register of the corresponding channel is transferred to the general register.

   This operation is illustrated in figure 13.35.

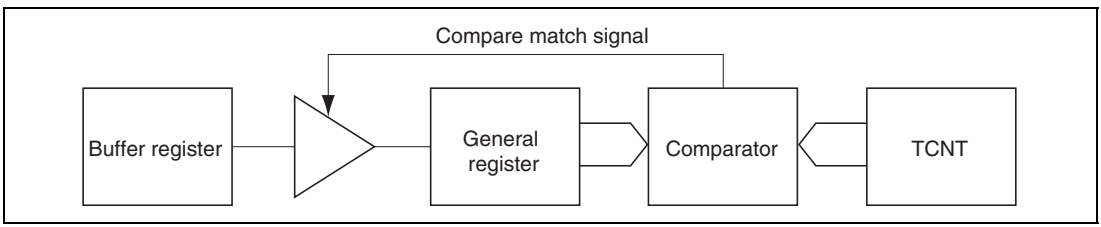


Figure 13.35 Compare Match Buffer Operation

2. When GR is an input capture register

   When an input capture occurs, the value in TCNT is transferred to the general register and the value previously stored in the general register is transferred to the buffer register.

   This operation is illustrated in figure 13.36.

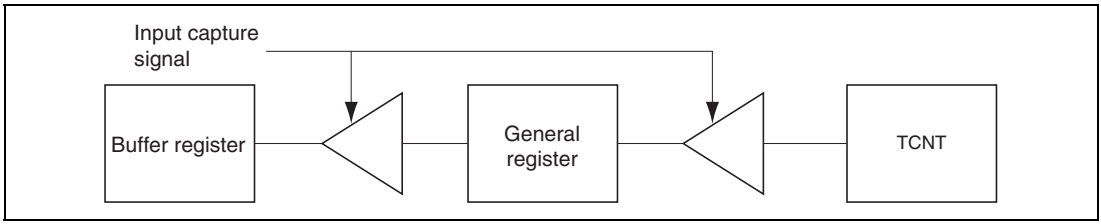


Figure 13.36 Input Capture Buffer Operation

3. Complementary PWM Mode

   When the counter switches from counting up to counting down or vice versa, the value of the buffer register is transferred to the general register. Here, the value of the buffer register is transferred to the general register in the following timing:

   A. When TCNT_0 and GRA_0 are compared and their contents match

   B. When TCNT_1 underflows

4. Reset Synchronous PWM Mode

   The value of the buffer register is transferred from compare match A0 to the general register.

5. Example of Buffer Operation Setting Procedure

   Figure 13.37 shows an example of the buffer operation setting procedure.

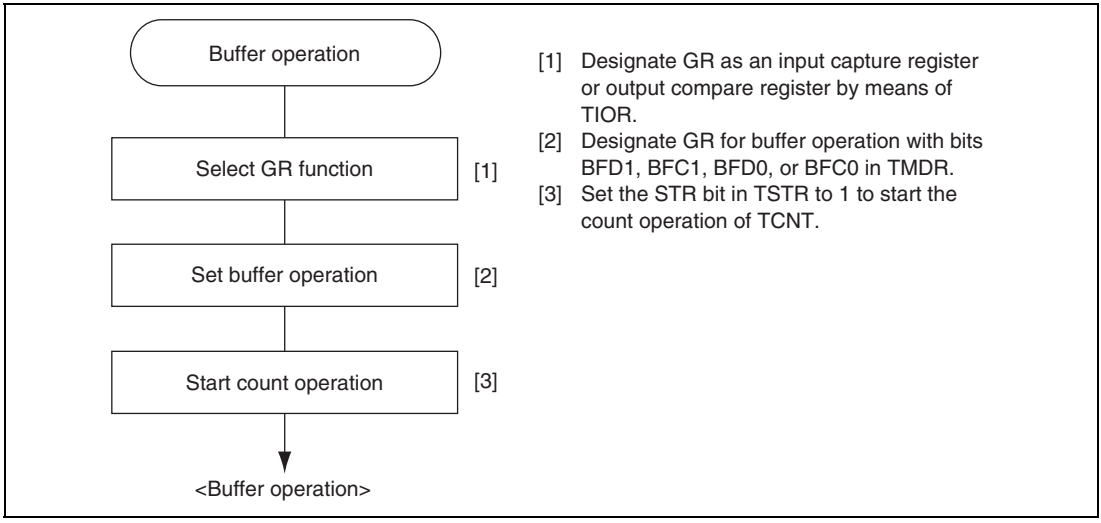


Figure 13.37 Example of Buffer Operation Setting Procedure

6. Examples of Buffer Operation

Figure 13.38 shows an operation example in which GRA has been designated as an output compare register, and buffer operation has been designated for GRA and GRC.

This is an example of TCNT operating as a periodic counter cleared by compare match B.

Pins FTIOA and FTIOB are set for toggle output by compare match A and B.

As buffer operation has been set, when compare match A occurs, the FTIOA pin performs toggle outputs and the value in buffer register is simultaneously transferred to the general register. This operation is repeated each time that compare match A occurs.

The timing to transfer data is shown in figure 13.39.

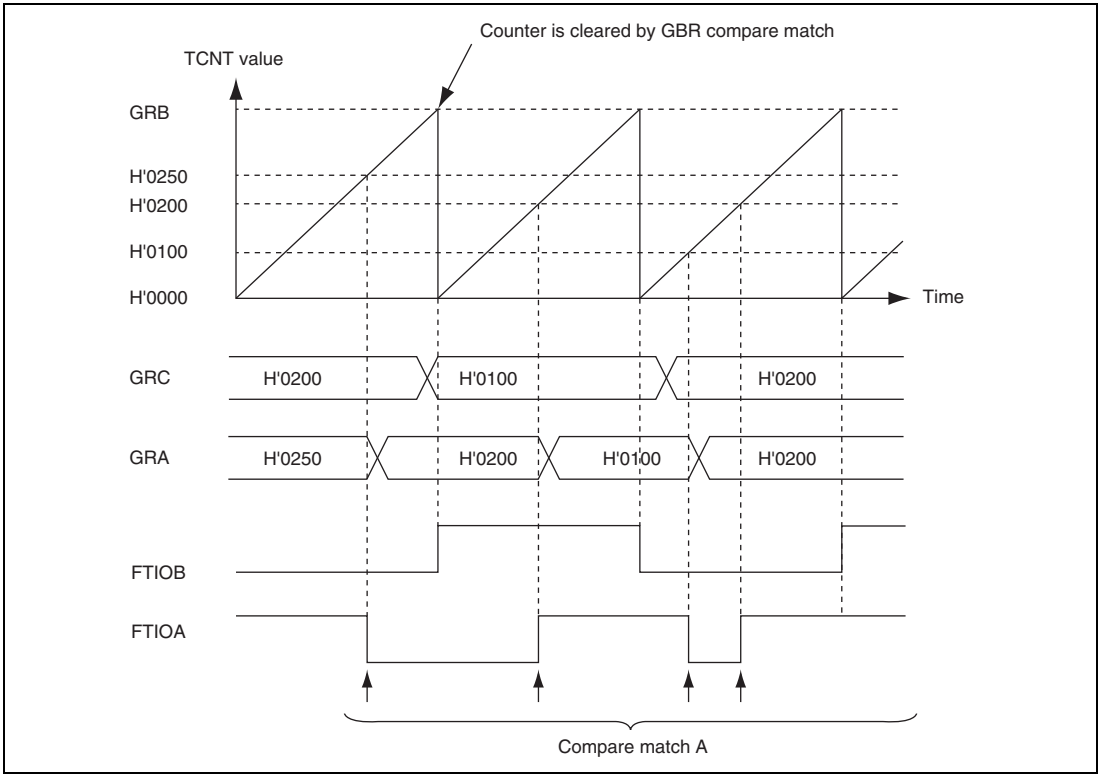


Figure 13.38 Example of Buffer Operation (1)
(Buffer Operation for Output Compare Register)

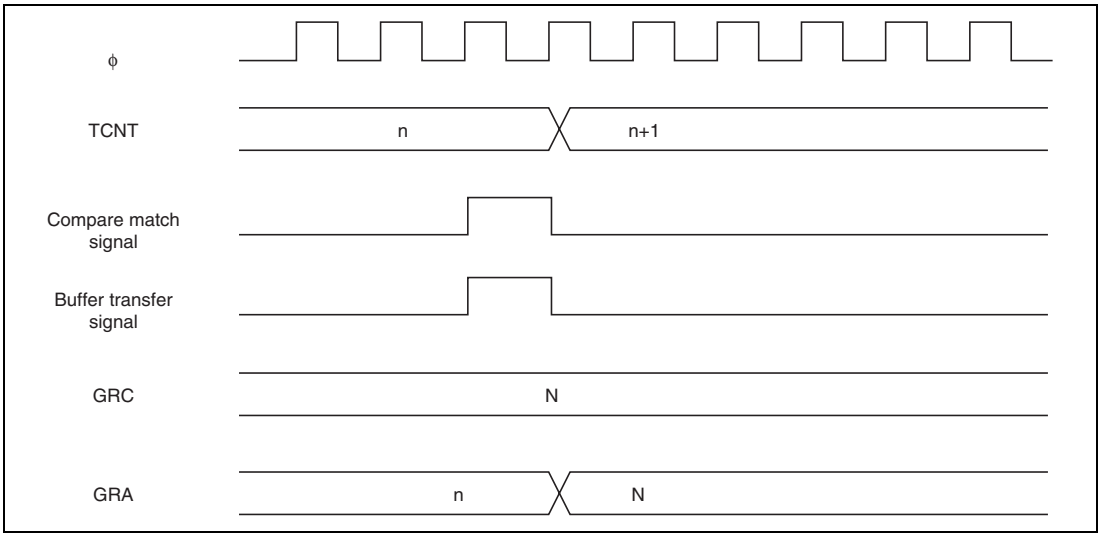


Figure 13.39 Example of Compare Match Timing for Buffer Operation

Figure 13.40 shows an operation example in which GRA has been designated as an input capture register, and buffer operation has been designated for GRA and GRC.

Counter clearing by input capture B has been set for TCNT, and falling edges have been selected as the FIOCB pin input capture input edge. And both rising and falling edges have been selected as the FIOCA pin input capture input edge.

As buffer operation has been set, when the TCNT value is stored in GRA upon the occurrence of input capture A, the value previously stored in GRA is simultaneously transferred to GRC. The transfer timing is shown in figure 13.41.

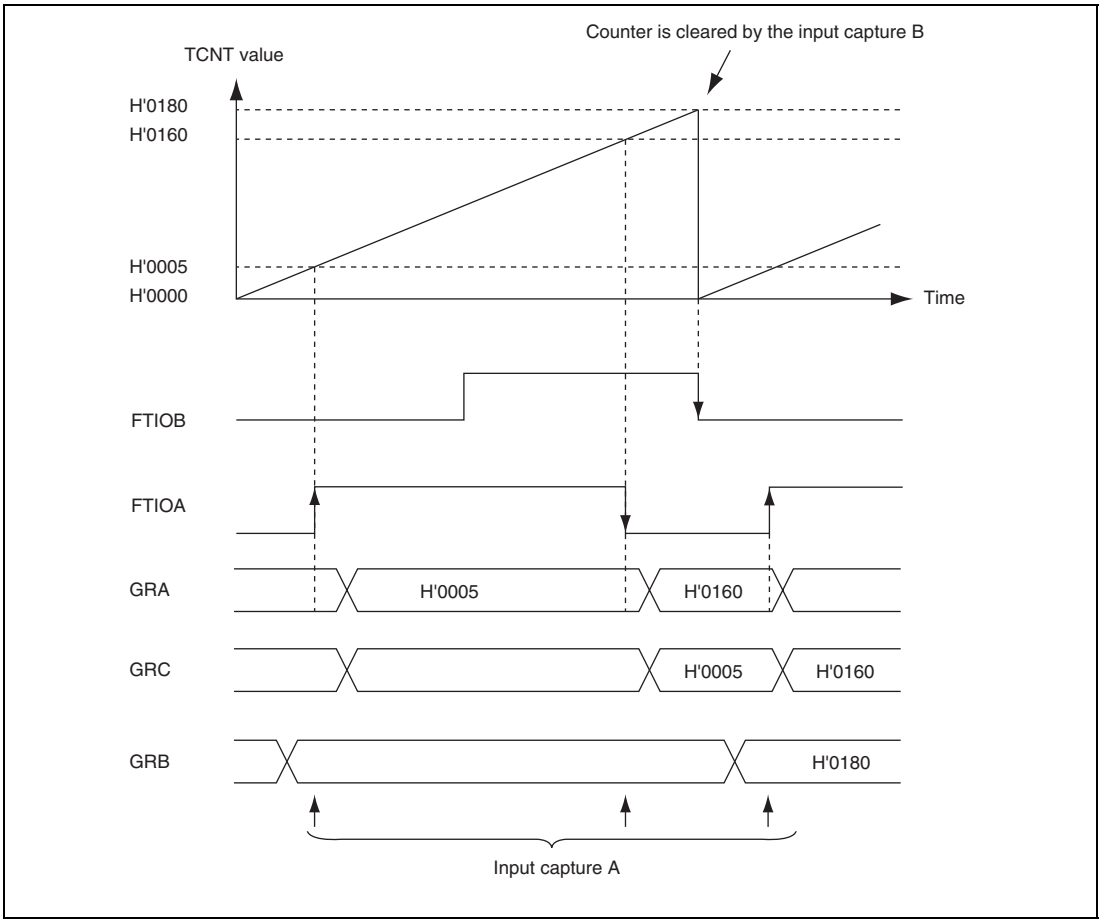


Figure 13.40 Example of Buffer Operation (2)
(Buffer Operation for Input Capture Register)

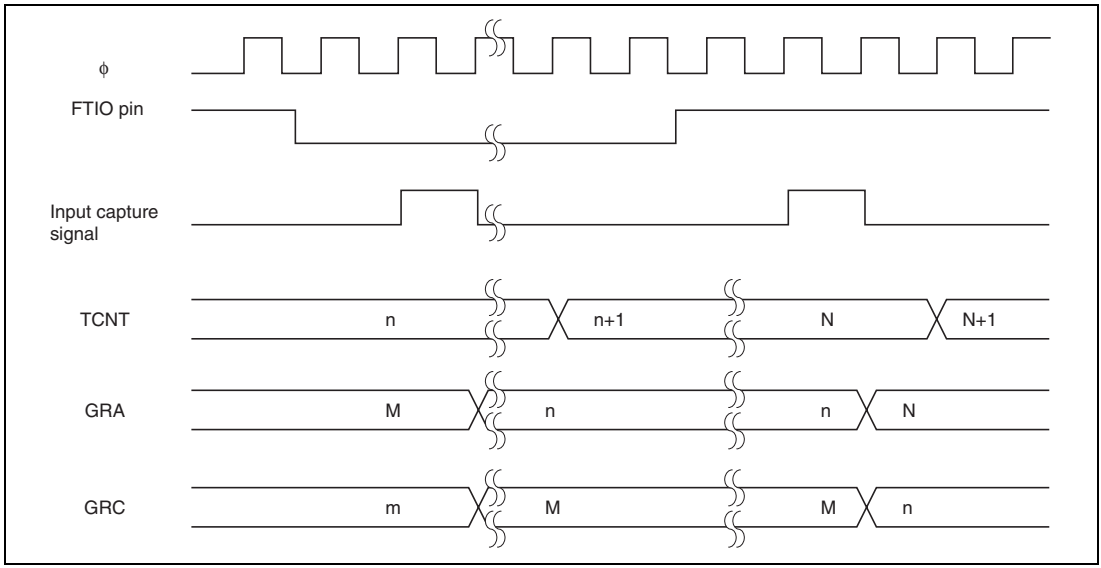


Figure 13.41 Input Capture Timing of Buffer Operation

Figures 13.42 and 13.43 show the operation examples when buffer operation has been designated for GRB_0 and GRD_0 in complementary PWM mode. These are examples when a PWM waveform of 0% duty is created by using the buffer operation and performing GRD_0 ≥ GRA_0. Data is transferred from GRD_0 to GRB_0 according to the settings of CMD_0 and CMD_1 when TCNT_0 and GRA_0 are compared and their contents match or when TCNT_1 underflows. However, when GRD_0 ≥ GRA_0, data is transferred from GRD_0 to GRB_0 when TCNT_1 underflows regardless of the setting of CMD_0 and CMD_1. When GRD_0 = H'0000, data is transferred from GRD_0 to GRB_0 when TCNT_0 and GRA_0 are compared and their contents match regardless of the settings of CMD_0 and CMD_1.

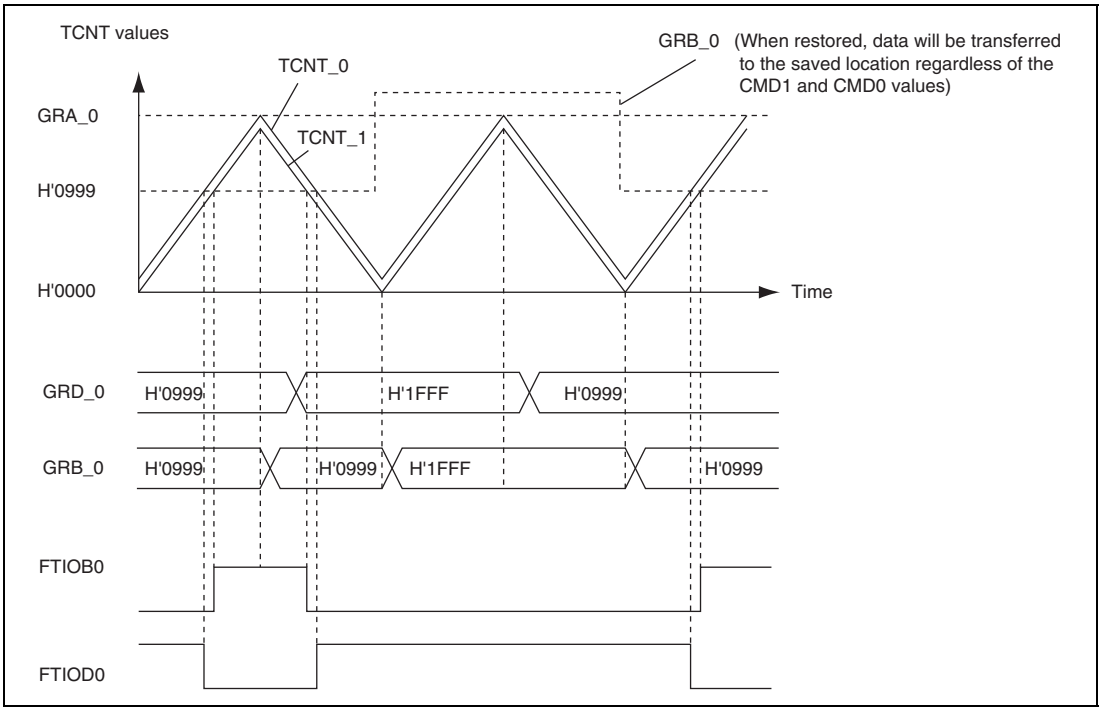


Figure 13.42 Buffer Operation (3)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1)

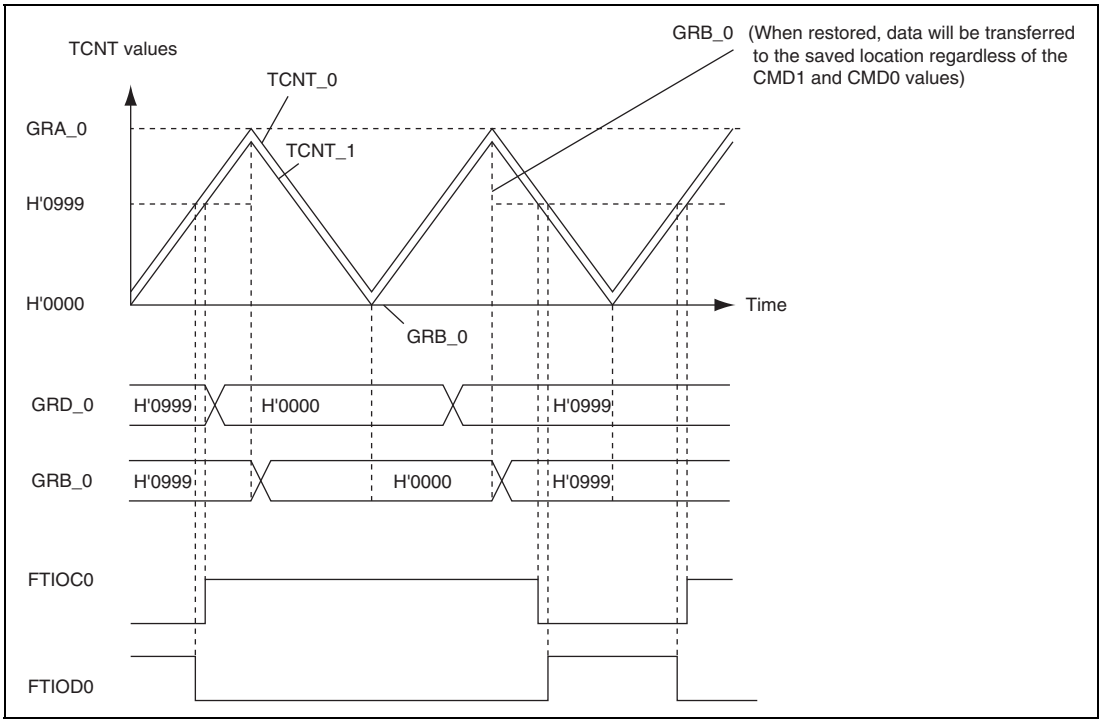


Figure 13.43 Buffer Operation (4)
(Buffer Operation in Complementary PWM Mode CMD1 = CMD0 = 1)

13.4.9 Timer Z Output Timing

The outputs of channels 0 and 1 can be disabled or inverted by the settings of TOER and TOCR and the external level.

1. Output Disable/Enable Timing of Timer Z by TOER: Setting the master enable bit in TOER to 1 disables the output of timer Z. By setting the PCR and PDR of the corresponding I/O port beforehand, any value can be output. Figure 13.44 shows the timing to enable or disable the output of timer Z by TOER.

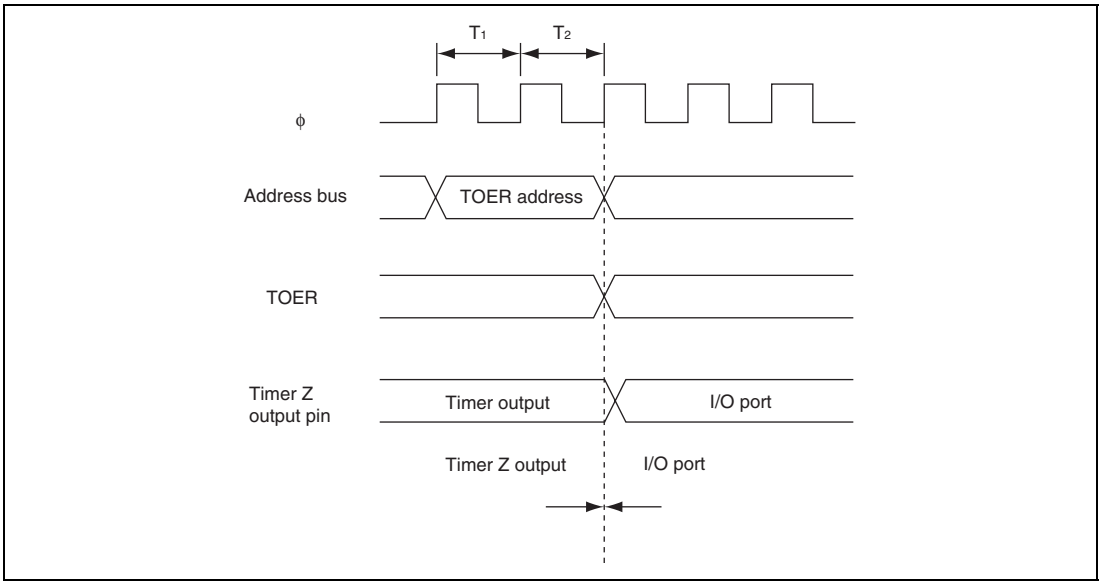


Figure 13.44 Example of Output Disable Timing of Timer Z by Writing to TOER

2. Output Disable Timing of Timer Z by External Trigger: When P54/$\overline{\text{WKP4}}$ is set as a $\overline{\text{WKP4}}$ input pin, and low level is input to $\overline{\text{WKP4}}$, the master enable bit in TOER is set to 1 and the output of timer Z will be disabled.

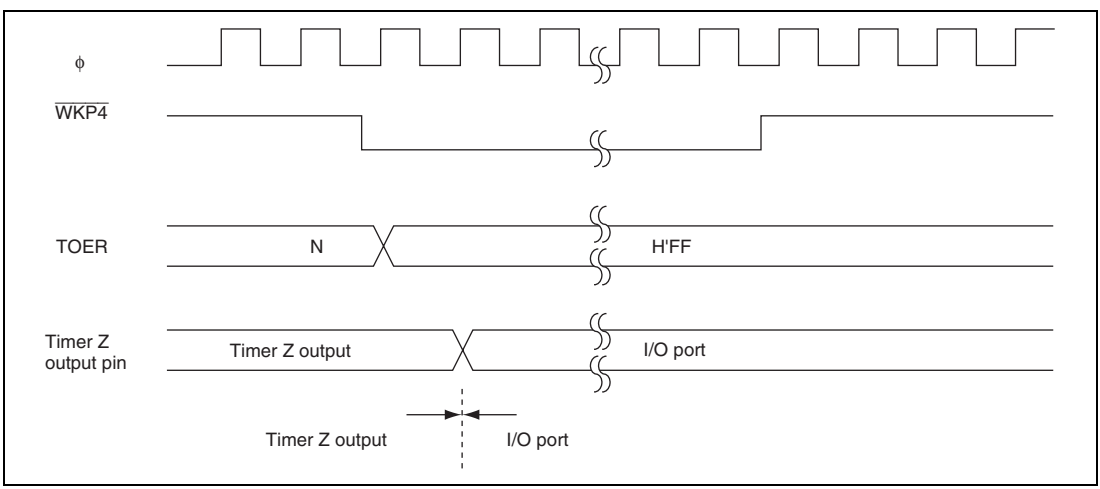


Figure 13.45 Example of Output Disable Timing of Timer Z by External Trigger

3. Output Inverse Timing by TFCR: The output level can be inverted by inverting the OLS1 and OLS0 bits in TFCR in reset synchronous PWM mode or complementary PWM mode. Figure 13.46 shows the timing.

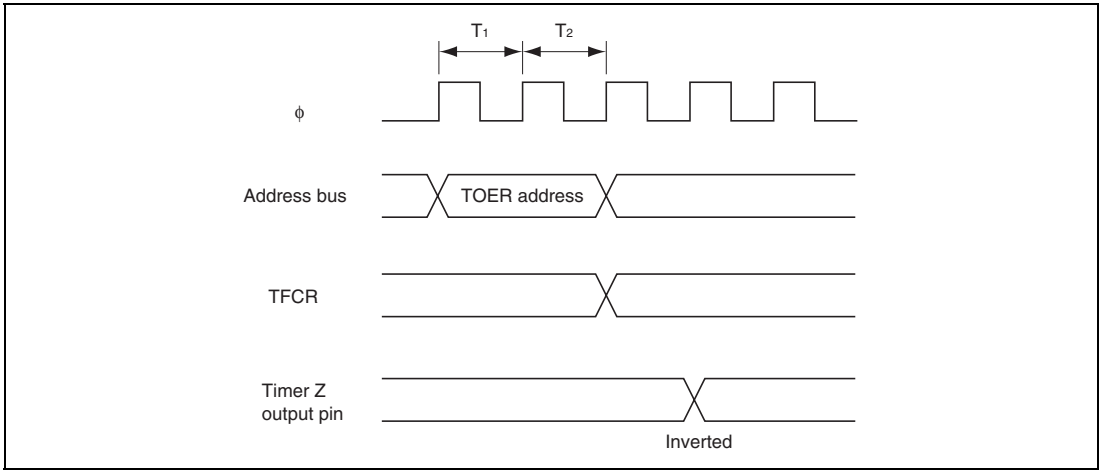


Figure 13.46 Example of Output Inverse Timing of Timer Z by Writing to TFCR

4. Output Inverse Timing by POCR: The output level can be inverted by inverting the POLD, POLC, and POLB bits in POCR in PWM mode. Figure 13.47 shows the timing.

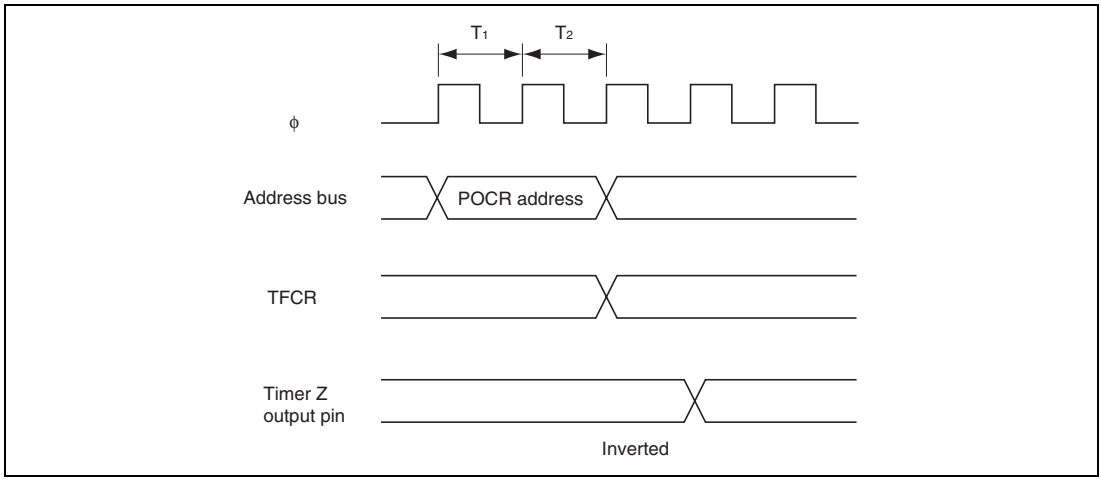


Figure 13.47 Example of Output Inverse Timing of Timer Z by Writing to POCR

13.5 Interrupts

There are three kinds of timer Z interrupt sources; input capture/compare match, overflow, and underflow. An interrupt is requested when the corresponding interrupt request flag is set to 1 while the corresponding interrupt enable bit is set to 1.

13.5.1 Status Flag Set Timing

1. IMF Flag Set Timing: The IMF flag is set to 1 by the compare match signal that is generated when the GR matches with the TCNT. The compare match signal is generated at the last state of matching (timing to update the counter value when the GR and TCNT match). Therefore, when the TCNT and GR matches, the compare match signal will not be generated until the TCNT input clock is generated. Figure 13.48 shows the timing to set the IMF flag.

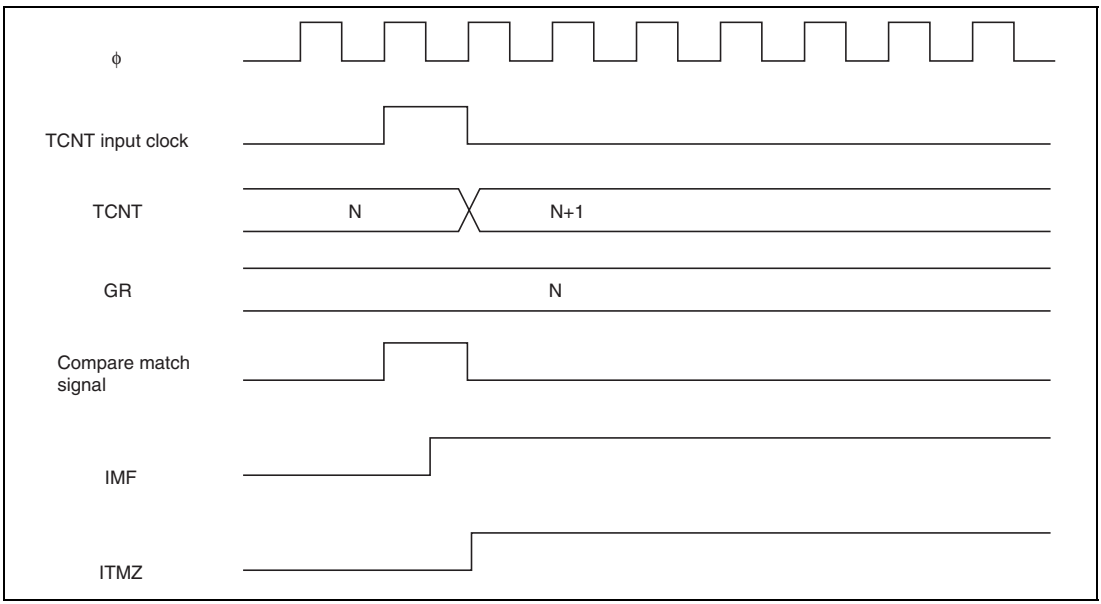


Figure 13.48 IMF Flag Set Timing when Compare Match Occurs

2. IMF Flag Set Timing at Input Capture: When an input capture signal is generated, the IMF flag is set to 1 and the value of TCNT is simultaneously transferred to corresponding GR. Figure 13.49 shows the timing.

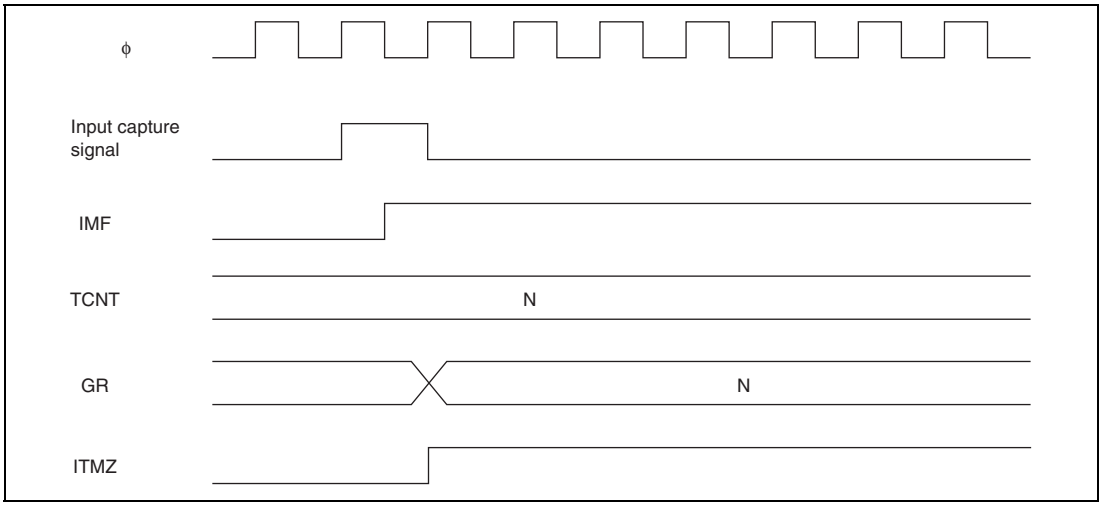


Figure 13.49 IMF Flag Set Timing at Input Capture

3. Overflow Flag (OVF) Set Timing: The overflow flag is set to 1 when the TCNT overflows. Figure 13.50 shows the timing.

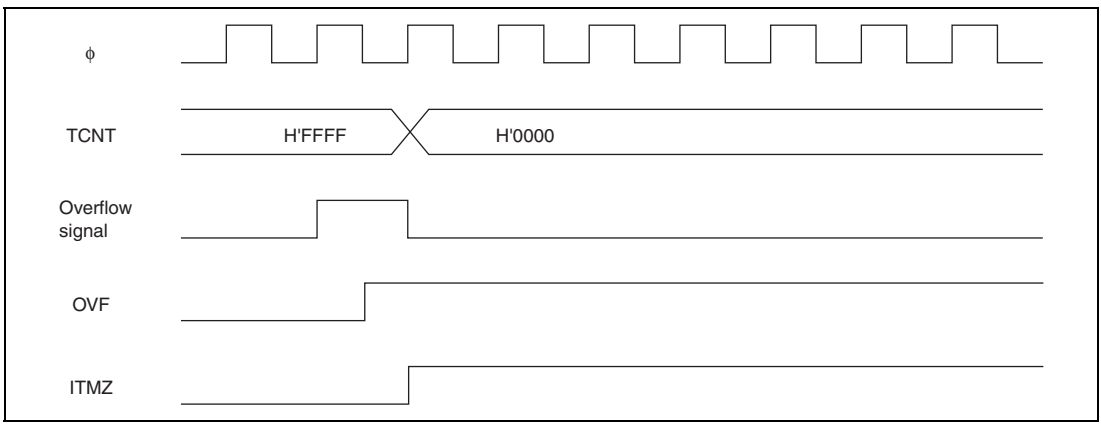


Figure 13.50 OVF Flag Set Timing

13.5.2 Status Flag Clearing Timing

The status flag can be cleared by writing 0 after reading 1 from the CPU. Figure 13.51 shows the timing in this case.

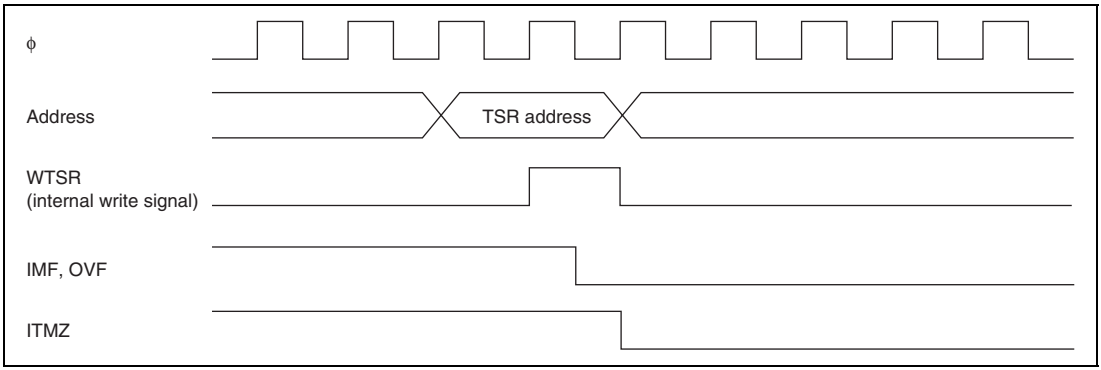


Figure 13.51 Status Flag Clearing Timing

13.6 Usage Notes

1. Contention between TCNT Write and Clear Operations: If a counter clear signal is generated in the T_2 state of a TCNT write cycle, TCNT clearing has priority and the TCNT write is not performed. Figure 13.52 shows the timing in this case.

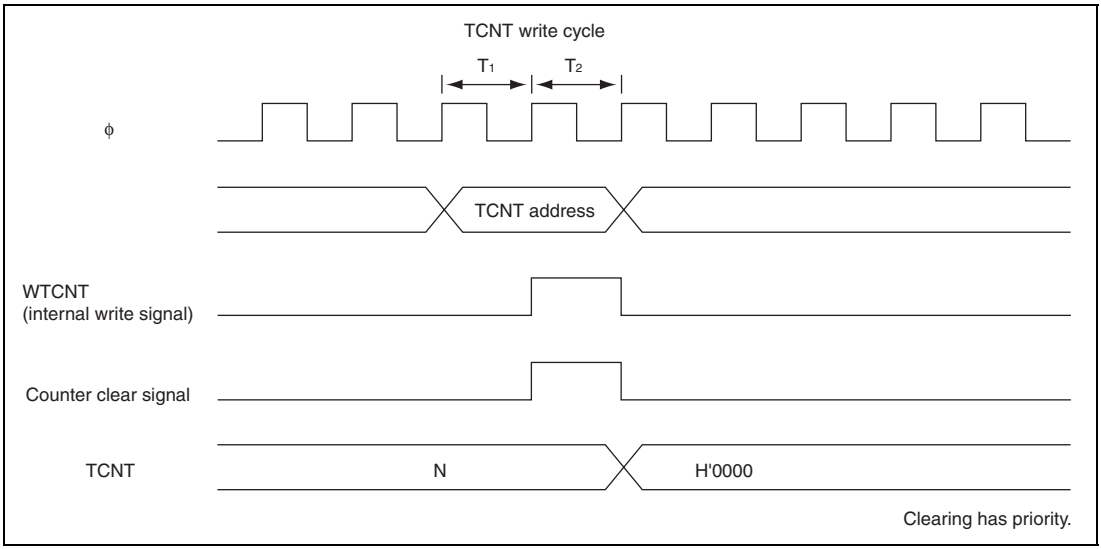


Figure 13.52 Contention between TCNT Write and Clear Operations

2. Contention between TCNT Write and Increment Operations: If incrementation is done in T_2 state of a TCNT write cycle, TCNT writing has priority. Figure 13.53 shows the timing in this case.

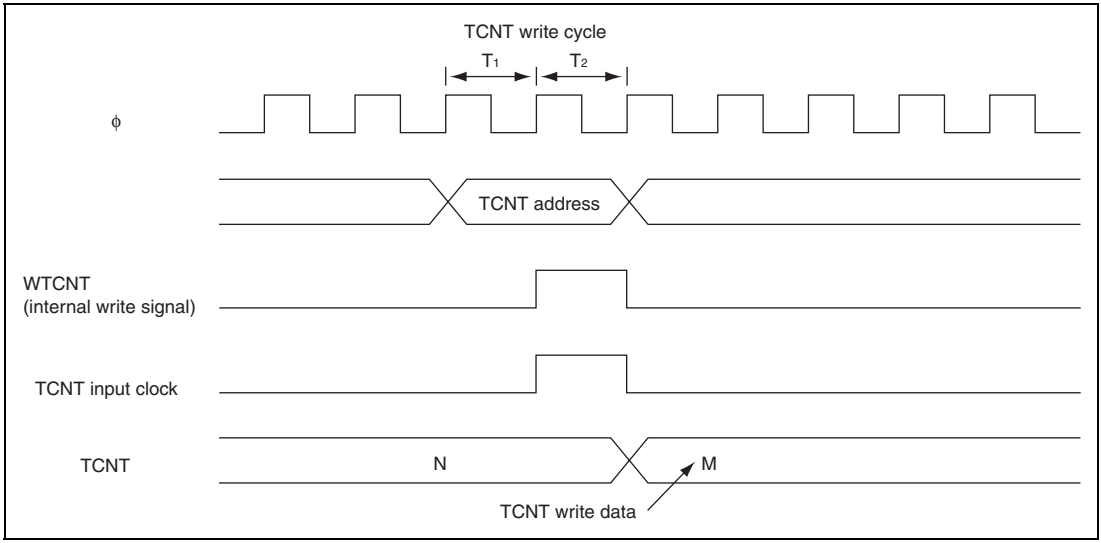


Figure 13.53 Contention between TCNT Write and Increment Operations

3. Contention between GR Write and Compare Match: If a compare match occurs in the T_2 state of a GR write cycle, GR write has priority and the compare match signal is disabled. Figure 13.54 shows the timing in this case.

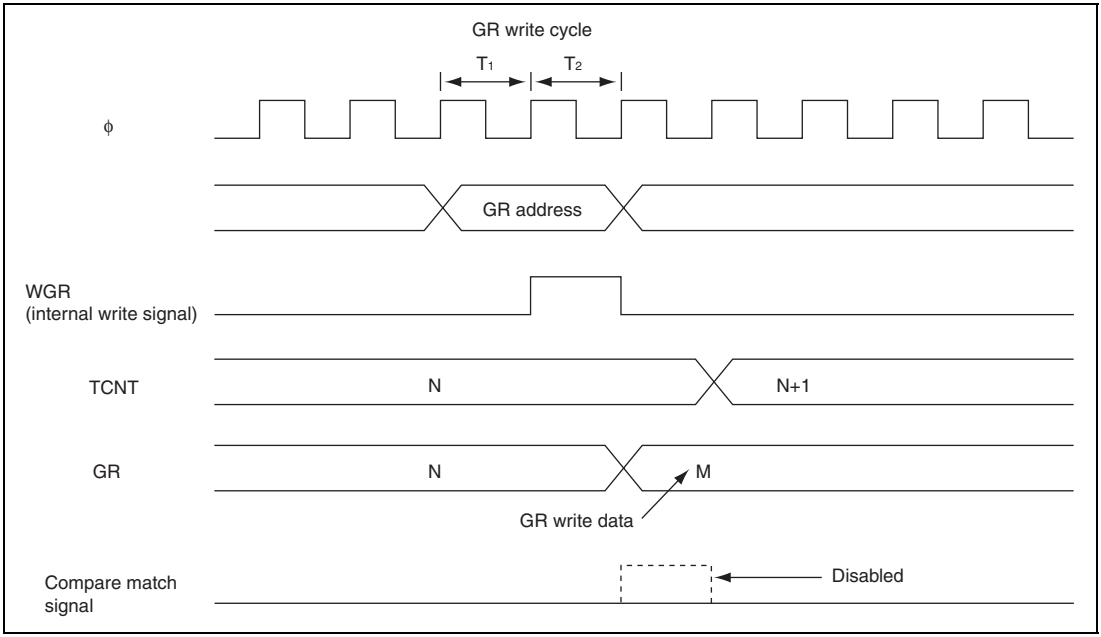


Figure 13.54 Contention between GR Write and Compare Match

4. Contention between TCNT Write and Overflow/Underflow: If overflow/underflow occurs in the T_2 state of a TCNT write cycle, TCNT write has priority without an increment operation. At this time, the OVF flag is set to 1. Figure 13.55 shows the timing in this case.

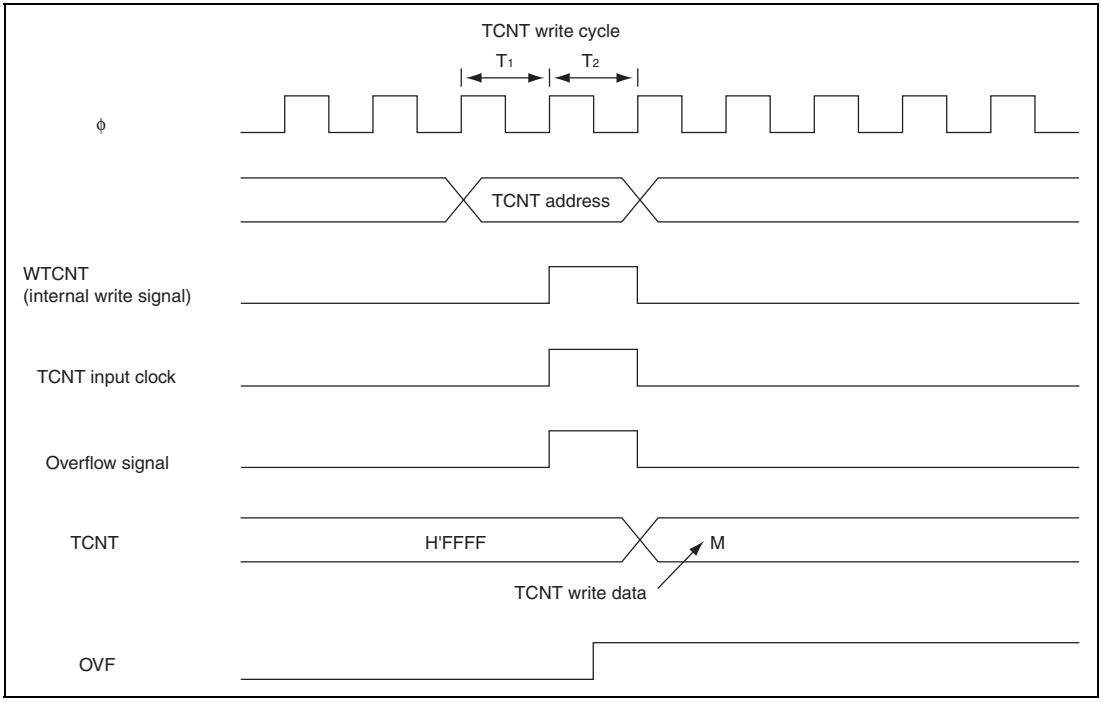


Figure 13.55 Contention between TCNT Write and Overflow

5. Contention between GR Read and Input Capture: If an input capture signal is generated in the T_1 state of a GR read cycle, the data that is read will be transferred before input capture transfer. Figure 13.56 shows the timing in this case.

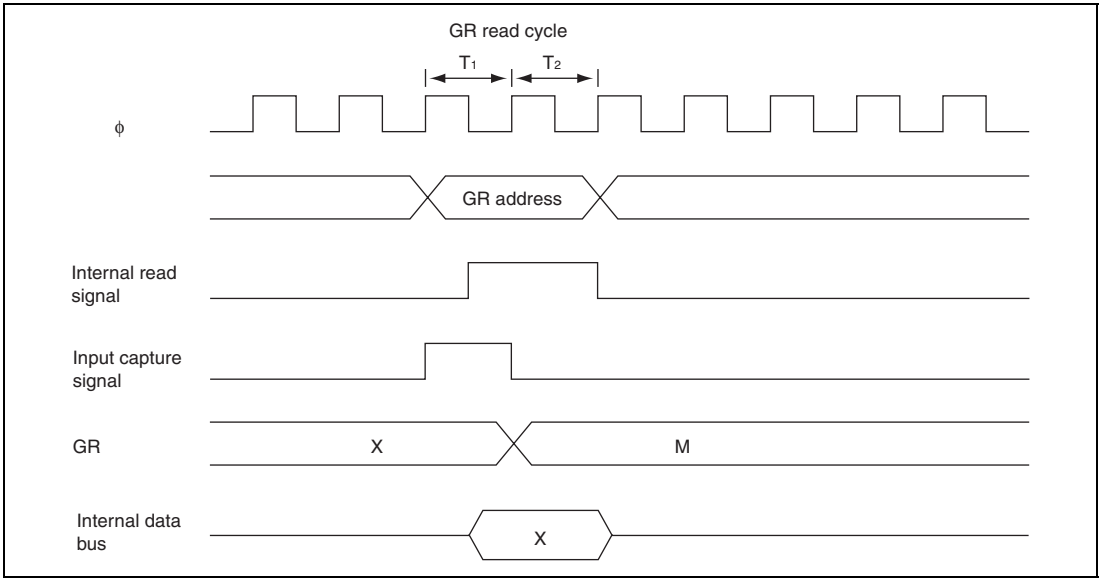


Figure 13.56 Contention between GR Read and Input Capture

6. Contention between Count Clearing and Increment Operations by Input Capture: If an input capture and increment signals are simultaneously generated, count clearing by the input capture operation has priority without an increment operation. The TCNT contents before clearing counter are transferred to GR. Figure 13.57 shows the timing in this case.

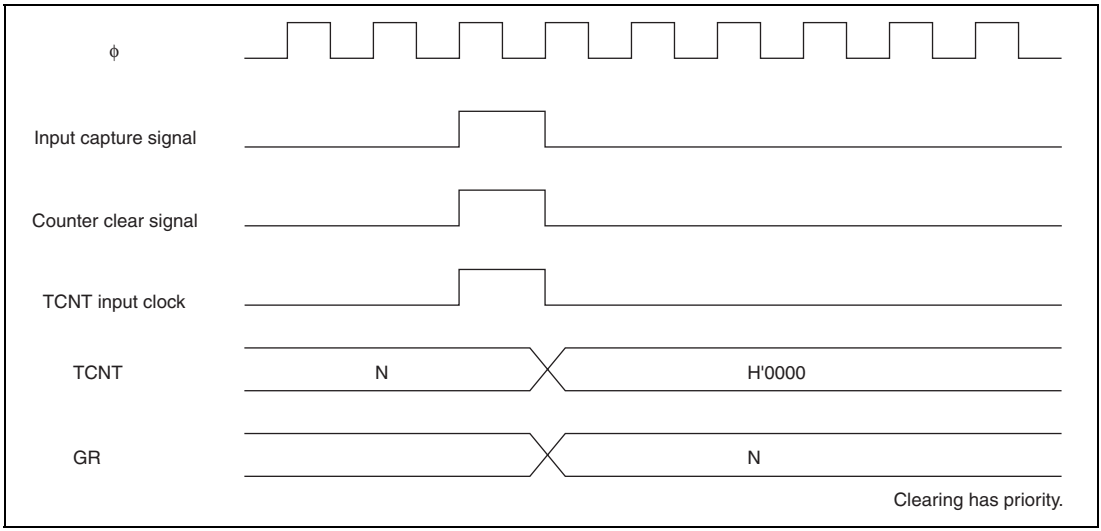


Figure 13.57 Contention between Count Clearing and Increment Operations by Input Capture

7. Contention between GR Write and Input Capture: If an input capture signal is generated in the T_2 state of a GR write cycle, the input capture operation has priority and the write to GR is not performed. Figure 13.58 shows the timing in this case.

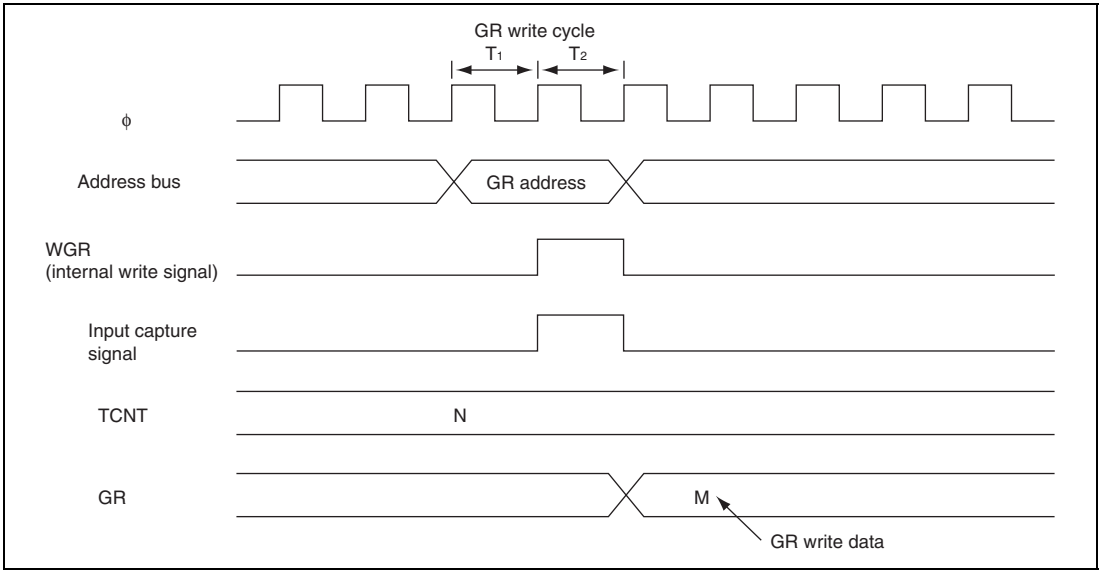


Figure 13.58 Contention between GR Write and Input Capture

8. Notes on Setting Reset Synchronous PWM Mode/Complementary PWM Mode: When bits CMD1 and CMD0 in TFCR are set, note the following:
   A. Write bits CMD1 and CMD0 while TCNT_1 and TCNT_0 are halted.
   B. Changing the settings of reset synchronous PWM mode to complementary PWM mode or vice versa is disabled. Set reset synchronous PWM mode or complementary PWM mode after the normal operation (bits CMD1 and CMD0 are cleared to 0) has been set.

9. Note on Clearing TSR Flag: When a specific flag in TSR is cleared, a combination of the BCLR or MOV instructions is used to read 1 from the flag and then write 0 to the flag. However, if another bit is set during this processing, the bit may also be cleared simultaneously. To avoid this, the following processing that does not use the BCLR instruction must be executed. Note that this note is only applied to the F-ZTAT version. This problem has already been solved in the mask ROM version.

   Example: When clearing bit 4 (OVF) in TSR

   ```
   MOV.B @TSR,R0L
   MOV.B #B'11101111, R0L <---- Only the bit to be cleared is 0 and
                                the other bits are all set to 1.
   MOV.B R0L,@TSR
   ```

10. Note on Writing to the TOA0 to TOD0 Bits and the TOA1 to TOD1 Bits in TOCR:

    The TOA0 to TOD0 bits and the TOA1 to TOD1 bits in TOCR decide the value of the FTIO pin, which is output until the first compare match occurs. Once a compare match occurs and this compare match changes the values of FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 output, the values of the FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pin output and the values read from the TOA0 to TOD0 and TOA1 to TOD1 bits may differ. Moreover, when the writing to TOCR and the generation of the compare match A0 to D0 and A1 to D1 occur at the same timing, the writing to TOCR has the priority. Thus, output change due to the compare match is not reflected to the FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pins. Therefore, when bit manipulation instruction is used to write to TOCR, the values of the FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 pin output may result in an unexpected result. When TOCR is to be written to while compare match is operating, stop the counter once before accessing to TOCR, read the port 6 state to reflect the values of FTIOA0 to FTIOD0 and FTIOA1 to FTIOD1 output, to TOA0 to TOD0 and TOA1 to TOD1, and then restart the counter. Figure 13.59 shows an example when the compare match and the bit manipulation instruction to TOCR occur at the same timing.

TOCR has been set to H'06. Compare match B0 and compare match C0 are used. The FTIOB0 pin is in the 1 output state, and is set to the toggle output or the 0 output by compare match B0.
When BCLR#2, @TOCR is executed to clear the TOC0 bit (the FTIOC0 signal is low) and compare match B0 occurs at the same timing as shown below, the H'02 writing to TOCR has priority and compare match B0 does not drive the FTIOB0 signal low; the FTIOB0 signal remains high.

Bit	7	6	5	4	3	2	1	0
TOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0
Set value	0	0	0	0	0	1	1	0

BCLR#2, @TOCR
(1) TOCR read operation: Read H'06
(2) Modify operation: Modify H'06 to H'02
(3) Write operation to TOCR: Write H'02

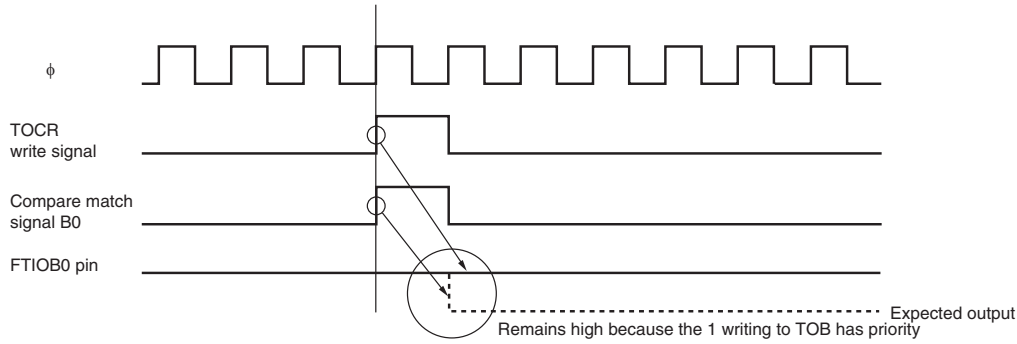


Figure 13.59 When Compare Match and Bit Manipulation Instruction to TOCR Occur at the Same Timing

Section 14 Watchdog Timer

The watchdog timer 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.

The block diagram of the watchdog timer is shown in figure 14.1.

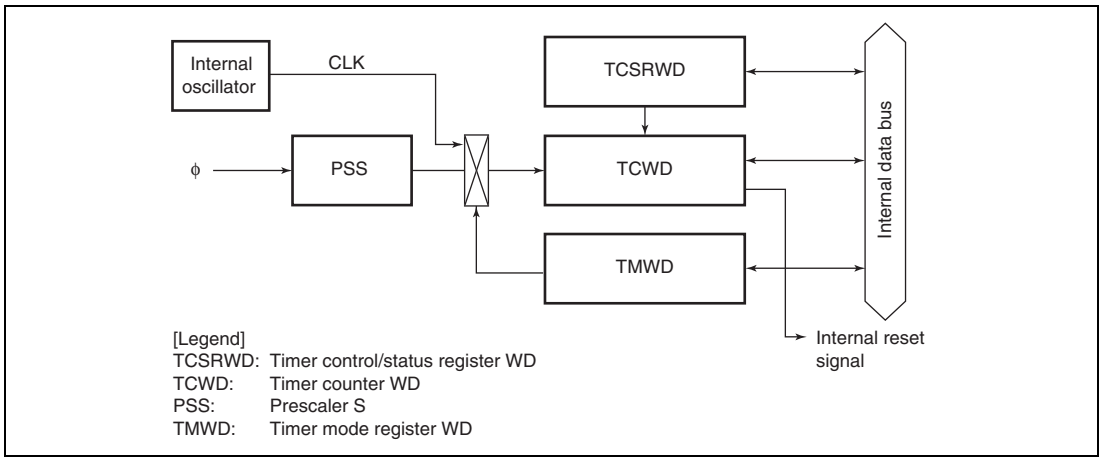


Figure 14.1 Block Diagram of Watchdog Timer

14.1 Features

- Selectable from nine counter input clocks.
  Eight clock sources (ϕ/64, ϕ/128, ϕ/256, ϕ/512, ϕ/1024, ϕ/2048, ϕ/4096, and ϕ/8192) or the internal oscillator can be selected as the timer-counter clock. When the internal oscillator is selected, it can operate as the watchdog timer in any operating mode.
- Reset signal generated on counter overflow
  An overflow period of 1 to 256 times the selected clock can be set.

14.2 Register Descriptions

The watchdog timer has the following registers.

- Timer control/status register WD (TCSRWD)
- Timer counter WD (TCWD)
- Timer mode register WD (TMWD)

14.2.1 Timer Control/Status Register WD (TCSRWD)

TCSRWD performs the TCSRWD and TCWD write control. TCSRWD also controls the watchdog timer operation and indicates the operating state. TCSRWD must be rewritten by using the MOV instruction. The bit manipulation instruction cannot be used to change the setting value.

Bit	Bit Name	Initial Value	R/W	Description
7	B6WI	1	R/W	Bit 6 Write Inhibit The TCWE bit can be written only when the write value of the B6WI bit is 0. This bit is always read as 1.
6	TCWE	0	R/W	Timer Counter WD Write Enable TCWD can be written when the TCWE bit is set to 1. When writing data to this bit, the value for bit 7 must be 0.
5	B4WI	1	R/W	Bit 4 Write Inhibit The TCSRWE bit can be written only when the write value of the B4WI bit is 0. This bit is always read as 1.
4	TCSRWE	0	R/W	Timer Control/Status Register WD Write Enable The WDON and WRST bits can be written when the TCSRWE bit is set to 1. When writing data to this bit, the value for bit 5 must be 0.
3	B2WI	1	R/W	Bit 2 Write Inhibit This bit can be written to the WDON bit only when the write value of the B2WI bit is 0. This bit is always read as 1.
2	WDON	0	R/W	Watchdog Timer On TCWD starts counting up when WDON is set to 1 and halts when WDON is cleared to 0. [Setting condition] When 1 is written to the WDON bit while writing 0 to the B2WI bit when the TCSRWE bit=1 [Clearing conditions] • Reset by $\overline{\text{RES}}$ pin • When 0 is written to the WDON bit while writing 0 to the B2WI when the TCSRWE bit=1
1	B0WI	1	R/W	Bit 0 Write Inhibit This bit can be written to the WRST bit only when the write value of the B0WI bit is 0. This bit is always read as 1.

Bit	Bit Name	Initial Value	R/W	Description
0	WRST	0	R/W	Watchdog Timer Reset [Setting condition] When TCWD overflows and an internal reset signal is generated [Clearing conditions] • Reset by $\overline{\text{RES}}$ pin • When 0 is written to the WRST bit while writing 0 to the B0WI bit when the TCSRWE bit=1

14.2.2 Timer Counter WD (TCWD)

TCWD is an 8-bit readable/writable up-counter. When TCWD overflows from H'FF to H'00, the internal reset signal is generated and the WRST bit in TCSRWD is set to 1. TCWD is initialized to H'00.

14.2.3 Timer Mode Register WD (TMWD)

TMWD selects the input clock.

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 1	—	Reserved These bits are always read as 1.
3	CKS3	1	R/W	Clock Select 3 to 0
2	CKS2	1	R/W	Select the clock to be input to TCWD.
1	CKS1	1	R/W	1000: Internal clock: counts on ϕ/64
0	CKS0	1	R/W	1001: Internal clock: counts on ϕ/128 1010: Internal clock: counts on ϕ/256 1011: Internal clock: counts on ϕ/512 1100: Internal clock: counts on ϕ/1024 1101: Internal clock: counts on ϕ/2048 1110: Internal clock: counts on ϕ/4096 1111: Internal clock: counts on ϕ8192 0XXX: Internal oscillator For the internal oscillator overflow periods, see section 23, Electrical Characteristics.

Legend: X: Don't care.

14.3 Operation

The watchdog timer is provided with an 8-bit counter. If 1 is written to WDON while writing 0 to B2WI when the TCSRWE bit in TCSRWD is set to 1, TCWD begins counting up. (To operate the watchdog timer, two write accesses to TCSRWD are required.) When a clock pulse is input after the TCWD count value has reached H'FF, the watchdog timer overflows and an internal reset signal is generated. The internal reset signal is output for a period of 256 ϕ_{osc} clock cycles. TCWD is a writable counter, and when a value is set in TCWD, the count-up starts from that value. An overflow period in the range of 1 to 256 input clock cycles can therefore be set, according to the TCWD set value.

Figure 14.2 shows an example of watchdog timer operation.

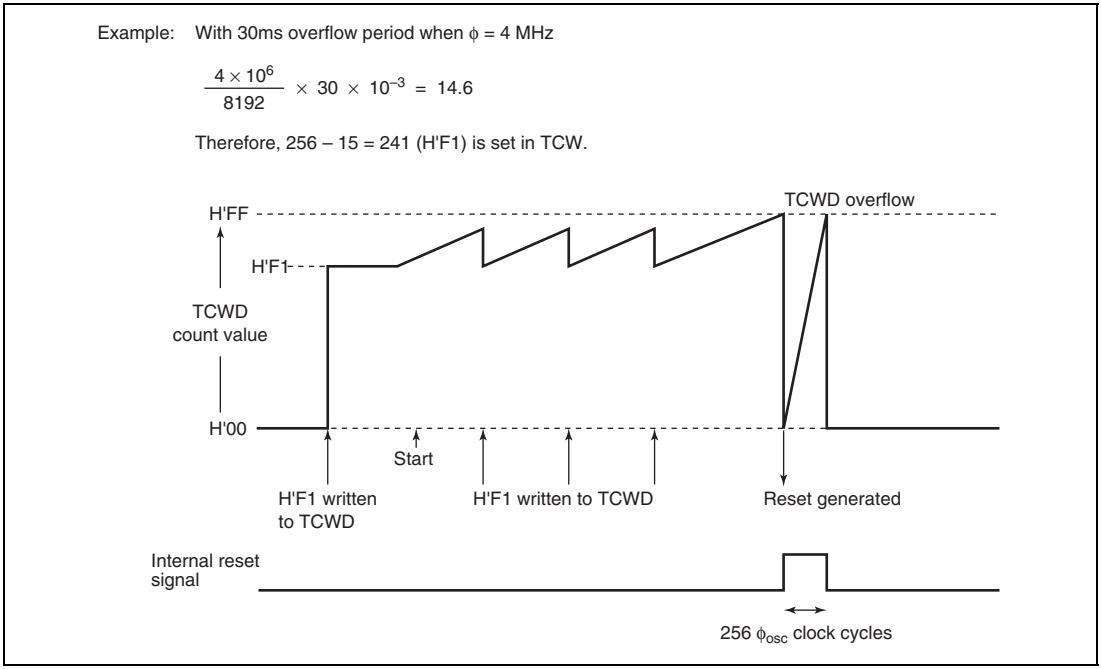


Figure 14.2 Watchdog Timer Operation Example

Section 15 14-Bit PWM

The 14-bit PWM is a pulse division type PWM that can be used for electronic tuner control, etc. Figure 15.1 shows a block diagram of the 14-bit PWM.

15.1 Features

- Choice of two conversion periods

  A conversion period of 32768/ϕ with a minimum modulation width of 2/ϕ, or a conversion period of 16384/ϕ with a minimum modulation width of 1/ϕ, can be selected.
- Pulse division method for less ripple

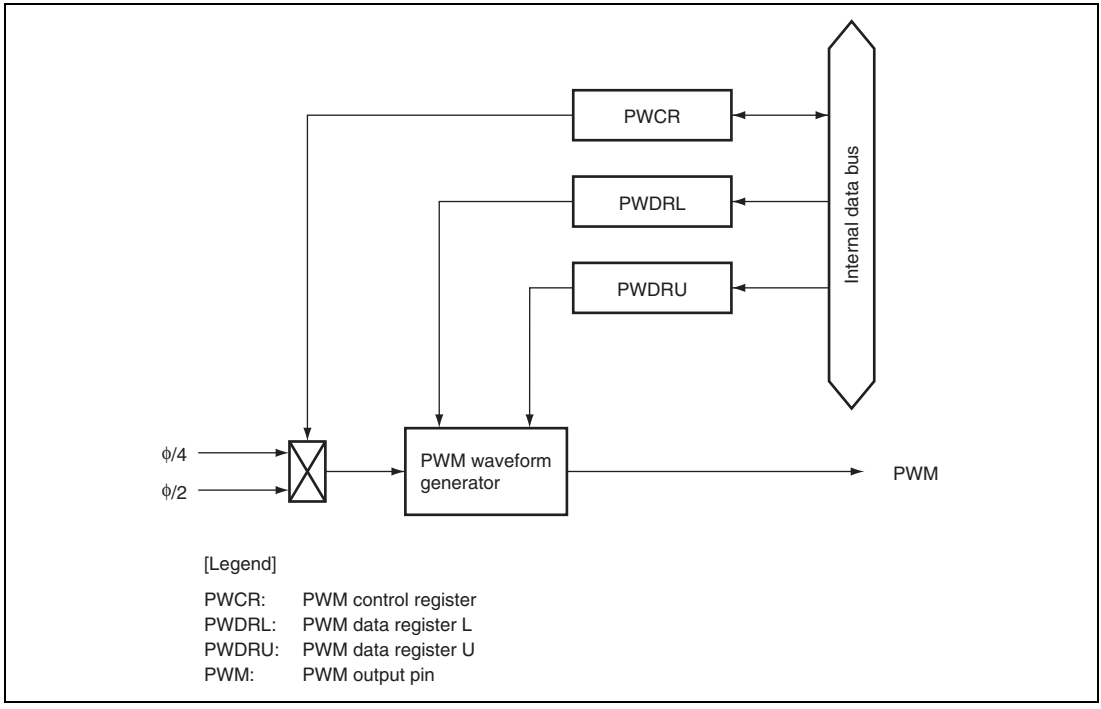


Figure 15.1 Block Diagram of 14-Bit PWM

15.2 Input/Output Pin

Table 15.1 shows the 14-bit PWM pin configuration.

Table 15.1 Pin Configuration

Name	Abbreviation	I/O	Function
14-bit PWM square-wave output	PWM	Output	14-bit PWM square-wave output pin

15.3 Register Descriptions

The 14-bit PWM has the following registers.

- PWM control register (PWCR)
- PWM data register U (PWDRU)
- PWM data register L (PWDRL)

15.3.1 PWM Control Register (PWCR)

PWCR selects the conversion period.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
0	PWCR0	0	R/W	Clock Select 0: The input clock is $\phi/2$ ($t\phi = 2/\phi$) — The conversion period is $16384/\phi$, with a minimum modulation width of $1/\phi$ 1: The input clock is $\phi/4$ ($t\phi = 4/\phi$) — The conversion period is $32768/\phi$, with a minimum modulation width of $2/\phi$

Legend: $t\phi$: Period of PWM clock input

15.3.2 PWM Data Registers U and L (PWDRU, PWDRL)

PWDRU and PWDRL indicate high level width in one PWM waveform cycle. PWDRU and PWDRL are 14-bit write-only registers, with the upper 6 bits assigned to PWDRU and the lower 8 bits to PWDRL. When read, all bits are always read as 1.

Both PWDRU and PWDRL are accessible only in bytes. Note that the operation is not guaranteed if word access is performed. When 14-bit data is written in PWDRU and PWDRL, the contents are latched in the PWM waveform generator and the PWM waveform generation data is updated. When writing the 14-bit data, the order is as follows: PWDRL to PWDRU.

PWDRU and PWDRL are initialized to H'C000.

15.4 Operation

When using the 14-bit PWM, set the registers in this sequence:

1. Set the PWM bit in the port mode register 1 (PMR1) to set the P11/PWM pin to function as a PWM output pin.
2. Set the PWCR0 bit in PWCR to select a conversion period of either.
3. Set the output waveform data in PWDRU and PWDRL. Be sure to write byte data first to PWDRL and then to PWDRU. When the data is written in PWDRU, the contents of these registers are latched in the PWM waveform generator, and the PWM waveform generation data is updated in synchronization with internal signals.

One conversion period consists of 64 pulses, as shown in figure 15.2. The total high-level width during this period (T_H) corresponds to the data in PWDRU and PWDRL. This relation can be expressed as follows:

$$T_H = (\text{data value in PWDRU and PWDRL} + 64) \times t\phi/2$$

where $t\phi$ is the period of PWM clock input: $2/\phi$ (bit PWCR0 = 0) or $4/\phi$ (bit PWCR0 = 1).
If the data value in PWDRU and PWDRL is from H'FFC0 to H'FFFF, the PWM output stays high. When the data value is H'C000, T_H is calculated as follows:

$$T_H = 64 \times t\phi/2 = 32\ t\phi$$

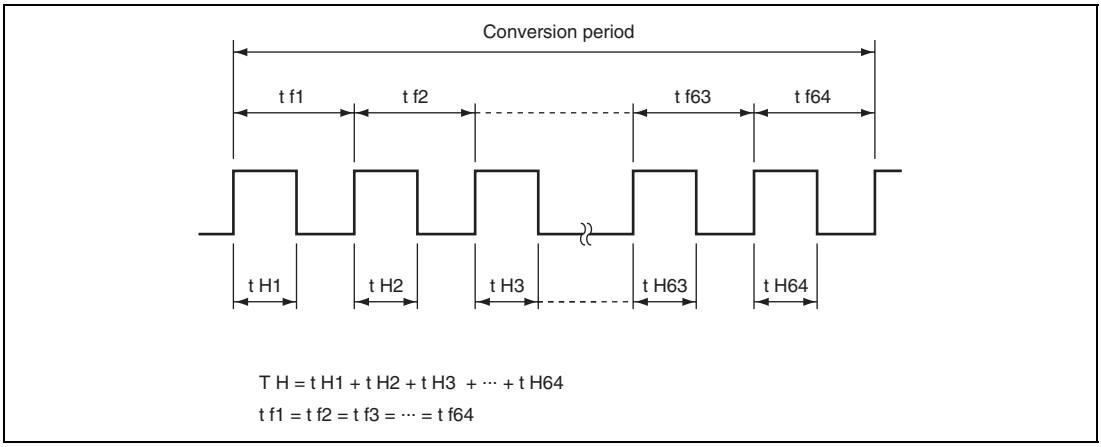


Figure 15.2 Waveform Output by 14-Bit PWM

Section 16 Serial Communication Interface 3 (SCI3)

This LSI includes a serial communication interface 3 (SCI3), which has independent two channels. The SCI3 can handle both asynchronous and clocked synchronous serial communication. In asynchronous mode, 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).

Table 16.1 shows the SCI3 channel configuration and figure 16.1 shows a block diagram of the SCI3. Since pin functions are identical for each of the two channels (SCI3 and SCI3_2), separate explanations are not given in this section.

16.1 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 or on-chip baud rate generator can be selected as a transfer clock source.
- Six interrupt sources

  Transmit-end, transmit-data-empty, receive-data-full, overrun error, framing error, and parity error.

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

Table 16.1 Channel Configuration

Channel	Abbreviation	Pin	Register	Register Address
Channel 1	SCI3*	SCK3 RXD TXD	SMR	H'FFA8
			BRR	H'FFA9
			SCR3	H'FFAA
			TDR	H'FFAB
			SSR	H'FFAC
			RDR	H'FFAD
			RSR	—
			TSR	—
Channel 2	SCI3_2	SCK3_2 RXD_2 TXD_2	SMR_2	H'F740
			BRR_2	H'F741
			SCR3_2	H'F742
			TDR_2	H'F743
			SSR_2	H'F744
			RDR_2	H'F745
			RSR_2	—
			TSR_2	—

Note: * The channel 1 of the SCI3 is used in on-board programming mode by boot mode.

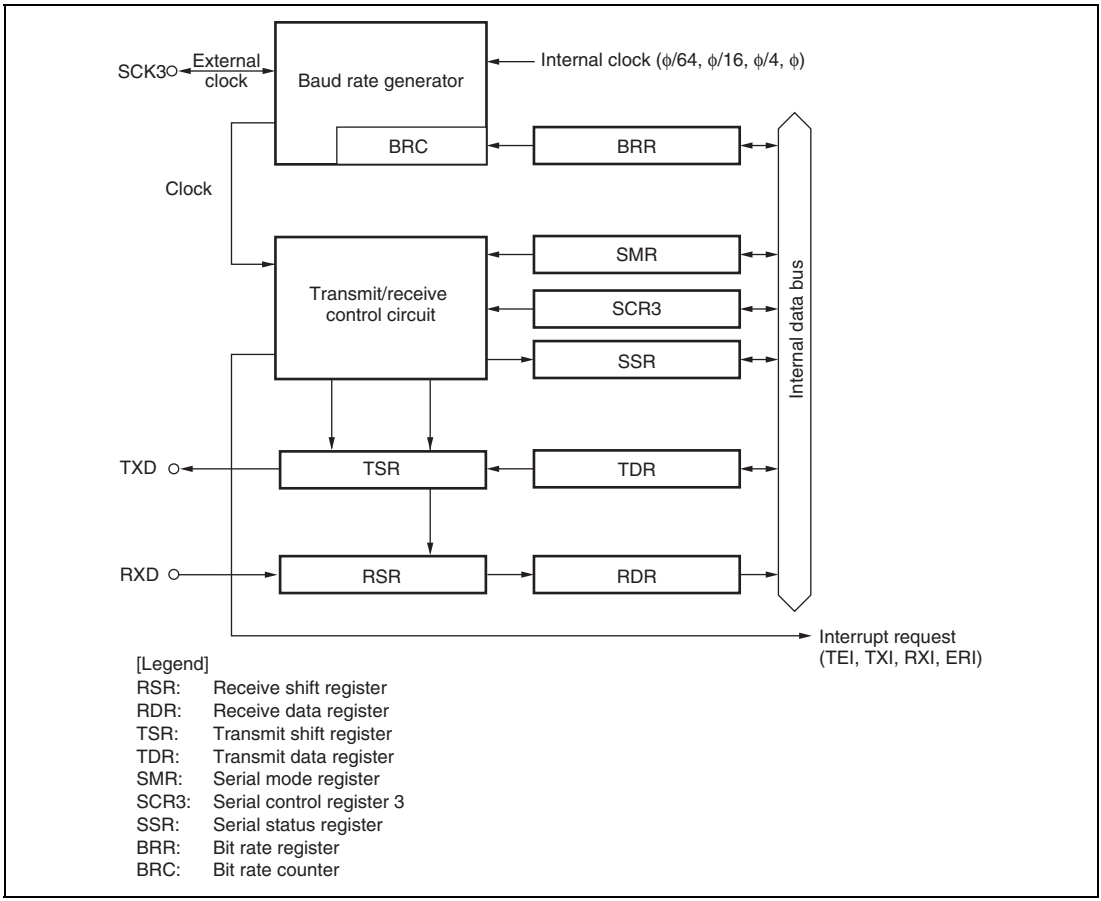


Figure 16.1 Block Diagram of SCI3

16.2 Input/Output Pins

Table 16.2 shows the SCI3 pin configuration.

Table 16.2 Pin Configuration

Pin Name	Abbreviation	I/O	Function
SCI3 clock	SCK3	I/O	SCI3 clock input/output
SCI3 receive data input	RXD	Input	SCI3 receive data input
SCI3 transmit data output	TXD	Output	SCI3 transmit data output

16.3 Register Descriptions

The SCI3 has the following registers for each channel.

- Receive Shift Register (RSR)
- Receive Data Register (RDR)
- Transmit Shift Register (TSR)
- Transmit Data Register (TDR)
- Serial Mode Register (SMR)
- Serial Control Register 3 (SCR3)
- Serial Status Register (SSR)
- Bit Rate Register (BRR)

16.3.1 Receive Shift Register (RSR)

RSR is a shift register that is used to receive serial data input from the RxD pin and convert it into parallel data. When one frame of data has been received, it is transferred to RDR automatically. RSR cannot be directly accessed by the CPU.

16.3.2 Receive Data Register (RDR)

RDR is an 8-bit register that stores received data. When the SCI3 has received one frame of serial data, it transfers the received serial data from RSR to RDR, where it is stored. After this, RSR is receive-enabled. As RSR and RDR function as a double buffer in this way, continuous receive operations are possible. After confirming that the RDRF bit in SSR is set to 1, read RDR only once. RDR cannot be written to by the CPU. RDR is initialized to H'00.

16.3.3 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data. To perform serial data transmission, the SCI3 first transfers transmit data from TDR to TSR automatically, then sends the data that starts from the LSB to the TXD pin. TSR cannot be directly accessed by the CPU.

16.3.4 Transmit Data Register (TDR)

TDR is an 8-bit register that stores data for transmission. When the SCI3 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 transmission of one-frame data, the SCI3 transfers the written data to TSR to continue transmission. To achieve reliable serial transmission, write transmit data to TDR only once after confirming that the TDRE bit in SSR is set to 1. TDR is initialized to H'FF.

16.3.5 Serial Mode Register (SMR)

SMR is used to set the SCI3's serial transfer format and select the baud rate generator clock source.

Bit	Bit Name	Initial Value	R/W	Description
7	COM	0	R/W	Communication Mode 0: Asynchronous mode 1: Clocked synchronous mode
6	CHR	0	R/W	Character Length (enabled only in asynchronous mode) 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length.
5	PE	0	R/W	Parity Enable (enabled only in asynchronous mode) When this bit is set to 1, the parity bit is added to transmit data before transmission, and the parity bit is checked in reception.
4	PM	0	R/W	Parity Mode (enabled only when the PE bit is 1 in asynchronous mode) 0: Selects even parity. 1: Selects odd parity.
3	STOP	0	R/W	Stop Bit Length (enabled only in asynchronous mode) Selects the stop bit length in transmission. 0: 1 stop bit 1: 2 stop bits For reception, only the first stop bit is checked, regardless of the value in the bit. If the second stop bit is 0, it is treated as the start bit of the next transmit character.
2	MP	0	R/W	Multiprocessor Mode When this bit is set to 1, the multiprocessor communication function is enabled. The PE bit and PM bit settings are invalid in multiprocessor mode. In clocked synchronous mode, clear this bit to 0.

Bit	Bit Name	Initial Value	R/W	Description
1	CKS1	0	R/W	Clock Select 0 and 1
0	CKS0	0	R/W	These bits select the clock source for the baud rate generator. 00: ϕ clock (n = 0) 01: ϕ/4 clock (n = 1) 10: ϕ/16 clock (n = 2) 11: ϕ/64 clock (n = 3) For the relationship between the bit rate register setting and the baud rate, see section 16.3.8, Bit Rate Register (BRR). n is the decimal representation of the value of n in BRR (see section 16.3.8, Bit Rate Register (BRR)).

16.3.6 Serial Control Register 3 (SCR3)

SCR3 is a register that enables or disables SCI3 transfer operations and interrupt requests, and is also used to select the transfer clock source. For details on interrupt requests, refer to section 16.7, Interrupts.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When this bit is set to 1, the TXI interrupt request is enabled.
6	RIE	0	R/W	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt requests are enabled.
5	TE	0	R/W	Transmit Enable When this bit s set to 1, transmission is enabled.
4	RE	0	R/W	Receive Enable When this bit is set to 1, reception is enabled.

Bit	Bit Name	Initial Value	R/W	Description
3	MPIE	0	R/W	Multiprocessor Interrupt Enable (enabled only when the MP bit in SMR is 1 in asynchronous mode) 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 OER status flags in SSR is disabled. 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 16.6, Multiprocessor Communication Function.
2	TEIE	0	R/W	Transmit End Interrupt Enable When this bit is set to 1, TEI interrupt request is enabled.
1	CKE1	0	R/W	Clock Enable 0 and 1
0	CKE0	0	R/W	Selects the clock source. • Asynchronous mode 00: On-chip baud rate generator 01: On-chip baud rate generator Outputs a clock of the same frequency as the bit rate from the SCK3 pin. 10: External clock Inputs a clock with a frequency 16 times the bit rate from the SCK3 pin. 11: Reserved • Clocked synchronous mode 00: On-chip clock (SCK3 pin functions as clock output) 01: Reserved 10: External clock (SCK3 pin functions as clock input) 11: Reserved

16.3.7 Serial Status Register (SSR)

SSR is a register containing status flags of the SCI3 and multiprocessor bits for transfer. 1 cannot be written to flags TDRE, RDRF, OER, PER, and FER; they can only be cleared.

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	1	R/W	Transmit Data Register Empty Indicates whether TDR contains transmit data. [Setting conditions] • When the TE bit in SCR3 is 0 • When data is transferred from TDR to TSR [Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the transmit data is written to TDR
6	RDRF	0	R/W	Receive Data Register Full Indicates that the received data is stored in RDR. [Setting condition] • When serial reception ends normally and receive data is transferred from RSR to RDR [Clearing conditions] • When 0 is written to RDRF after reading RDRF = 1 • When data is read from RDR
5	OER	0	R/W	Overrun Error [Setting condition] • When an overrun error occurs in reception [Clearing condition] • When 0 is written to OER after reading OER = 1
4	FER	0	R/W	Framing Error [Setting condition] • When a framing error occurs in reception [Clearing condition] • When 0 is written to FER after reading FER = 1

Bit	Bit Name	Initial Value	R/W	Description
3	PER	0	R/W	Parity Error [Setting condition] • When a parity error is detected during reception [Clearing condition] • When 0 is written to PER after reading PER = 1
2	TEND	1	R	Transmit End [Setting conditions] • When the TE bit in SCR3 is 0 • When TDRE = 1 at transmission of the last bit of a 1-frame serial transmit character [Clearing conditions] • When 0 is written to TDRE after reading TDRE = 1 • When the transmit data is written to TDR
1	MPBR	0	R	Multiprocessor Bit Receive MPBR stores the multiprocessor bit in the receive character data. When the RE bit in SCR3 is cleared to 0, its state is retained.
0	MPBT	0	R/W	Multiprocessor Bit Transfer MPBT stores the multiprocessor bit to be added to the transmit character data.

16.3.8 Bit Rate Register (BRR)

BRR is an 8-bit register that adjusts the bit rate. The initial value of BRR is H'FF. Table 16.3 shows the relationship between the N setting in BRR and the n setting in bits CKS1 and CKS0 of SMR in asynchronous mode. Table 16.4 shows the maximum bit rate for each frequency in asynchronous mode. The values shown in both tables 16.3 and 16.4 are values in active (high-speed) mode. Table 16.5 shows the relationship between the N setting in BRR and the n setting in bits CKS1 and CKS0 of SMR in clocked synchronous mode. The values shown in table 16.5 are values in active (high-speed) mode. The N setting in BRR and error for other operating frequencies and bit rates can be obtained by the following formulas:

[Asynchronous Mode]

$$N = \frac{\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1$$

$$\text{Error (\%)} = \left\{ \frac{\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100$$

[Clocked Synchronous Mode]

$$N = \frac{\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

Legend B: Bit rate (bit/s)

N: BRR setting for baud rate generator ($0 \leq N \leq 255$)

ϕ: Operating frequency (MHz)

n: CSK1 and CSK0 settings in SMR ($0 \leq n \leq 3$)

Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (1)

	Operating Frequency ϕ (MHz)											
	2			2.097152			2.4576			3		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	1	141	0.03	1	148	–0.04	1	174	–0.26	1	212	0.03
150	1	103	0.16	1	108	0.21	1	127	0.00	1	155	0.16
300	0	207	0.16	0	217	0.21	0	255	0.00	1	77	0.16
600	0	103	0.16	0	108	0.21	0	127	0.00	0	155	0.16
1200	0	51	0.16	0	54	–0.70	0	63	0.00	0	77	0.16
2400	0	25	0.16	0	26	1.14	0	31	0.00	0	38	0.16
4800	0	12	0.16	0	13	–2.48	0	15	0.00	0	19	–2.34
9600	0	6	–6.99	0	6	–2.48	0	7	0.00	0	9	–2.34
19200	0	2	8.51	0	2	13.78	0	3	0.00	0	4	–2.34
31250	0	1	0.00	0	1	4.86	0	1	22.88	0	2	0.00
38400	0	1	–18.62	0	1	–14.67	0	1	0.00	—	—	—

	Operating Frequency ϕ (MHz)											
	3.6864			4			4.9152			5		
Bit Rate (bits/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	64	0.70	2	70	0.03	2	86	0.31	2	88	–0.25
150	1	191	0.00	1	207	0.16	1	255	0.00	2	64	0.16
300	1	95	0.00	1	103	0.16	1	127	0.00	1	129	0.16
600	0	191	0.00	0	207	0.16	0	255	0.00	1	64	0.16
1200	0	95	0.00	0	103	0.16	0	127	0.00	0	129	0.16
2400	0	47	0.00	0	51	0.16	0	63	0.00	0	64	0.16
4800	0	23	0.00	0	25	0.16	0	31	0.00	0	32	–1.36
9600	0	11	0.00	0	12	0.16	0	15	0.00	0	15	1.73
19200	0	5	0.00	0	6	–6.99	0	7	0.00	0	7	1.73
31250	—	—	—	0	3	0.00	0	4	–1.70	0	4	0.00
38400	0	2	0.00	0	2	8.51	0	3	0.00	0	3	1.73

Legend:

—: A setting is available but error occurs

Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (2)

	Operating Frequency ϕ (MHz)								
	6			6.144			7.3728		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	106	–0.44	2	108	0.08	2	130	–0.07
150	2	77	0.16	2	79	0.00	2	95	0.00
300	1	155	0.16	1	159	0.00	1	191	0.00
600	1	77	0.16	1	79	0.00	1	95	0.00
1200	0	155	0.16	0	159	0.00	0	191	0.00
2400	0	77	0.16	0	79	0.00	0	95	0.00
4800	0	38	0.16	0	39	0.00	0	47	0.00
9600	0	19	–2.34	0	19	0.00	0	23	0.00
19200	0	9	–2.34	0	9	0.00	0	11	0.00
31250	0	5	0.00	0	5	2.40	0	6	5.33
38400	0	4	–2.34	0	4	0.00	0	5	0.00

	Operating Frequency ϕ (MHz)											
	8			9.8304			10			12		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	141	0.03	2	174	–0.26	2	177	–0.25	2	212	0.03
150	2	103	0.16	2	127	0.00	2	129	0.16	2	155	0.16
300	1	207	0.16	1	255	0.00	2	64	0.16	2	77	0.16
600	1	103	0.16	1	127	0.00	1	129	0.16	1	155	0.16
1200	0	207	0.16	0	255	0.00	1	64	0.16	1	77	0.16
2400	0	103	0.16	0	127	0.00	0	129	0.16	0	155	0.16
4800	0	51	0.16	0	63	0.00	0	64	0.16	0	77	0.16
9600	0	25	0.16	0	31	0.00	0	32	–1.36	0	38	0.16
19200	0	12	0.16	0	15	0.00	0	15	1.73	0	19	–2.34
31250	0	7	0.00	0	9	–1.70	0	9	0.00	0	11	0.00
38400	0	6	-6.99	0	7	0.00	0	7	1.73	0	9	–2.34

Legend:

—: A setting is available but error occurs.

Table 16.3 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode) (3)

	Operating Frequency ϕ (MHz)											
	12.888			14			14.7456			16		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	2	217	0.08	2	248	–0.17	3	64	0.70	3	70	0.03
150	2	159	0.00	2	181	0.16	2	191	0.00	2	207	0.16
300	2	79	0.00	2	90	0.16	2	95	0.00	2	103	0.16
600	1	159	0.00	1	181	0.16	1	191	0.00	1	207	0.16
1200	1	79	0.00	1	90	0.16	1	95	0.00	1	103	0.16
2400	0	159	0.00	0	181	0.16	0	191	0.00	0	207	0.16
4800	0	79	0.00	0	90	0.16	0	95	0.00	0	103	0.16
9600	0	39	0.00	0	45	–0.93	0	47	0.00	0	51	0.16
19200	0	19	0.00	0	22	–0.93	0	23	0.00	0	25	0.16
31250	0	11	2.40	0	13	0.00	0	14	–1.70	0	15	0.00
38400	0	9	0.00	—	—	—	0	11	0.00	0	12	0.16

	Operating Frequency ϕ (MHz)					
	18			20		
Bit Rate (bit/s)	n	N	Error (%)	n	N	Error (%)
110	3	79	–0.12	3	88	–0.25
150	2	233	0.16	3	64	0.16
300	2	116	0.16	2	129	0.16
600	1	233	0.16	2	64	0.16
1200	1	116	0.16	1	129	0.16
2400	0	233	0.16	1	64	0.16
4800	0	116	0.16	0	129	0.16
9600	0	58	–0.96	0	64	0.16
19200	0	28	1.02	0	32	–1.36
31250	0	17	0.00	0	19	0.00
38400	0	14	–2.34	0	15	1.73

Legend:

—: A setting is available but error occurs.

Table 16.4 Maximum Bit Rate for Each Frequency (Asynchronous Mode)

ϕ (MHz)	Maximum Bit Rate (bit/s)	n	N	ϕ (MHz)	Maximum Bit Rate (bit/s)	n	N
2	62500	0	0	8	250000	0	0
2.097152	65536	0	0	9.8304	307200	0	0
2.4576	76800	0	0	10	312500	0	0
3	93750	0	0	12	375000	0	0
3.6864	115200	0	0	12.288	384000	0	0
4	125000	0	0	14	437500	0	0
4.9152	153600	0	0	14.7456	460800	0	0
5	156250	0	0	16	500000	0	0
6	187500	0	0	17.2032	537600	0	0
6.144	192000	0	0	18	562500	0	0
7.3728	230400	0	0	20	625000	0	0

Table 16.5 Examples of BRR Settings for Various Bit Rates (Clocked Synchronous Mode) (1)

Bit Rate (bit/s)	Operating Frequency ϕ (MHz)									
	2		4		8		10		16	
	n	N	n	N	n	N	n	N	n	N
110	3	70	—	—	—	—	—	—		
250	2	124	2	249	3	124	—	—	3	249
500	1	249	2	124	2	249	—	—	3	124
1k	1	124	1	249	2	124	—	—	2	249
2.5k	0	199	1	99	1	199	1	249	2	99
5k	0	99	0	199	1	99	1	124	1	199
10k	0	49	0	99	0	199	0	249	1	99
25k	0	19	0	39	0	79	0	99	0	159
50k	0	9	0	19	0	39	0	49	0	79
100k	0	4	0	9	0	19	0	24	0	39
250k	0	1	0	3	0	7	0	9	0	15
500k	0	0*	0	1	0	3	0	4	0	7
1M			0	0*	0	1	—	—	0	3
2M					0	0*	—	—	0	1
2.5M							0	0*	—	—
4M									0	0*

Legend:

Blank : No setting is available.

— : A setting is available but error occurs.

* : Continuous transfer is not possible.

Table 16.5 Examples of BRR Settings for Various Bit Rates (Clocked Synchronous Mode) (2)

Bit Rate (bit/s)	Operating Frequency φ (MHz)			
	18		20	
	n	N	n	N
110	—	—	—	—
250	—	—	—	—
500	3	140	3	155
1k	3	69	3	77
2.5k	2	112	2	124
5k	1	224	1	249
10k	1	112	1	124
25k	0	179	0	199
50k	0	89	0	99
100k	0	44	0	49
250k	0	17	0	19
500k	0	8	0	9
1M	0	4	0	4
2M	—	—	—	—
2.5M	—	—	0	1
4M	—	—	—	—

Legend:

Blank : No setting is available.

— : A setting is available but error occurs.

* : Continuous transfer is not possible.

16.4 Operation in Asynchronous Mode

Figure 16.2 shows the general format for asynchronous serial communication. One character (or 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). Inside the SCI3, 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.

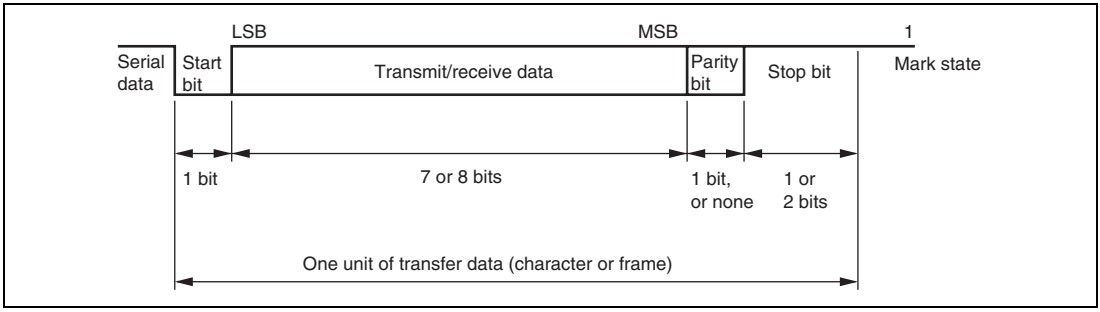


Figure 16.2 Data Format in Asynchronous Communication

16.4.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input at the SCK3 pin can be selected as the SCI3's serial clock, according to the setting of the COM bit in SMR and the CKE0 and CKE1 bits in SCR3. When an external clock is input at the SCK3 pin, the clock frequency should be 16 times the bit rate used.

When the SCI3 is operated on an internal clock, the clock can be output from the SCK3 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 16.3.

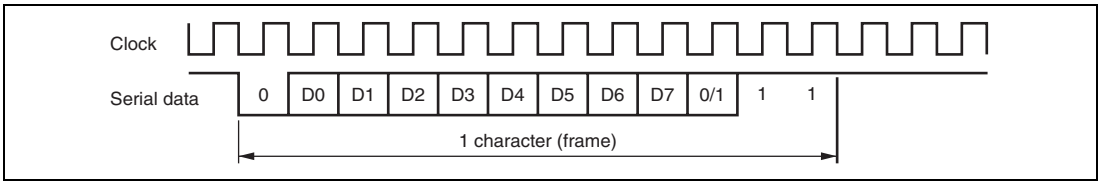


Figure 16.3 Relationship between Output Clock and Transfer Data Phase (Asynchronous Mode) (Example with 8-Bit Data, Parity, Two Stop Bits)

16.4.2 SCI3 Initialization

Before transmitting and receiving data, you should first clear the TE and RE bits in SCR3 to 0, then initialize the SCI3 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 OER flags, or the contents of RDR. When the external clock is used in asynchronous mode, the clock must be supplied even during initialization.

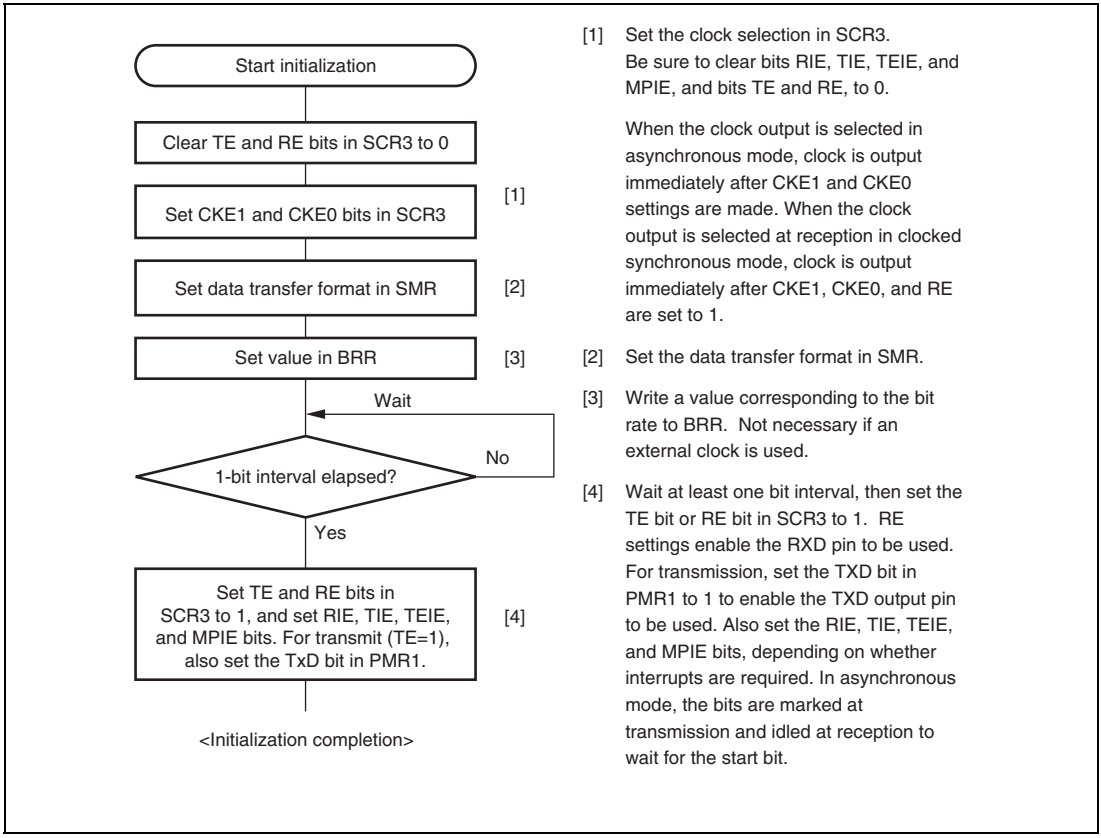


Figure 16.4 Sample SCI3 Initialization Flowchart

16.4.3 Data Transmission

Figure 16.5 shows an example of operation for transmission in asynchronous mode. In transmission, the SCI3 operates as described below.

1. The SCI3 monitors the TDRE flag in SSR. If the flag is cleared to 0, the SCI3 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 SCI3 sets the TDRE flag to 1 and starts transmission. If the TIE bit is set to 1 at this time, a TXI interrupt request 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. The SCI3 checks the TDRE flag at the timing for sending the stop bit.
4. 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.
5. 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 SCR3 is set to 1 at this time, a TEI interrupt request is generated.
6. Figure 16.6 shows a sample flowchart for transmission in asynchronous mode.

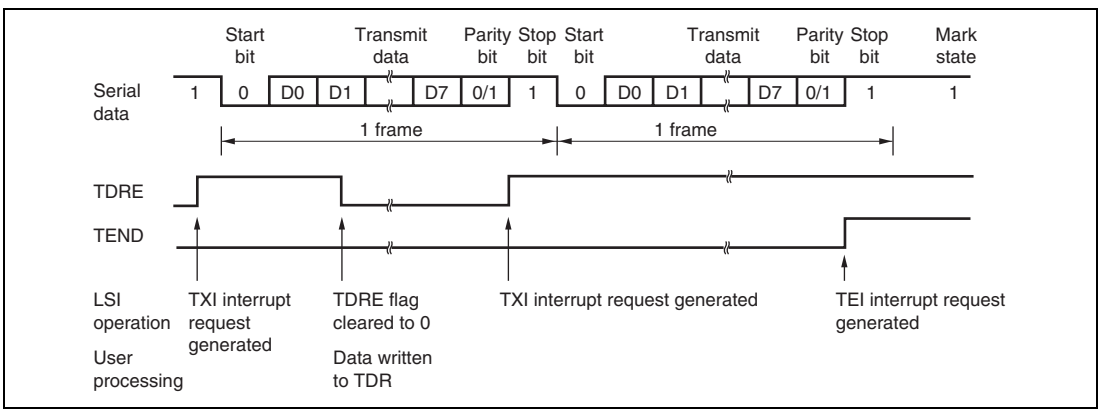
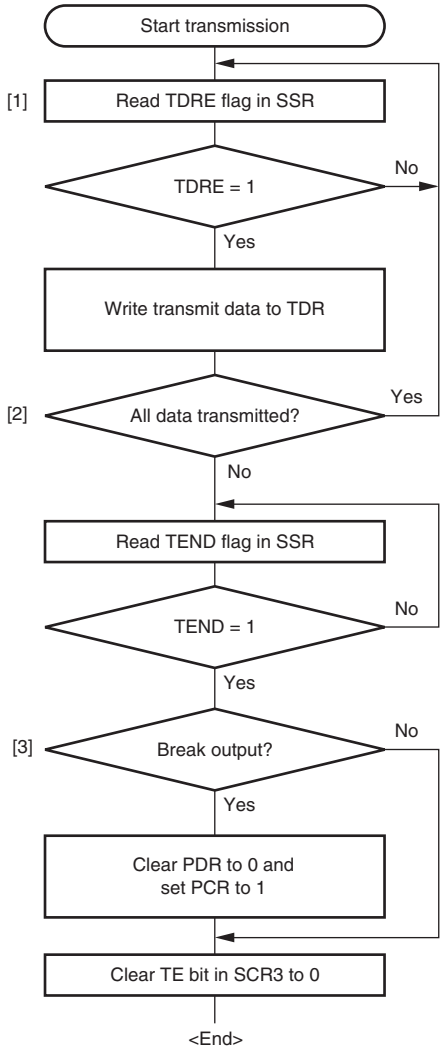


Figure 16.5 Example of SCI3 Transmission in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)



[1] Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR. When data is written to TDR, the TDRE flag is automaticaly cleared to 0.

[2] To continue serial transmission, read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR. When data is written to TDR, the TDRE flag is automaticaly cleared to 0.

[3] To output a break in serial transmission, after setting PCR to 1 and PDR to 0, clear TxD in PMR1 to 0, then clear the TE bit in SCR3 to 0.

Figure 16.6 Sample Serial Transmission Data Flowchart (Asynchronous Mode)

16.4.4 Serial Data Reception

Figure 16.7 shows an example of operation for reception in asynchronous mode. In serial reception, the SCI3 operates as described below.

1. The SCI3 monitors the communication line. If a start bit is detected, the SCI3 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 OER bit in SSR is set to 1. If the RIE bit in SCR3 is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
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 SCR3 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 SCR3 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 SCR3 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.

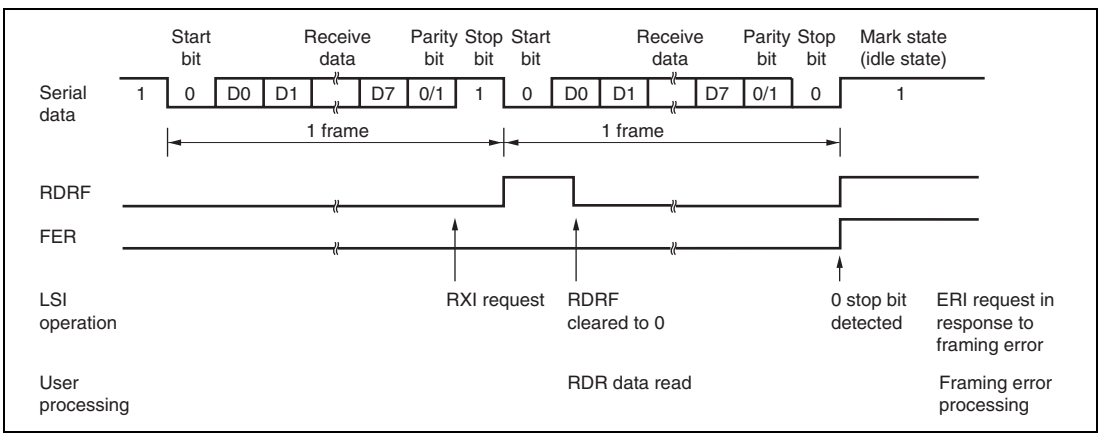


Figure 16.7 Example of SCI3 Reception in Asynchronous Mode (8-Bit Data, Parity, One Stop Bit)

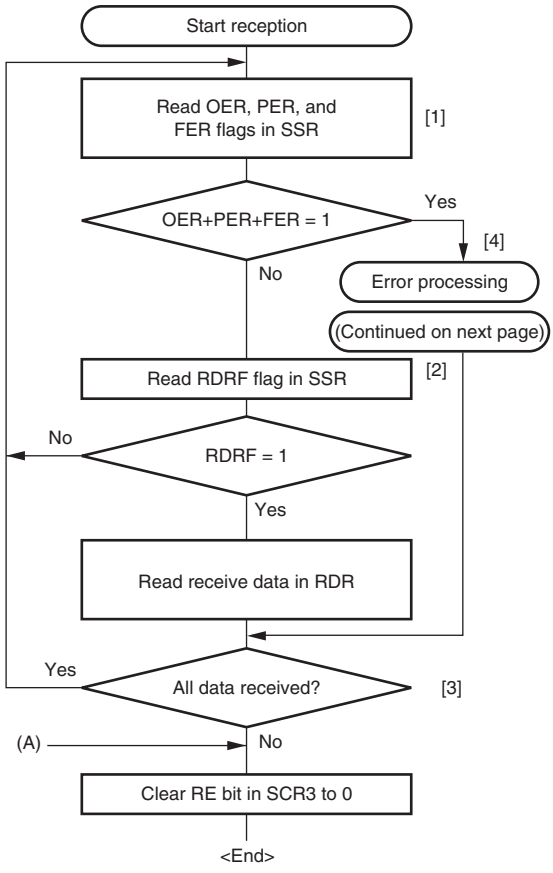
Table 16.6 shows the states of the SSR status flags and receive data handling when a receive error is detected. If a receive error is detected, the RDRF flag retains its state before receiving data. Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the OER,

FER, PER, and RDRF bits to 0 before resuming reception. Figure 16.8 shows a sample flow chart for serial data reception.

Table 16.6 SSR Status Flags and Receive Data Handling

SSR Status Flag					
RDRF*	**OER**	**FER**	**PER**	**Receive Data**	**Receive Error Type**
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] Read the OER, PER, and FER flags in SSR to identify the error. If a receive error occurs, performs the appropriate error processing.

[2] Read SSR and check that RDRF = 1, then read the receive data in RDR. The RDRF flag is cleared automatically.

[3] To continue serial reception, before the stop bit for the current frame is received, read the RDRF flag and read RDR.
The RDRF flag is cleared automatically.

[4] If a receive error occurs, read the OER, PER, and FER flags in SSR to identify the error. After performing the appropriate error processing, ensure that the OER, 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.

Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode) (1)

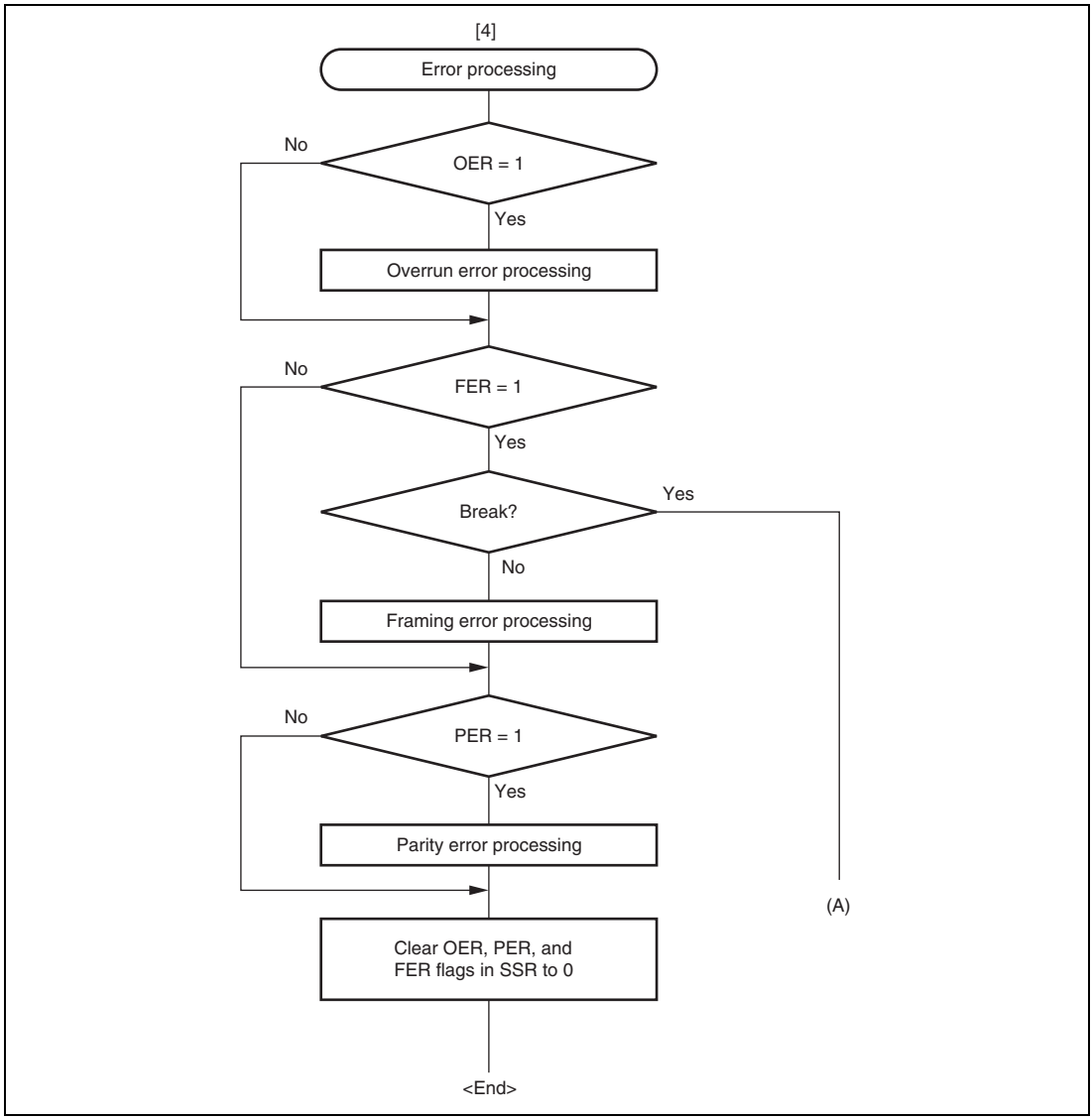


Figure 16.8 Sample Serial Reception Data Flowchart (Asynchronous Mode) (2)

16.5 Operation in Clocked Synchronous Mode

Figure 16.9 shows the general format for clocked synchronous communication. In clocked synchronous mode, data is transmitted or received synchronous with clock pulses. A single character in the transmit data consists of the 8-bit data starting from the LSB. In clocked synchronous serial communication, data on the transmission line is output from one falling edge of the synchronization clock to the next. In clocked synchronous mode, the SCI3 receives data in synchronous with the rising edge of the synchronization 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 SCI3, 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.

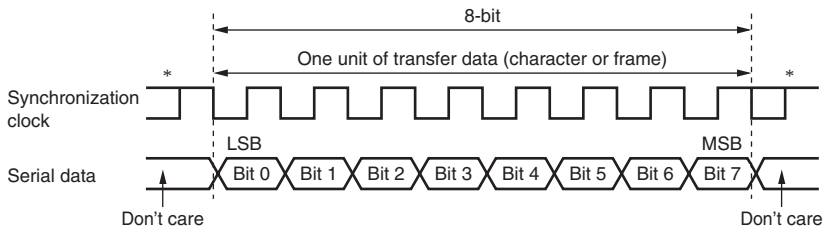


Figure 16.9 Data Format in Clocked Synchronous Communication

16.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCK3 pin can be selected, according to the setting of the COM bit in SMR and CKE0 and CKE1 bits in SCR3. When the SCI3 is operated on an internal clock, the synchronization clock is output from the SCK3 pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is fixed high.

16.5.2 SCI3 Initialization

Before transmitting and receiving data, the SCI3 should be initialized as described in a sample flowchart in figure 16.4.

16.5.3 Serial Data Transmission

Figure 16.10 shows an example of SCI3 operation for transmission in clocked synchronous mode. In serial transmission, the SCI3 operates as described below.

1. The SCI3 monitors the TDRE flag in SSR, and if the flag is 0, the SCI3 recognizes that data has been written to TDR, and transfers the data from TDR to TSR.
2. The SCI3 sets the TDRE flag to 1 and starts transmission. If the TIE bit in SCR3 is set to 1 at this time, a transmit data empty interrupt (TXI) is generated.
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. Serial data is transmitted sequentially from the LSB (bit 0), from the TxD pin.
4. The SCI3 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 SCR3 is set to 1 at this time, a TEI interrupt request is generated.
7. The SCK3 pin is fixed high at the end of transmission.

Figure 16.11 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 (OER, FER, or PER) is set to 1. Make sure that the receive error flags are cleared to 0 before starting transmission.

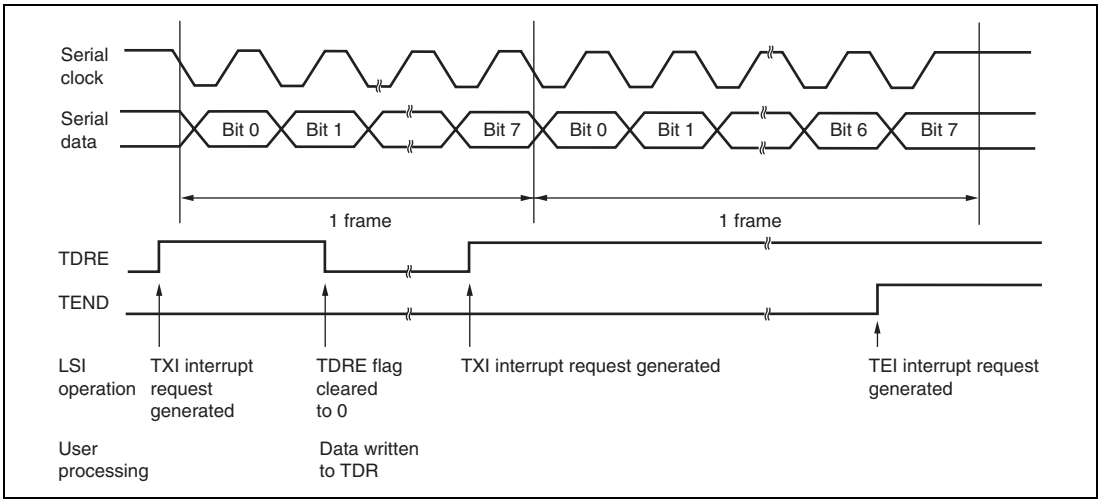
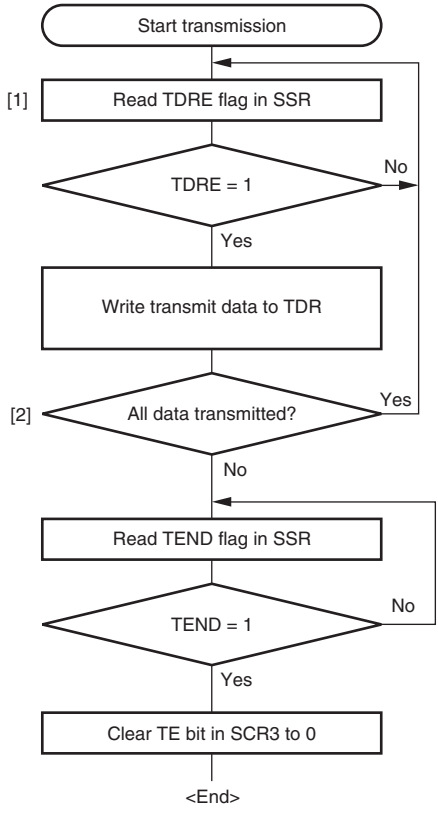


Figure 16.10 Example of SCI3 Transmission in Clocked Synchronous Mode



[1] Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR. When data is written to TDR, the TDRE flag is automatically cleared to 0 and clocks are output to start the data transmission.

[2] To continue serial transmission, be sure to read 1 from the TDRE flag to confirm that writing is possible, then write data to TDR. When data is written to TDR, the TDRE flag is automatically cleared to 0.

Figure 16.11 Sample Serial Transmission Flowchart (Clocked Synchronous Mode)

16.5.4 Serial Data Reception (Clocked Synchronous Mode)

Figure 16.12 shows an example of SCI3 operation for reception in clocked synchronous mode. In serial reception, the SCI3 operates as described below.

1. The SCI3 performs internal initialization synchronous with a synchronization clock input or output, starts receiving data.
2. The SCI3 stores the receive data in RSR.
3. 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 OER bit in SSR is set to 1. If the RIE bit in SCR3 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.
4. 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 SCR3 is set to 1 at this time, an RXI interrupt request is generated.

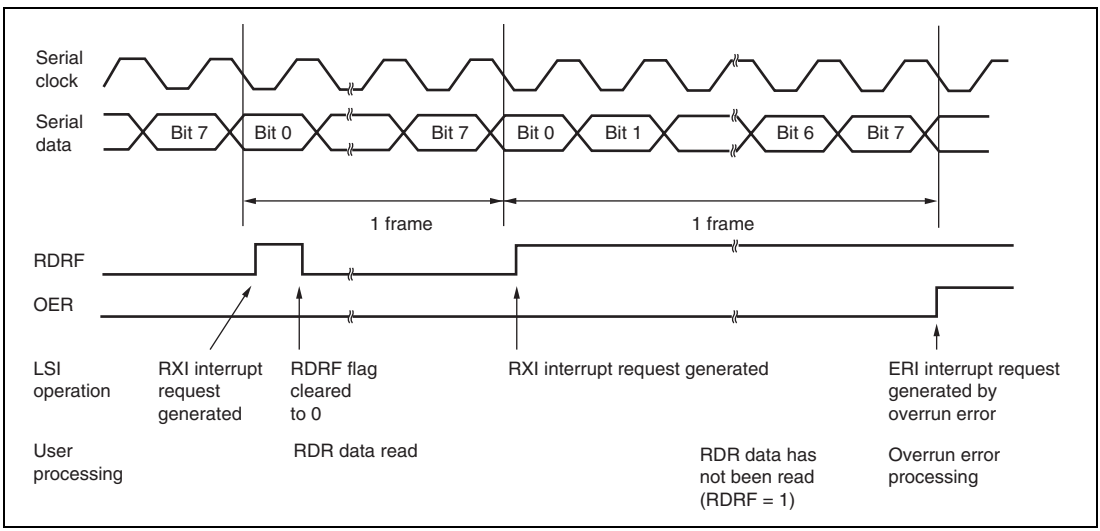
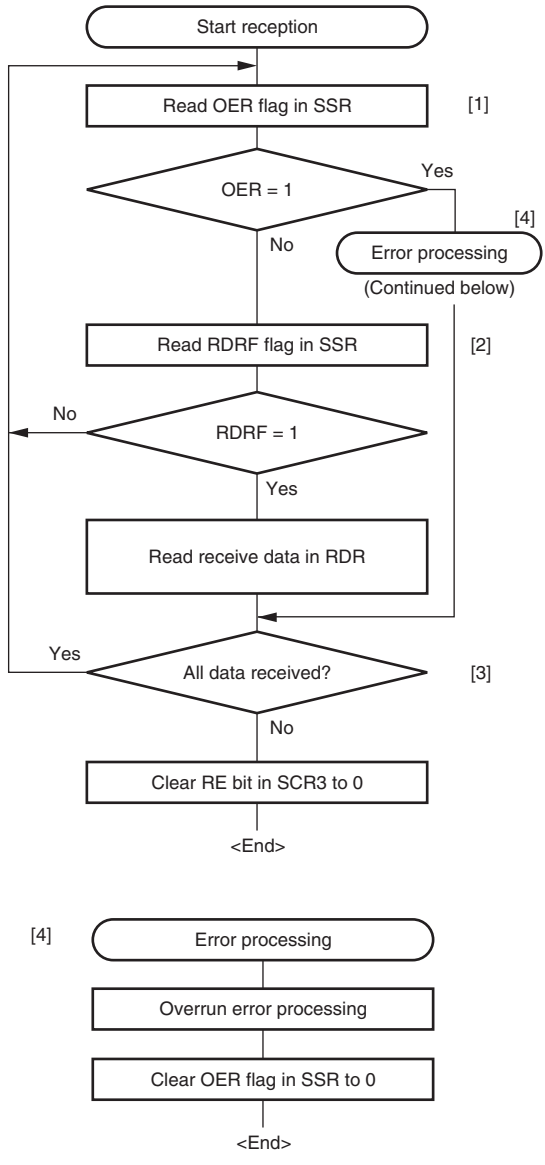


Figure 16.12 Example of SCI3 Reception in Clocked Synchronous Mode

Reception cannot be resumed while a receive error flag is set to 1. Accordingly, clear the OER, FER, PER, and RDRF bits to 0 before resuming reception. Figure 16.13 shows a sample flow chart for serial data reception.



[1] Read the OER flag in SSR to determine if there is an error. If an overrun error has occurred, execute overrun error processing.

[2] Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR. When data is read from RDR, the RDRF flag is automatically cleared to 0.

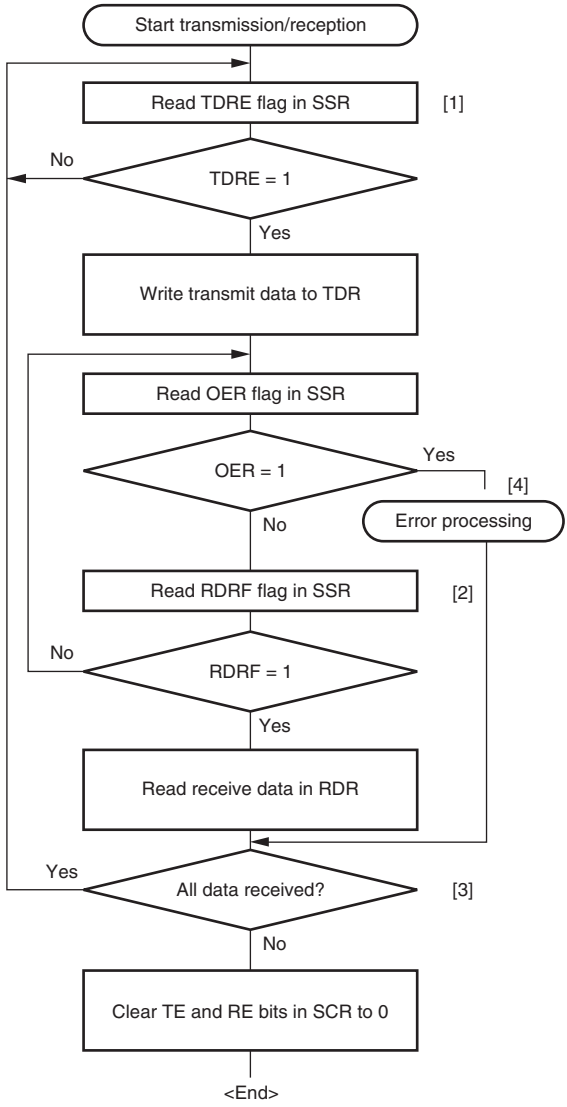
[3] To continue serial reception, before the MSB (bit 7) of the current frame is received, reading the RDRF flag and reading RDR should be finished. When data is read from RDR, the RDRF flag is automatically cleared to 0.

[4] If an overrun error occurs, read the OER flag in SSR, and after performing the appropriate error processing, clear the OER flag to 0. Reception cannot be resumed if the OER flag is set to 1.

Figure 16.13 Sample Serial Reception Flowchart (Clocked Synchronous Mode)

16.5.5 Simultaneous Serial Data Transmission and Reception

Figure 16.14 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 SCI3 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 SCI3 has finished reception, clear RE to 0. Then after checking that the RDRF and receive error flags (OER, FER, and PER) are cleared to 0, simultaneously set TE and RE to 1 with a single instruction.



[1] Read SSR and check that the TDRE flag is set to 1, then write transmit data to TDR.
When data is written to TDR, the TDRE flag is automatically cleared to 0.

[2] Read SSR and check that the RDRF flag is set to 1, then read the receive data in RDR.
When data is read from RDR, the RDRF flag is automatically cleared to 0.

[3] To continue serial transmission/ reception, before the MSB (bit 7) of the current frame is received, finish reading the RDRF flag, reading RDR. 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.
When data is written to TDR, the TDRE flag is automatically cleared to 0. When data is read from RDR, the RDRF flag is automatically cleared to 0.

[4] If an overrun error occurs, read the OER flag in SSR, and after performing the appropriate error processing, clear the OER flag to 0. Transmission/reception cannot be resumed if the OER flag is set to 1. For overrun error processing, see figure 16.13.

Figure 16.14 Sample Flowchart of Simultaneous Serial Transmit and Receive Operations (Clocked Synchronous Mode)

16.6 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 16.15 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 SCI3 uses the MPIE bit in SCR3 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 OER, 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 MPBR bit in SSR is set to 1 and the MPIE bit is automatically cleared, thus normal reception is resumed. If the RIE bit in SCR3 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.

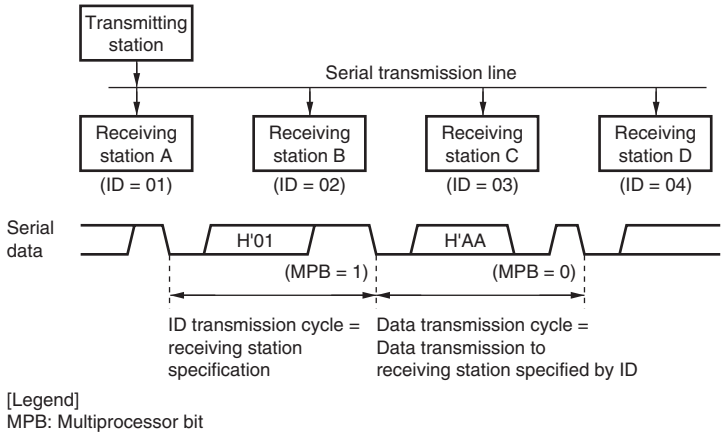


Figure 16.15 Example of Inter-Processor Communication Using Multiprocessor Format (Transmission of Data H'AA to Receiving Station A)

16.6.1 Multiprocessor Serial Data Transmission

Figure 16.16 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 SCI3 operations are the same as those in asynchronous mode.

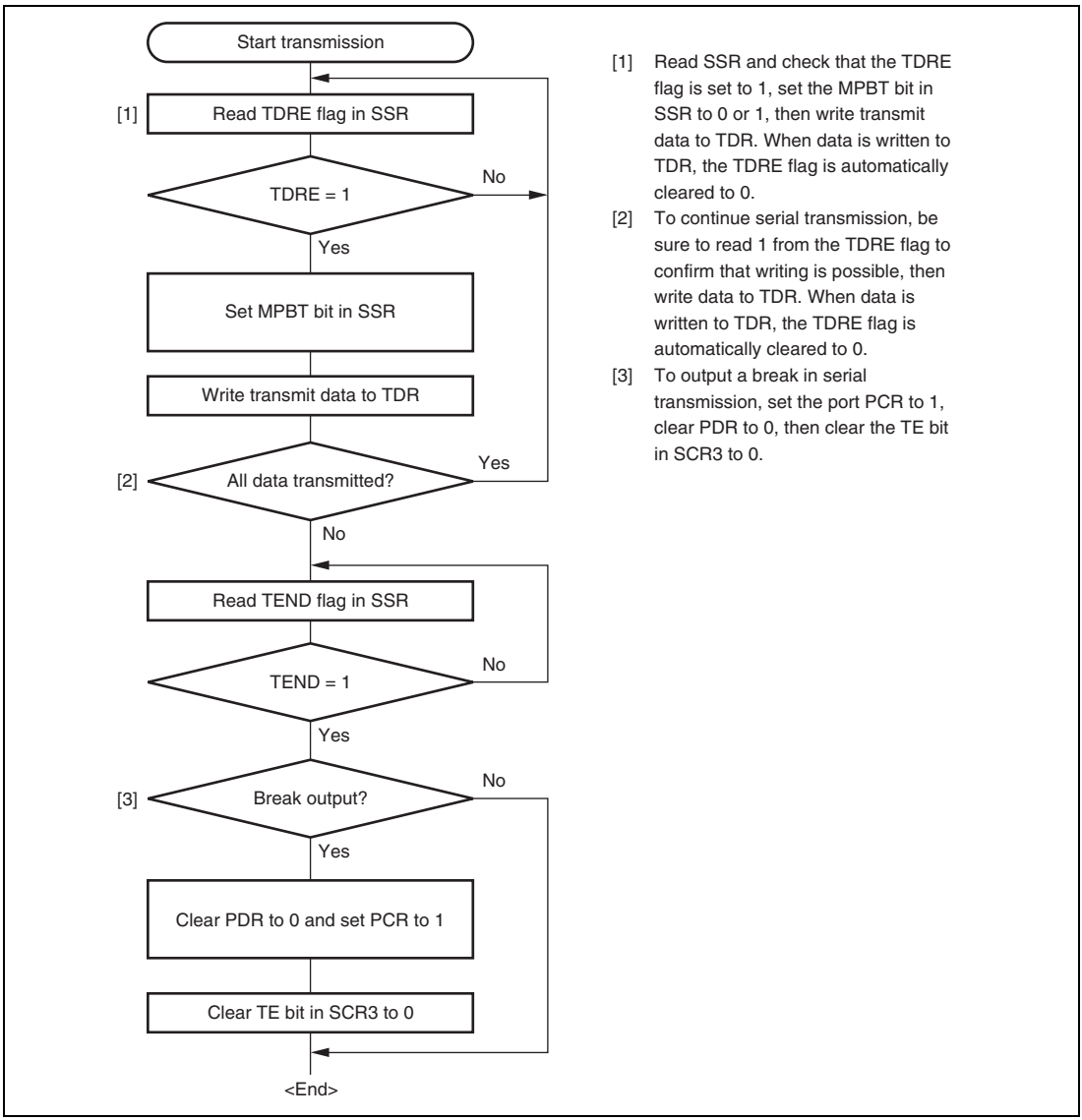


Figure 16.16 Sample Multiprocessor Serial Transmission Flowchart

16.6.2 Multiprocessor Serial Data Reception

Figure 16.17 shows a sample flowchart for multiprocessor serial data reception. If the MPIE bit in SCR3 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 SCI3 operations are the same as those in asynchronous mode. Figure 16.18 shows an example of SCI3 operation for multiprocessor format reception.

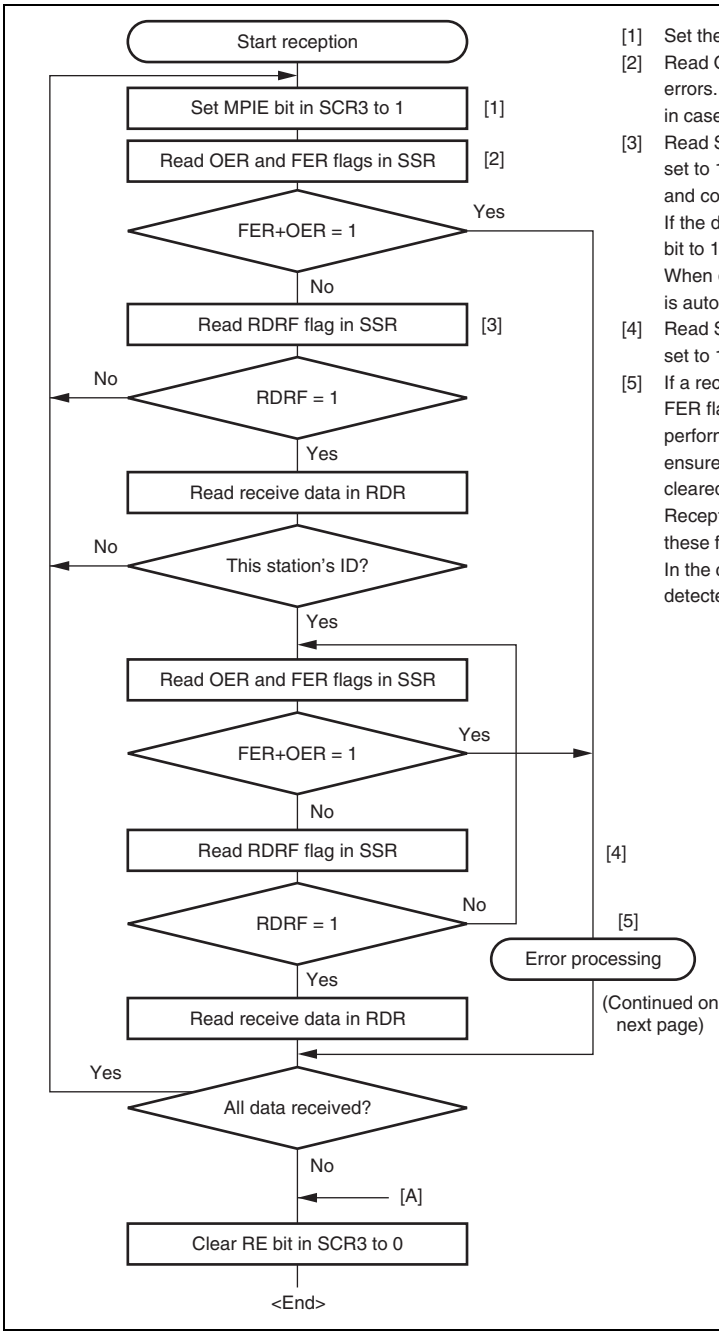


Figure 16.17 Sample Multiprocessor Serial Reception Flowchart (1)

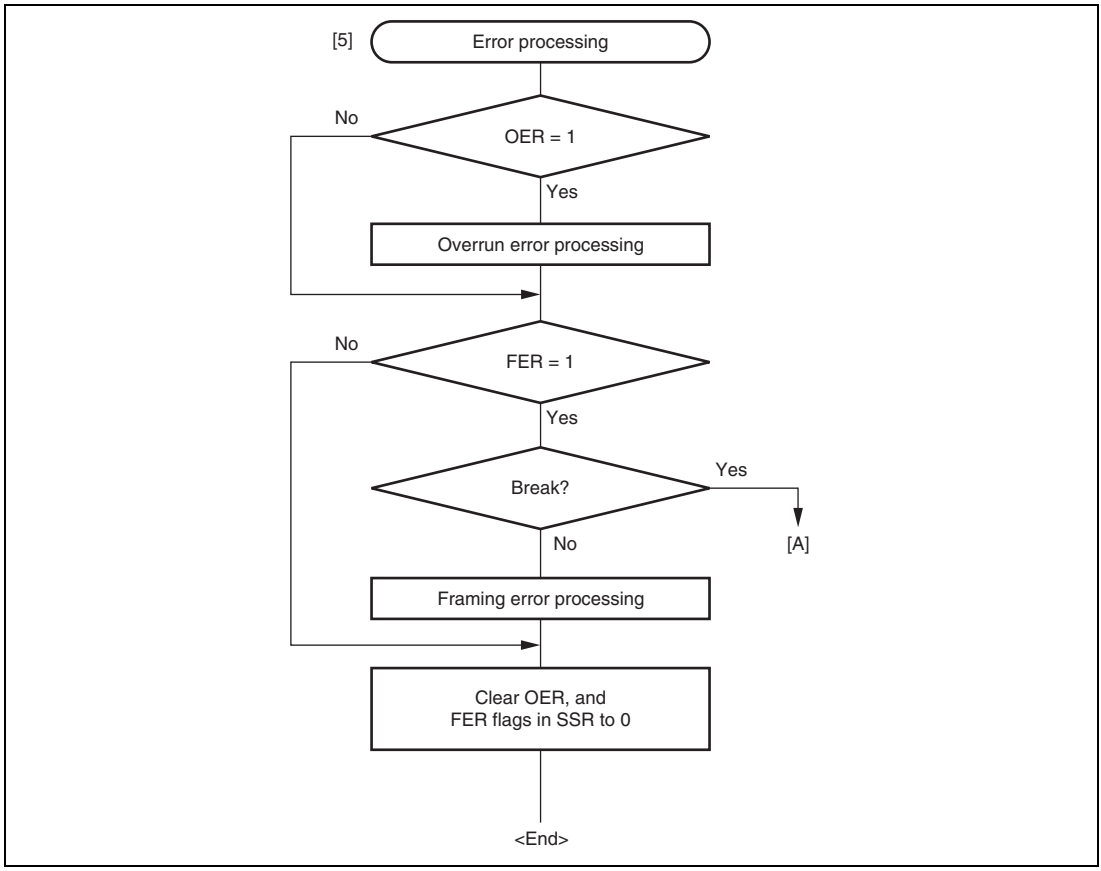
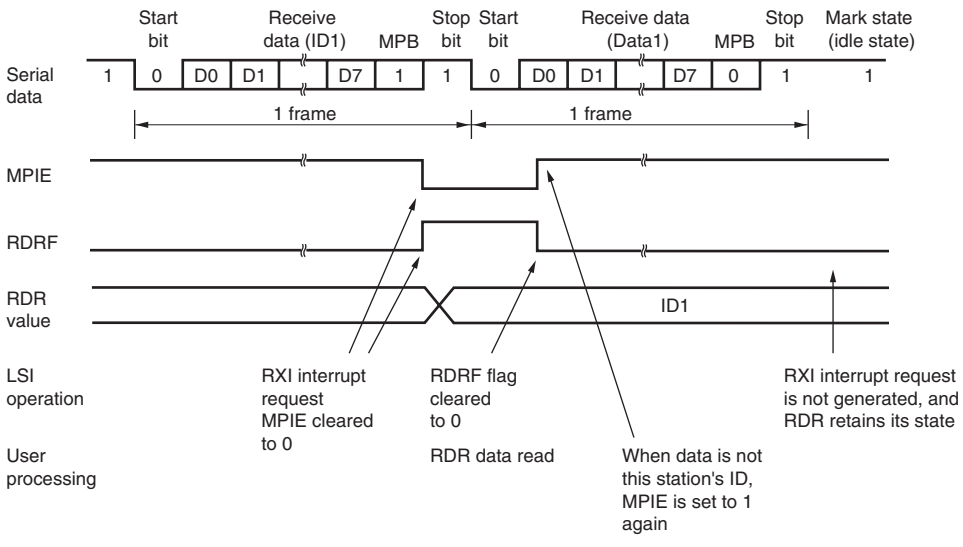
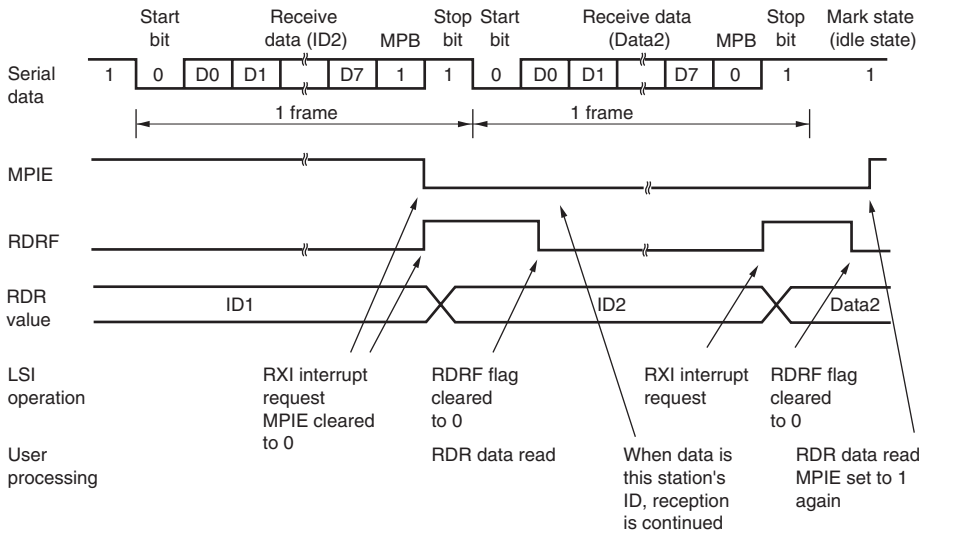


Figure 16.17 Sample Multiprocessor Serial Reception Flowchart (2)



(a) When data does not match this receiver's ID



(b) When data matches this receiver's ID

Figure 16.18 Example of SCI3 Reception Using Multiprocessor Format (Example with 8-Bit Data, Multiprocessor Bit, One Stop Bit)

16.7 Interrupts

SCI3 creates the following six interrupt requests: transmission end, transmit data empty, receive data full, and receive errors (overrun error, framing error, and parity error). Table 16.7 shows the interrupt sources.

Table 16.7 SCI3 Interrupt Requests

Interrupt Requests	Abbreviation	Interrupt Sources
Receive Data Full	RXI	Setting RDRF in SSR
Transmit Data Empty	TXI	Setting TDRE in SSR
Transmission End	TEI	Setting TEND in SSR
Receive Error	ERI	Setting OER, FER, and PER in SSR

The initial value of the TDRE flag in SSR is 1. Thus, when the TIE bit in SCR3 is set to 1 before transferring the transmit data to TDR, a TXI interrupt request is generated even if the transmit data is not ready. The initial value of the TEND flag in SSR is 1. Thus, when the TEIE bit in SCR3 is set to 1 before transferring the transmit data to TDR, a TEI interrupt request is generated even if the transmit data has not been sent. It is possible to make use of the most of these interrupt requests efficiently by transferring the transmit data to TDR in the interrupt routine. To prevent the generation of these interrupt requests (TXI and TEI), set the enable bits (TIE and TEIE) that correspond to these interrupt requests to 1, after transferring the transmit data to TDR.

16.8 Usage Notes

16.8.1 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 SCI3 continues the receive operation after receiving a break, even if the FER flag is cleared to 0, it will be set to 1 again.

16.8.2 Mark State and Break Sending

When TE is 0, the TxD pin is used as an I/O port whose direction (input or output) and level are determined by PCR and PDR. 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 clear 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.

16.8.3 Receive Error Flags and Transmit Operations (Clocked Synchronous Mode Only)

Transmission cannot be started when a receive error flag (OER, 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.

16.8.4 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI3 operates on a basic clock with a frequency of 16 times the transfer rate. In reception, the SCI3 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 16.19. Thus, the reception margin in asynchronous mode is given by formula (1) below.

$$M = \left\{(0.5 - \frac{1}{2N}) - \frac{D - 0.5}{N} - (L - 0.5)\,F\right\} \times 100(\%)$$

... Formula (1)

Legend 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.

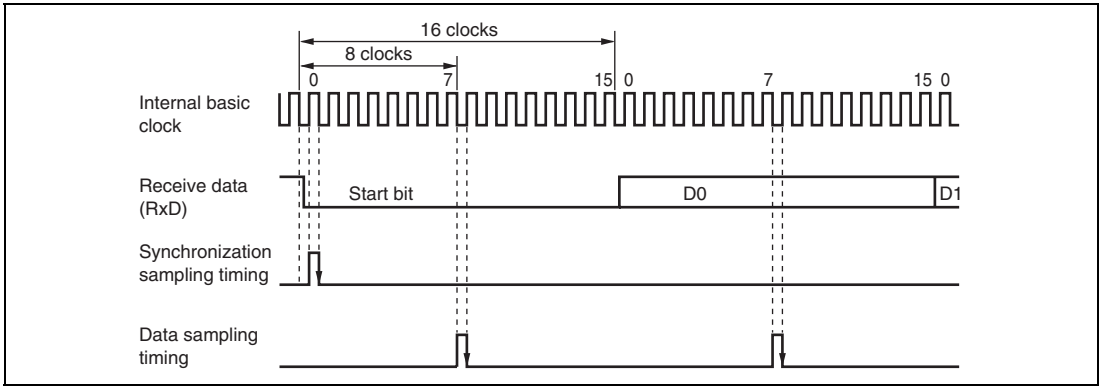


Figure 16.19 Receive Data Sampling Timing in Asynchronous Mode

Section 17 I²C Bus Interface 2 (IIC2)

The I²C bus interface 2 conforms to and provides a subset of the Philips I²C bus (inter-IC bus) interface functions. The register configuration that controls the I²C bus differs partly from the Philips configuration, however.

Figure 17.1 shows a block diagram of the I²C bus interface 2.

Figure 17.2 shows an example of I/O pin connections to external circuits.

17.1 Features

- Selection of I²C format or clocked synchronous serial format
- Continuous transmission/reception

  Since the shift register, transmit data register, and receive data register are independent from each other, the continuous transmission/reception can be performed.

I²C bus format

- Start and stop conditions generated automatically in master mode
- Selection of acknowledge output levels when receiving
- Automatic loading of acknowledge bit when transmitting
- Bit synchronization/wait function

  In master mode, the state of SCL is monitored per bit, and the timing is synchronized automatically.

  If transmission/reception is not yet possible, set the SCL to low until preparations are completed.
- Six interrupt sources

  Transmit data empty (including slave-address match), transmit end, receive data full (including slave-address match), arbitration lost, NACK detection, and stop condition detection
- Direct bus drive

  Two pins, SCL and SDA pins, function as NMOS open-drain outputs when the bus drive function is selected.

Clocked synchronous format

- Four interrupt sources

  Transmit-data-empty, transmit-end, receive-data-full, and overrun error

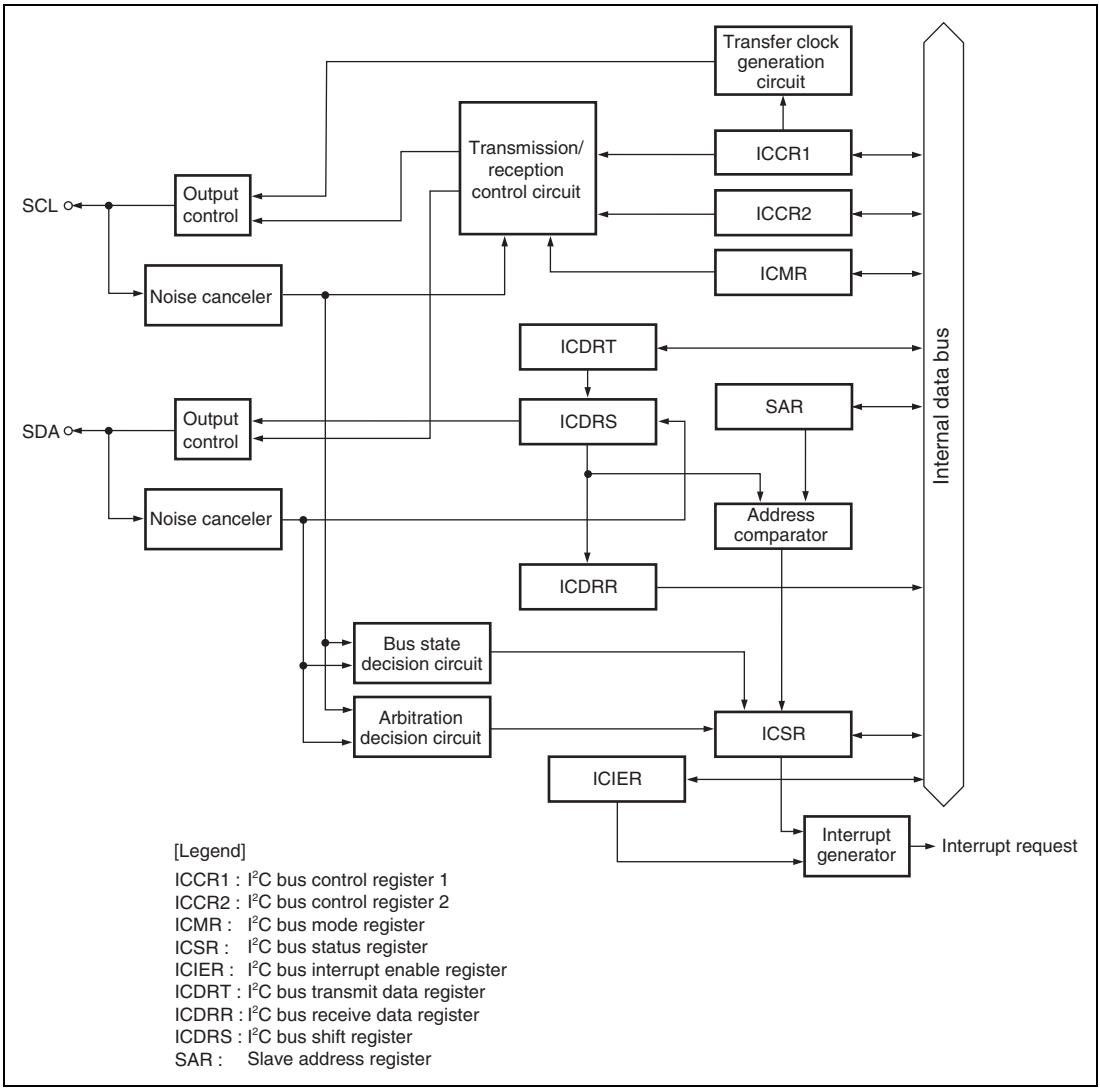


Figure 17.1 Block Diagram of I²C Bus Interface 2

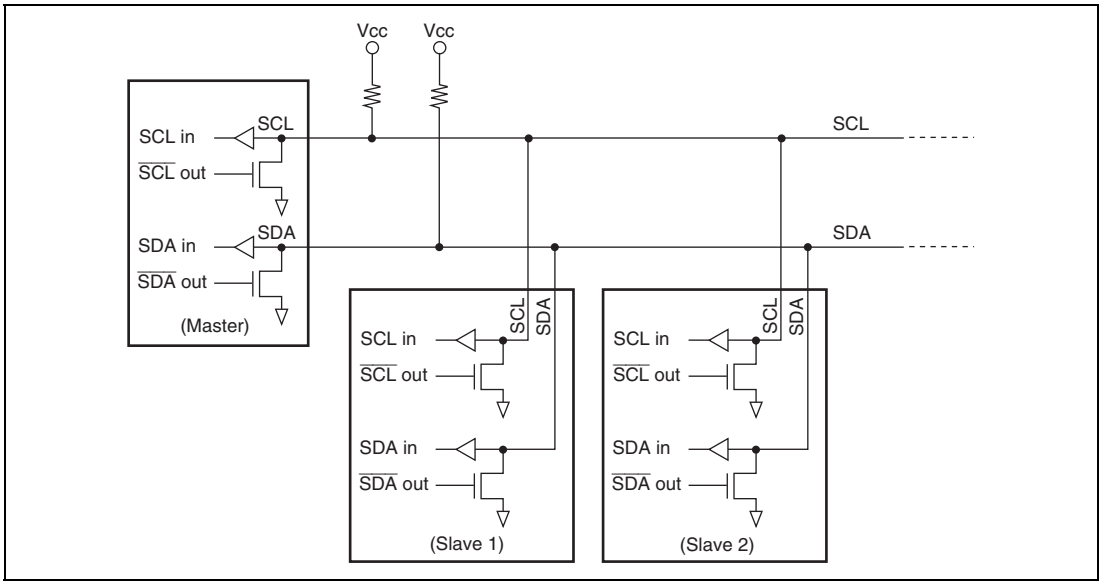


Figure 17.2 External Circuit Connections of I/O Pins

17.2 Input/Output Pins

Table 17.1 summarizes the input/output pins used by the I²C bus interface 2.

Table 17.1 I²C Bus Interface Pins

Name	Abbreviation	I/O	Function
Serial clock	SCL	I/O	IIC serial clock input/output
Serial data	SDA	I/O	IIC serial data input/output

17.3 Register Descriptions

The I²C bus interface 2 has the following registers:

- I²C bus control register 1 (ICCR1)
- I²C bus control register 2 (ICCR2)
- I²C bus mode register (ICMR)
- I²C bus interrupt enable register (ICIER)
- I²C bus status register (ICSR)
- I²C bus slave address register (SAR)
- I²C bus transmit data register (ICDRT)

- I²C bus receive data register (ICDRR)
- I²C bus shift register (ICDRS)

17.3.1 I²C Bus Control Register 1 (ICCR1)

ICCR1 enables or disables the I²C bus interface 2, controls transmission or reception, and selects master or slave mode, transmission or reception, and transfer clock frequency in master mode.

Bit	Bit Name	Initial Value	R/W	Description
7	ICE	0	R/W	I²C Bus Interface Enable 0: This module is halted. (SCL and SDA pins are set to port function.) 1: This bit is enabled for transfer operations. (SCL and SDA pins are bus drive state.)
6	RCVD	0	R/W	Reception Disable This bit enables or disables the next operation when TRS is 0 and ICDRR is read. 0: Enables next reception 1: Disables next reception
5	MST	0	R/W	Master/Slave Select
4	TRS	0	R/W	Transmit/Receive Select In master mode with the I²C bus format, when arbitration is lost, MST and TRS are both reset by hardware, causing a transition to slave receive mode. Modification of the TRS bit should be made between transfer frames. After data receive has been started in slave receive mode, when the first seven bits of the receive data agree with the slave address that is set to SAR and the eighth bit is 1, TRS is automatically set to 1. If an overrun error occurs in master mode with the clock synchronous serial format, MST is cleared to 0 and slave receive mode is entered. Operating modes are described below according to MST and TRS combination. When clocked synchronous serial format is selected and MST is 1, clock is output. 00: Slave receive mode 01: Slave transmit mode 10: Master receive mode 11: Master transmit mode

Bit	Bit Name	Initial Value	R/W	Description
3	CKS3	0	R/W	Transfer Clock Select 3 to 0
2	CKS2	0	R/W	These bits should be set according to the necessary transfer rate (see table 17.2) in master mode. In slave mode, these bits are used for reservation of the setup time in transmit mode. The time is 10 t_{cyc} when CKS3 = 0 and 20 t_{cyc} when CKS3 = 1.
1	CKS1	0	R/W	
0	CKS0	0	R/W	

Table 17.2 Transfer Rate

Bit 3	Bit 2	Bit 1	Bit 0		Transfer Rate				
CKS3	CKS2	CKS1	CKS0	Clock	φ = 5 MHz	φ = 8 MHz	φ = 10 MHz	φ = 16 MHz	φ = 20 MHz
0	0	0	0	φ/28	179 kHz	286 kHz	357 kHz	571 kHz	714 kHz
			1	φ/40	125 kHz	200 kHz	250 kHz	400 kHz	500 kHz
		1	0	φ/48	104 kHz	167 kHz	208 kHz	333 kHz	417 kHz
			1	φ/64	78.1 kHz	125 kHz	156 kHz	250 kHz	313 kHz
	1	0	0	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz	250 kHz
			1	φ/100	50.0 kHz	80.0 kHz	100 kHz	160 kHz	200 kHz
		1	0	φ/112	44.6 kHz	71.4 kHz	89.3 kHz	143 kHz	179 kHz
			1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz	156 kHz
1	0	0	0	φ/56	89.3 kHz	143 kHz	179 kHz	286 kHz	357 kHz
			1	φ/80	62.5 kHz	100 kHz	125 kHz	200 kHz	250 kHz
		1	0	φ/96	52.1 kHz	83.3 kHz	104 kHz	167 kHz	208 kHz
			1	φ/128	39.1 kHz	62.5 kHz	78.1 kHz	125 kHz	156 kHz
	1	0	0	φ/160	31.3 kHz	50.0 kHz	62.5 kHz	100 kHz	125 kHz
			1	φ/200	25.0 kHz	40.0 kHz	50.0 kHz	80.0 kHz	100 kHz
		1	0	φ/224	22.3 kHz	35.7 kHz	44.6 kHz	71.4 kHz	89.3 kHz
			1	φ/256	19.5 kHz	31.3 kHz	39.1 kHz	62.5 kHz	78.1 kHz

17.3.2 I²C Bus Control Register 2 (ICCR2)

ICCR1 issues start/stop conditions, manipulates the SDA pin, monitors the SCL pin, and controls reset in the control part of the I²C bus interface 2.

Bit	Bit Name	Initial Value	R/W	Description
7	BBSY	0	R/W	Bus Busy This bit enables to confirm whether the I²C bus is occupied or released and to issue start/stop conditions in master mode. With the clocked synchronous serial format, this bit has no meaning. With the I²C bus format, this bit is set to 1 when the SDA level changes from high to low under the condition of SCL = high, assuming that the start condition has been issued. This bit is cleared to 0 when the SDA level changes from low to high under the condition of SCL = high, assuming that the stop condition has been issued. Write 1 to BBSY and 0 to SCP to issue a start condition. Follow this procedure when also re-transmitting a start condition. Write 0 in BBSY and 0 in SCP to issue a stop condition. To issue start/stop conditions, use the MOV instruction.
6	SCP	1	W	Start/Stop Issue Condition Disable The SCP bit controls the issue of start/stop conditions in master mode. To issue a start condition, write 1 in BBSY and 0 in SCP. A retransmit start condition is issued in the same way. To issue a stop condition, write 0 in BBSY and 0 in SCP. This bit is always read as 1. If 1 is written, the data is not stored.
5	SDAO	1	R/W	SDA Output Value Control This bit is used with SDAOP when modifying output level of SDA. This bit should not be manipulated during transfer. 0: When reading, SDA pin outputs low. When writing, SDA pin is changed to output low. 1: When reading, SDA pin outputs high. When writing, SDA pin is changed to output Hi-Z (outputs high by external pull-up resistance).

Bit	Bit Name	Initial Value	R/W	Description
4	SDAOP	1	R/W	SDAO Write Protect This bit controls change of output level of the SDA pin by modifying the SDAO bit. To change the output level, clear SDAO and SDAOP to 0 or set SDAO to 1 and clear SDAOP to 0 by the MOV instruction. This bit is always read as 1.
3	SCLO	1	R	This bit monitors SCL output level. When SCLO is 1, SCL pin outputs high. When SCLO is 0, SCL pin outputs low.
2	—	1	—	Reserved This bit is always read as 1, and cannot be modified.
1	IICRST	0	R/W	IIC Control Part Reset This bit resets the control part except for I²C registers. If this bit is set to 1 when hang-up occurs because of communication failure during I²C operation, I²C control part can be reset without setting ports and initializing registers.
0	—	1	—	Reserved This bit is always read as 1, and cannot be modified.

17.3.3 I²C Bus Mode Register (ICMR)

ICMR selects whether the MSB or LSB is transferred first, performs master mode wait control, and selects the transfer bit count.

Bit	Bit Name	Initial Value	R/W	Description
7	MLS	0	R/W	MSB-First/LSB-First Select 0: MSB-first 1: LSB-first Set this bit to 0 when the I²C bus format is used.
6	WAIT	0	R/W	Wait Insertion Bit In master mode with the I²C bus format, this bit selects whether to insert a wait after data transfer except the acknowledge bit. When WAIT is set to 1, after the fall of the clock for the final data bit, low period is extended for two transfer clocks. If WAIT is cleared to 0, data and acknowledge bits are transferred consecutively with no wait inserted. The setting of this bit is invalid in slave mode with the I²C bus format or with the clocked synchronous serial format.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
3	BCWP	1	R/W	BC Write Protect This bit controls the BC2 to BC0 modifications. When modifying BC2 to BC0, this bit should be cleared to 0 and use the MOV instruction. In clock synchronous serial mode, BC should not be modified. 0: When writing, values of BC2 to BC0 are set. 1: When reading, 1 is always read. When writing, settings of BC2 to BC0 are invalid.
2	BC2	0	R/W	Bit Counter 2 to 0 These bits specify the number of bits to be transferred next. When read, the remaining number of transfer bits is indicated. With the I²C bus format, the data is transferred with one addition acknowledge bit. Bit BC2 to BC0 settings should be made during an interval between transfer frames. If bits BC2 to BC0 are set to a value other than 000, the setting should be made while the SCL pin is low. The value returns to 000 at the end of a data transfer, including the acknowledge bit. With the clock synchronous serial format, these bits should not be modified. I²C Bus Format / Clock Synchronous Serial Format 000: 9 bits / 000: 8 bits 001: 2 bits / 001: 1 bits 010: 3 bits / 010: 2 bits 011: 4 bits / 011: 3 bits 100: 5 bits / 100: 4 bits 101: 6 bits / 101: 5 bits 110: 7 bits / 110: 6 bits 111: 8 bits / 111: 7 bits
1	BC1	0	R/W	
0	BC0	0	R/W	

17.3.4 I²C Bus Interrupt Enable Register (ICIER)

ICIER enables or disables interrupt sources and acknowledge bits, sets acknowledge bits to be transferred, and confirms acknowledge bits to be received.

Bit	Bit Name	Initial Value	R/W	Description
7	TIE	0	R/W	Transmit Interrupt Enable When the TDRE bit in ICSR is set to 1, this bit enables or disables the transmit data empty interrupt (TXI). 0: Transmit data empty interrupt request (TXI) is disabled. 1: Transmit data empty interrupt request (TXI) is enabled.
6	TEIE	0	R/W	Transmit End Interrupt Enable This bit enables or disables the transmit end interrupt (TEI) at the rising of the ninth clock while the TDRE bit in ICSR is 1. TEI can be canceled by clearing the TEND bit or the TEIE bit to 0. 0: Transmit end interrupt request (TEI) is disabled. 1: Transmit end interrupt request (TEI) is enabled.
5	RIE	0	R/W	Receive Interrupt Enable This bit enables or disables the receive data full interrupt request (RXI) and the overrun error interrupt request (ERI) with the clocked synchronous format, when a receive data is transferred from ICDRS to ICDRR and the RDRF bit in ICSR is set to 1. RXI can be canceled by clearing the RDRF or RIE bit to 0. 0: Receive data full interrupt request (RXI) and overrun error interrupt request (ERI) with the clocked synchronous format are disabled. 1: Receive data full interrupt request (RXI) and overrun error interrupt request (ERI) with the clocked synchronous format are enabled.
4	NAKIE	0	R/W	NACK Receive Interrupt Enable This bit enables or disables the NACK receive interrupt request (NAKI) and the overrun error (setting of the OVE bit in ICSR) interrupt request (ERI) with the clocked synchronous format, when the NACKF and AL bits in ICSR are set to 1. NAKI can be canceled by clearing the NACKF, OVE, or NAKIE bit to 0. 0: NACK receive interrupt request (NAKI) is disabled. 1: NACK receive interrupt request (NAKI) is enabled.

Bit	Bit Name	Initial Value	R/W	Description
3	STIE	0	R/W	Stop Condition Detection Interrupt Enable 0: Stop condition detection interrupt request (STPI) is disabled. 1: Stop condition detection interrupt request (STPI) is enabled.
2	ACKE	0	R/W	Acknowledge Bit Judgement Select 0: The value of the receive acknowledge bit is ignored, and continuous transfer is performed. 1: If the receive acknowledge bit is 1, continuous transfer is halted.
1	ACKBR	0	R	Receive Acknowledge In transmit mode, this bit stores the acknowledge data that are returned by the receive device. This bit cannot be modified. 0: Receive acknowledge = 0 1: Receive acknowledge = 1
0	ACKBT	0	R/W	Transmit Acknowledge In receive mode, this bit specifies the bit to be sent at the acknowledge timing. 0: 0 is sent at the acknowledge timing. 1: 1 is sent at the acknowledge timing.

17.3.5 I²C Bus Status Register (ICSR)

ICSR performs confirmation of interrupt request flags and status.

Bit	Bit Name	Initial Value	R/W	Description
7	TDRE	0	R/W	Transmit Data Register Empty [Setting conditions] • When data is transferred from ICDRT to ICDRS and ICDRT becomes empty • When TRS is set • When a start condition (including re-transfer) has been issued • When transmit mode is entered from receive mode in slave mode [Clearing conditions] • When 0 is written in TDRE after reading TDRE = 1 • When data is written to ICDRT with an instruction
6	TEND	0	R/W	Transmit End [Setting conditions] • When the ninth clock of SCL rises with the I²C bus format while the TDRE flag is 1 • When the final bit of transmit frame is sent with the clock synchronous serial format [Clearing conditions] • When 0 is written in TEND after reading TEND = 1 • When data is written to ICDRT with an instruction
5	RDRF	0	R/W	Receive Data Register Full [Setting condition] • When a receive data is transferred from ICDRS to ICDRR [Clearing conditions] • When 0 is written in RDRF after reading RDRF = 1 • When ICDRR is read with an instruction

Bit	Bit Name	Initial Value	R/W	Description
4	NACKF	0	R/W	No Acknowledge Detection Flag [Setting condition] • When no acknowledge is detected from the receive device in transmission while the ACKE bit in ICIER is 1 [Clearing condition] • When 0 is written in NACKF after reading NACKF = 1
3	STOP	0	R/W	Stop Condition Detection Flag [Setting Conditions] • In master mode, when a stop condition is detected after frame transfer • In slave mode, when a stop condition is detected after the general call address or the first byte slave address, next to detection of start condition, accords with the address set in SAR [Clearing Condition] • When 0 is written in STOP after reading STOP = 1

Bit	Bit Name	Initial Value	R/W	Description
2	AL/OVE	0	R/W	Arbitration Lost Flag/Overrun Error Flag This flag indicates that arbitration was lost in master mode with the I²C bus format and that the final bit has been received while RDRF = 1 with the clocked synchronous format. When two or more master devices attempt to seize the bus at nearly the same time, if the I²C bus interface detects data differing from the data it sent, it sets AL to 1 to indicate that the bus has been taken by another master. [Setting conditions] • If the internal SDA and SDA pin disagree at the rise of SCL in master transmit mode • When the SDA pin outputs high in master mode while a start condition is detected • When the final bit is received with the clocked synchronous format while RDRF = 1 [Clearing condition] • When 0 is written in AL/OVE after reading AL/OVE=1
1	AAS	0	R/W	Slave Address Recognition Flag In slave receive mode, this flag is set to 1 if the first frame following a start condition matches bits SVA6 to SVA0 in SAR. [Setting conditions] • When the slave address is detected in slave receive mode • When the general call address is detected in slave receive mode. [Clearing condition] • When 0 is written in AAS after reading AAS=1

Bit	Bit Name	Initial Value	R/W	Description
0	ADZ	0	R/W	General Call Address Recognition Flag This bit is valid in I²C bus format slave receive mode. [Setting condition] • When the general call address is detected in slave receive mode [Clearing condition] • When 0 is written in ADZ after reading ADZ=1

17.3.6 Slave Address Register (SAR)

SAR selects the communication format and sets the slave address. When the chip is in slave mode with the I²C bus format, if the upper 7 bits of SAR match the upper 7 bits of the first frame received after a start condition, the chip operates as the slave device.

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	SVA6 to SVA0	All 0	R/W	Slave Address 6 to 0 These bits set a unique address in bits SVA6 to SVA0, differing form the addresses of other slave devices connected to the I²C bus.
0	FS	0	R/W	Format Select 0: I²C bus format is selected. 1: Clocked synchronous serial format is selected.

17.3.7 I²C Bus Transmit Data Register (ICDRT)

ICDRT is an 8-bit readable/writable register that stores the transmit data. When ICDRT detects the space in the shift register (ICDRS), it transfers the transmit data which is written in ICDRT to ICDRS and starts transferring data. If the next transfer data is written to ICDRT during transferring data of ICDRS, continuous transfer is possible. If the MLS bit of ICMR is set to 1 and when the data is written to ICDRT, the MSB/LSB inverted data is read. The initial value of ICDRT is H'FF.

17.3.8 I²C Bus Receive Data Register (ICDRR)

ICDRR is an 8-bit register that stores the receive data. When data of one byte is received, ICDRR transfers the receive data from ICDRS to ICDRR and the next data can be received. ICDRR is a receive-only register, therefore the CPU cannot write to this register. The initial value of ICDRR is H'FF.

17.3.9 I²C Bus Shift Register (ICDRS)

ICDRS is a register that is used to transfer/receive data. In transmission, data is transferred from ICDRT to ICDRS and the data is sent from the SDA pin. In reception, data is transferred from ICDRS to ICDRR after data of one byte is received. This register cannot be read directly from the CPU.

17.4 Operation

The I²C bus interface can communicate either in I²C bus mode or clocked synchronous serial mode by setting FS in SAR.

17.4.1 I²C Bus Format

Figure 17.3 shows the I²C bus formats. Figure 17.4 shows the I²C bus timing. The first frame following a start condition always consists of 8 bits.

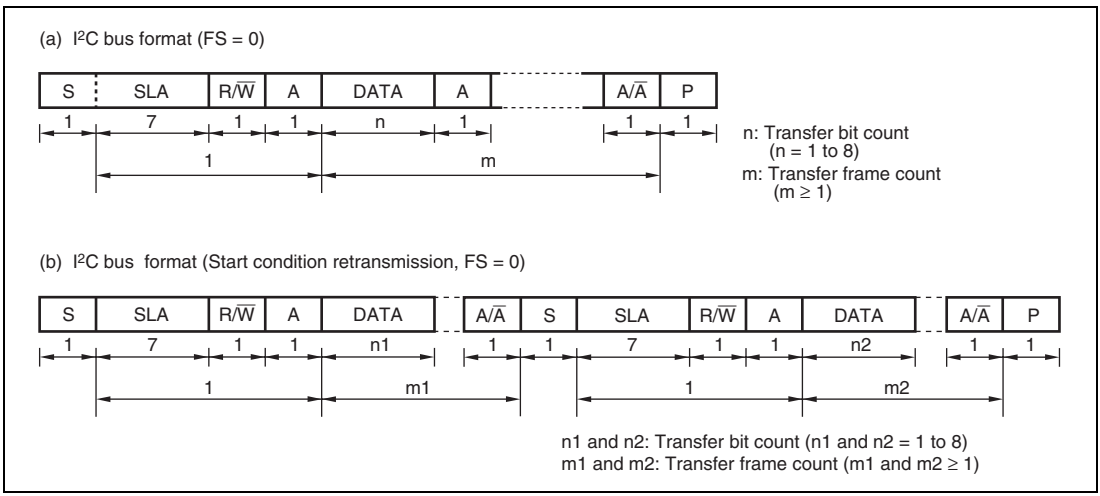


Figure 17.3 I²C Bus Formats

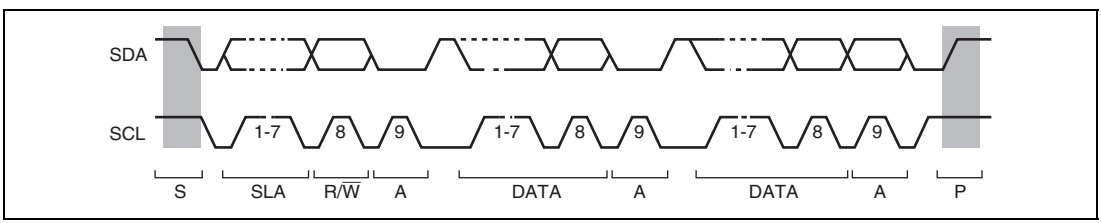


Figure 17.4 I²C Bus Timing

Legend

S: Start condition. The master device drives SDA from high to low while SCL is high.
SLA: Slave address

R/$\overline{W}$: Indicates the direction of data transfer: from the slave device to the master device when R/$\overline{W}$ is 1, or from the master device to the slave device when R/$\overline{W}$ is 0.
A: Acknowledge. The receive device drives SDA to low.
DATA: Transfer data
P: Stop condition. The master device drives SDA from low to high while SCL is high.

17.4.2 Master Transmit Operation

In master transmit mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For master transmit mode operation timing, refer to figures 17.5 and 17.6. The transmission procedure and operations in master transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 to CKS0 bits in ICCR1 to 1. (Initial setting)
2. Read the BBSY flag in ICCR2 to confirm that the bus is free. Set the MST and TRS bits in ICCR1 to select master transmit mode. Then, write 1 to BBSY and 0 to SCP using MOV instruction. (Start condition issued) This generates the start condition.
3. After confirming that TDRE in ICSR has been set, write the transmit data (the first byte data show the slave address and R/$\overline{W}$) to ICDRT. At this time, TDRE is automatically cleared to 0, and data is transferred from ICDRT to ICDRS. TDRE is set again.
4. When transmission of one byte data is completed while TDRE is 1, TEND in ICSR is set to 1 at the rise of the 9th transmit clock pulse. Read the ACKBR bit in ICIER, and confirm that the slave device has been selected. Then, write second byte data to ICDRT. When ACKBR is 1, the slave device has not been acknowledged, so issue the stop condition. To issue the stop condition, write 0 to BBSY and SCP using MOV instruction. SCL is fixed low until the transmit data is prepared or the stop condition is issued.
5. The transmit data after the second byte is written to ICDRT every time TDRE is set.
6. Write the number of bytes to be transmitted to ICDRT. Wait until TEND is set (the end of last byte data transmission) while TDRE is 1, or wait for NACK (NACKF in ICSR = 1) from the receive device while ACKE in ICIER is 1. Then, issue the stop condition to clear TEND or NACKF.
7. When the STOP bit in ICSR is set to 1, the operation returns to the slave receive mode.

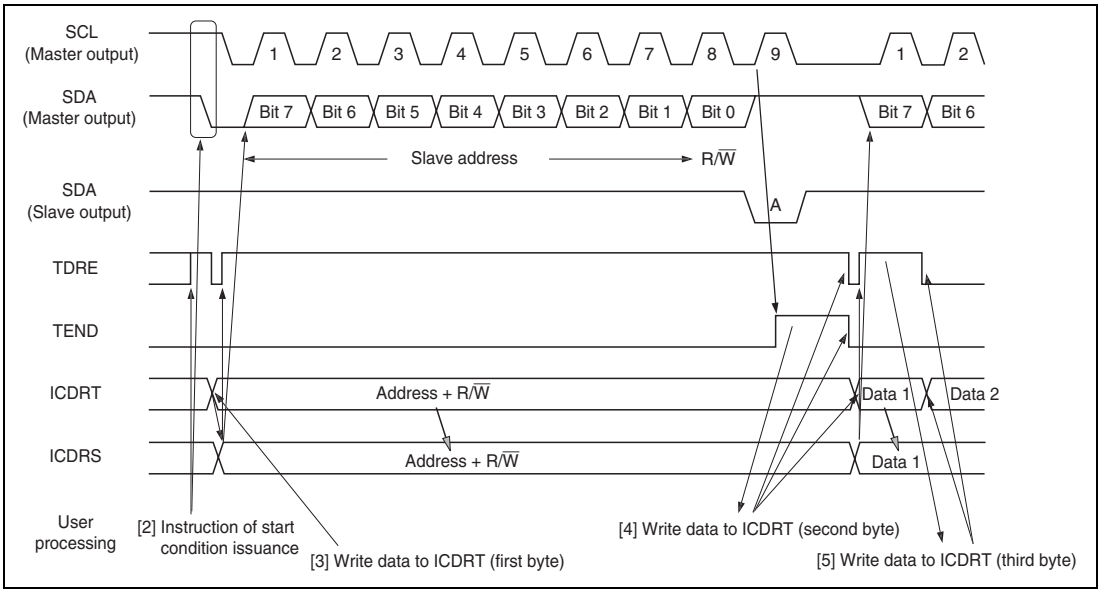


Figure 17.5 Master Transmit Mode Operation Timing (1)

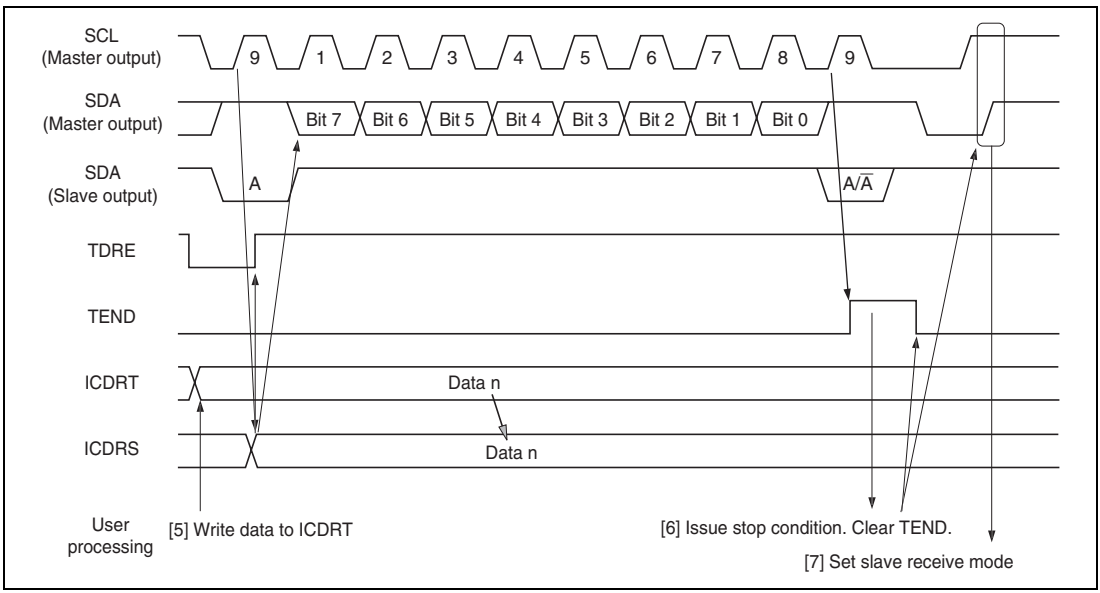


Figure 17.6 Master Transmit Mode Operation Timing (2)

17.4.3 Master Receive Operation

In master receive mode, the master device outputs the receive clock, receives data from the slave device, and returns an acknowledge signal. For master receive mode operation timing, refer to figures 17.7 and 17.8. The reception procedure and operations in master receive mode are shown below.

1. Clear the TEND bit in ICSR to 0, then clear the TRS bit in ICCR1 to 0 to switch from master transmit mode to master receive mode. Then, clear the TDRE bit to 0.
2. When ICDRR is read (dummy data read), reception is started, and the receive clock is output, and data received, in synchronization with the internal clock. The master device outputs the level specified by ACKBT in ICIER to SDA, at the 9th receive clock pulse.
3. After the reception of first frame data is completed, the RDRF bit in ICST is set to 1 at the rise of 9th receive clock pulse. At this time, the receive data is read by reading ICDRR, and RDRF is cleared to 0.
4. The continuous reception is performed by reading ICDRR every time RDRF is set. If 8th receive clock pulse falls after reading ICDRR by the other processing while RDRF is 1, SCL is fixed low until ICDRR is read.
5. If next frame is the last receive data, set the RCVD bit in ICCR1 to 1 before reading ICDRR. This enables the issuance of the stop condition after the next reception.
6. When the RDRF bit is set to 1 at rise of the 9th receive clock pulse, issue the stage condition.
7. When the STOP bit in ICSR is set to 1, read ICDRR. Then clear the RCVD bit to 0.
8. The operation returns to the slave receive mode.

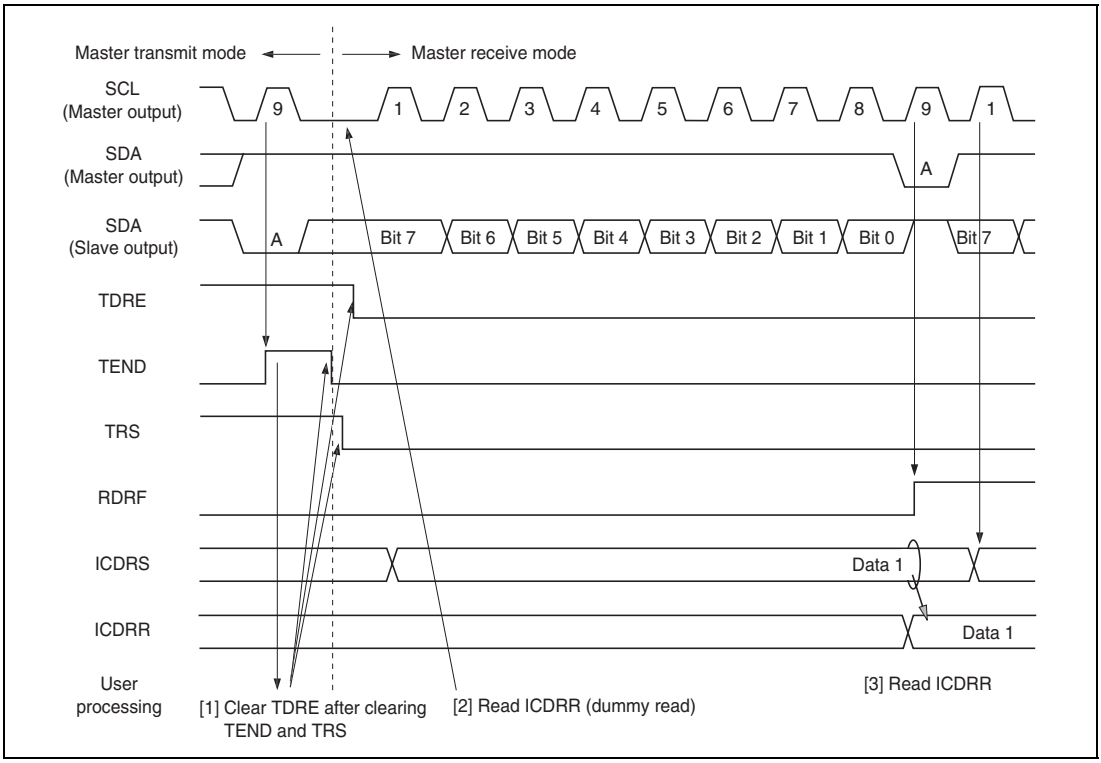


Figure 17.7 Master Receive Mode Operation Timing (1)

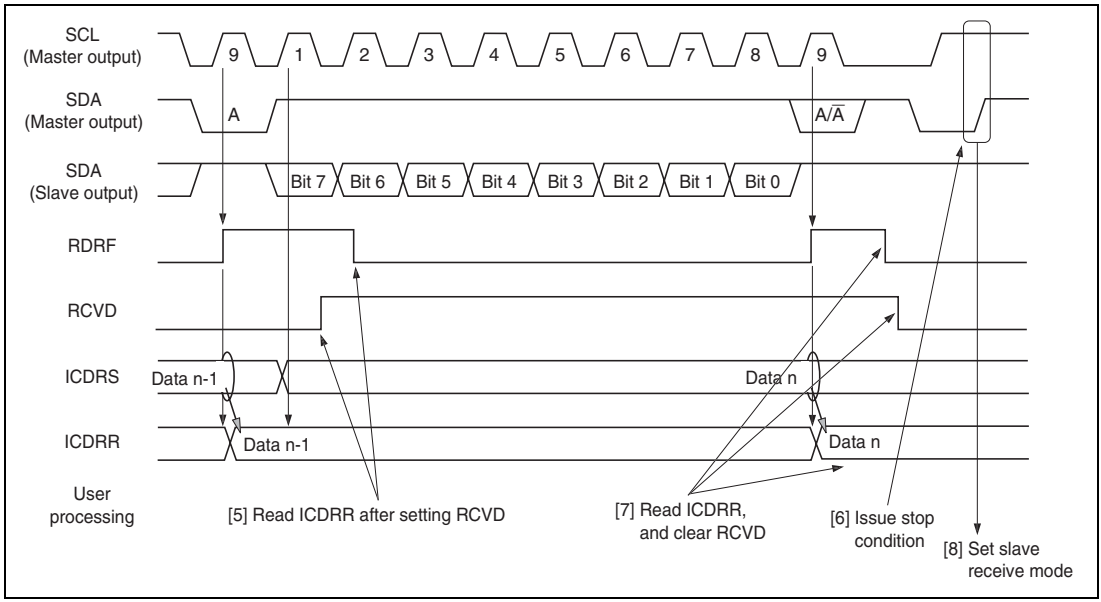


Figure 17.8 Master Receive Mode Operation Timing (2)

17.4.4 Slave Transmit Operation

In slave transmit mode, the slave device outputs the transmit data, while the master device outputs the receive clock and returns an acknowledge signal. For slave transmit mode operation timing, refer to figures 17.9 and 17.10.

The transmission procedure and operations in slave transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 to CKS0 bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the 9th clock pulse. At this time, if the 8th bit data (R/$\overline{W}$) is 1, the TRS and ICSR bits in ICCR1 are set to 1, and the mode changes to slave transmit mode automatically. The continuous transmission is performed by writing transmit data to ICDRT every time TDRE is set.
3. If TDRE is set after writing last transmit data to ICDRT, wait until TEND in ICSR is set to 1, with TDRE = 1. When TEND is set, clear TEND.
4. Clear TRS for the end processing, and read ICDRR (dummy read). SCL is free.
5. Clear TDRE.

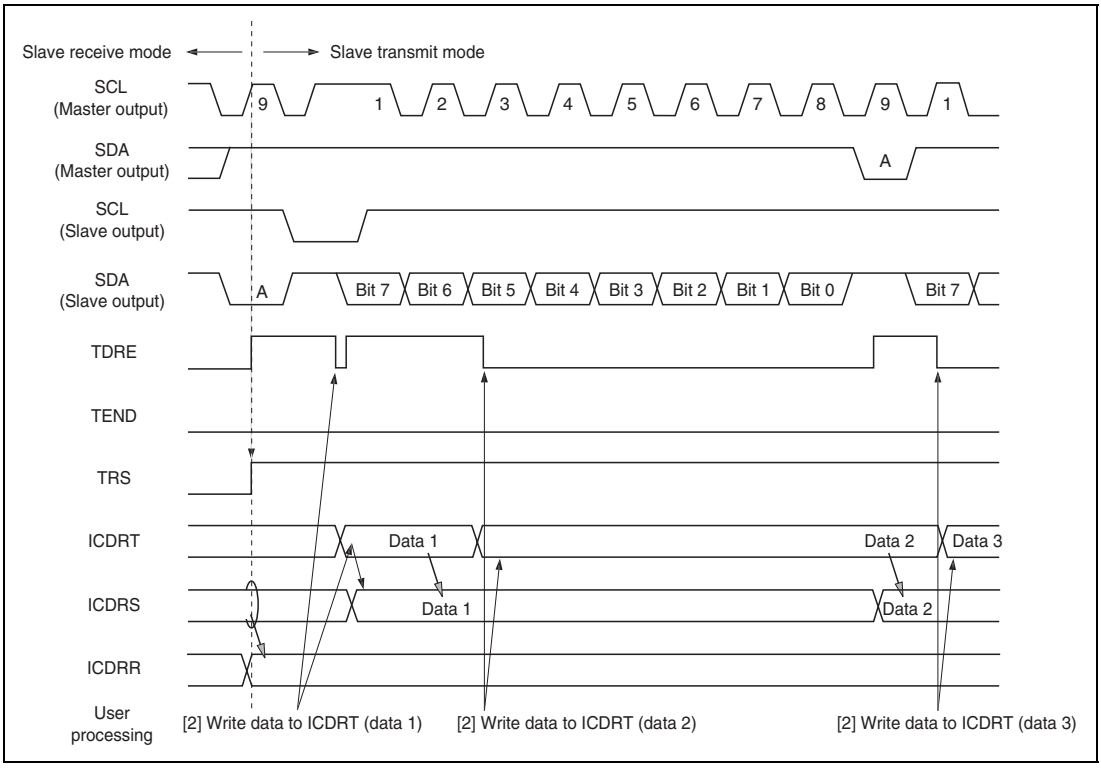


Figure 17.9 Slave Transmit Mode Operation Timing (1)

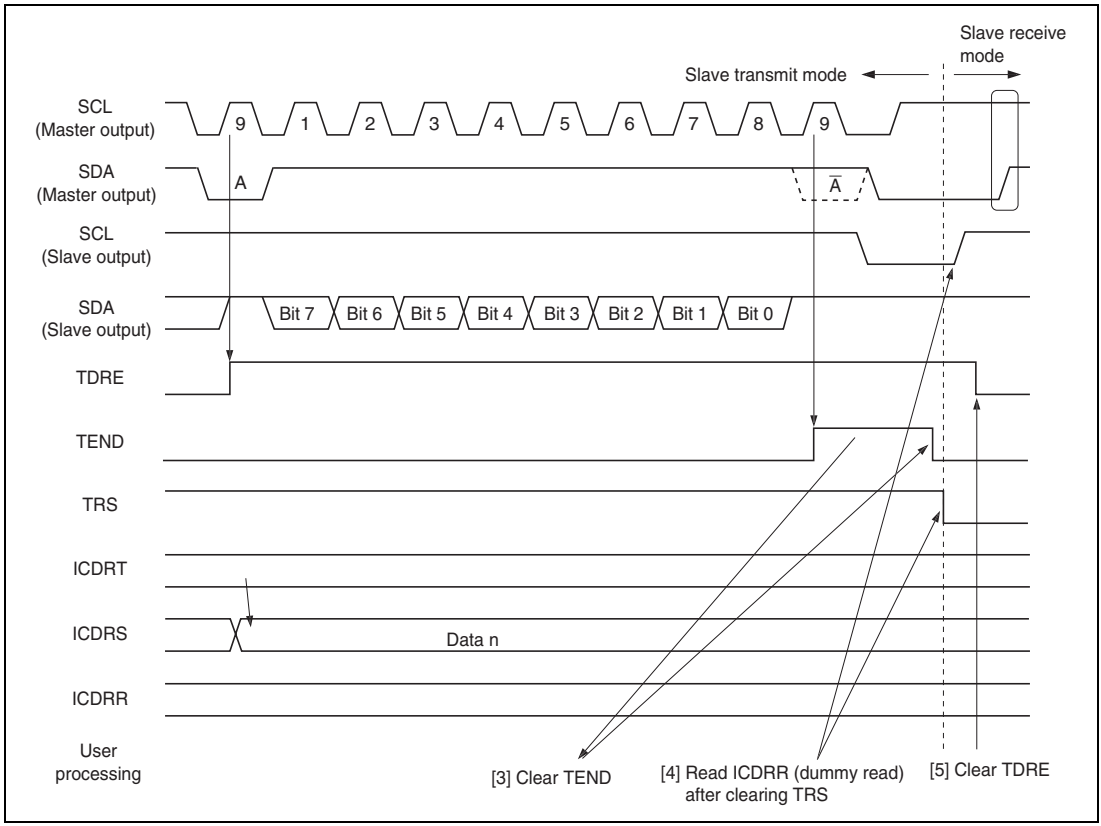


Figure 17.10 Slave Transmit Mode Operation Timing (2)

17.4.5 Slave Receive Operation

In slave receive mode, the master device outputs the transmit clock and transmit data, and the slave device returns an acknowledge signal. For slave receive mode operation timing, refer to figures 17.11 and 17.12. The reception procedure and operations in slave receive mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MLS and WAIT bits in ICMR and the CKS3 to CKS0 bits in ICCR1 to 1. (Initial setting) Set the MST and TRS bits in ICCR1 to select slave receive mode, and wait until the slave address matches.
2. When the slave address matches in the first frame following detection of the start condition, the slave device outputs the level specified by ACKBT in ICIER to SDA, at the rise of the 9th clock pulse. At the same time, RDRF in ICSR is set to read ICDRR (dummy read). (Since the read data show the slave address and R/$\overline{W}$, it is not used.)

3. Read ICDRR every time RDRF is set. If 8th receive clock pulse falls while RDRF is 1, SCL is fixed low until ICDRR is read. The change of the acknowledge before reading ICDRR, to be returned to the master device, is reflected to the next transmit frame.
4. The last byte data is read by reading ICDRR.

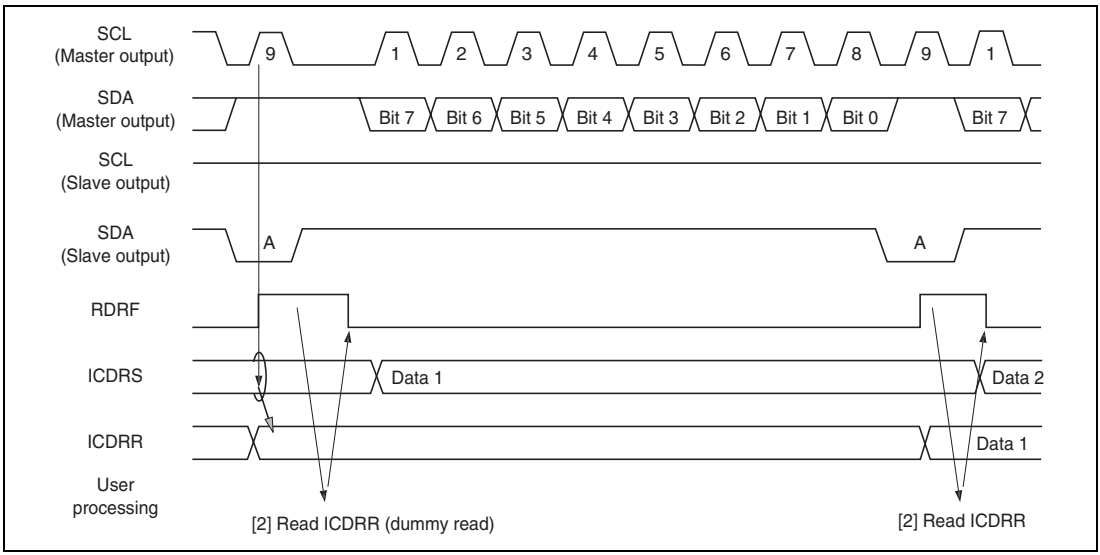


Figure 17.11 Slave Receive Mode Operation Timing (1)

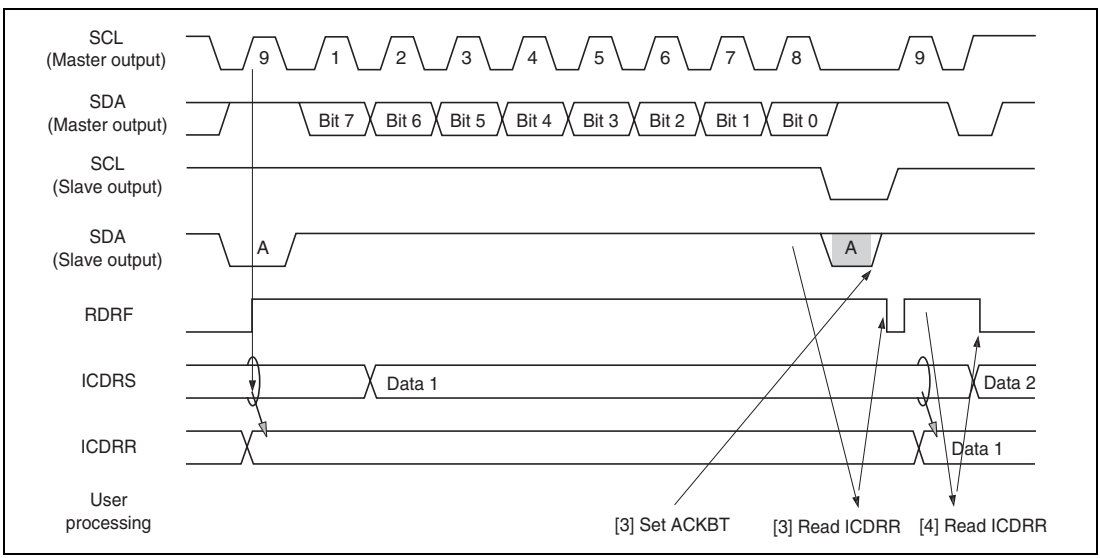


Figure 17.12 Slave Receive Mode Operation Timing (2)

17.4.6 Clocked Synchronous Serial Format

This module can be operated with the clocked synchronous serial format, by setting the FS bit in SAR to 1. When the MST bit in ICCR1 is 1, the transfer clock output from SCL is selected. When MST is 0, the external clock input is selected.

Data Transfer Format

Figure 17.13 shows the clocked synchronous serial transfer format.

The transfer data is output from the rise to the fall of the SCL clock, and the data at the rising edge of the SCL clock is guaranteed. The MLS bit in ICMR sets the order of data transfer, in either the MSB first or LSB first. The output level of SDA can be changed during the transfer wait, by the SDAO bit in ICCR2.

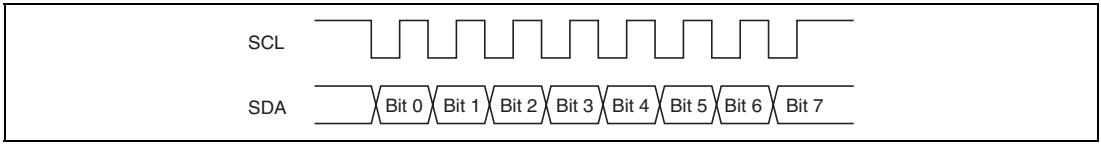


Figure 17.13 Clocked Synchronous Serial Transfer Format

Transmit Operation

In transmit mode, transmit data is output from SDA, in synchronization with the fall of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For transmit mode operation timing, refer to figure 17.14. The transmission procedure and operations in transmit mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS3 to CKS0 bits in ICCR1 to 1. (Initial setting)
2. Set the TRS bit in ICCR1 to select the transmit mode. Then, TDRE in ICSR is set.
3. Confirm that TDRE has been set. Then, write the transmit data to ICDRT. The data is transferred from ICDRT to ICDRS, and TDRE is set automatically. The continuous transmission is performed by writing data to ICDRT every time TDRE is set. When changing from transmit mode to receive mode, clear TRS while TDRE is 1.

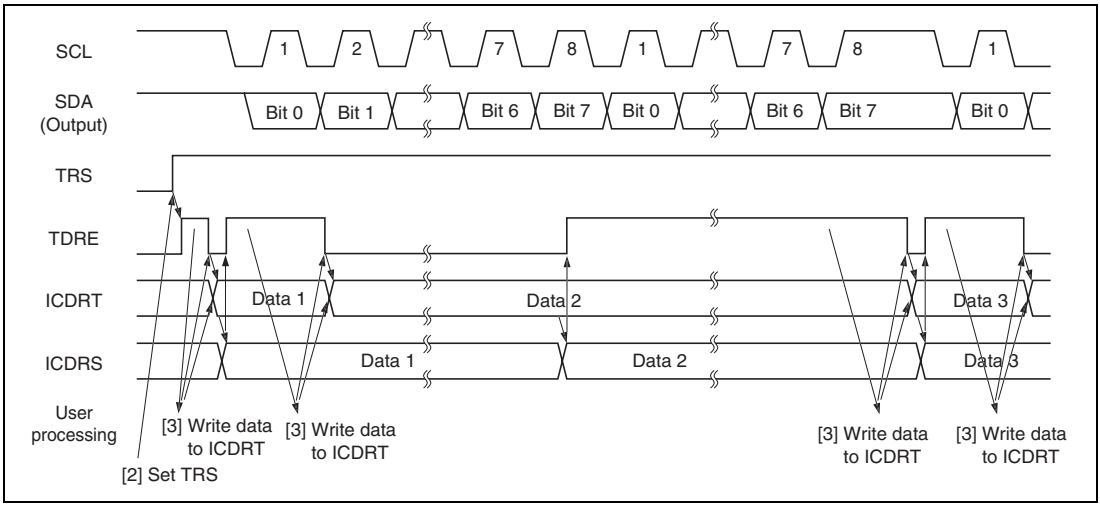


Figure 17.14 Transmit Mode Operation Timing

Receive Operation

In receive mode, data is latched at the rise of the transfer clock. The transfer clock is output when MST in ICCR1 is 1, and is input when MST is 0. For receive mode operation timing, refer to figure 17.15. The reception procedure and operations in receive mode are described below.

1. Set the ICE bit in ICCR1 to 1. Set the MST and CKS3 to CKS0 bits in ICCR1 to 1. (Initial setting)
2. When the transfer clock is output, set MST to 1 to start outputting the receive clock.
3. When the receive operation is completed, data is transferred from ICDRS to ICDRR and RDRF in ICSR is set. When MST = 1, the next byte can be received, so the clock is continually output. The continuous reception is performed by reading ICDRR every time RDRF is set. When the 8th clock is risen while RDRF is 1, the overrun is detected and AL/OVE in ICSR is set. At this time, the previous reception data is retained in ICDRR.
4. To stop receiving when MST = 1, set RCVD in ICCR1 to 1, then read ICDRR. Then, SCL is fixed high after receiving the next byte data.

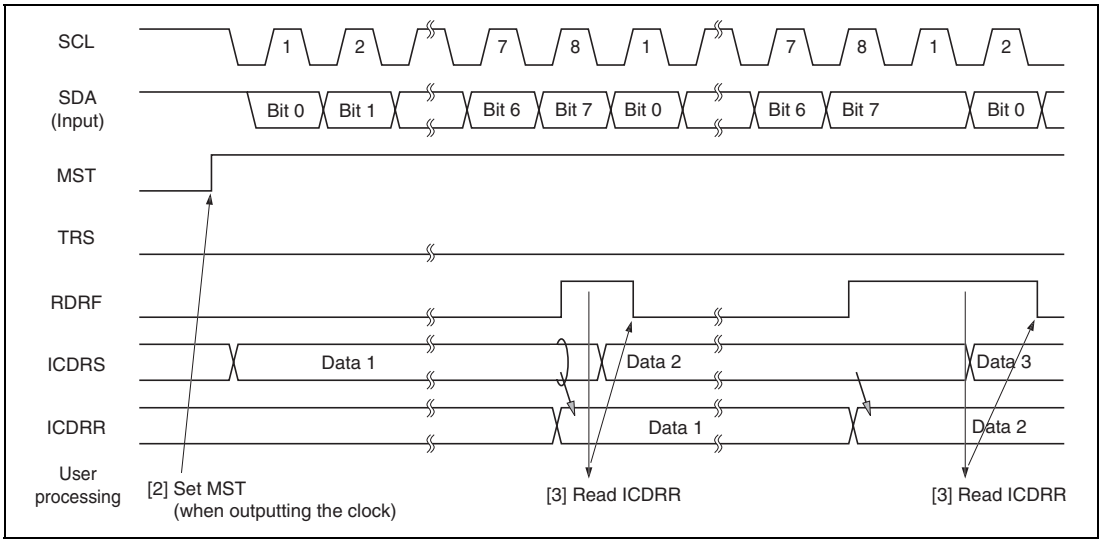


Figure 17.15 Receive Mode Operation Timing

17.4.7 Noise Canceler

The logic levels at the SCL and SDA pins are routed through noise cancelers before being latched internally. Figure 17.16 shows a block diagram of the noise canceler circuit.

The noise canceler consists of two cascaded latches and a match detector. The SCL (or SDA) input signal is sampled on the system clock, but is not passed forward to the next circuit unless the outputs of both latches agree. If they do not agree, the previous value is held.

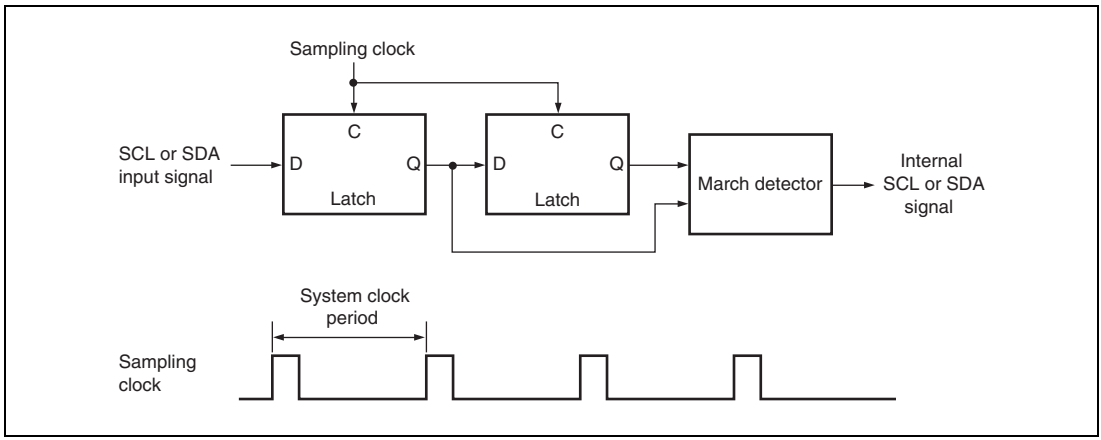


Figure 17.16 Block Diagram of Noise Conceler

17.4.8 Example of Use

Flowcharts in respective modes that use the I²C bus interface are shown in figures 17.17 to 17.20.

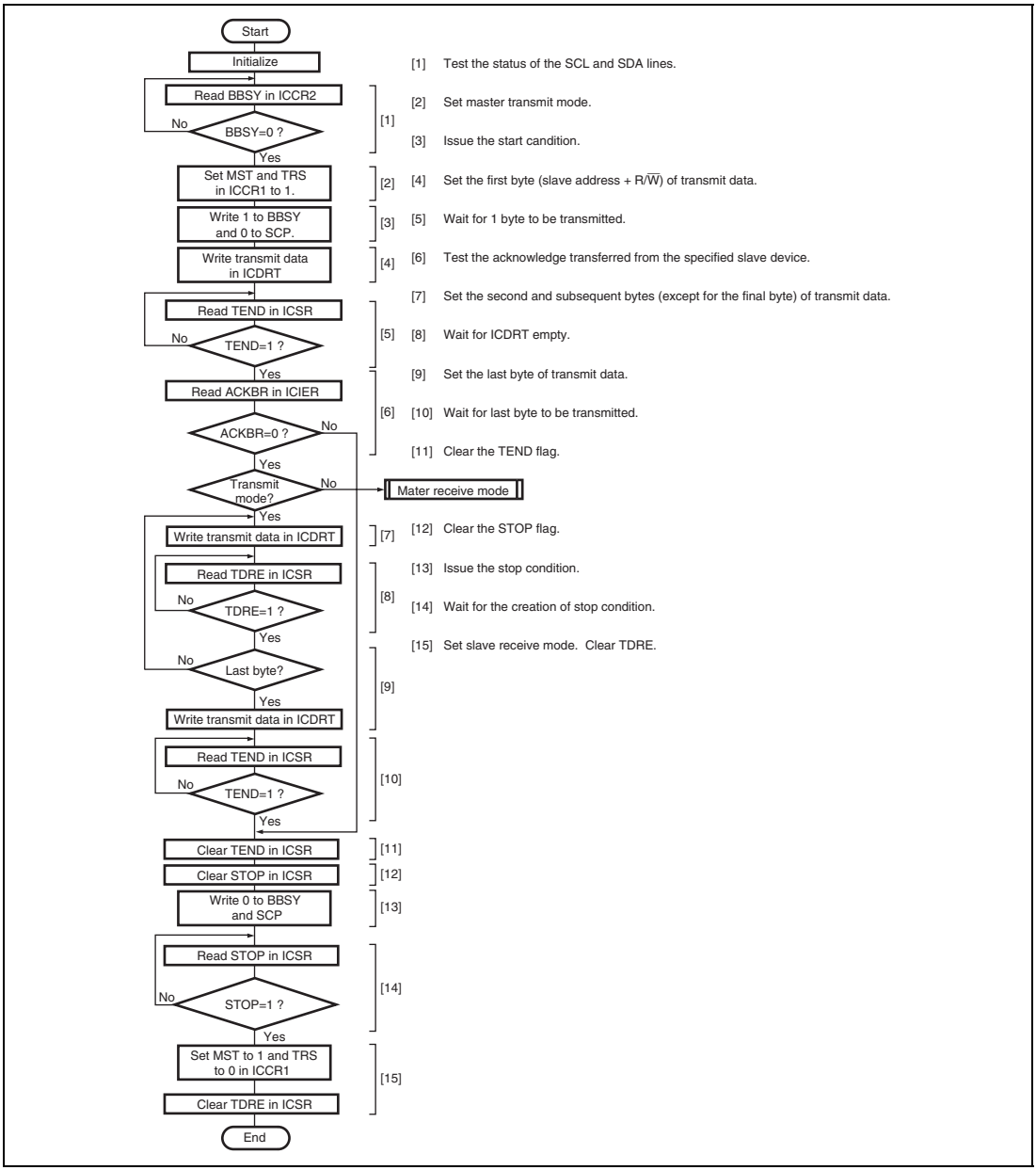
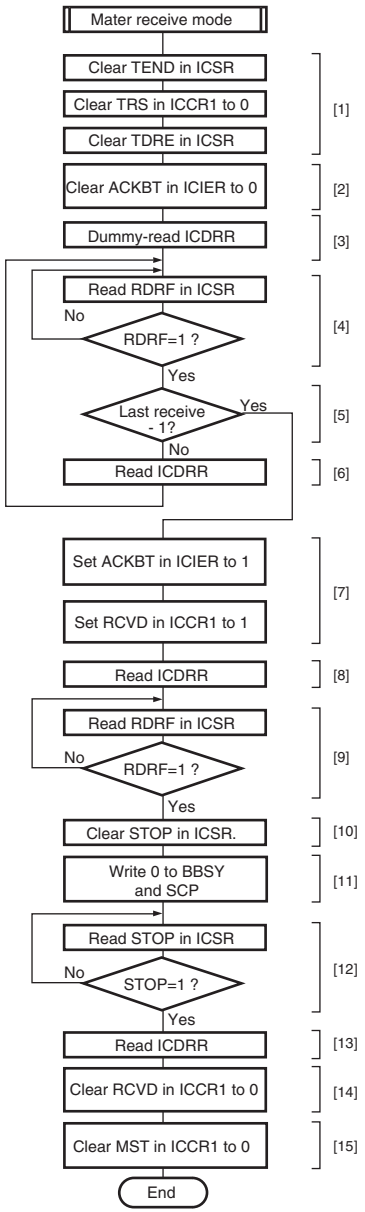


Figure 17.17 Sample Flowchart for Master Transmit Mode



[1] Clear TEND, select master receive mode, and then clear TDRE.*

[2] Set acknowledge to the transmit device.*

[3] Dummy-read ICDDR.*

[4] Wait for 1 byte to be received

[5] Check whether it is the (last receive - 1).

[6] Read the receive data last.

[7] Set acknowledge of the final byte. Disable continuous reception (RCVD = 1).

[8] Read the (final byte - 1) of receive data.

[9] Wait for the last byte to be receive.

[10] Clear the STOP flag.

[11] Issue the stop condition.

[12] Wait for the creation of stop condition.

[13] Read the last byte of receive data.

[14] Clear RCVD.

[15] Set slave receive mode.

Note: Do not activate an interrupt during the execution of steps [1] to [3].

Supplementary explanation: When one byte is received, steps [2] to [6] are skipped after step [1], before jumping to step [7]. The step [8] is dummy-read in ICDRR.

Figure 17.18 Sample Flowchart for Master Receive Mode

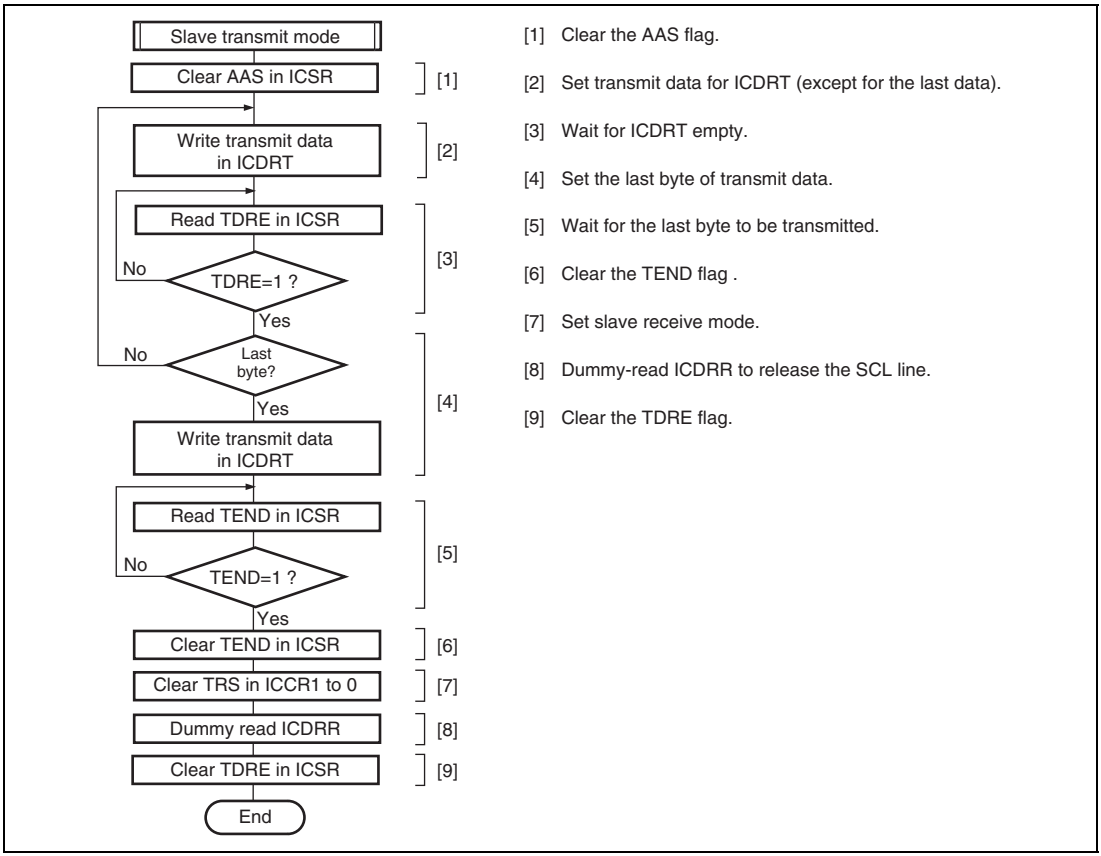
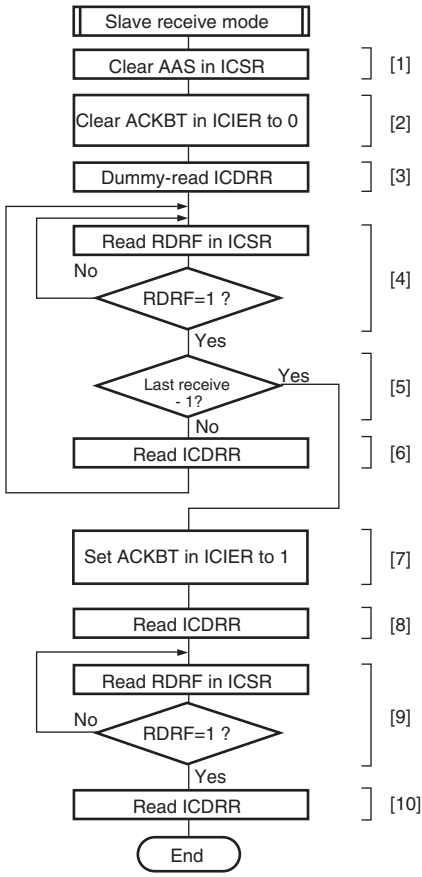


Figure 17.19 Sample Flowchart for Slave Transmit Mode



[1] Clear the AAS flag.

[2] Set acknowledge to the transmit device.

[3] Dummy-read ICDRR.

[4] Wait for 1 byte to be received.

[5] Check whether it is the (last receive - 1).

[6] Read the receive data.

[7] Set acknowledge of the last byte.

[8] Read the (last byte - 1) of receive data.

[9] Wait the last byte to be received.

[10] Read for the last byte of receive data.

Supplementary explanation: When one byte is received, steps [2] to [6] are skipped after step [1], before jumping to step [7]. The step [8] is dummy-read in ICDRR.

Figure 17.20 Sample Flowchart for Slave Receive Mode

17.5 Interrupt Request

There are six interrupt requests in this module; transmit data empty, transmit end, receive data full, NACK receive, STOP recognition, and arbitration lost/overrun error. Table 17.3 shows the contents of each interrupt request.

Table 17.3 Interrupt Requests

Interrupt Request	Abbreviation	Interrupt Condition	I²C Mode	Clocked Synchronous Mode
Transmit Data Empty	TXI	(TDRE=1) • (TIE=1)	○	○
Transmit End	TEI	(TEND=1) • (TEIE=1)	○	○
Receive Data Full	RXI	(RDRF=1) • (RIE=1)	○	○
STOP Recognition	STPI	(STOP=1) • (STIE=1)	○	×
NACK Receive	NAKI	{(NACKF=1)+(AL=1)} • (NAKIE=1)	○	×
Arbitration Lost/Overrun Error			○	○

When interrupt conditions described in table 17.3 are 1 and the I bit in CCR is 0, the CPU executes an interrupt exception processing. Interrupt sources should be cleared in the exception processing. TDRE and TEND are automatically cleared to 0 by writing the transmit data to ICDRT. RDRF are automatically cleared to 0 by reading ICDRR. TDRE is set to 1 again at the same time when transmit data is written to ICDRT. When TDRE is cleared to 0, then an excessive data of one byte may be transmitted.

17.6 Bit Synchronous Circuit

In master mode, this module has a possibility that high level period may be short in the two states described below.

- When SCL is driven to low by the slave device
- When the rising speed of SCL is lowered by the load of the SCL line (load capacitance or pull-up resistance)

Therefore, it monitors SCL and communicates by bit with synchronization.

Figure 17.21 shows the timing of the bit synchronous circuit and table 17.4 shows the time when SCL output changes from low to Hi-Z then SCL is monitored.

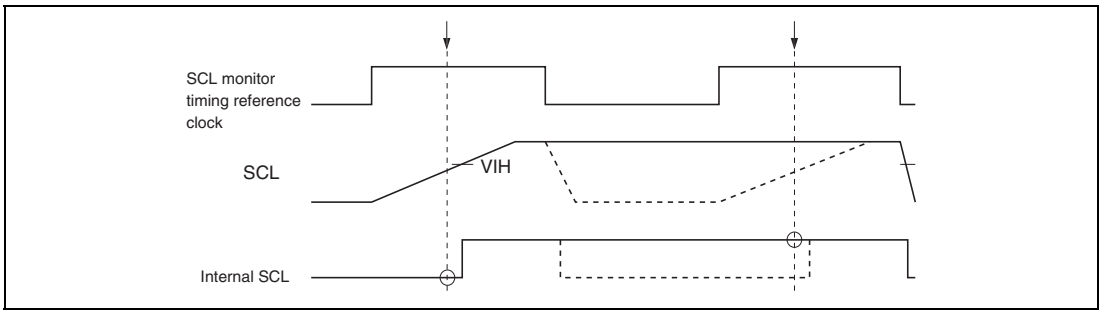


Figure 17.21 The Timing of the Bit Synchronous Circuit

Table 17.4 Time for Monitoring SCL

CKS3	CKS2	Time for Monitoring SCL
0	0	7.5 tcyc
	1	19.5 tcyc
1	0	17.5 tcyc
	1	41.5 tcyc

17.7 Usage Notes

17.7.1 Issue (Retransmission) of Start/Stop Conditions

In master mode, when the start/stop conditions are issued (retransmitted) at the specific timing under the following condition 1 or 2, such conditions may not be output successfully. To avoid this, issue (retransmit) the start/stop conditions after the fall of the ninth clock is confirmed. Check the SCLO bit in the I²C control register 2 (IICR2) to confirm the fall of the ninth clock.

1. When the rising of SCL falls behind the time specified in section 17.6, Bit Synchronous Circuit, by the load of the SCL bus (load capacitance or pull-up resistance)
2. When the bit synchronous circuit is activated by extending the low period of eighth and ninth clocks, that is driven by the slave device

17.7.2 WAIT Setting in I²C Bus Mode Register (ICMR)

If the WAIT bit is set to 1, and the SCL signal is driven low for two or more transfer clocks by the slave device at the eighth and ninth clocks, the high period of ninth clock may be shortened. To avoid this, set the WAIT bit in ICMR to 0.

Section 18 A/D Converter

This LSI includes a successive approximation type 10-bit A/D converter that allows up to eight analog input channels to be selected. The block diagram of the A/D converter is shown in figure 18.1.

18.1 Features

- 10-bit resolution
- Eight input channels
- Conversion time: at least 3.5 µs 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 data register for each channel
- Sample-and-hold function
- Two conversion start methods
  - — Software
  - — External trigger signal
- Interrupt request
  - — An A/D conversion end interrupt request (ADI) can be generated

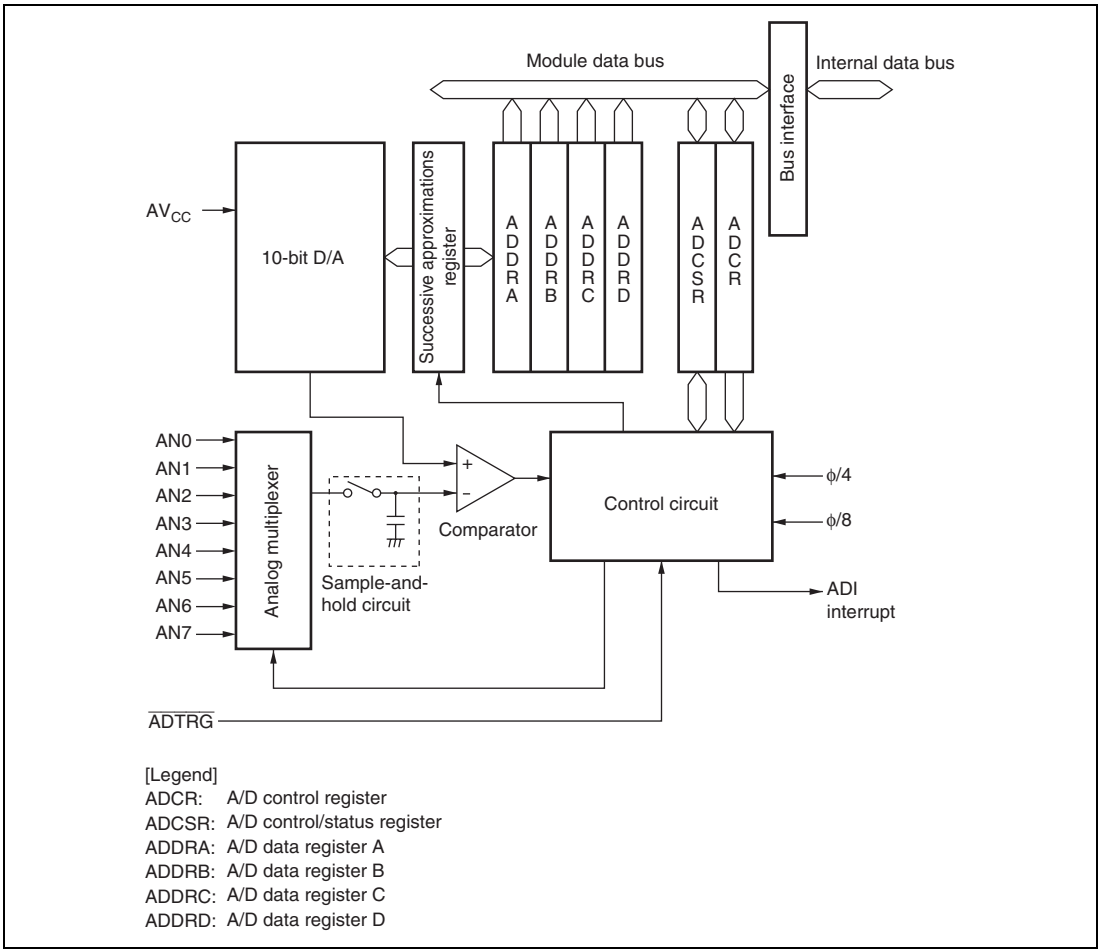


Figure 18.1 Block Diagram of A/D Converter

18.2 Input/Output Pins

Table 18.1 summarizes the input pins used by the A/D converter. The 8 analog input pins are divided into two 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. The AVcc pin is the power supply pin for the analog block in the A/D converter.

Table 18.1 Pin Configuration

Pin Name	Abbreviation	I/O	Function
Analog power supply pin	AV_{CC}	Input	Analog block power supply
Analog input pin 0	AN0	Input	Group 0 analog input
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
Analog input pin 5	AN5	Input	
Analog input pin 6	AN6	Input	
Analog input pin 7	AN7	Input	
A/D external trigger input pin	$\overline{\text{ADTRG}}$	Input	External trigger input for starting A/D conversion

18.3 Register Descriptions

The A/D converter has the following registers.

- 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)

18.3.1 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 analog input channel, are shown in table 18.2.

The converted 10-bit data is stored in bits 15 to 6. The lower 6 bits are always read as 0.

The data bus width between the CPU and the A/D converter is 8 bits. 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. Therefore byte access to ADDR should be done by reading the upper byte first then the lower one. Word access is also possible. ADDR is initialized to H'0000.

Table 18.2 Analog Input Channels and Corresponding ADDR Registers

Analog Input Channel		
Group 0	**Group 1**	**A/D Data Register to Be Stored Results of A/D Conversion**
AN0	AN4	ADDRA
AN1	AN5	ADDRB
AN2	AN6	ADDRC
AN3	AN7	ADDRD

18.3.2 A/D Control/Status Register (ADCSR)

ADCSR consists of the control bits and conversion end status bits of the A/D converter.

Bit	Bit Name	Initial Value	R/W	Description
7	ADF	0	R/W	A/D End Flag [Setting conditions] • When A/D conversion ends in single mode • When A/D conversion ends once on all the channels selected in scan mode [Clearing condition] • When 0 is written after reading ADF = 1
6	ADIE	0	R/W	A/D Interrupt Enable A/D conversion end interrupt request (ADI) is enabled by ADF when this bit is set to 1
5	ADST	0	R/W	A/D Start 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 standby mode.
4	SCAN	0	R/W	Scan Mode Selects single mode or scan mode as the A/D conversion operating mode. 0: Single mode 1: Scan mode
3	CKS	0	R/W	Clock Select Selects the A/D conversions time. 0: Conversion time = 134 states (max.) 1: Conversion time = 70 states (max.) Clear the ADST bit to 0 before switching the conversion time.

Bit	Bit Name	Initial Value	R/W	Description	
2	CH2	0	R/W	Channel Select 2 to 0	
1	CH1	0	R/W	Select analog input channels.	
0	CH0	0	R/W	When SCAN = 0	When SCAN = 1
				000: AN0	000: AN0
				001: AN1	001: AN0 and AN1
				010: AN2	010: AN0 to AN2
				011: AN3	011: AN0 to AN3
				100: AN4	100: AN4
				101: AN5	101: AN4 and AN5
				110: AN6	110: AN4 to AN6
				111: AN7	111: AN4 to AN7

18.3.3 A/D Control Register (ADCR)

ADCR enables A/D conversion started by an external trigger signal.

Bit	Bit Name	Initial Value	R/W	Description
7	TRGE	0	R/W	Trigger Enable A/D conversion is started at the falling edge and the rising edge of the external trigger signal ($\overline{\text{ADTRG}}$) when this bit is set to 1. The selection between the falling edge and rising edge of the external trigger pin ($\overline{\text{ADTRG}}$) conforms to the WPEG5 bit in the interrupt edge select register 2 (IEGR2)
6 to 1	—	All 1	—	Reserved These bits are always read as 1.
0	—	0	R/W	Reserved Do not set this bit to 1, though the bit is readable/writable.

18.4 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 in ADCSR to 0. The ADST bit can be set at the same time as the operating mode or analog input channel is changed.

18.4.1 Single Mode

In single mode, A/D conversion is performed once for the analog input of the specified single channel as follows:

1. A/D conversion is started when the ADST bit in ADCSR 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 of 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.

18.4.2 Scan Mode

In scan mode, A/D conversion is performed sequentially for the analog input of the specified channels (four channels maximum) as follows:

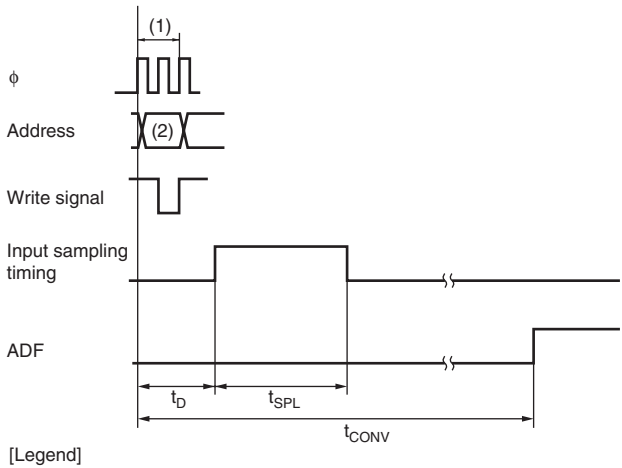
1. When the ADST bit in ADCSR is set to 1 by software or external trigger input, A/D conversion starts on the first channel in the group (AN0 when CH2 = 0, AN4 when CH2 = 1).
2. When A/D conversion for each channel is completed, the result is sequentially transferred to the A/D data register corresponding to each channel.
3. When conversion of all the selected channels is completed, the ADF flag in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt requested is generated. A/D conversion starts again on the first channel in the group.
4. The ADST bit is not automatically cleared to 0. Steps [2] and [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.

18.4.3 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 18.2 shows the A/D conversion timing. Table 18.3 shows the A/D conversion time.

As indicated in figure 18.2, the A/D conversion time includes t_D and the input sampling time. 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 18.3.

In scan mode, the values given in table 18.3 apply to the first conversion time. In the second and subsequent conversions, the conversion time is 128 states (fixed) when CKS = 0 and 66 states (fixed) when CKS = 1.



[Legend]
(1) : ADCSR write cycle
(2) : ADCSR address
t_D : A/D conversion start delay time
t_{SPL} : Input sampling time
t_{CONV} : A/D conversion time

Figure 18.2 A/D Conversion Timing

Table 18.3 A/D Conversion Time (Single Mode)

		CKS = 0			CKS = 1		
Item	**Symbol**	**Min**	**Typ**	**Max**	**Min**	**Typ**	**Max**
A/D conversion start delay time	t_D	6	—	9	4	—	5
Input sampling time	t_{SPL}	—	31	—	—	15	—
A/D conversion time	t_{CONV}	131	—	134	69	—	70

Note: All values represent the number of states.

18.4.4 External Trigger Input Timing

A/D conversion can also be started by an external trigger input. When the TRGE bit in ADCR is set to 1, external trigger input is enabled at the $\overline{\text{ADTRG}}$ pin. A falling edge at the $\overline{\text{ADTRG}}$ input pin sets the ADST bit in ADCSR to 1, 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 18.3 shows the timing.

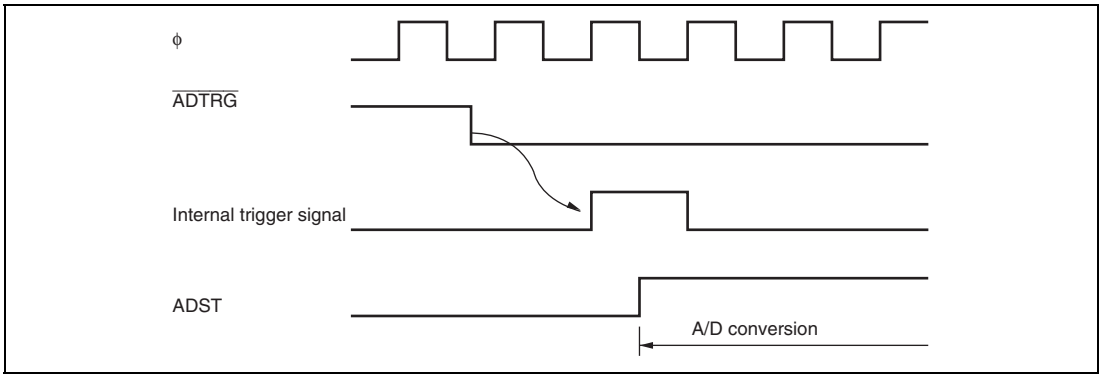


Figure 18.3 External Trigger Input Timing

18.5 A/D Conversion Accuracy Definitions

This LSI's A/D conversion accuracy definitions are given below.

- Resolution

  The number of A/D converter digital output codes
- Quantization error

  The deviation inherent in the A/D converter, given by 1/2 LSB (see figure 18.4).
- 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 0000000000 to 0000000001 (see figure 18.5).
- Full-scale error

  The deviation of the analog input voltage value from the ideal A/D conversion characteristic when the digital output changes from 1111111110 to 1111111111 (see figure 18.5).
- Nonlinearity error

  The deviation from the ideal A/D conversion characteristic as the voltage changes from zero to full scale. This does not include the offset error, full-scale error, or quantization error.
- Absolute accuracy

  The deviation between the digital value and the analog input value. Includes offset error, full-scale error, quantization error, and nonlinearity error.

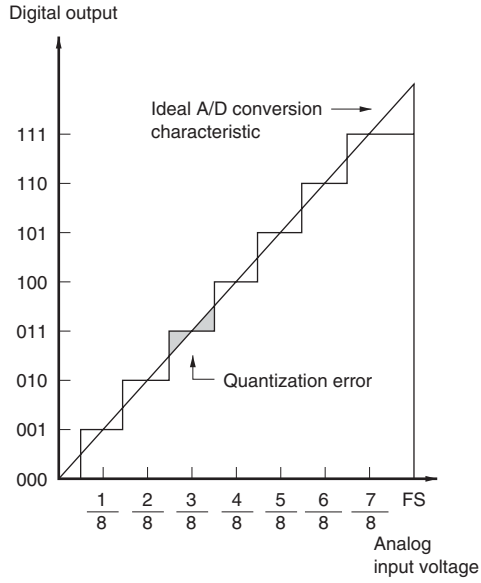


Figure 18.4 A/D Conversion Accuracy Definitions (1)

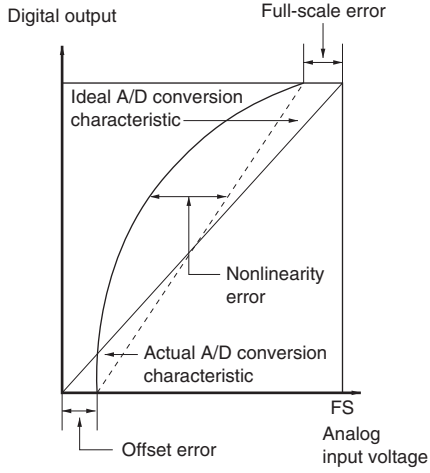


Figure 18.5 A/D Conversion Accuracy Definitions (2)

18.6 Usage Notes

18.6.1 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion accuracy is guaranteed for an input signal for which the signal source impedance is 5 kΩ 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 kΩ, charging may be insufficient and it may not be possible to guarantee A/D conversion accuracy. However, for A/D conversion in single mode with a large capacitance provided externally, the input load will essentially comprise only the internal input resistance of 10 kΩ, 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 mV/µs or greater) (see figure 18.6). When converting a high-speed analog signal or converting in scan mode, a low-impedance buffer should be inserted.

18.6.2 Influences on Absolute Accuracy

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute accuracy. Be sure to make the connection to an electrically stable GND.

Care is also required to ensure that filter circuits do not interfere with digital signals or act as antennas on the mounting board.

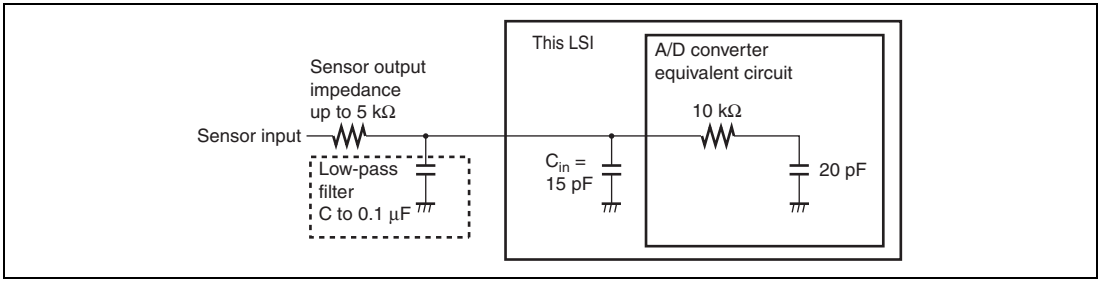


Figure 18.6 Analog Input Circuit Example

Section 19 EEPROM

The H8/3687N has an on-chip 512-byte EEPROM. The block diagram of the EEPROM is shown in figure 19.1.

19.1 Features

- Two writing methods:
  1-byte write
  Page write: Page size 8 bytes
- Three reading methods:
  Current address read
  Random address read
  Sequential read
- Acknowledge polling possible
- Write cycle time:
  10 ms (power supply voltage Vcc = 2.7 V or more)
- Write/Erase endurance:
  10^4 cycles/byte (byte write mode), 10^5 cycles/page (page write mode)
- Data retention:
  10 years after the write cycle of 10^4 cycles (page write mode)
- Interface with the CPU
  I^2C bus interface (complies with the standard of Philips Corporation)
  Device code 1010
  Sleep address code can be changed (initial value: 000)
  The I^2C bus is open to the outside, so the EEPROM can be directly accessed from the outside.

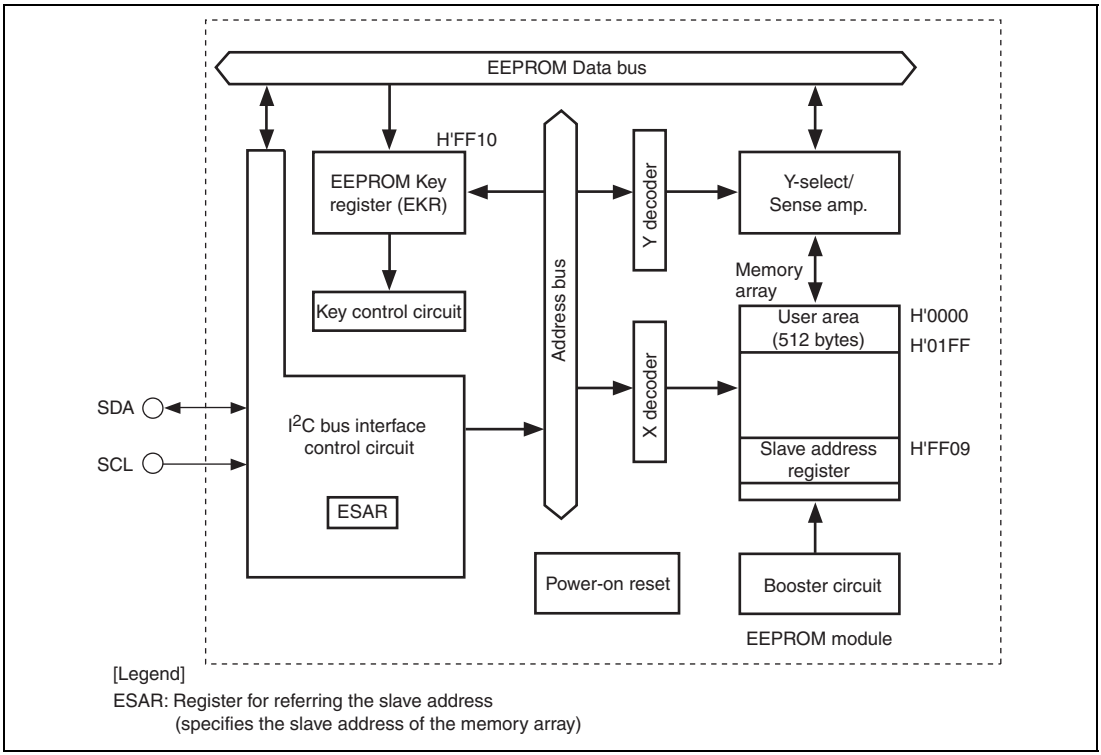


Figure 19.1 Block Diagram of EEPROM

19.2 Input/Output Pins

Pins used in the EEPROM are listed in table 19.1.

Table 19.1 Pin Configuration

Pin name	Symbol	Input/Output	Function
Serial clock pin	SCL	Input	The SCL pin is used to control serial input/output data timing. The data is input at the rising edge of the clock and output at the falling edge of the clock. The SCL pin needs to be pulled up by resistor as that pin is open-drain driven structure of the I^2C pin. Use proper resistor value for your system by considering V_{OL}, I_{OL}, and the C_{IN} pin capacitance in section 23.2.2, DC Characteristics and in section 23.2.3, AC Characteristics. Maximum clock frequency is 400 kHz.
Serial data pin	SDA	Input/Output	The SDA pin is bidirectional for serial data transfer. The SDA pin needs to be pulled up by resistor as that pin is open-drain driven structure. Use proper resistor value for your system by considering V_{OL}, I_{OL}, and the C_{IN} pin capacitance in section 23.2.2, DC Characteristics and in section 23.2.3, AC Characteristics. Except for a start condition and a stop condition which will be discussed later, the high-to-low and low-to-high change of SDA input should be done during SCL low periods.

19.3 Register Description

The EEPROM has a following register.

- EEPROM key register (EKR)

19.3.1 EEPROM Key Register (EKR)

EKR is an 8-bit readable/writable register, which changes the slave address code written in the EEPROM. The slave address code is changed by writing H'5F in EKR and then writing either of H'00 to H'07 as an address code to the H'FF09 address in the EEPROM by the byte write method. EKR is initialized to H'FF.

19.4 Operation

19.4.1 EEPROM Interface

The HD64N3687G has a multi-chip structure with two internal chips of the HD64F3687G (F-ZTAT™ version) and 512-byte EEPROM. The HD6483687G has a multi-chip structure with two internal chips of the HD6433687G (mask-ROM version) and 512-byte EEPROM.

The EEPROM interface is the I^2C bus interface. This I^2C bus is open to the outside, so the communication with the external devices connected to the I^2C bus can be made.

19.4.2 Bus Format and Timing

The I^2C bus format and the I^2C bus timing follow section 17.4.1, I^2C Bus Format. The bus formats specific for the EEPROM are the following two.

1. The EEPROM address is configured of two bytes, the write data is transferred in the order of upper address and lower address from each MSB side.
2. The write data is transmitted from the MSB side.

The bus format and bus timing of the EEPROM are shown in figure 19.2.

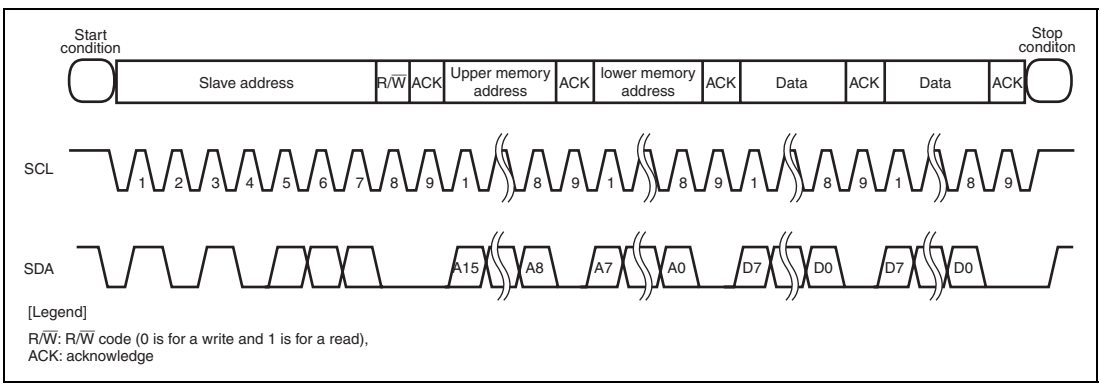


Figure 19.2 EEPROM Bus Format and Bus Timing

19.4.3 Start Condition

A high-to-low transition of the SDA input with the SCL input high is needed to generate the start condition for starting read, write operation.

19.4.4 Stop Condition

A low-to-high transition of the SDA input with the SCL input high is needed to generate the stop condition for stopping read, write operation.

The standby operation starts after a read sequence by a stop condition. In the case of write operation, a stop condition terminates the write data inputs and place the device in an internally-timed write cycle to the memories. After the internally-timed write cycle (t_{WC}) which is specified as t_{WC}, the device enters a standby mode.

19.4.5 Acknowledge

All address data and serial data such as read data and write data are transmitted to and from in 8-bit unit. The acknowledgement is the signal that indicates that this 8-bit data is normally transmitted to and from.

In the write operation, EEPROM sends "0" to acknowledge in the ninth cycle after receiving the data. In the read operation, EEPROM sends a read data following the acknowledgement after receiving the data. After sending read data, the EEPROM enters the bus open state. If the EEPROM receives "0" as an acknowledgement, it sends read data of the next address. If the EEPROM does not receive acknowledgement "0" and receives a following stop condition, it stops the read operation and enters a standby mode. If the EEPROM receives neither acknowledgement "0" nor a stop condition, the EEPROM keeps bus open without sending read data.

19.4.6 Slave Addressing

The EEPROM device receives a 7-bit slave address and a 1-bit R/$\overline{W}$ code following the generation of the start conditions. The EEPROM enables the chip for a read or a write operation with this operation.

The slave address consists of a former 4-bit device code and latter 3-bit slave address as shown in table 19.2. The device code is used to distinguish device type and this LSI uses "1010" fixed code in the same manner as in a general-purpose EEPROM. The slave address code selects one device out of all devices with device code 1010 (8 devices in maximum) which are connected to the I^2C bus. This means that the device is selected if the inputted slave address code received in the order of A2, A1, A0 is equal to the corresponding slave address reference register (ESAR).

The slave address code is stored in the address H'FF09 in the EEPROM. It is transferred to ESAR from the slave address register in the memory array during 10 ms after the reset is released. An access to the EEPROM is not allowed during transfer.

The initial value of the slave address code written in the EEPROM is H'00. It can be written in the range of H'00 to H'07. Be sure to write the data by the byte write method.

The next one bit of the slave address is the R/$\overline{W}$ code. 0 is for a write and 1 is for a read.

The EEPROM turns to a standby state if the device code is not "1010" or slave address code doesn't coincide.

Table 19.2 Slave Addresses

Bit	Bit name	Initial Value	Setting Value	Remarks
7	Device code D3	—	1	
6	Device code D2	—	0	
5	Device code D1	—	1	
4	Device code D0	—	0	
3	Slave address code A2	0	A2	The initial value can be changed
2	Slave address code A1	0	A1	The initial value can be changed
1	Slave address code A0	0	A0	The initial value can be changed

19.4.7 Write Operations

There are two types write operations; byte write operation and page write operation. To initiate the write operation, input 0 to R/$\overline{W}$ code following the slave address.

1. Byte Write

   A write operation requires an 8-bit data of a 7-bit slave address with R/$\overline{W}$ code = "0". Then the EEPROM sends acknowledgement "0" at the ninth bit. This enters the write mode. Then, two bytes of the memory address are received from the MSB side in the order of upper and lower. Upon receipt of one-byte memory address, the EEPROM sends acknowledgement "0" and receives a following a one-byte write data. After receipt of write data, the EEPROM sends acknowledgement "0". If the EEPROM receives a stop condition, the EEPROM enters an internally controlled write cycle and terminates receipt of SCL and SDA inputs until completion of the write cycle. The EEPROM returns to a standby mode after completion of the write cycle.

   The byte write operation is shown in figure 19.3.

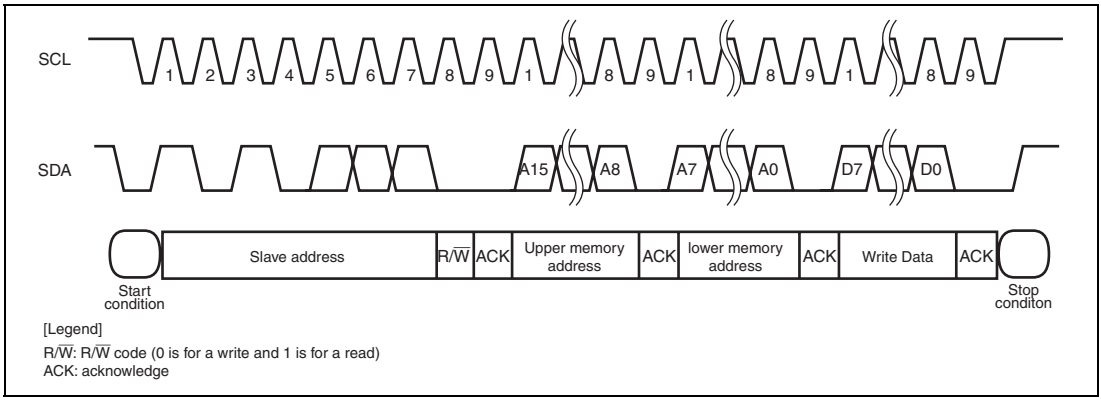


Figure 19.3 Byte Write Operation

2. Page Write

This LSI is capable of the page write operation which allows any number of bytes up to 8 bytes to be written in a single write cycle. The write data is input in the same sequence as the byte write in the order of a start condition, slave address + R/$\overline{W}$ code, memory address (n), and write data (Dn) with every ninth bit acknowledgement "0" output. The EEPROM enters the page write operation if the EEPROM receives more write data (Dn+1) is input instead of receiving a stop condition after receiving the write data (Dn). LSB 3 bits (A2 to A0) in the EEPROM address are automatically incremented to be the (n+1) address upon receiving write data (Dn+1). Thus the write data can be received sequentially.

Addresses in the page are incremented at each receipt of the write data and the write data can be input up to 8 bytes. If the LSB 3 bits (A2 to A0) in the EEPROM address reach the last address of the page, the address will roll over to the first address of the same page. When the address is rolled over, write data is received twice or more to the same address, however, the last received data is valid. At the receipt of the stop condition, write data reception is terminated and the write operation is entered.

The page write operation is shown in figure 19.4.

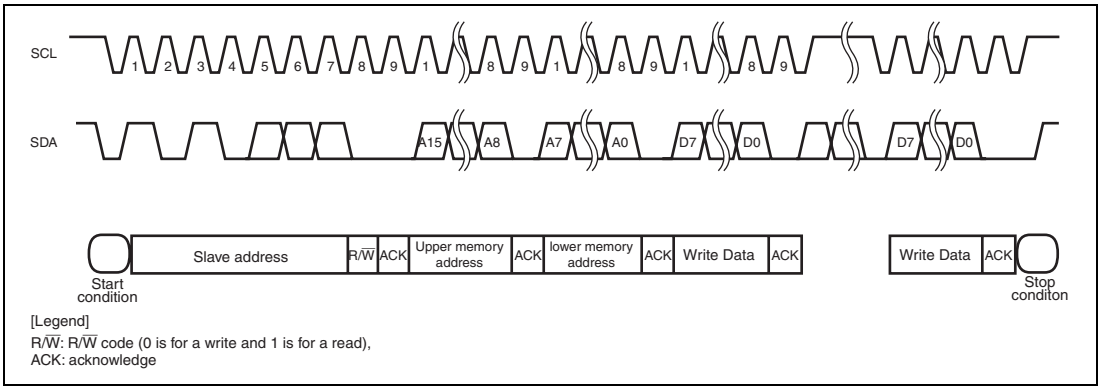


Figure 19.4 Page Write Operation

19.4.8 Acknowledge Polling

Acknowledge polling feature is used to show if the EEPROM is in an internally-timed write cycle or not. This feature is initiated by the input of the 8-bit slave address + R/$\overline{W}$ code following the start condition during an internally-timed write cycle. Acknowledge polling will operate R/W code = "0". The ninth acknowledgement judges if the EEPROM is an internally-timed write cycle or not. Acknowledgement "1" shows the EEPROM is in a internally-timed write cycle and acknowledgement "0" shows the internally-timed write cycle has been completed. The acknowledge polling starts to function after a write data is input, i.e., when the stop condition is input.

19.4.9 Read Operation

There are three read operations; current address read, random address read, and sequential read. Read operations are initiated in the same way as write operations with the exception of R/W = 1.

1. Current Address Read

   The internal address counter maintains the (n+1) address that is made by the last address (n) accessed during the last read or write operation, with incremented by one. Current address read accesses the (n+1) address kept by the internal address counter.

   After receiving in the order of a start condition and the slave address + R/$\overline{W}$ code (R/W = 1), the EEPROM outputs the 1-byte data of the (n+1) address from the most significant bit following acknowledgement "0". If the EEPROM receives in the order of acknowledgement "1" and a following stop condition, the EEPROM stops the read operation and is turned to a standby state.

In case the EEPROM has accessed the last address H'01FF at previous read operation, the current address will roll over and returns to zero address. In case the EEPROM has accessed the last address of the page at previous write operation, the current address will roll over within page addressing and returns to the first address in the same page.

The current address is valid while power is on. The current address after power on will be undefined. After power is turned on, define the address by the random address read operation described below is necessary.

The current address read operation is shown in figure 19.5.

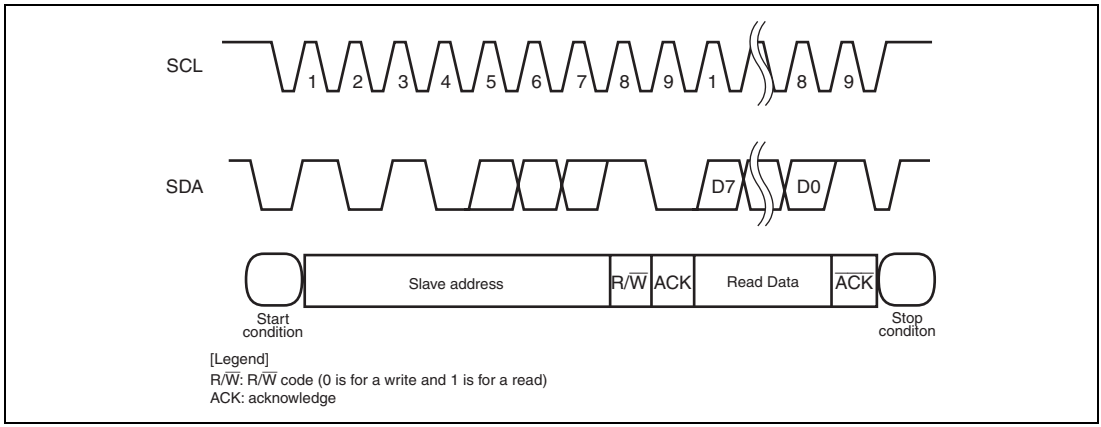


Figure 19.5 Current Address Read Operation

2. Random Address Read

This is a read operation with defined read address. A random address read requires a dummy write to set read address. The EEPROM receives a start condition, slave address + R/$\overline{W}$ code (R/$\overline{W}$ = 0), memory address (upper) and memory address (lower) sequentially. The EEPROM outputs acknowledgement "0" after receiving memory address (lower) then enters a current address read with receiving a start condition again. The EEPROM outputs the read data of the address which was defined in the dummy write operation. After receiving acknowledgement "1" and a following stop condition, the EEPROM stops the random read operation and returns to a standby state.

The random address read operation is shown in figure 19.6.

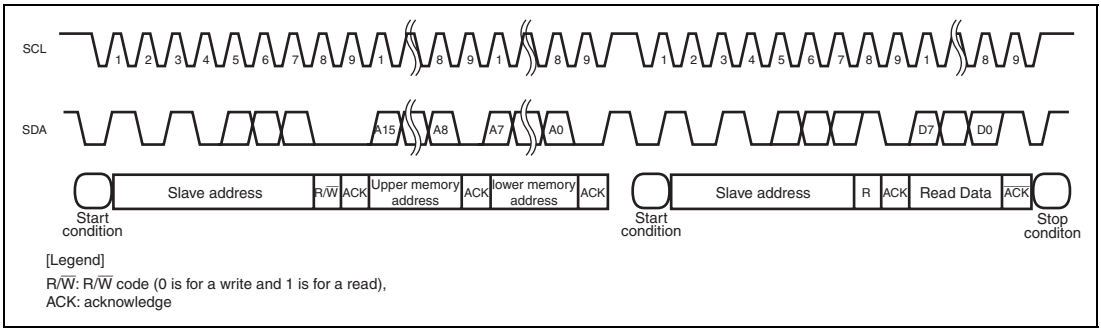


Figure 19.6 Random Address Read Operation

3. Sequential Read

This is a mode to read the data sequentially. Data is sequential read by either a current address read or a random address read. If the EEPROM receives acknowledgement "0" after 1-byte read data is output, the read address is incremented and the next 1-byte read data are coming out. Data is output sequentially by incrementing addresses as long as the EEPROM receives acknowledgement "0" after the data is output. The address will roll over and returns address zero if it reaches the last address H'01FF. The sequential read can be continued after roll over. The sequential read is terminated if the EEPROM receives acknowledgement "1" and a following stop condition as the same manner as in the random address read.

The condition of a sequential read when the current address read is used is shown in figure 19.7.

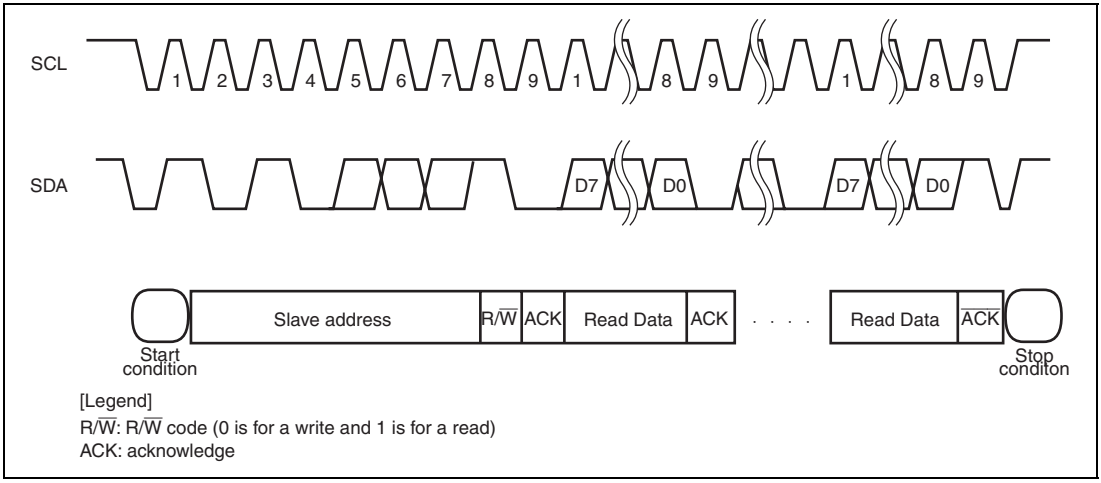


Figure 19.7 Sequential Read Operation (when current address read is used)

19.5 Usage Notes

19.5.1 Data Protection at V_{CC} On/Off

When V_{CC} is turned on or off, the data might be destroyed by malfunction. Be careful of the notices described below to prevent the data to be destroyed.

1. SCL and SDA should be fixed to V_{CC} or V_{SS} during V_{CC} on/off.
2. V_{CC} should be turned off after the EEPROM is placed in a standby state.
3. When V_{CC} is turned on from the intermediate level, malfunction is caused, so V_{CC} should be turned on from the ground level (V_{SS}).
4. V_{CC} turn on speed should be longer than 10 us.

19.5.2 Write/Erase Endurance

The endurance is 10^5 cycles/page (1% cumulative failure rate) in case of page programming and 10^4 cycles/byte in case of byte programming. The data retention time is more than 10 years when a device is page-programmed less than 10^4 cycles.

19.5.3 Noise Suppression Time

This EEPROM has a noise suppression function at SCL and SDA inputs, that cuts noise of width less than 50 ns. Be careful not to allow noise of width more than 50 ns because the noise of with more than 50 ms is recognized as an active pulse.

Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional)

This LSI can include a power-on reset circuit and low-voltage detection circuit as optional circuits.

The low-voltage detection circuit consists of two circuits: LVDI (interrupt by low voltage detect) and LVDR (reset by low voltage detect) circuits.

This circuit is used to prevent abnormal operation (runaway execution) from occurring due to the power supply voltage fall and to recreate the state before the power supply voltage fall when the power supply voltage rises again.

Even if the power supply voltage falls, the unstable state when the power supply voltage falls below the guaranteed operating voltage can be removed by entering standby mode when exceeding the guaranteed operating voltage and during normal operation. Thus, system stability can be improved. If the power supply voltage falls more, the reset state is automatically entered. If the power supply voltage rises again, the reset state is held for a specified period, then active mode is automatically entered.

Figure 20.1 is a block diagram of the power-on reset circuit and the low-voltage detection circuit.

20.1 Features

- Power-on reset circuit

  Uses an external capacitor to generate an internal reset signal when power is first supplied.
- Low-voltage detection circuit

  LVDR: Monitors the power-supply voltage, and generates an internal reset signal when the voltage falls below a specified value.

  LVDI: Monitors the power-supply voltage, and generates an interrupt when the voltage falls below or rises above respective specified values.

  Two pairs of detection levels for reset generation voltage are available: when only the LVDR circuit is used, or when the LVDI and LVDR circuits are both used.

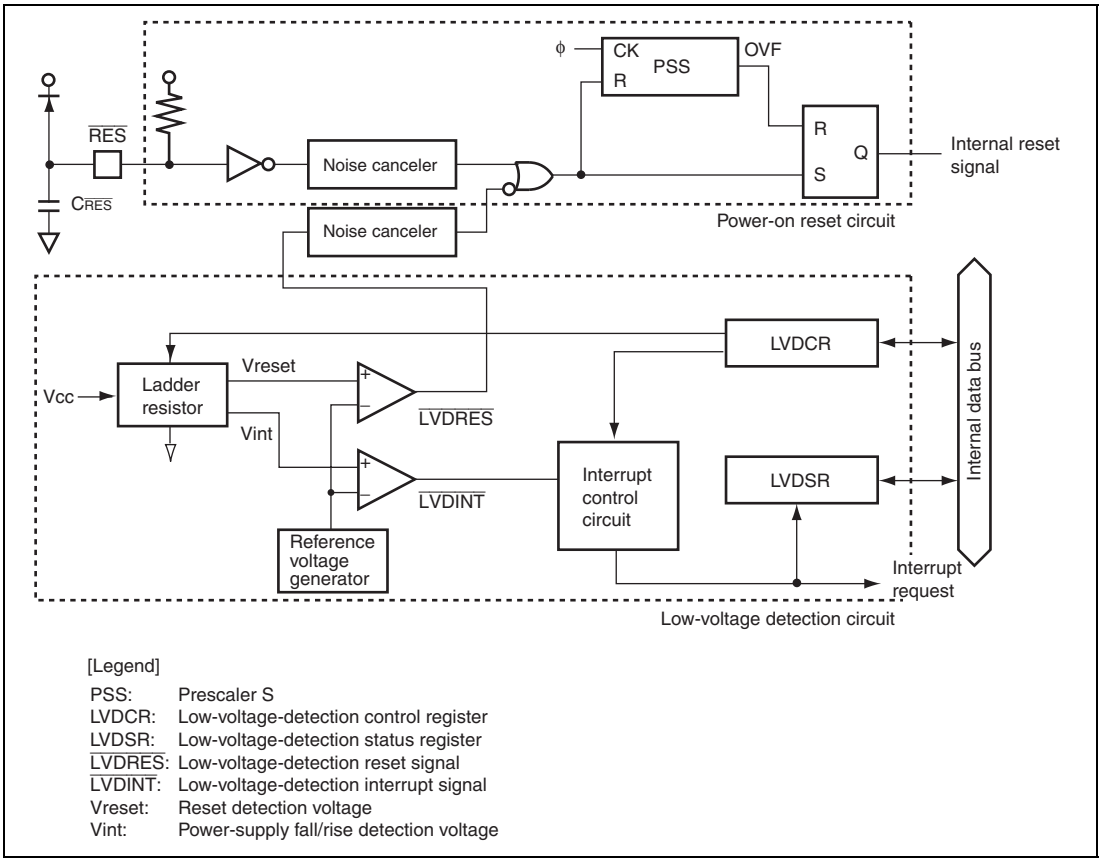


Figure 20.1 Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit

20.2 Register Descriptions

The low-voltage detection circuit has the following registers.

- Low-voltage-detection control register (LVDCR)
- Low-voltage-detection status register (LVDSR)

20.2.1 Low-Voltage-Detection Control Register (LVDCR)

LVDCR is used to enable or disable the low-voltage detection circuit, set the detection levels for the LVDR function, enable or disable the LVDR function, and enable or disable generation of an interrupt when the power-supply voltage rises above or falls below the respective levels.

Table 20.1 shows the relationship between the LVDCR settings and select functions. LVDCR should be set according to table 20.1.

Bit	Bit Name	Initial Value	R/W	Description
7	LVDE	0*	R/W	LVD Enable 0: The low-voltage detection circuit is not used (In standby mode) 1: The low-voltage detection circuit is used
6 to 4	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
3	LVDSEL	0*	R/W	LVDR Detection Level Select 0: Reset detection voltage is 2.3 V (typ.) 1: Reset detection voltage is 3.6 V (typ.) When the falling or rising voltage detection interrupt is used, reset detection voltage of 2.3 V (typ.) should be used. When only a reset detection interrupt is used, reset detection voltage of 3.6 V (typ.) should be used.
2	LVDRE	0*	R/W	LVDR Enable 0: Disables the LVDR function 1: Enables the LVDR function
1	LVDDE	0	R/W	Voltage-Fall-Interrupt Enable 0: Interrupt on the power-supply voltage falling below the selected detection level disabled 1: Interrupt on the power-supply voltage falling below the selected detection level enabled
0	LVDUE	0	R/W	Voltage-Rise-Interrupt Enable 0: Interrupt on the power-supply voltage rising above the selected detection level disabled 1: Interrupt on the power-supply voltage rising above the selected detection level enabled

Note: * Not initialized by LVDR but initialized by a power-on reset or WDT reset.

Table 20.1 LVDCR Settings and Select Functions

LVDCR Settings					Select Functions			
LVDE	LVDSEL	LVDRE	LVDDE	LVDUE	Power-On Reset	LVDR	Low-Voltage-Detection Falling Interrupt	Low-Voltage-Detection Rising Interrupt
0	*	*	*	*	O	—	—	—
1	1	1	0	0	O	O	—	—
1	0	0	1	0	O	—	O	—
1	0	0	1	1	O	—	O	O
1	0	1	1	1	O	O	O	O

Legend: *: means invalid.

20.2.2 Low-Voltage-Detection Status Register (LVDSR)

LVDSR indicates whether the power-supply voltage falls below or rises above the respective specified values.

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 1	—	Reserved These bits are always read as 1, and cannot be modified.
1	LVDDF	0*	R/W	LVD Power-Supply Voltage Fall Flag [Setting condition] When the power-supply voltage falls below Vint (D) (typ. = 3.7 V) [Clearing condition] Writing 0 to this bit after reading it as 1
0	LVDUF	0*	R/W	LVD Power-Supply Voltage Rise Flag [Setting condition] When the power supply voltage falls below Vint (D) while the LVDUE bit in LVDCR is set to 1, then rises above Vint (U) (typ. = 4.0 V) before falling below Vreset1 (typ. = 2.3 V) [Clearing condition] Writing 0 to this bit after reading it as 1

Note: * Initialized by LVDR.

20.3 Operation

20.3.1 Power-On Reset Circuit

Figure 20.2 shows the timing of the operation of the power-on reset circuit. As the power-supply voltage rises, the capacitor which is externally connected to the $\overline{\text{RES}}$ pin is gradually charged via the on-chip pull-up resistor (typ. 150 kΩ). Since the state of the $\overline{\text{RES}}$ pin is transmitted within the chip, the prescaler S and the entire chip are in their reset states. When the level on the $\overline{\text{RES}}$ pin reaches the specified value, the prescaler S is released from its reset state and it starts counting. The OVF signal is generated to release the internal reset signal after the prescaler S has counted 131,072 clock (ϕ) cycles. The noise cancellation circuit of approximately 100 ns is incorporated to prevent the incorrect operation of the chip by noise on the $\overline{\text{RES}}$ pin.

To achieve stable operation of this LSI, the power supply needs to rise to its full level and settles within the specified time. The maximum time required for the power supply to rise and settle after power has been supplied (t_{PWON}) is determined by the oscillation frequency (f_{OSC}) and capacitance which is connected to $\overline{\text{RES}}$ pin ($C_{\overline{\text{RES}}}$). If t_{PWON} means the time required to reach 90 % of power supply voltage, the power supply circuit should be designed to satisfy the following formula.

$$t_{PWON}\ (\text{ms}) \leq 90 \times C_{\overline{\text{RES}}}\ (\mu\text{F}) + 162/f_{OSC}\ (\text{MHz})$$

($t_{PWON} \leq 3000$ ms, $C_{\overline{\text{RES}}} \geq 0.22$ μF, and $f_{OSC} = 10$ in 2-MHz to 10-MHz operation)

Note that the power supply voltage (Vcc) must fall below Vpor = 100 mV and rise after charge on the $\overline{\text{RES}}$ pin is removed. To remove charge on the $\overline{\text{RES}}$ pin, it is recommended that the diode should be placed near Vcc. If the power supply voltage (Vcc) rises from the point above Vpor, a power-on reset may not occur.

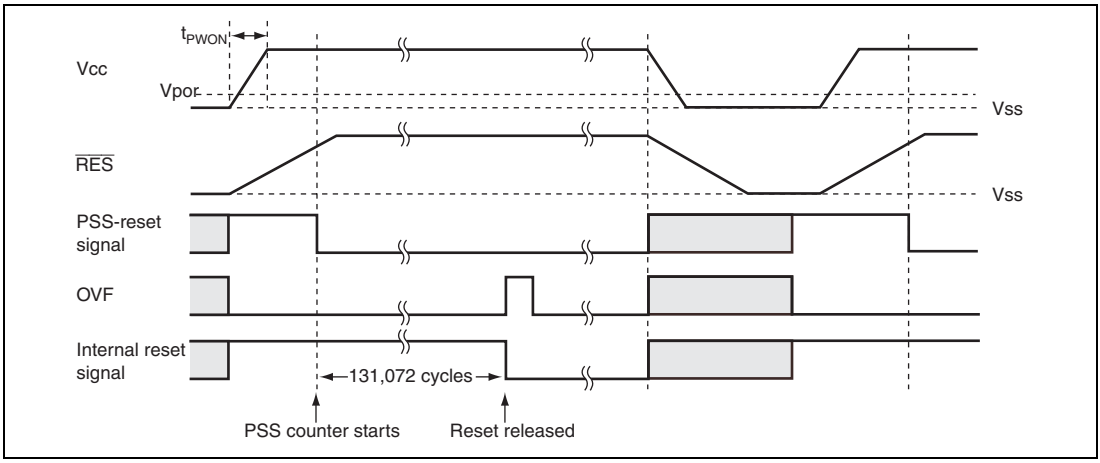


Figure 20.2 Operational Timing of Power-On Reset Circuit

20.3.2 Low-Voltage Detection Circuit

LVDR (Reset by Low Voltage Detect) Circuit:

Figure 20.3 shows the timing of the LVDR function. The LVDR enters the module-standby state after a power-on reset is canceled. To operate the LVDR, set the LVDE bit in LVDCR to 1, wait for 50 µs (t_{LVDON}) until the reference voltage and the low-voltage-detection power supply have stabilized by a software timer, etc., then set the LVDRE bit in LVDCR to 1. After that, the output settings of ports must be made. To cancel the low-voltage detection circuit, first the LVDRE bit should be cleared to 0 and then the LVDE bit should be cleared to 0. The LVDE and LVDRE bits must not be cleared to 0 simultaneously because incorrect operation may occur.

When the power-supply voltage falls below the Vreset voltage (typ. = 2.3 V or 3.6 V), the LVDR clears the $\overline{\text{LVDRES}}$ signal to 0, and resets the prescaler S. The low-voltage detection reset state remains in place until a power-on reset is generated. When the power-supply voltage rises above the Vreset voltage again, the prescaler S starts counting. It counts 131,072 clock (ϕ) cycles, and then releases the internal reset signal. In this case, the LVDE, LVDSEL, and LVDRE bits in LVDCR are not initialized.

Note that if the power supply voltage (Vcc) falls below $V_{LVDRmin}$ = 1.0 V and then rises from that point, the low-voltage detection reset may not occur.

If the power supply voltage (Vcc) falls below Vpor = 100 mV, a power-on reset occurs.

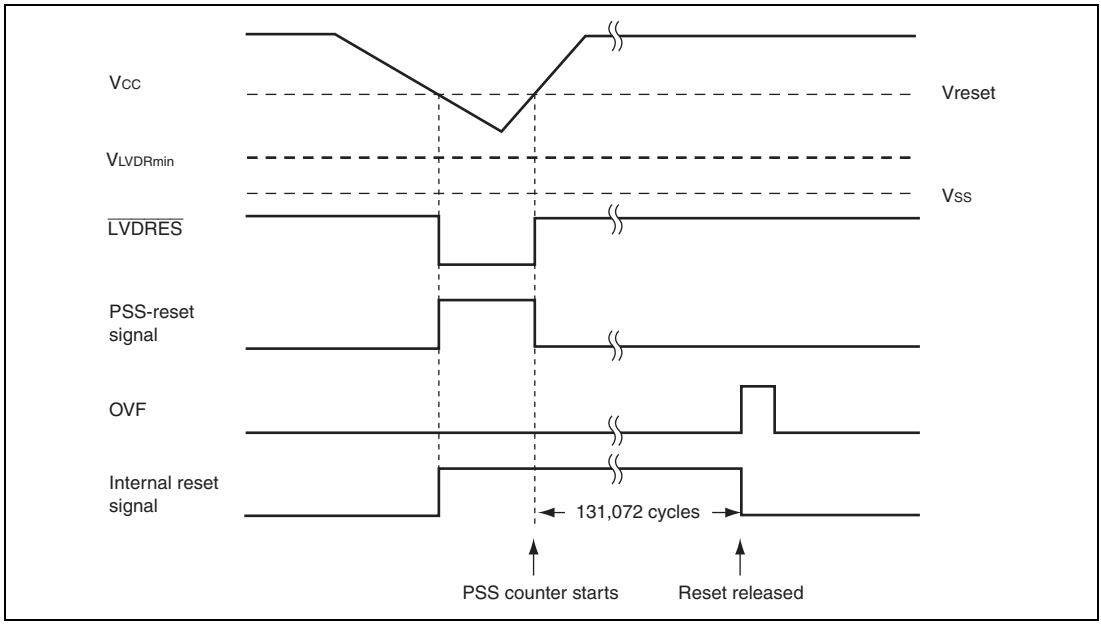


Figure 20.3 Operational Timing of LVDR Circuit

LVDI (Interrupt by Low Voltage Detect) Circuit:

Figure 20.4 shows the timing of LVDI functions. The LVDI enters the module-standby state after a power-on reset is canceled. To operate the LVDI, set the LVDE bit in LVDCR to 1, wait for 50 µs (t_{LVDON}) until the reference voltage and the low-voltage-detection power supply have stabilized by a software timer, etc., then set the LVDDE and LVDUE bits in LVDCR to 1. After that, the output settings of ports must be made. To cancel the low-voltage detection circuit, first the LVDDE and LVDUE bits should all be cleared to 0 and then the LVDE bit should be cleared to 0. The LVDE bit must not be cleared to 0 at the same timing as the LVDDE and LVDUE bits because incorrect operation may occur.

When the power-supply voltage falls below Vint (D) (typ. = 3.7 V) voltage, the LVDI clears the $\overline{\text{LVDINT}}$ signal to 0 and the LVDDF bit in LVDSR is set to 1. If the LVDDE bit is 1 at this time, an IRQ0 interrupt request is simultaneously generated. In this case, the necessary data must be saved in the external EEPROM, etc, and a transition must be made to standby mode or subsleep mode. Until this processing is completed, the power supply voltage must be higher than the lower limit of the guaranteed operating voltage.

When the power-supply voltage does not fall below Vreset1 (typ. = 2.3 V) voltage but rises above Vint (U) (typ. = 4.0 V) voltage, the LVDI sets the $\overline{\text{LVDINT}}$ signal to 1. If the LVDUE bit is 1 at

this time, the LVDUF bit in LVDSR is set to 1 and an IRQ0 interrupt request is simultaneously generated.

If the power supply voltage (Vcc) falls below Vreset1 (typ. = 2.3 V) voltage, the LVDR function is performed.

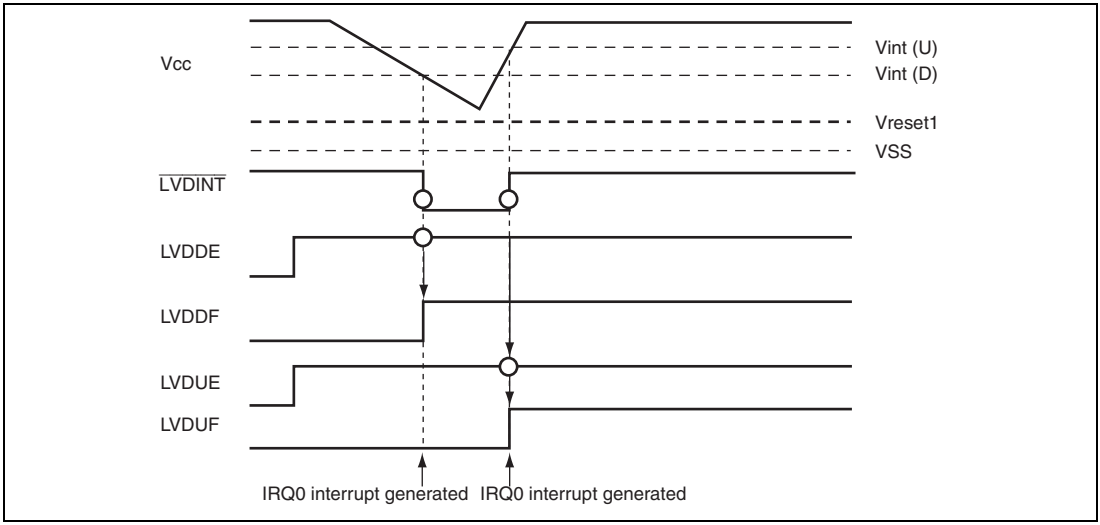


Figure 20.4 Operational Timing of LVDI Circuit

Procedures for Clearing Settings when Using LVDR and LVDI:

To operate or release the low-voltage detection circuit normally, follow the procedure described below. Figure 20.5 shows the timing for the operation and release of the low-voltage detection circuit.

1. To operate the low-voltage detection circuit, set the LVDE bit in LVDCR to 1.
2. Wait for 50 µs (t_{LVDON}) until the reference voltage and the low-voltage-detection power supply have stabilized by a software timer, etc. Then, clear the LVDDF and LVDUF bits in LVDSR to 0 and set the LVDRE, LVDDE, and LVDUE bits in LVDCR to 1, as required.
3. To release the low-voltage detection circuit, start by clearing all of the LVDRE, LVDDE, and LVDUE bits to 0. Then clear the LVDE bit to 0. The LVDE bit must not be cleared to 0 at the same timing as the LVDRE, LVDDE, and LVDUE bits because incorrect operation may occur.

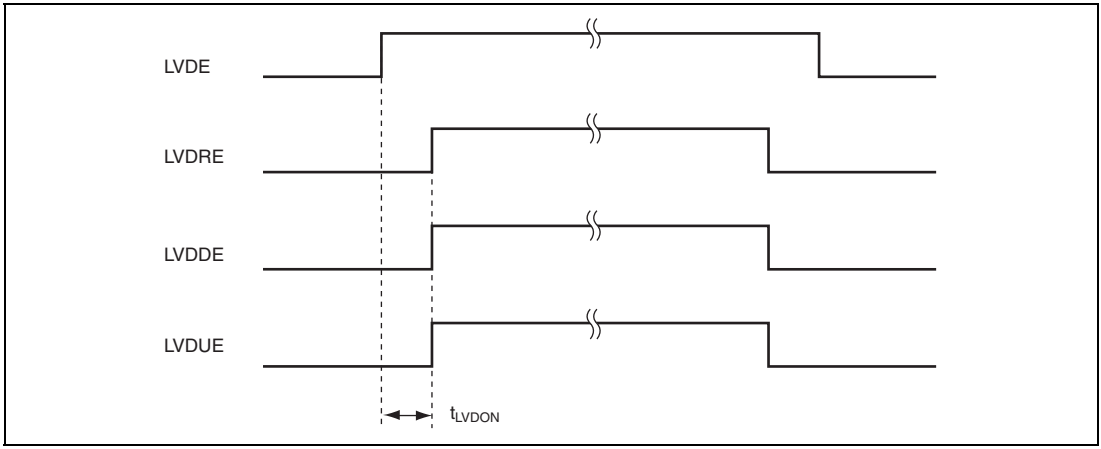


Figure 20.5 Timing for Operation/Release of Low-Voltage Detection Circuit

Section 21 Power Supply Circuit

This LSI incorporates an internal power supply step-down circuit. Use of this circuit enables the internal power supply to be fixed at a constant level of approximately 3.0 V, independently of the voltage of the power supply connected to the external V_{CC} pin. As a result, the current consumed when an external power supply is used at 3.0 V or above can be held down to virtually the same low level as when used at approximately 3.0 V. If the external power supply is 3.0 V or below, the internal voltage will be practically the same as the external voltage. It is, of course, also possible to use the same level of external power supply voltage and internal power supply voltage without using the internal power supply step-down circuit.

21.1 When Using Internal Power Supply Step-Down Circuit

Connect the external power supply to the V_{CC} pin, and connect a capacitance of approximately 0.1 µF between V_{CL} and V_{SS}, as shown in figure 21.1. The internal step-down circuit is made effective simply by adding this external circuit. In the external circuit interface, the external power supply voltage connected to V_{CC} and the GND potential connected to V_{SS} are the reference levels. For example, for port input/output levels, the V_{CC} level is the reference for the high level, and the V_{SS} level is that for the low level. The A/D converter analog power supply is not affected by the internal step-down circuit.

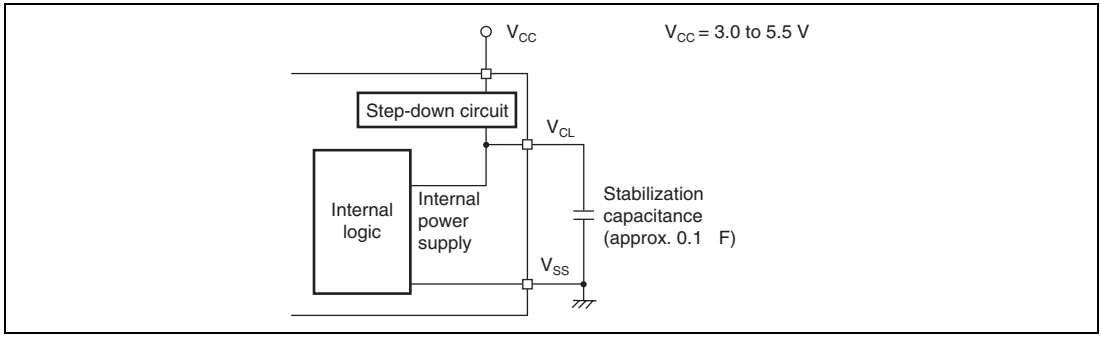


Figure 21.1 Power Supply Connection when Internal Step-Down Circuit is Used

21.2 When Not Using Internal Power Supply Step-Down Circuit

When the internal power supply step-down circuit is not used, connect the external power supply to the V_{CL} pin and V_{CC} pin, as shown in figure 21.2. The external power supply is then input directly to the internal power supply. The permissible range for the power supply voltage is 3.0 V to 3.6 V. Operation cannot be guaranteed if a voltage outside this range (less than 3.0 V or more than 3.6 V) is input.

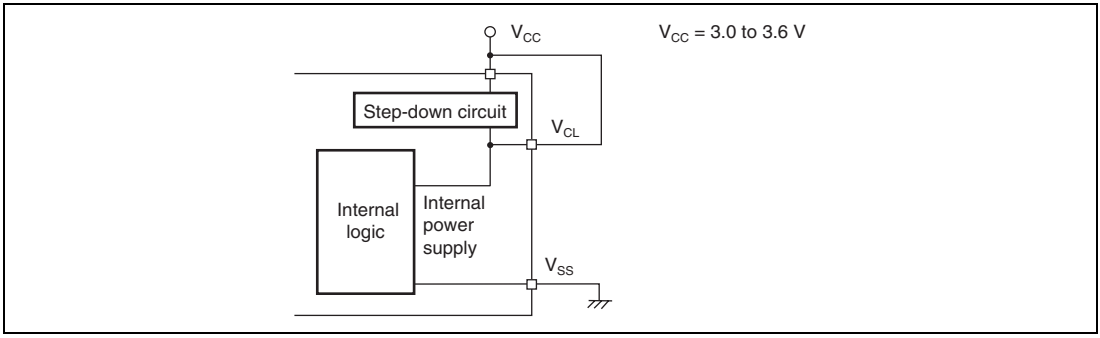


Figure 21.2 Power Supply Connection when Internal Step-Down Circuit is Not Used

Section 22 List of Registers

The register list gives information on the on-chip I/O register addresses, how the register bits are configured, and the register states in each operating mode. The information is given as shown below.

1. Register addresses (address order)

- Registers are listed from the lower allocation addresses.
- The symbol — in the register-name column represents a reserved address or range of reserved addresses.
  Do not attempt to access reserved addresses.
- When the address is 16-bit wide, the address of the upper byte is given in the list.
- Registers are classified by functional modules.
- The data bus width is indicated.
- The number of access states is indicated.

2. Register bits

- Bit configurations of the registers are described in the same order as the register addresses.
- Reserved bits are indicated by — in the bit name column.
- When registers consist of 16 bits, bits are described from the MSB side.

3. Register states in each operating mode

- Register states are described in the same order as the register addresses.
- The register states described here are for the basic operating modes. If there is a specific reset for an on-chip peripheral module, refer to the section on that on-chip peripheral module.

22.1 Register Addresses (Address Order)

The data-bus width column indicates the number of bits. The access-state column shows the number of states of the selected basic clock that is required for access to the register.

Note: Access to undefined or reserved addresses should not take place. Correct operation of the access itself or later operations is not guaranteed when such a register is accessed.

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
—	—	—	H'F000 to H'F6FF	—	—	—
Timer control register_0	TCR_0	8	H'F700	Timer Z	8	2
Timer I/O control register A_0	TIORA_0	8	H'F701	Timer Z	8	2
Timer I/O control register C_0	TIORC_0	8	H'F702	Timer Z	8	2
Timer status register_0	TSR_0	8	H'F703	Timer Z	8	2
Timer interrupt enable register_0	TIER_0	8	H'F704	Timer Z	8	2
PWM mode output level control register_0	POCR_0	8	H'F705	Timer Z	8	2
Timer counter_0	TCNT_0	16	H'F706	Timer Z	16	2
General register A_0	GRA_0	16	H'F708	Timer Z	16	2
General register B_0	GRB_0	16	H'F70A	Timer Z	16	2
General register C_0	GRC_0	16	H'F70C	Timer Z	16	2
General register D_0	GRD_0	16	H'F70E	Timer Z	16	2
Timer control register_1	TCR_1	8	H'F710	Timer Z	8	2
Timer I/O control register A_1	TIORA_1	8	H'F711	Timer Z	8	2
Timer I/O control register C_1	TIORC_1	8	H'F712	Timer Z	8	2
Timer status register_1	TSR_1	8	H'F713	Timer Z	8	2
Timer interrupt enable register_1	TIER_1	8	H'F714	Timer Z	8	2
PWM mode output level control register_1	POCR_1	8	H'F715	Timer Z	8	2
Timer counter_1	TCNT_1	16	H'F716	Timer Z	16	2
General register A_1	GRA_1	16	H'F718	Timer Z	16	2
General register B_1	GRB_1	16	H'F71A	Timer Z	16	2

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
General register C_1	GRC_1	16	H'F71C	Timer Z	16	2
General register D_1	GRD_1	16	H'F71E	Timer Z	16	2
Timer start register	TSTR	8	H'F720	Timer Z	8	2
Timer mode register	TMDR	8	H'F721	Timer Z	8	2
Timer PWM mode register	TPMR	8	H'F722	Timer Z	8	2
Timer Z, for common use	TFCR	8	H'F723	Timer Z	8	2
Timer output master enable register	TOER	8	H'F724	Timer Z	8	2
Timer output control register	TOCR	8	H'F725	Timer Z	8	2
—	—	—	H'F726, H'F727	Timer Z	—	—
Second data register/free running counter data register	RSECDR	8	H'F728	RTC	8	2
Minute data register	RMINDR	8	H'F729	RTC	8	2
Hour data register	RHRDR	8	H'F72A	RTC	8	2
Day-of-week data register	RWKDR	8	H'F72B	RTC	8	2
RTC control register 1	RTCCR1	8	H'F72C	RTC	8	2
RTC control register 2	RTCCR2	8	H'F72D	RTC	8	2
—	—	—	H'F72E	RTC	—	—
Clock source select register	RTCCSR	8	H'F72F	RTC	8	2
Low-voltage-detection control register	LVDCR	8	H'F730	LVDC*[1]	8	2
Low-voltage-detection status register	LVDSR	8	H'F731	LVDC*[1]	8	2
—	—	—	H'F732 to H'F73F	—	—	—
Serial mode register_2	SMR_2	8	H'F740	SCI3_2	8	3
Bit rate register_2	BRR_2	8	H'F741	SCI3_2	8	3
Serial control register 3_2	SCR3_2	8	H'F742	SCI3_2	8	3
Transmit data register_2	TDR_2	8	H'F743	SCI3_2	8	3
Serial status register_2	SSR_2	8	H'F744	SCI3_2	8	3

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
Receive data register_2	RDR_2	8	H'F745	SCI3_2	8	3
—	—	—	H'F746, H'F747	SCI3_2	—	—
I2C bus control register 1	ICCR1	8	H'F748	IIC2	8	2
I2C bus control register 2	ICCR2	8	H'F749	IIC2	8	2
I2C bus mode register	ICMR	8	H'F74A	IIC2	8	2
I2C bus interrupt enable register	ICIER	8	H'F74B	IIC2	8	2
I2C status register	ICSR	8	H'F74C	IIC2	8	2
Slave address register	SAR	8	H'F74D	IIC2	8	2
I2C bus transmit data register	ICDRT	8	H'F74E	IIC2	8	2
I2C bus receive data register	ICDRR	8	H'F74F	IIC2	8	2
—	—	—	H'F750 to H'F75F	—	—	—
Timer mode register B1	TMB1	8	H'F760	Timer B1	8	2
Timer counter B1	TCB1	8	H'F761	Timer B1	8	2
—	—	—	H'F762 to H'FF8F	—	—	—
Flash memory control register 1	FLMCR1	8	H'FF90	ROM	8	2
Flash memory control register 2	FLMCR2	8	H'FF91	ROM	8	2
Flash memory power control register	FLPWCR	8	H'FF92	ROM	8	2
Erase block register 1	EBR1	8	H'FF93	ROM	8	2
—	—	—	H'FF94 to H'FF9A	ROM	—	—
Flash memory enable register	FENR	8	H'FF9B	ROM	8	2
—	—	—	H'FF9C to H'FF9F	ROM	—	—
Timer control register V0	TCRV0	8	H'FFA0	Timer V	8	3
Timer control/status register V	TCSRV	8	H'FFA1	Timer V	8	3

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
Time constant register A	TCORA	8	H'FFA2	Timer V	8	3
Time constant register B	TCORB	8	H'FFA3	Timer V	8	3
Timer counter V	TCNTV	8	H'FFA4	Timer V	8	3
Timer control register V1	TCRV1	8	H'FFA5	Timer V	8	3
—	—	—	H'FFA6, H'FFA7	—	—	—
Serial mode register	SMR	8	H'FFA8	SCI3	8	3
Bit rate register	BRR	8	H'FFA9	SCI3	8	3
Serial control register 3	SCR3	8	H'FFAA	SCI3	8	3
Transmit data register	TDR	8	H'FFAB	SCI3	8	3
Serial status register	SSR	8	H'FFAC	SCI3	8	3
Receive data register	RDR	8	H'FFAD	SCI3	8	3
—	—	—	H'FFAE, H'FFAF	SCI3	—	—
A/D data register	ADDRA	16	H'FFB0	A/D converter	8	3
A/D data register	ADDRB	16	H'FFB2	A/D converter	8	3
A/D data register	ADDRC	16	H'FFB4	A/D converter	8	3
A/D data register	ADDRD	16	H'FFB6	A/D converter	8	3
A/D control/status register	ADCSR	8	H'FFB8	A/D converter	8	3
A/D control register	ADCR	8	H'FFB9	A/D converter	8	3
—	—	—	H'FFBA, H'FFBB	—	—	—
PWM data register L	PWDRL	8	H'FFBC	14-bit PWM	8	2
PWM data register U	PWDRU	8	H'FFBD	14-bit PWM	8	2
PWM control register	PWCR	8	H'FFBE	14-bit PWM	8	2
—	—	—	H'FFBF	14-bit PWM	—	—

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
Timer control/status register WD	TCSRWD	8	H'FFC0	WDT*[2]	8	2
Timer counter WD	TCWD	8	H'FFC1	WDT*[2]	8	2
Timer mode register WD	TMWD	8	H'FFC2	WDT*[2]	8	2
—	—	—	H'FFC3	WDT*[2]	—	—
—	—	—	H'FFC4 to H'FFC7	—	—	—
Address break control register	ABRKCR	8	H'FFC8	Address break	8	2
Address break status register	ABRKSR	8	H'FFC9	Address break	8	2
Break address register H	BARH	8	H'FFCA	Address break	8	2
Break address register L	BARL	8	H'FFCB	Address break	8	2
Break data register H	BDRH	8	H'FFCC	Address break	8	2
Break data register L	BDRL	8	H'FFCD	Address break	8	2
Port pull-up control register 1	PUCR1	8	H'FFD0	I/O port	8	2
Port pull-up control register 5	PUCR5	8	H'FFD1	I/O port	8	2
—	—	—	H'FFD2, H'FFD3	I/O port	—	—
Port data register 1	PDR1	8	H'FFD4	I/O port	8	2
Port data register 2	PDR2	8	H'FFD5	I/O port	8	2
Port data register 3	PDR3	8	H'FFD6	I/O port	8	2
—	—	—	H'FFD7	I/O port	—	—
Port data register 5	PDR5	8	H'FFD8	I/O port	8	2
Port data register 6	PDR6	8	H'FFD9	I/O port	8	2
Port data register 7	PDR7	8	H'FFDA	I/O port	8	2
Port data register 8	PDR8	8	H'FFDB	I/O port	8	2
—	—	—	H'FFDC	I/O port	—	—

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
Port data register B	PDRB	8	H'FFDD	I/O port	8	2
—	—	—	H'FFDE, H'FFDF	I/O port	—	—
Port mode register 1	PMR1	8	H'FFE0	I/O port	8	2
Port mode register 5	PMR5	8	H'FFE1	I/O port	8	2
Port mode register 3	PMR3	8	H'FFE2	I/O port	8	2
—	—	—	H'FFD3	I/O port	—	—
Port control register 1	PCR1	8	H'FFE4	I/O port	8	2
Port control register 2	PCR2	8	H'FFE5	I/O port	8	2
Port control register 3	PCR3	8	H'FFE6	I/O port	8	2
—	—	—	H'FFE7	I/O port	—	—
Port control register 5	PCR5	8	H'FFE8	I/O port	8	2
Port control register 6	PCR6	8	H'FFE9	I/O port	8	2
Port control register 7	PCR7	8	H'FFEA	I/O port	8	2
Port control register 8	PCR8	8	H'FFEB	I/O port	8	2
—	—	—	H'FFEC to H'FFEF	I/O port	—	—
System control register 1	SYSCR1	8	H'FFF0	Low power	8	2
System control register 2	SYSCR2	8	H'FFF1	Low power	8	2
Interrupt edge select register 1	IEGR1	8	H'FFF2	Interrupt	8	2
Interrupt edge select register 2	IEGR2	8	H'FFF3	Interrupt	8	2
Interrupt enable register 1	IENR1	8	H'FFF4	Interrupt	8	2
Interrupt enable register 2	IENR2	8	H'FFF5	Interrupt	8	2
Interrupt flag register 1	IRR1	8	H'FFF6	Interrupt	8	2
Interrupt flag register 2	IRR2	8	H'FFF7	Interrupt	8	2
Wakeup interrupt flag register	IWPR	8	H'FFF8	Interrupt	8	2
Module standby control register 1	MSTCR1	8	H'FFF9	Low power	8	2
Module standby control register 2	MSTCR2	8	H'FFFA	Low power	8	2
—	—	—	H'FFEB	Low power	—	—

Register	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
—	—	—	H'FFFC to H'FFFF	—	—	—

- EEPROM

Register Name	Abbre-viation	Bit No	Address	Module Name	Data Bus Width	Access State
EEPROM slave address register	—	8	H'FF09	EEPROM	—	—
EEPROM key register	EKR	8	H'FF10	EEPROM	8	2

Notes: 1. LVDC: Low-voltage detection circuits (optional)
2. WDT: Watchdog timer

22.2 Register Bits

The addresses and bit names of the registers in the on-chip peripheral modules are listed below. The 16-bit register is indicated in two rows, 8 bits for each row.

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
—	—	—	—	—	—	—	—	—	—
TCR_0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	Timer Z
TIORA_0	—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0	
TIORC_0	—	IOD2	IOD1	IOD0	—	IOC2	IOC1	IOC0	
TSR_0	—	—	—	OVF	IMFD	IMFC	IMFB	IMFA	
TIER_0	—	—	—	OVIE	IMIED	IMIEC	IMIEB	IMIEA	
POCR_0	—	—	—	—	—	POLD	POLC	POLB	
TCNT_0	TCNT0H7	TCNT0H6	TCNT0H5	TCNT0H4	TCNT0H3	TCNT0H2	TCNT0H1	TCNT0H0	
	TCNT0L7	TCNT0L6	TCNT0L5	TCNT0L4	TCNT0L3	TCNT0L2	TCNT0L1	TCNT0L0	
GRA_0	GRA0H7	GRA0H6	GRA0H5	GRA0H4	GRA0H3	GRA0H2	GRA0H1	GRA0H0	
	GRA0L7	GRA0L6	GRA0L5	GRA0L4	GRA0L3	GRA0L2	GRA0L1	GRA0L0	
GRB_0	GRB0H7	GRB0H6	GRB0H5	GRB0H4	GRB0H3	GRB0H2	GRB0H1	GRB0H0	
	GRB0L7	GRB0L6	GRB0L5	GRB0L4	GRB0L3	GRB0L2	GRB0L1	GRB0L0	
GRC_0	GRC0H7	GRC0H6	GRC0H5	GRC0H4	GRC0H3	GRC0H2	GRC0H1	GRC0H0	
	GRC0L7	GRC0L6	GRC0L5	GRC0L4	GRC0L3	GRC0L2	GRC0L1	GRC0L0	
GRD_0	GRD0H7	GRD0H6	GRD0H5	GRD0H4	GRD0H3	GRD0H2	GRD0H1	GRD0H0	
	GRD0L7	GRD0L6	GRD0L5	GRD0L4	GRD0L3	GRD0L2	GRD0L1	GRD0L0	
TCR_1	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TPSC2	TPSC1	TPSC0	
TIORA_1	—	IOB2	IOB1	IOB0	—	IOA2	IOA1	IOA0	
TIORC_1	—	IOD2	IOD1	IOD0	—	IOC2	IOC1	IOC0	
TSR_1	—	—	UDF	OVF	IMFD	IMFC	IMFB	IMFA	
TIER_1	—	—	—	OVIE	IMIED	IMIEC	IMIEB	IMIEA	
POCR_1	—	—	—	—	—	POLD	POLC	POLB	
TCNT_1	TCNT1H7	TCNT1H6	TCNT1H5	TCNT1H4	TCNT1H3	TCNT1H2	TCNT1H1	TCNT1H0	
	TCNT1L7	TCNT1L6	TCNT1L5	TCNT1L4	TCNT1L3	TCNT1L2	TCNT1L1	TCNT1L0	
GRA_1	GRA1H7	GRA1H6	GRA1H5	GRA1H4	GRA1H3	GRA1H2	GRA1H1	GRA1H0	
	GRA1L7	GRA1L6	GRA1L5	GRA1L4	GRA1L3	GRA1L2	GRA1L1	GRA1L0	

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
GRB_1	GRB1H7	GRB1H6	GRB1H5	GRB1H4	GRB1H3	GRB1H2	GRB1H1	GRB1H0	Timer Z
	GRB1L7	GRB1L6	GRB1L5	GRB1L4	GRB1L3	GRB1L2	GRB1L1	GRB1L0	
GRC_1	GRC1H7	GRC1H6	GRC1H5	GRC1H4	GRC1H3	GRC1H2	GRC1H1	GRC1H0	
	GRC1L7	GRC1L6	GRC1L5	GRC1L4	GRC1L3	GRC1L2	GRC1L1	GRC1L0	
GRD_1	GRD1H7	GRD1H6	GRD1H5	GRD1H4	GRD1H3	GRD1H2	GRD1H1	GRD1H0	
	GRD1L7	GRD1L6	GRD1L5	GRD1L4	GRD1L3	GRD1L2	GRD1L1	GRD1L0	
TSTR	—	—	—	—	—	—	STR1	STR0	
TMDR	BFD1	BFC1	BFD0	BFC0	—	—	—	SYNC	
TPMR	—	PWMD1	PWMC1	PWMB1	—	PWMD0	PWMC0	PWMB0	
TFCR	—	STCLK	ADEG	ADTRG	OLS1	OLS0	CMD1	CMD0	
TOER	ED1	EC1	EB1	EA1	ED0	EC0	EB0	EA0	
TOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0	
RSECDR	BSY	SC12	SC11	SC10	SC03	SC02	SC01	SC00	RTC
RMINDR	BSY	MN12	MN11	MN10	MN03	MN02	MN01	MN00	
RHRDR	BSY	—	HR11	HR10	HR03	HR02	HR01	HR00	
RWKDR	BSY	—	—	—	—	WK2	WK1	WK0	
RTCCR1	RUN	12/24	PM	RST	—	—	—	—	
RTCCR2	—	—	FOIE	WKIE	DYIE	HRIE	MNIE	SEIE	
RTCCSR	—	RCS6	RCS5	—	RCS3	RCS2	RCS1	RCS0	
LVDCR	LVDE	—	—	—	LVDSEL	LVDRE	LVDDE	LVDUE	LVDC
LVDSR	—	—	—	—	—	—	LVDDF	LVDUF	(optional)*[1]
—	—	—	—	—	—	—	—	—	—
SMR_2	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0	SCI3_2
BRR_2	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0	
SCR3_2	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR_2	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	
SSR_2	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT	
RDR_2	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0	
ICCR1	ICE	RCVD	MST	TRS	CKS3	CKS2	CKS1	CKS0	IIC2
ICCR2	BBSY	SCP	SDAO	SDAOP	SCLO	—	IICRST	—	
ICMR	MLS	WAIT	—	—	BCWP	BC2	BC1	BC0	

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
ICIER	TIE	TEIE	RIE	NAKIE	STIE	ACKE	ACKBR	ACKBT	IIC2
ICSR	TDRE	TEND	RDRF	NACKF	STOP	AL/OVE	AAS	ADZ	
SAR	SVA6	SVA5	SVA4	SVA3	SVA2	SVA1	SVA0	FS	
ICDRT	ICDRT7	ICDRT6	ICDRT5	ICDRT4	ICDRT3	ICDRT2	ICDRT1	ICDRT0	
ICDRR	ICDRR7	ICDRR6	ICDRR5	ICDRR4	ICDRR3	ICDRR2	ICDRR1	ICDRR0	
—	—	—	—	—	—	—	—	—	—
TMB1	TMB17	—	—	—	—	TMB12	TMB11	TMB10	Timer B1
TCB1	TCB17	TCB16	TCB15	TCB14	TCB13	TCB12	TCB11	TCB10	
—	—	—	—	—	—	—	—	—	—
FLMCR1	—	SWE	ESU	PSU	EV	PV	E	P	ROM
FLMCR2	FLER	—	—	—	—	—	—	—	
FLPWCR	PDWND	—	—	—	—	—	—	—	
EBR1	—	EB6	EB5	EB4	EB3	EB2	EB1	EB0	
FENR	FLSHE	—	—	—	—	—	—	—	
TCRV0	CMIEB	CMIEA	OVIE	CCLR1	CCLR0	CKS2	CKS1	CKS0	Timer V
TCSRV	CMFB	CMFA	OVF	—	OS3	OS2	OS1	OS0	
TCORA	TCORA7	TCORA6	TCORA5	TCORA4	TCORA3	TCORA2	TCORA1	TCORA0	
TCORB	TCORB7	TCORB6	TCORB5	TCORB4	TCORB3	TCORB2	TCORB1	TCORB0	
TCNTV	TCNTV7	TCNTV6	TCNTV5	TCNTV4	TCNTV3	TCNTV2	TCNTV1	TCNTV0	
TCRV1	—	—	—	TVEG1	TVEG0	TRGE	—	ICKS0	
—	—	—	—	—	—	—	—	—	—
SMR	COM	CHR	PE	PM	STOP	MP	CKS1	CKS0	SCI3
BRR	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1	BRR0	
SCR3	TIE	RIE	TE	RE	MPIE	TEIE	CKE1	CKE0	
TDR	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1	TDR0	
SSR	TDRE	RDRF	OER	FER	PER	TEND	MPBR	MPBT	
RDR	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1	RDR0	
ADDRA	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D converter
	AD1	AD0	—	—	—	—	—	—	

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
ADDRB	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	A/D converter
	AD1	AD0	—	—	—	—	—	—	
ADDRC	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADDRD	AD9	AD8	AD7	AD6	AD5	AD4	AD3	AD2	
	AD1	AD0	—	—	—	—	—	—	
ADCSR	ADF	ADIE	ADST	SCAN	CKS	CH2	CH1	CH0	
ADCR	TRGE	—	—	—	—	—	—	—	
—	—	—	—	—	—	—	—	—	—
PWDRL	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1	PWDRL0	14-bit PWM
PWDRU	—	—	PWDRU5	PWDRU4	PWDRU3	PWDRU2	PWDRU1	PWDRU0	
PWCR	—	—	—	—	—	—	—	PWCR0	
TCSRWD	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	B0WI	WRST	WDT*[2]
TCWD	TCWD7	TCWD6	TCWD5	TCWD4	TCWD3	TCWD2	TCWD1	TCWD0	
TMWD	—	—	—	—	CKS3	CKS2	CKS1	CKS0	
—	—	—	—	—	—	—	—	—	—
ABRKCR	RTINTE	CSEL1	CSEL0	ACMP2	ACMP1	ACMP0	DCMP1	DCMP0	Address break
ABRKSR	ABIF	ABIE	—	—	—	—	—	—	
BARH	BARH7	BARH6	BARH5	BARH4	BARH3	BARH2	BARH1	BARH0	
BARL	BARL7	BARL6	BARL5	BARL4	BARL3	BARL2	BARL1	BARL0	
BDRH	BDRH7	BDRH6	BDRH5	BDRH4	BDRH3	BDRH2	BDRH1	BDRH0	
BDRL	BDRL7	BDRL6	BDRL5	BDRL4	BDRL3	BDRL2	BDRL1	BDRL0	
—	—	—	—	—	—	—	—	—	—
PUCR1	PUCR17	PUCR16	PUCR15	PUCR14	—	PUCR12	PUCR11	PUCR10	I/O port
PUCR5	—	—	PUCR55	PUCR54	PUCR53	PUCR52	PUCR51	PUCR50	
PDR1	P17	P16	P15	P14	—	P12	P11	P10	
PDR2	—	—	—	P24	P23	P22	P21	P20	
PDR3	P37	P36	P35	P34	P33	P32	P31	P30	
PDR5	P57*[3]	P56*[3]	P55	P54	P53	P52	P51	P50	
PDR6	P67	P66	P65	P64	P63	P62	P61	P60	
PDR7	—	P76	P75	P74	—	P72	P71	P70	

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
PDR8	P87	P86	P85	—	—	—	—	—	I/O port
PDRB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0	
PMR1	IRQ3	IRQ2	IRQ1	IRQ0	TXD2	PWM	TXD	TMOW	
PMR5	POF57	POF56	WKP5	WKP4	WKP3	WKP2	WKP1	WKP0	
PMR3	—	—	—	POF24	POF23	—	—	—	
PCR1	PCR17	PCR16	PCR15	PCR14	—	PCR12	PCR11	PCR10	
PCR2	—	—	—	PCR24	PCR23	PCR22	PCR21	PCR20	
PCR3	PCR37	PCR36	PCR35	PCR34	PCR33	PCR32	PCR31	PCR30	
PCR5	PCR57*[3]	PCR56*[3]	PCR55	PCR54	PCR53	PCR52	PCR51	PCR50	
PCR6	PCR67	PCR66	PCR65	PCR64	PCR63	PCR62	PCR61	PCR60	
PCR7	—	PCR76	PCR75	PCR74	—	PCR72	PCR71	PCR70	
PCR8	PCR87	PCR86	PCR85	—	—	—	—	—	
SYSCR1	SSBY	STS2	STS1	STS0	NESEL	—	—	—	Low power
SYSCR2	SMSEL	LSON	DTON	MA2	MA1	MA0	SA1	SA0	
IEGR1	NMIEG	—	—	—	IEG3	IEG2	IEG1	IEG0	Interrupt
IEGR2	—	—	WPEG5	WPEG4	WPEG3	WPEG2	WPEG1	WPEG0	
IENR1	IENDT	IENTA	IENWP	—	IEN3	IEN2	IEN1	IEN0	
IENR2	—	—	IENTB1	—	—	—	—	—	
IRR1	IRRDT	IRRTA	—	—	IRRI3	IRRI2	IRRI1	IRRI0	
IRR2	—	—	IRRTB1	—	—	—	—	—	
IWPR	—	—	IWPF5	IWPF4	IWPF3	IWPF2	IWPF1	IWPF0	Interrupt
MSTCR1	—	MSTIIC	MSTS3	MSTAD	MSTWD	—	MSTTV	MSTTA	Low power
MSTCR2	MSTS3_2	—	—	MSTTB1	—	—	MSTTZ	MSTPWM	
—	—	—	—	—	—	—	—	—	—

- EEPROM

Register Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Module Name
EKR									EEPROM

Notes: 1. LVDC: Low-voltage detection circuits (optional)
2. WDT: Watchdog timer
3. These bits are reserved in the EEPROM stacked F-ZTAT™ and mask-ROM versions.

22.3 Registers States in Each Operating Mode

Register Name	Reset	Active	Sleep	Subactive	Subsleep	Standby	Module
TCR_0	Initialized	—	—	—	—	—	Timer Z
TIORA_0	Initialized	—	—	—	—	—	
TIORC_0	Initialized	—	—	—	—	—	
TSR_0	Initialized	—	—	—	—	—	
TIER_0	Initialized	—	—	—	—	—	
POCR_0	Initialized	—	—	—	—	—	
TCNT_0	Initialized	—	—	—	—	—	
GRA_0	Initialized	—	—	—	—	—	
GRB_0	Initialized	—	—	—	—	—	
GRC_0	Initialized	—	—	—	—	—	
GRD_0	Initialized	—	—	—	—	—	
TCR_1	Initialized	—	—	—	—	—	
TIORA_1	Initialized	—	—	—	—	—	
TIORC_1	Initialized	—	—	—	—	—	
TSR_1	Initialized	—	—	—	—	—	
TIER_1	Initialized	—	—	—	—	—	
POCR_1	Initialized	—	—	—	—	—	
TCNT_1	Initialized	—	—	—	—	—	
GRA_1	Initialized	—	—	—	—	—	
GRB_1	Initialized	—	—	—	—	—	
GRC_1	Initialized	—	—	—	—	—	
GRD_1	Initialized	—	—	—	—	—	
TSTR	Initialized	—	—	—	—	—	
TMDR	Initialized	—	—	—	—	—	
TPMR	Initialized	—	—	—	—	—	
TFCR	Initialized	—	—	—	—	—	
TOER	Initialized	—	—	—	—	—	
TOCR	Initialized	—	—	—	—	—	
RSECDR	—	—	—	—	—	—	RTC

Register Name	Reset	Active	Sleep	Subactive	Subsleep	Standby	Module
RMINDR	—	—	—	—	—	—	RTC
RHRDR	—	—	—	—	—	—	
RWKDR	—	—	—	—	—	—	
RTCCR1	—	—	—	—	—	—	
RTCCR2	—	—	—	—	—	—	
RTCCSR	Initialized	—	—	—	—	—	
LVDCR	Initialized	—	—	—	—	—	LVDC
LVDSR	Initialized	—	—	—	—	—	(optional)*[1]
SMR_2	Initialized	—	—	Initialized	Initialized	Initialized	SCI3_2
BRR_2	Initialized	—	—	Initialized	Initialized	Initialized	
SCR3_2	Initialized	—	—	Initialized	Initialized	Initialized	
TDR_2	Initialized	—	—	Initialized	Initialized	Initialized	
SSR_2	Initialized	—	—	Initialized	Initialized	Initialized	
RDR_2	Initialized	—	—	Initialized	Initialized	Initialized	
ICCR1	Initialized	—	—	—	—	—	IIC2
ICCR2	Initialized	—	—	—	—	—	
ICMR	Initialized	—	—	—	—	—	
ICIER	Initialized	—	—	—	—	—	
ICSR	Initialized	—	—	—	—	—	
SAR	Initialized	—	—	—	—	—	
ICDRT	Initialized	—	—	—	—	—	
ICDRR	Initialized	—	—	—	—	—	
TMB1	Initialized	—	—	—	—	—	Timer B1
TCB1	Initialized	—	—	—	—	—	
FLMCR1	Initialized	—	—	Initialized	Initialized	Initialized	ROM
FLMCR2	Initialized	—	—	—	—	—	
FLPWCR	Initialized	—	—	—	—	—	
EBR1	Initialized	—	—	Initialized	Initialized	Initialized	
FENR	Initialized	—	—	—	—	—	
TCRV0	Initialized	—	—	Initialized	Initialized	Initialized	Timer V
TCSRV	Initialized	—	—	Initialized	Initialized	Initialized	

Register Name	Reset	Active	Sleep	Subactive	Subsleep	Standby	Module
TCORA	Initialized	—	—	Initialized	Initialized	Initialized	Timer V
TCORB	Initialized	—	—	Initialized	Initialized	Initialized	
TCNTV	Initialized	—	—	Initialized	Initialized	Initialized	
TCRV1	Initialized	—	—	Initialized	Initialized	Initialized	
SMR	Initialized	—	—	Initialized	Initialized	Initialized	SCI3
BRR	Initialized	—	—	Initialized	Initialized	Initialized	
SCR3	Initialized	—	—	Initialized	Initialized	Initialized	
TDR	Initialized	—	—	Initialized	Initialized	Initialized	
SSR	Initialized	—	—	Initialized	Initialized	Initialized	
RDR	Initialized	—	—	Initialized	Initialized	Initialized	
ADDRA	Initialized	—	—	Initialized	Initialized	Initialized	A/D converter
ADDRB	Initialized	—	—	Initialized	Initialized	Initialized	
ADDRC	Initialized	—	—	Initialized	Initialized	Initialized	
ADDRD	Initialized	—	—	Initialized	Initialized	Initialized	
ADCSR	Initialized	—	—	Initialized	Initialized	Initialized	
ADCR	Initialized	—	—	Initialized	Initialized	Initialized	
PWDRL	Initialized	—	—	—	—	—	14bit PWM
PWDRU	Initialized	—	—	—	—	—	
PWCR	Initialized	—	—	—	—	—	
TCSRWD	Initialized	—	—	—	—	—	WDT*[2]
TCWD	Initialized	—	—	—	—	—	
TMWD	Initialized	—	—	—	—	—	
ABRKCR	Initialized	—	—	—	—	—	Address break
ABRKSR	Initialized	—	—	—	—	—	
BARH	Initialized	—	—	—	—	—	
BARL	Initialized	—	—	—	—	—	
BDRH	Initialized	—	—	—	—	—	
BDRL	Initialized	—	—	—	—	—	
PUCR1	Initialized	—	—	—	—	—	I/O port
PUCR5	Initialized	—	—	—	—	—	
PDR1	Initialized	—	—	—	—	—	

Register Name	Reset	Active	Sleep	Subactive	Subsleep	Standby	Module
PDR2	Initialized	—	—	—	—	—	I/O port
PDR3	Initialized	—	—	—	—	—	
PDR5	Initialized	—	—	—	—	—	
PDR6	Initialized	—	—	—	—	—	
PDR7	Initialized	—	—	—	—	—	
PDR8	Initialized	—	—	—	—	—	
PDRB	Initialized	—	—	—	—	—	
PMR1	Initialized	—	—	—	—	—	
PMR5	Initialized	—	—	—	—	—	
PMR3	Initialized	—	—	—	—	—	
PCR1	Initialized	—	—	—	—	—	
PCR2	Initialized	—	—	—	—	—	
PCR3	Initialized	—	—	—	—	—	
PCR5	Initialized	—	—	—	—	—	
PCR6	Initialized	—	—	—	—	—	
PCR7	Initialized	—	—	—	—	—	
PCR8	Initialized	—	—	—	—	—	
SYSCR1	Initialized	—	—	—	—	—	Low power
SYSCR2	Initialized	—	—	—	—	—	
IEGR1	Initialized	—	—	—	—	—	Interrupt
IEGR2	Initialized	—	—	—	—	—	
IENR1	Initialized	—	—	—	—	—	
IENR2	Initialized	—	—	—	—	—	
IRR1	Initialized	—	—	—	—	—	
IRR2	Initialized	—	—	—	—	—	
IWPR	Initialized	—	—	—	—	—	
MSTCR1	Initialized	—	—	—	—	—	Low power
MSTCR2	Initialized	—	—	—	—	—	

- EEPROM

Register Name	Reset	Active	Sleep	Subactive	Subsleep	Standby	Module
EKR	—	—	—	—	—	—	EEPROM

Notes: — is not initialized

1. LVDC: Low-voltage detection circuits (optional)
2. WDT: Watchdog timer

Section 23 Electrical Characteristics

23.1 Absolute Maximum Ratings

Table 23.1 Absolute Maximum Ratings

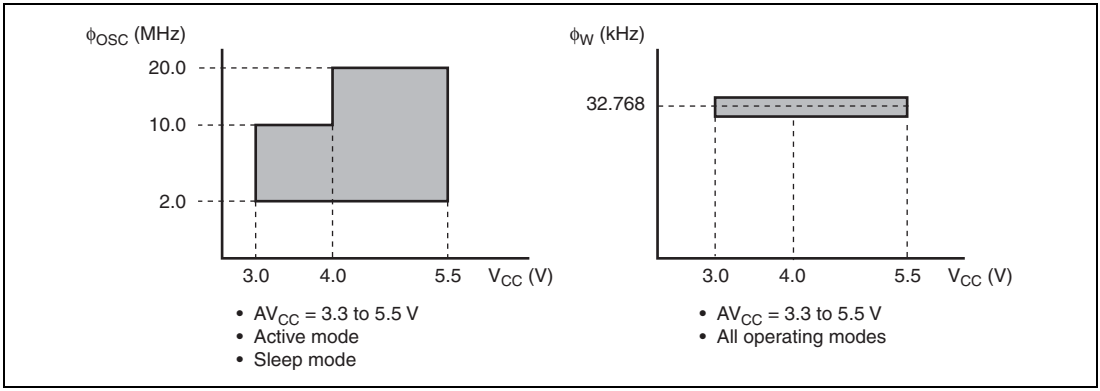
Item		Symbol	Value	Unit	Note
Power supply voltage		V_{CC}	–0.3 to +7.0	V	*
Analog power supply voltage		AV_{CC}	–0.3 to +7.0	V	
Input voltage	Ports other than ports B and X1	V_{IN}	–0.3 to V_{CC} +0.3	V	
	Port B		–0.3 to AV_{CC} +0.3	V	
	X1		–0.3 to 4.3	V	
Operating temperature		T_{opr}	–20 to +75	°C	
Storage temperature		T_{stg}	–55 to +125	°C	

Note: * Permanent damage may result if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

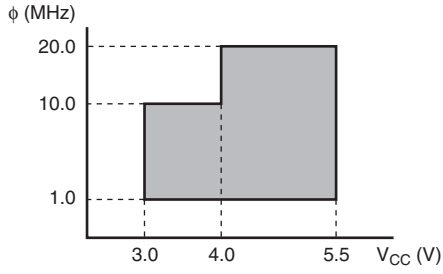
23.2 Electrical Characteristics (F-ZTAT™ Version, EEPROM Stacked F-ZTAT™ Version)

23.2.1 Power Supply Voltage and Operating Ranges

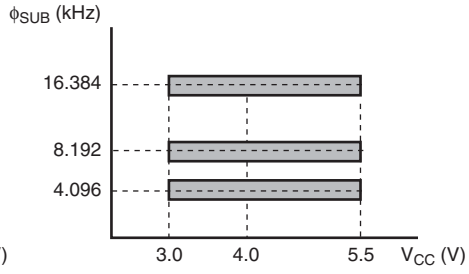
Power Supply Voltage and Oscillation Frequency Range



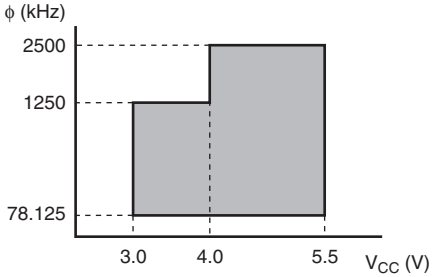
Power Supply Voltage and Operating Frequency Range



- AV_{CC} = 3.3 to 5.5 V
- Active mode
- Sleep mode
  (When MA2 in SYSCR2 = 0)

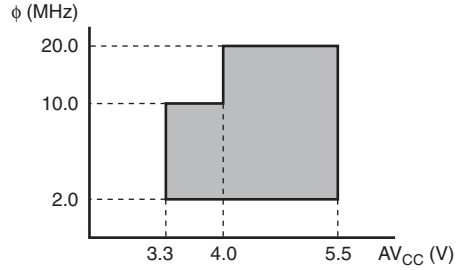


- AV_{CC} = 3.3 to 5.5 V
- Subactive mode
- Subsleep mode

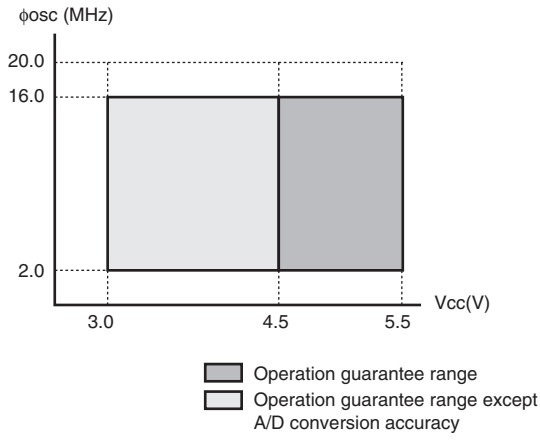


- AV_{CC} = 3.3 to 5.5 V
- Active mode
- Sleep mode
  (When MA2 in SYSCR2 = 1)

Analog Power Supply Voltage and A/D Converter Accuracy Guarantee Range



Range of Power Supply Voltage and Oscillation Frequency when Low-Voltage Detection Circuit is Used



23.2.2 DC Characteristics

Table 23.2 DC Characteristics (1)

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TMIB1, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1,SCK3, SCK3_2, TRGV	V_{CC} = 4.0 to 5.5 V	$V_{CC}\times 0.8$	—	V_{CC} + 0.3	V	
				$V_{CC}\times 0.9$	—	V_{CC} + 0.3		
		RXD, RXD_2, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72 P74 to P76, P85 to P87	V_{CC} = 4.0 to 5.5 V	$V_{CC}\times 0.7$	—	V_{CC} + 0.3	V	
				$V_{CC}\times 0.8$	—	V_{CC} + 0.3		
		PB0 to PB7	V_{CC} = 4.0 to 5.5 V	$V_{CC}\times 0.7$	—	AV_{CC} + 0.3	V	
				$V_{CC}\times 0.8$	—	AV_{CC} + 0.3		
		OSC1	V_{CC} = 4.0 to 5.5 V	V_{CC} – 0.5	—	V_{CC} + 0.3	V	
				V_{CC} – 0.3	—	V_{CC} + 0.3		
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TMIB1, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV	V_{CC} = 4.0 to 5.5 V	–0.3	—	$V_{CC}\times 0.2$	V	
				–0.3	—	$V_{CC}\times 0.1$		

Note: Connect the TEST pin to Vss.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Input low voltage	V_{IL}	RXD, RXD_2, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87, PB0 to PB7	V_{CC} = 4.0 to 5.5 V	–0.3	—	$V_{CC} \times 0.3$	V	
				–0.3	—	$V_{CC} \times 0.2$	V	
		OSC1	V_{CC} = 4.0 to 5.5 V	–0.3	—	0.5	V	
				–0.3	—	0.3		
Output high voltage	V_{OH}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P55, P60 to P67, P70 to P72, P74 to P76, P85 to P87,	V_{CC} = 4.0 to 5.5 V $-I_{OH}$ = 1.5 mA	V_{CC} – 1.0	—	—	V	
			$-I_{OH}$ = 0.1 mA	V_{CC} – 0.5	—	—		
		P56, P57	V_{CC} = 4.0 to 5.5 V $-I_{OH}$ = 0.1 mA	V_{CC} – 2.5	—	—	V	
			V_{CC} = 3.0 to 4.0 V $-I_{OH}$ = 0.1 mA	V_{CC} – 2.0	—	—		
Output low voltage	V_{OL}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P70 to P72, P74 to P76, P85 to P87	V_{CC} = 4.0 to 5.5 V I_{OL} = 1.6 mA	—	—	0.6	V	
			I_{OL} = 0.4 mA	—	—	0.4		
		P60 to P67	V_{CC} = 4.0 to 5.5 V I_{OL} = 20.0 mA	—	—	1.5	V	
			V_{CC} = 4.0 to 5.5 V I_{OL} = 10.0 mA	—	—	1.0		
			V_{CC} = 4.0 to 5.5 V I_{OL} = 1.6 mA	—	—	0.4		
			I_{OL} = 0.4 mA	—	—	0.4		

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Output low voltage	V_{OL}	SCL, SDA	V_{CC} = 4.0 to 5.5 V I_{OL} = 6.0 mA	—	—	0.6	V	
			I_{OL} = 3.0 mA	—	—	0.4		
Input/ output leakage current	$\lvert I_{IL} \rvert$	OSC1, TMIB1, $\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TRGV, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1 RXD, SCK3, RXD_2, SCK3_2, SCL, SDA	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA	
		P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87,	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA	
		PB0 to PB7	V_{IN} = 0.5 V to (AV_{CC} – 0.5 V)	—	—	1.0	µA	
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17, P50 to P55	V_{CC} = 5.0 V, V_{IN} = 0.0 V	50.0	—	300.0	µA	
			V_{CC} = 3.0 V, V_{IN} = 0.0 V	—	60.0	—		Reference value
Input capacitance	C_{in}	All input pins except power supply pins	f = 1 MHz, V_{IN} = 0.0 V, T_a = 25°C	—	—	15.0	pF	
Active mode current consumption	I_{OPE1}	V_{CC}	Active mode 1 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	21.0	30.0	mA	*
			Active mode 1 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	9.0	—		* Reference value
	I_{OPE2}	V_{CC}	Active mode 2 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	1.8	3.0	mA	*
			Active mode 2 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	1.2	—		* Reference value

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Sleep mode current consumption	I_{SLEEP1}	V_{CC}	Sleep mode 1 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	17.5	22.5	mA	*
			Sleep mode 1 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	7.5	—		* Reference value
	I_{SLEEP2}	V_{CC}	Sleep mode 2 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	1.7	2.7	mA	*
			Sleep mode 2 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	1.1	—		* Reference value
Subactive mode current consumption	I_{SUB}	V_{CC}	V_{CC} = 3.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	35.0	70.0	µA	*
			V_{CC} = 3.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/8$)	—	25.0	—		* Reference value
Subsleep mode current consumption	I_{SUBSP}	V_{CC}	V_{CC} = 3.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	25.0	50.0	µA	*
Standby mode current consumption	I_{STBY}	V_{CC}	32-kHz crystal resonator not used	—	—	5.0	µA	*
RAM data retaining voltage	V_{RAM}	V_{CC}		2.0	—	—	V	

Note: * Pin states during current consumption measurement are given below (excluding current in the pull-up MOS transistors and output buffers).

Mode	$\overline{RES}$ Pin	Internal State	Other Pins	Oscillator Pins
Active mode 1	V_{CC}	Operates	V_{CC}	Main clock: ceramic or crystal resonator
Active mode 2		Operates (ϕOSC/64)		Subclock: Pin X1 = V_{SS}
Sleep mode 1	V_{CC}	Only timers operate	V_{CC}	
Sleep mode 2		Only timers operate (ϕOSC/64)		
Subactive mode	V_{CC}	Operates	V_{CC}	Main clock: ceramic or crystal resonator
Subsleep mode	V_{CC}	Only timers operate	V_{CC}	Subclock: crystal resonator
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Main clock: ceramic or crystal resonator Subclock: Pin X1 = V_{SS}

Table 23.2 DC Characteristics (2)

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
EEPROM current consumption	I_{EEW}	V_{CC}	V_{CC} = 5.0 V, t_{SCL} = 2.5 µs (when writing)	—	—	2.0	mA	*
	I_{EER}	V_{CC}	V_{CC} = 5.0 V, t_{SCL} = 2.5 µs (when reading)	—	—	0.3	mA	
	I_{EESTBY}	V_{CC}	V_{CC} = 5.0 V, t_{SCL} = 2.5 µs (at standby)	—	—	3.0	µA	

Note: * The current consumption of the EEPROM chip is shown.
For the current consumption of H8/3687N, add the above current values to the current consumption of H8/3687F.

Table 23.2 DC Characteristics (3)

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit
Allowable output low current (per pin)	I_{OL}	Output pins except port 6, SCL, and SDA	V_{CC} = 4.0 to 5.5 V	—	—	2.0	mA
		Port 6		—	—	20.0	
		Output pins except port 6, SCL, and SDA		—	—	0.5	
		Port 6		—	—	10.0	
		SCL, SDA		—	—	6.0	
Allowable output low current (total)	ΣI_{OL}	Output pins except port 6, SCL, and SDA	V_{CC} = 4.0 to 5.5 V	—	—	40.0	mA
		Port 6, SCL, and SDA		—	—	80.0	
		Output pins except port 6, SCL, and SDA		—	—	20.0	
		Port 6, SCL, and SDA		—	—	40.0	
Allowable output high current (per pin)	$\lvert -I_{OH} \rvert$	All output pins	V_{CC} = 4.0 to 5.5 V	—	—	2.0	mA
				—	—	0.2	
Allowable output high current (total)	$\lvert -\Sigma I_{OH} \rvert$	All output pins	V_{CC} = 4.0 to 5.5 V	—	—	30.0	mA
				—	—	8.0	

23.2.3 AC Characteristics

Table 23.3 AC Characteristics

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values			Unit	Reference Figure
				Min	Typ	Max		
System clock oscillation frequency	f_{OSC}	OSC1, OSC2	V_{CC} = 4.0 to 5.5 V	2.0	—	20.0	MHz	*1
				2.0	—	10.0		
System clock (φ) cycle time	t_{cyc}			1	—	64	t_{OSC}	*2
				—	—	12.8	µs	
Subclock oscillation frequency	f_W	X1, X2		—	32.768	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X1, X2		—	30.5	—	µs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}			2	—	8	t_W	*2
Instruction cycle time				2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time (crystal resonator)	t_{rc}	OSC1, OSC2		—	—	10.0	ms	
Oscillation stabilization time (ceramic resonator)	t_{rc}	OSC1, OSC2		—	—	5.0	ms	
Oscillation stabilization time	t_{rcx}	X1, X2		—	—	2.0	s	
External clock high width	t_{CPH}	OSC1	V_{CC} = 4.0 to 5.5 V	20.0	—	—	ns	Figure 23.1
				40.0	—	—		
External clock low width	t_{CPL}	OSC1	V_{CC} = 4.0 to 5.5 V	20.0	—	—	ns	
				40.0	—	—		
External clock rise time	t_{CPr}	OSC1	V_{CC} = 4.0 to 5.5 V	—	—	10.0	ns	
				—	—	15.0		
External clock fall time	t_{CPf}	OSC1	V_{CC} = 4.0 to 5.5 V	—	—	10.0	ns	
				—	—	15.0		

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
$\overline{RES}$ pin low width	t_{REL}	$\overline{RES}$	At power-on and in modes other than those below	t_{rc}	—	—	ms	Figure 23.2
			In active mode and sleep mode operation	200	—	—	ns	
Input pin high width	t_{IH}	$\overline{NMI}$, TMIB1, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{WKP0}$ to $\overline{WKP5}$, TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1		2	—	—	t_{cyc} t_{subcyc}	Figure 23.3
Input pin low width	t_{IL}	$\overline{NMI}$, TMIB1, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{WKP0}$ to $\overline{WKP5}$, TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1		2	—	—	t_{cyc} t_{subcyc}	

Notes: 1. When an external clock is input, the minimum system clock oscillation frequency is 1.0 MHz.

2. Determined by MA2, MA1, MA0, SA1, and SA0 of system control register 2 (SYSCR2).

Table 23.4 I²C Bus Interface Timing

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Test Condition	Values			Unit	Reference Figure
			Min	Typ	Max		
SCL input cycle time	t_{SCL}		$12t_{cyc}$ + 600	—	—	ns	Figure 23.4
SCL input high width	t_{SCLH}		$3t_{cyc}$ + 300	—	—	ns	
SCL input low width	t_{SCLL}		$5t_{cyc}$ + 300	—	—	ns	
SCL and SDA input fall time	t_{Sf}		—	—	300	ns	
SCL and SDA input spike pulse removal time	t_{SP}		—	—	$1t_{cyc}$	ns	
SDA input bus-free time	t_{BUF}		$5t_{cyc}$	—	—	ns	
Start condition input hold time	t_{STAH}		$3t_{cyc}$	—	—	ns	
Retransmission start condition input setup time	t_{STAS}		$3t_{cyc}$	—	—	ns	
Setup time for stop condition input	t_{STOS}		$3t_{cyc}$	—	—	ns	
Data-input setup time	t_{SDAS}		$1t_{cyc}$+20	—	—	ns	
Data-input hold time	t_{SDAH}		0	—	—	ns	
Capacitive load of SCL and SDA	c_b		0	—	400	pF	
SCL and SDA output fall time	t_{Sf}	V_{CC} = 4.0 to 5.5 V	—	—	250	ns	
			—	—	300		

Table 23.5 Serial Communication Interface (SCI) Timing

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item		Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
Input clock cycle	Asynchronous	t_{Scyc}	SCK3		4	—	—	t_{cyc}	Figure 23.5
	Clocked synchronous				6	—	—		
Input clock pulse width		t_{SCKW}	SCK3		0.4	—	0.6	t_{Scyc}	
Transmit data delay time (clocked synchronous)		t_{TXD}	TXD	V_{CC} = 4.0 to 5.5 V	—	—	1	t_{cyc}	Figure 23.6
					—	—	1		
Receive data setup time (clocked synchronous)		t_{RXS}	RXD	V_{CC} = 4.0 to 5.5 V	50.0	—	—	ns	
					100.0	—	—		
Receive data hold time (clocked synchronous)		t_{RXH}	RXD	V_{CC} = 4.0 to 5.5 V	50.0	—	—	ns	
					100.0	—	—		

23.2.4 A/D Converter Characteristics

Table 23.6 A/D Converter Characteristics

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values			Unit	Reference Figure
				Min	Typ	Max		
Analog power supply voltage	AV_{CC}	AV_{CC}		3.3	V_{CC}	5.5	V	*1
Analog input voltage	AV_{IN}	AN0 to AN7		V_{SS} – 0.3	—	AV_{CC} + 0.3	V	
Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 5.0 V f_{OSC} = 20 MHz	—	—	2.0	mA	
	AI_{STOP1}	AV_{CC}		—	50	—	µA	*2 Reference value
	AI_{STOP2}	AV_{CC}		—	—	5.0	µA	*3
Analog input capacitance	C_{AIN}	AN0 to AN7		—	—	30.0	pF	
Allowable signal source impedance	R_{AIN}	AN0 to AN7		—	—	5.0	kΩ	
Resolution (data length)				10	10	10	bit	
Conversion time (single mode)			AV_{CC} = 3.3 to 5.5 V	134	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	
Conversion time (single mode)			AV_{CC} = 4.0 to 5.5 V	70	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
Conversion time (single mode)			AV_{CC} = 4.0 to 5.5 V	134	—	—	t_{cyc}	
Nonlinearity error				—	—	±3.5	LSB	
Offset error				—	—	±3.5	LSB	
Full-scale error				—	—	±3.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±4.0	LSB	

Notes: 1. Set $AV_{CC} = V_{CC}$ when the A/D converter is not used.
2. AI_{STOP1} is the current in active and sleep modes while the A/D converter is idle.
3. AI_{STOP2} is the current at reset and in standby, subactive, and subsleep modes while the A/D converter is idle.

23.2.5 Watchdog Timer Characteristics

Table 23.7 Watchdog Timer Characteristics

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
On-chip oscillator overflow time	t_{OVF}			0.2	0.4	—	s	*

Note: * Shows the time to count from 0 to 255, at which point an internal reset is generated, when the internal oscillator is selected.

23.2.6 Flash Memory Characteristics

Table 23.8 Flash Memory Characteristics

V_{CC} = 3.0 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item		Symbol	Test Condition	Values Min	Typ	Max	Unit
Programming time (per 128 bytes)*1*2*4		t_P		—	7	200	ms
Erase time (per block) *1*3*6		t_E		—	100	1200	ms
Reprogramming count		N_{WEC}		1000	10000	—	Times
Programming	Wait time after SWE bit setting*1	x		1	—	—	µs
	Wait time after PSU bit setting*1	y		50	—	—	µs
	Wait time after P bit setting *1*4	z1	1 ≤ n ≤ 6	28	30	32	µs
		z2	7 ≤ n ≤ 1000	198	200	202	µs
		z3	Additional-programming	8	10	12	µs
	Wait time after P bit clear*1	α		5	—	—	µs
	Wait time after PSU bit clear*1	β		5	—	—	µs
	Wait time after PV bit setting*1	γ		4	—	—	µs
	Wait time after dummy write*1	ε		2	—	—	µs
	Wait time after PV bit clear*1	η		2	—	—	µs
	Wait time after SWE bit clear*1	θ		100	—	—	µs
	Maximum programming count *1*4*5	N		—	—	1000	Times

Item		Symbol	Test Condition	Values Min	Typ	Max	Unit
Erasing	Wait time after SWE bit setting*[1]	x		1	—	—	µs
	Wait time after ESU bit setting*[1]	y		100	—	—	µs
	Wait time after E bit setting*[1]*[6]	z		10	—	100	ms
	Wait time after E bit clear*[1]	α		10	—	—	µs
	Wait time after ESU bit clear*[1]	β		10	—	—	µs
	Wait time after EV bit setting*[1]	γ		20	—	—	µs
	Wait time after dummy write*[1]	ε		2	—	—	µs
	Wait time after EV bit clear*[1]	η		4	—	—	µs
	Wait time after SWE bit clear*[1]	θ		100	—	—	µs
	Maximum erase count *[1]*[6]*[7]	N		—	—	120	Times

Notes:
1. Make the time settings in accordance with the program/erase algorithms.
2. The programming time for 128 bytes. (Indicates the total time for which the P bit in flash memory control register 1 (FLMCR1) is set. The program-verify time is not included.)
3. The time required to erase one block. (Indicates the time for which the E bit in flash memory control register 1 (FLMCR1) is set. The erase-verify time is not included.)
4. Programming time maximum value (t_P(max.)) = wait time after P bit setting (z) × maximum programming count (N)
5. Set the maximum programming count (N) according to the actual set values of z1, z2, and z3, so that it does not exceed the programming time maximum value (t_P(max.)). The wait time after P bit setting (z1, z2) should be changed as follows according to the value of the programming count (n).

   Programming count (n)

   $1 \leq n \leq 6$ z1 = 30 µs

   $7 \leq n \leq 1000$ z2 = 200 µs
6. Erase time maximum value (t_E(max.)) = wait time after E bit setting (z) × maximum erase count (N)
7. Set the maximum erase count (N) according to the actual set value of (z), so that it does not exceed the erase time maximum value (t_E(max.)).

23.2.7 EEPROM Characteristics

Table 23.9 EEPROM Characteristics

V_{CC} = 3.0 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C, unless otherwise specified.

Item	Symbol	Test Condition	Values			Unit	Reference Figure
			Min	Typ	Max		
SCL input cycle time	t_{SCL}		2500	—		ns	Figure 23.7
SCL input high pulse width	t_{SCLH}		600	—	—	µs	
SCL input low pulse width	t_{SCLL}		1200	—	—	ns	
SCL, SDA input spike pulse removal time	t_{SP}		—	—	50	ns	
SDA input bus-free time	t_{BUF}		1200	—	—	ns	
Start condition input hold time	t_{STAH}		600	—	—	ns	
Retransmit start condition input setup time	t_{STAS}		600	—	—	ns	
Stop condition input setup time	t_{STOS}		600	—	—	ns	
Data input setup time	t_{SDAS}		160	—	—	ns	
Data input hold time	t_{SDAH}		0	—	—	ns	
SCL, SDA input fall time	t_{Sf}		—	—	300	ns	
SDA input rise time	t_{Sr}		—	—	300	ns	
Data output hold time	t_{DH}		50	—	—	ns	
SCL, SDA capacitive load	C_b		0	—	400	pF	
Access time	t_{AA}		100	—	900	ns	
Cycle time at writing*	t_{WC}		—	—	10	ms	
Reset release time	t_{RES}		—	—	13	ms	

Note: * Cycle time at writing is a time from the stop condition to write completion (internal control).

23.2.8 Power-Supply-Voltage Detection Circuit Characteristics (Optional)

Table 23.10 Power-Supply-Voltage Detection Circuit Characteristics

V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

			Values			
Item	**Symbol**	**Test Condition**	**Min**	**Typ**	**Max**	**Unit**
Power-supply falling detection voltage	Vint (D)	LVDSEL = 0	3.3	3.7	—	V
Power-supply rising detection voltage	Vint (U)	LVDSEL = 0	—	4.0	4.5	V
Reset detection voltage 1*[1]	Vreset1	LVDSEL = 0	—	2.3	2.7	V
Reset detection voltage 2*[2]	Vreset2	LVDSEL = 1	3.0	3.6	4.2	V
Lower-limit voltage of LVDR operation*[3]	$V_{LVDRmin}$		1.0	—	—	V
LVD stabilization time	t_{LVDON}		50	—	—	µs
Current consumption in standby mode	I_{STBY}	LVDE = 1, Vcc = 5.0 V, When a 32-kHz crystal resonator is not used	—	—	350	µA

Notes: 1. This voltage should be used when the falling and rising voltage detection function is used.
2. Select the low-voltage reset 2 when only the low-voltage detection reset is used.
3. When the power-supply voltage (Vcc) falls below $V_{LVDRmin}$ = 1.0 V and then rises, a reset may not occur. Therefore sufficient evaluation is required.

23.2.9 Power-On Reset Circuit Characteristics (Optional)

Table 23.11 Power-On Reset Circuit Characteristics

V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

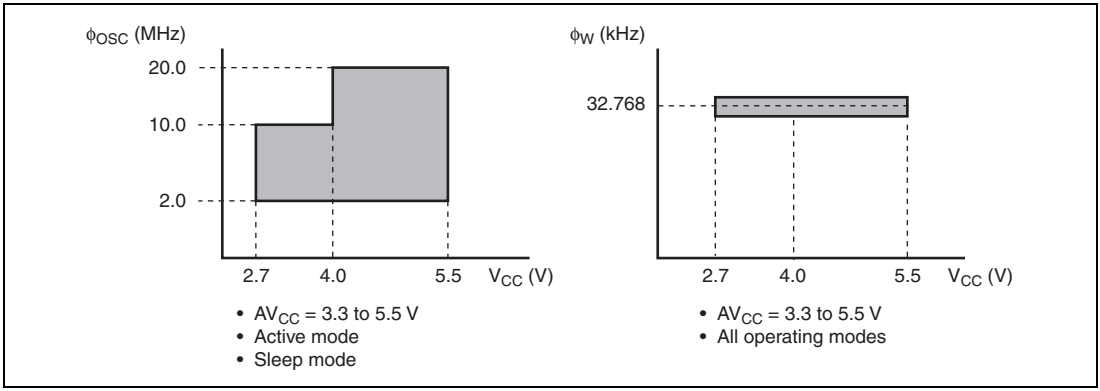
			Values			
Item	**Symbol**	**Test Condition**	**Min**	**Typ**	**Max**	**Unit**
Pull-up resistance of $\overline{RES}$ pin	R_{RES}		100	150	—	kΩ
Power-on reset start voltage*	V_{por}		—	—	100	mV

Note: * The power-supply voltage (Vcc) must fall below Vpor = 100 mV and then rise after charge of the $\overline{RES}$ pin is removed completely. In order to remove charge of the $\overline{RES}$ pin, it is recommended that the diode be placed in the Vcc side. If the power-supply voltage (Vcc) rises from the point over 100 mV, a power-on reset may not occur.

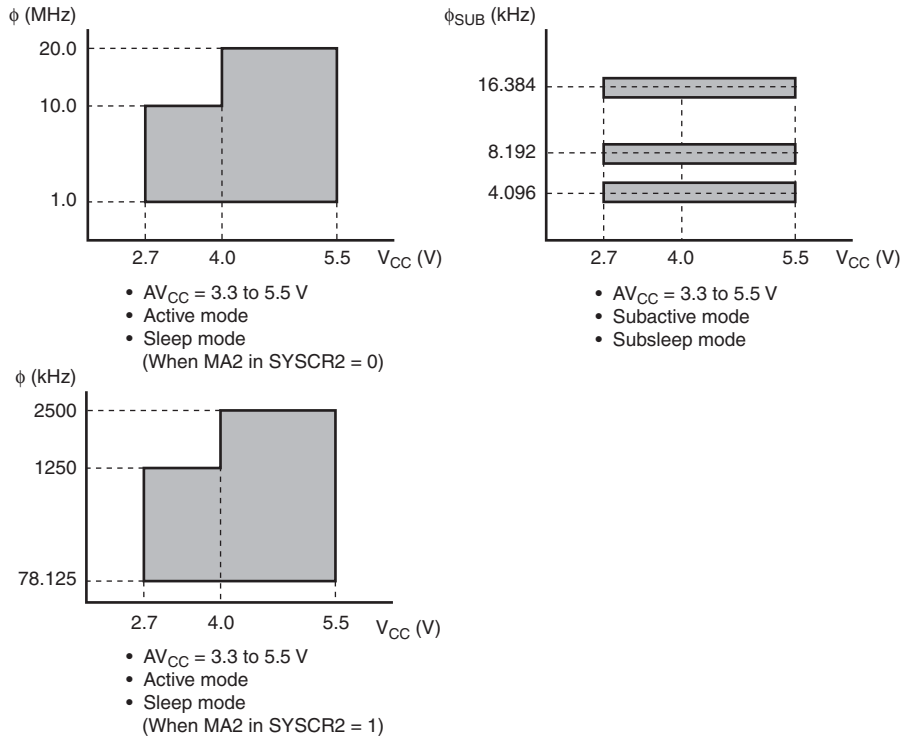
23.3 Electrical Characteristics (Mask-ROM Version, EEPROM Stacked Mask-ROM Version)

23.3.1 Power Supply Voltage and Operating Ranges

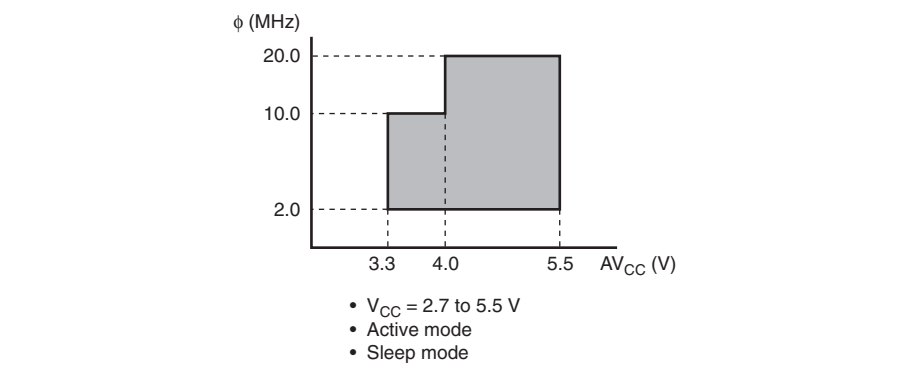
Power Supply Voltage and Oscillation Frequency Range



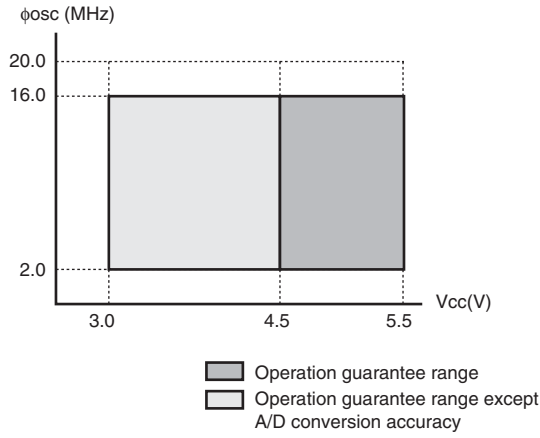
Power Supply Voltage and Operating Frequency Range



Analog Power Supply Voltage and A/D Converter Accuracy Guarantee Range



Range of Power Supply Voltage and Oscillation Frequency when Low-Voltage Detection Circuit is Used


φosc (MHz)
20.0
16.0
2.0
3.0
4.5
5.5
Vcc(V)
Operation guarantee range
Operation guarantee range except A/D conversion accuracy

23.3.2 DC Characteristics

Table 23.12 DC Characteristics (1)

V_{CC} = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$,TMIB1, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV	V_{CC} = 4.0 to 5.5 V	$V_{CC}\times 0.8$	—	V_{CC} + 0.3	V	
				$V_{CC}\times 0.9$	—	V_{CC} + 0.3		
		RXD, RXD_2 SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37	V_{CC} = 4.0 to 5.5 V	$V_{CC}\times 0.7$	—	V_{CC} + 0.3	V	
		P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87		$V_{CC}\times 0.8$	—	V_{CC} + 0.3		
		PB0 to PB7	V_{CC} = 4.0 to 5.5 V	$V_{CC}\times 0.7$	—	AV_{CC} + 0.3	V	
				$V_{CC}\times 0.8$	—	AV_{CC} + 0.3		
		OSC1	V_{CC} = 4.0 to 5.5 V	V_{CC} – 0.5	—	V_{CC} + 0.3	V	
				V_{CC} – 0.3	—	V_{CC} + 0.3		

Note: Connect the TEST pin to Vss.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Input low voltage	V_{IL}	$\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, SCK3, SCK3_2, TRGV	V_{CC} = 4.0 to 5.5 V	–0.3	—	$V_{CC} \times 0.2$	V	
				–0.3	—	$V_{CC} \times 0.1$		
		RXD, RXD_2, SCL, SDA, P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67,. P70 to P72, P74 to P76, P85 to P87, PB0 to PB7	V_{CC} = 4.0 to 5.5 V	–0.3	—	$V_{CC} \times 0.3$	V	
				–0.3	—	$V_{CC} \times 0.2$		
		OSC1	V_{CC} = 4.0 to 5.5 V	–0.3	—	0.5	V	
				–0.3	—	0.3		
Output high voltage	V_{OH}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P55, P60 to P67, P70 to P72, P74 to P76, P85 to P87	V_{CC} = 4.0 to 5.5 V $-I_{OH}$ = 1.5 mA	V_{CC} – 1.0	—	—	V	
			$-I_{OH}$ = 0.1 mA	V_{CC} – 0.5	—	—		
		P56, P57	V_{CC} = 4.0 to 5.5 V $-I_{OH}$ = 0.1 mA	V_{CC} – 2.5	—	—	V	
			V_{CC} =2.7 to 4.0 V $-I_{OH}$ = 0.1 mA	V_{CC} – 2.0	—	—		

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Output low voltage	V_{OL}	P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P70 to P72, P74 to P76, P85 to P87	V_{CC} = 4.0 to 5.5 V, I_{OL} = 1.6 mA	—	—	0.6	V	
			I_{OL} = 0.4 mA	—	—	0.4		
		P60 to P67	V_{CC} = 4.0 to 5.5 V, I_{OL} = 20.0 mA	—	—	1.5	V	
			V_{CC} = 4.0 to 5.5 V, I_{OL} = 10.0 mA	—	—	1.0		
			V_{CC} = 4.0 to 5.5 V, I_{OL} = 1.6 mA	—	—	0.4		
			I_{OL} = 0.4 mA	—	—	0.4		
		SCL, SDA	V_{CC} = 4.0 to 5.5 V, I_{OL} = 6.0 mA	—	—	0.6	V	
			I_{OL} = 3.0 mA	—	—	0.4		
Input/ output leakage current	$\lvert I_{IL} \rvert$	OSC1, TMIB1, $\overline{RES}$, $\overline{NMI}$, $\overline{WKP0}$ to $\overline{WKP5}$, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{ADTRG}$, TRGV, TMRIV, TMCIV, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1, RXD, SCK3, RXD_2, SCK3_2, SCL, SDA	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA	
		P10 to P12, P14 to P17, P20 to P24, P30 to P37, P50 to P57, P60 to P67, P70 to P72, P74 to P76, P85 to P87,	V_{IN} = 0.5 V to (V_{CC} – 0.5 V)	—	—	1.0	µA	
		PB0 to PB7	V_{IN} = 0.5 V to (AV_{CC} – 0.5 V)	—	—	1.0	µA	

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Pull-up MOS current	$-I_p$	P10 to P12, P14 to P17, P50 to P55	V_{CC} = 5.0 V, V_{IN} = 0.0 V	50.0	—	300.0	µA	
			V_{CC} = 3.0 V, V_{IN} = 0.0 V	—	60.0	—		Reference value
Input capacitance	C_{in}	All input pins except power supply pins	f = 1 MHz, V_{IN} = 0.0 V, T_a = 25°C	—	—	15.0	pF	
Active mode current consumption	I_{OPE1}	V_{CC}	Active mode 1 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	21.0	30.0	mA	*
			Active mode 1 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	9.0	—		* Reference value
	I_{OPE2}	V_{CC}	Active mode 2 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	1.8	3.0	mA	*
			Active mode 2 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	1.2	—		* Reference value
Sleep mode current consumption	I_{SLEEP1}	V_{CC}	Sleep mode 1 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	17.5	22.5	mA	*
			Sleep mode 1 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	7.5	—		* Reference value
	I_{SLEEP2}	V_{CC}	Sleep mode 2 V_{CC} = 5.0 V, f_{OSC} = 20 MHz	—	1.7	2.7	mA	*
			Sleep mode 2 V_{CC} = 3.0 V, f_{OSC} = 10 MHz	—	1.1	—		* Reference value
Subactive mode current consumption	I_{SUB}	V_{CC}	V_{CC} = 3.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	35.0	70.0	µA	*
			V_{CC} = 3.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/8$)	—	25.0	—		* Reference value
Subsleep mode current consumption	I_{SUBSP}	V_{CC}	V_{CC} = 3.0 V 32-kHz crystal resonator ($\phi_{SUB} = \phi_W/2$)	—	25.0	50.0	µA	*

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Notes
Standby mode current consumption	I_{STBY}	V_{CC}	32-kHz crystal resonator not used	—	—	5.0	µA	*
RAM data retaining voltage	V_{RAM}	V_{CC}		2.0	—	—	V	

Note: * Pin states during current consumption measurement are given below (excluding current in the pull-up MOS transistors and output buffers).

Mode	$\overline{RES}$ Pin	Internal State	Other Pins	Oscillator Pins
Active mode 1	V_{CC}	Operates	V_{CC}	Main clock: ceramic or crystal resonator
Active mode 2		Operates (φOSC/64)		Subclock: Pin X1 = V_{SS}
Sleep mode 1	V_{CC}	Only timers operate	V_{CC}	
Sleep mode 2		Only timers operate (φOSC/64)		
Subactive mode	V_{CC}	Operates	V_{CC}	Main clock: ceramic or crystal resonator
Subsleep mode	V_{CC}	Only timers operate	V_{CC}	Subclock resonator: crystal
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Main clock: ceramic or crystal resonator Subclock: Pin X1 = V_{SS}

Table 23.12 DC Characteristics (2)

V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Values Typ	Values Max	Unit	Notes
EEPROM current consumption	I_{EEW}	V_{CC}	V_{CC} = 5.0 V, t_{SCL} = 2.5 µs (when writing)	—	—	2.0	mA	*
	I_{EER}	V_{CC}	V_{CC} = 5.0 V, t_{SCL} = 2.5 µs (when reading)	—	—	0.3	mA	
	I_{EESTBY}	V_{CC}	V_{CC} = 5.0 V, t_{SCL} = 2.5 µs (at standby)	—	—	3.0	µA	

Note: * The current consumption of the EEPROM chip is shown.
For the current consumption of H8/3687N, add the above current values to the current consumption of H8/3687.

Table 23.12 DC Characteristics (3)

V_{CC} = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

				Values			
Item	**Symbol**	**Applicable Pins**	**Test Condition**	**Min**	**Typ**	**Max**	**Unit**
Allowable output low current (per pin)	I_{OL}	Output pins except port 6, SCL, and SDA	V_{CC} = 4.0 to 5.5 V	—	—	2.0	mA
		Port 6		—	—	20.0	
		Output pins except port 6, SCL, and SDA		—	—	0.5	
		Port 6		—	—	10.0	
		SCL, SDA		—	—	6.0	
Allowable output low current (total)	ΣI_{OL}	Output pins except port 6, SCL, and SDA	V_{CC} = 4.0 to 5.5 V	—	—	40.0	mA
		Port 6, SCL, and SDA		—	—	80.0	
		Output pins except port 6, SCL, and SDA		—	—	20.0	
		Port 6, SCL, and SDA		—	—	40.0	
Allowable output high current (per pin)	$\lvert -I_{OH} \rvert$	All output pins	V_{CC} = 4.0 to 5.5 V	—	—	2.0	mA
				—	—	0.2	
Allowable output high current (total)	$\lvert -\Sigma I_{OH} \rvert$	All output pins	V_{CC} = 4.0 to 5.5 V	—	—	30.0	mA
				—	—	8.0	

23.3.3 AC Characteristics

Table 23.13 AC Characteristics

V_{CC} = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
System clock oscillation frequency	f_{OSC}	OSC1, OSC2	V_{CC} = 4.0 to 5.5 V	2.0	—	20.0	MHz	*1
				2.0		10.0		
System clock (ϕ) cycle time	t_{cyc}			1	—	64	t_{OSC}	*2
				—	—	12.8	µs	
Subclock oscillation frequency	f_W	X1, X2		—	32.768	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X1, X2		—	30.5	—	µs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}			2	—	8	t_W	*2
Instruction cycle time				2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time (crystal resonator)	t_{rc}	OSC1, OSC2		—	—	10.0	ms	
Oscillation stabilization time (ceramic resonator)	t_{rc}	OSC1, OSC2		—	—	5.0	ms	
Oscillation stabilization time	t_{rcx}	X1, X2		—	—	2.0	s	
External clock high width	t_{CPH}	OSC1	V_{CC} = 4.0 to 5.5 V	20.0	—	—	ns	Figure 23.1
				40.0	—	—		
External clock low width	t_{CPL}	OSC1	V_{CC} = 4.0 to 5.5 V	20.0	—	—	ns	
				40.0	—	—		
External clock rise time	t_{CPr}	OSC1	V_{CC} = 4.0 to 5.5 V	—	—	10.0	ns	
				—	—	15.0		
External clock fall time	t_{CPf}	OSC1	V_{CC} = 4.0 to 5.5 V	—	—	10.0	ns	
				—	—	15.0		

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
$\overline{RES}$ pin low width	t_{REL}	$\overline{RES}$	At power-on and in modes other than those below	t_{rc}	—	—	ms	Figure 23.2
			In active mode and sleep mode operation	200	—	—	ns	
Input pin high width	t_{IH}	$\overline{NMI}$, TMIB1, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{WKP0}$ to $\overline{WKP5}$, TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1		2	—	—	t_{cyc} t_{subcyc}	Figure 23.3
Input pin low width	t_{IL}	$\overline{NMI}$, TMIB1, $\overline{IRQ0}$ to $\overline{IRQ3}$, $\overline{WKP0}$ to $\overline{WKP5}$, TMCIV, TMRIV, TRGV, $\overline{ADTRG}$, FTIOA0 to FTIOD0, FTIOA1 to FTIOD1		2	—	—	t_{cyc} t_{subcyc}	

Notes: 1. When an external clock is input, the minimum system clock oscillation frequency is 1.0 MHz.

2. Determined by MA2, MA1, MA0, SA1, and SA0 of system control register 2 (SYSCR2).

Table 23.14 I²C Bus Interface Timing

V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Test Condition	Values			Unit	Reference Figure
			Min	Typ	Max		
SCL input cycle time	t_{SCL}		$12t_{cyc}$ + 600	—	—	ns	Figure 23.4
SCL input high width	t_{SCLH}		$3t_{cyc}$ + 300	—	—	ns	
SCL input low width	t_{SCLL}		$5t_{cyc}$ + 300	—	—	ns	
SCL and SDA input fall time	t_{Sf}		—	—	300	ns	
SCL and SDA input spike pulse removal time	t_{SP}		—	—	$1t_{cyc}$	ns	
SDA input bus-free time	t_{BUF}		$5t_{cyc}$	—	—	ns	
Start condition input hold time	t_{STAH}		$3t_{cyc}$	—	—	ns	
Retransmission start condition input setup time	t_{STAS}		$3t_{cyc}$	—	—	ns	
Setup time for stop condition input	t_{STOS}		$3t_{cyc}$	—	—	ns	
Data-input setup time	t_{SDAS}		$1t_{cyc}$+20	—	—	ns	
Data-input hold time	t_{SDAH}		0	—	—	ns	
Capacitive load of SCL and SDA	c_b		0	—	400	pF	
SCL and SDA output fall time	t_{Sf}	V_{CC} = 4.0 to 5.5 V	—	—	250	ns	
			—	—	300		

Table 23.15 Serial Communication Interface (SCI) Timing

V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item		Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
Input clock cycle	Asynchronous	t_{Scyc}	SCK3		4	—	—	t_{cyc}	Figure 23.5
	Clocked synchronous				6	—	—		
Input clock pulse width		t_{SCKW}	SCK3		0.4	—	0.6	t_{Scyc}	
Transmit data delay time (clocked synchronous)		t_{TXD}	TXD	V_{CC} = 4.0 to 5.5 V	—	—	1	t_{cyc}	Figure 23.6
					—	—	1		
Receive data setup time (clocked synchronous)		t_{RXS}	RXD	V_{CC} = 4.0 to 5.5 V	50.0	—	—	ns	
					100.0	—	—		
Receive data hold time (clocked synchronous)		t_{RXH}	RXD	V_{CC} = 4.0 to 5.5 V	50.0	—	—	ns	
					100.0	—	—		

23.3.4 A/D Converter Characteristics

Table 23.16 A/D Converter Characteristics

V_{CC} = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values			Unit	Reference Figure
				Min	Typ	Max		
Analog power supply voltage	AV_{CC}	AV_{CC}		3.3	V_{CC}	5.5	V	*1
Analog input voltage	AV_{IN}	AN0 to AN7		V_{SS} – 0.3	—	AV_{CC} + 0.3	V	
Analog power supply current	AI_{OPE}	AV_{CC}	AV_{CC} = 5.0 V f_{OSC} = 20 MHz	—	—	2.0	mA	
	AI_{STOP1}	AV_{CC}		—	50	—	µA	*2 Reference value
	AI_{STOP2}	AV_{CC}		—	—	5.0	µA	*3
Analog input capacitance	C_{AIN}	AN0 to AN7		—	—	30.0	pF	
Allowable signal source impedance	R_{AIN}	AN0 to AN7		—	—	5.0	kΩ	
Resolution (data length)				10	10	10	bit	
Conversion time (single mode)			AV_{CC} = 3.3 to 5.5 V	134	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	
Conversion time (single mode)			AV_{CC} = 4.0 to 5.5 V	70	—	—	t_{cyc}	
Nonlinearity error				—	—	±7.5	LSB	
Offset error				—	—	±7.5	LSB	
Full-scale error				—	—	±7.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±8.0	LSB	

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
Conversion time (single mode)			AV_{CC} = 4.0 to 5.5 V	134	—	—	t_{cyc}	
Nonlinearity error				—	—	±3.5	LSB	
Offset error				—	—	±3.5	LSB	
Full-scale error				—	—	±3.5	LSB	
Quantization error				—	—	±0.5	LSB	
Absolute accuracy				—	—	±4.0	LSB	

Notes: 1. Set $AV_{CC} = V_{CC}$ when the A/D converter is not used.
2. AI_{STOP1} is the current in active and sleep modes while the A/D converter is idle.
3. AI_{STOP2} is the current at reset and in standby, subactive, and subsleep modes while the A/D converter is idle.

23.3.5 Watchdog Timer Characteristics

Table 23.17 Watchdog Timer Characteristics

V_{CC} = 2.7 to 5.5 V, V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Applicable Pins	Test Condition	Values Min	Typ	Max	Unit	Reference Figure
On-chip oscillator overflow time	t_{OVF}			0.2	0.4	—	s	*

Note: * Shows the time to count from 0 to 255, at which point an internal reset is generated, when the internal oscillator is selected.

23.3.6 EEPROM Characteristics

Table 23.18 EEPROM Characteristics

V_{CC} = 2.7 V to 5.5 V, V_{SS} = 0.0 V, T_a = –20°C to +75°C, unless otherwise indicated.

Item	Symbol	Test Condition	Values			Unit	Reference Figure
			Min	Typ	Max		
SCL input cycle time	t_{SCL}		2500	—		ns	Figure 23.7
SCL input high pulse width	t_{SCLH}		600	—	—	µs	
SCL input low pulse width	t_{SCLL}		1200	—	—	ns	
SCL, SDA input spike pulse removal time	t_{SP}		—	—	50	ns	
SDA input bus-free time	t_{BUF}		1200	—	—	ns	
Start condition input hold time	t_{STAH}		600	—	—	ns	
Retransmit start condition input setup time	t_{STAS}		600	—	—	ns	
Stop condition input setup time	t_{STOS}		600	—	—	ns	
Data input setup time	t_{SDAS}		160	—	—	ns	
Data input hold time	t_{SDAH}		0	—	—	ns	
SCL, SDA input fall time	t_{Sf}		—	—	300	ns	
SDA input rise time	t_{Sr}		—	—	300	ns	
Data output hold time	t_{DH}		50	—	—	ns	
SCL, SDA capacitive load	C_b		0	—	400	pF	
Access time	t_{AA}		100	—	900	ns	
Cycle time at writing*	t_{WC}		—	—	10	ms	
Reset release time	t_{RES}		—	—	13	ms	

Note: * Cycle time at writing is a time from the stop condition to write completion (internal control).

23.3.7 Power-Supply-Voltage Detection Circuit Characteristics (Optional)

Table 23.19 Power-Supply-Voltage Detection Circuit Characteristics

V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Test Condition	Values			Unit
			Min	Typ	Max	
Power-supply falling detection voltage	Vint (D)	LVDSEL = 0	3.3	3.7	—	V
Power-supply rising detection voltage	Vint (U)	LVDSEL = 0	—	4.0	4.5	V
Reset detection voltage 1*[1]	Vreset1	LVDSEL = 0	—	2.3	2.7	V
Reset detection voltage 2*[2]	Vreset2	LVDSEL = 1	3.0	3.6	4.2	V
Lower-limit voltage of LVDR operation*[3]	$V_{LVDRmin}$		1.0	—	—	V
LVD stabilization time	t_{LVDON}		50	—	—	µs
Current consumption in standby mode	I_{STBY}	LVDE = 1, Vcc = 5.0 V, When a 32-kHz crystal resonator is not used	—	—	350	µA

Notes: 1. This voltage should be used when the falling and rising voltage detection function is used.
2. Select the low-voltage reset 2 when only the low-voltage detection reset is used.
3. When the power-supply voltage (Vcc) falls below $V_{LVDRmin}$ = 1.0 V and then rises, a reset may not occur. Therefore sufficient evaluation is required.

23.3.8 Power-On Reset Circuit Characteristics (Optional)

Table 23.20 Power-On Reset Circuit Characteristics

V_{SS} = 0.0 V, T_a = –20 to +75°C, unless otherwise indicated.

Item	Symbol	Test Condition	Values			Unit
			Min	Typ	Max	
Pull-up resistance of $\overline{RES}$ pin	R_{RES}		100	150	—	kΩ
Power-on reset start voltage*	V_{por}		—	—	100	mV

Note: * The power-supply voltage (Vcc) must fall below Vpor = 100 mV and then rise after charge of the $\overline{RES}$ pin is removed completely. In order to remove charge of the $\overline{RES}$ pin, it is recommended that the diode be placed in the Vcc side. If the power-supply voltage (Vcc) rises from the point over 100 mV, a power-on reset may not occur.

23.4 Operation Timing

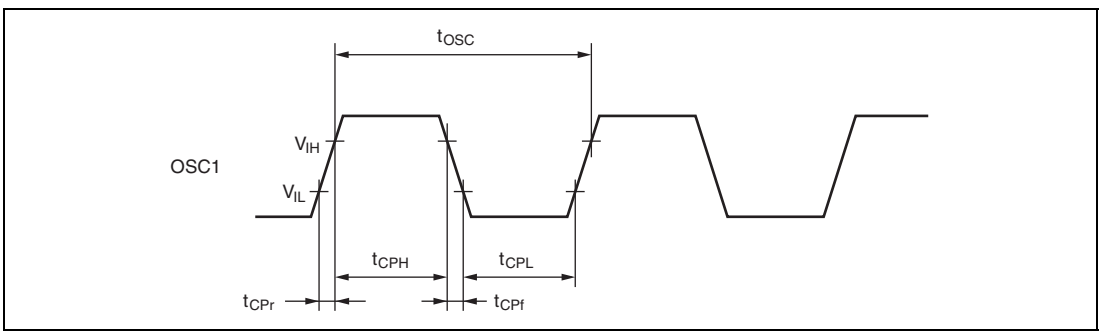


Figure 23.1 System Clock Input Timing

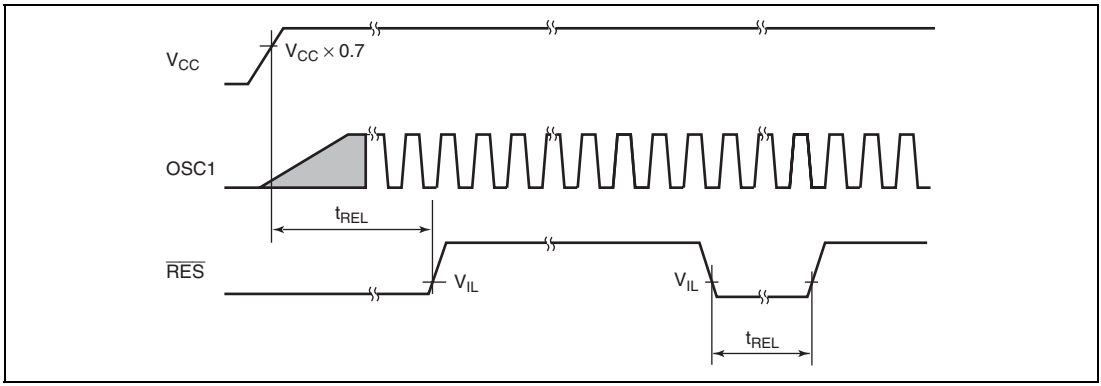


Figure 23.2 $\overline{RES}$ Low Width Timing

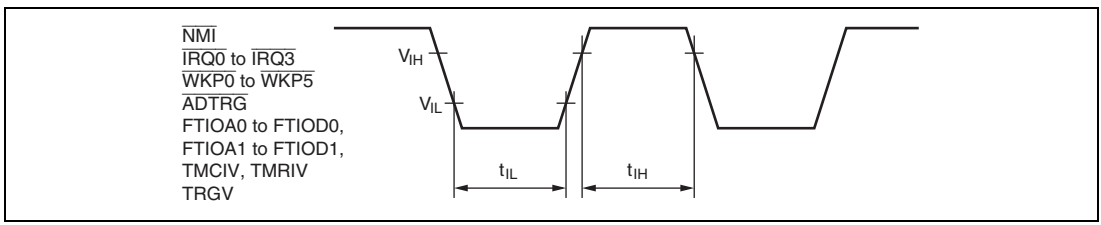
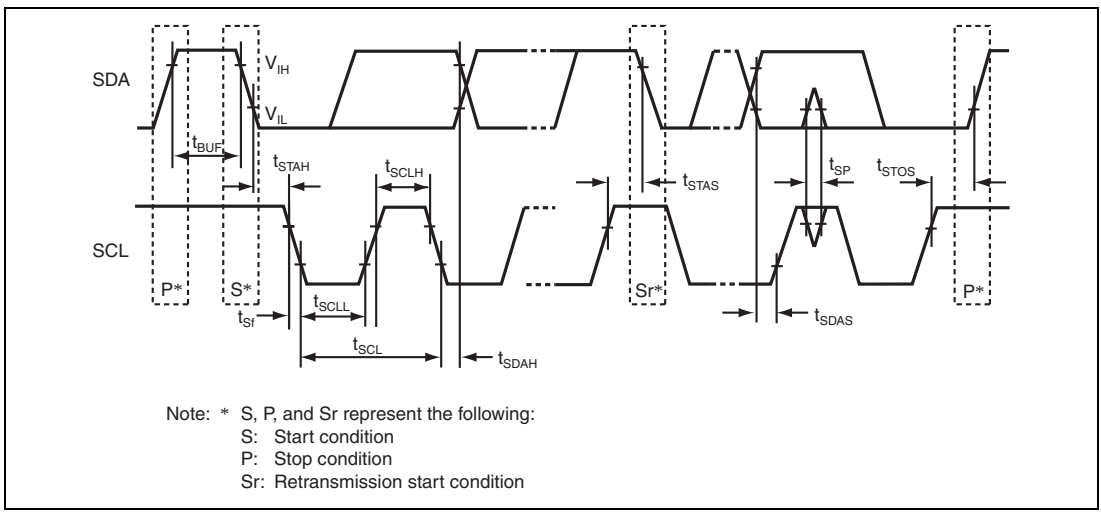


Figure 23.3 Input Timing



Note: * S, P, and Sr represent the following:
S: Start condition
P: Stop condition
Sr: Retransmission start condition

Figure 23.4 I^2C Bus Interface Input/Output Timing

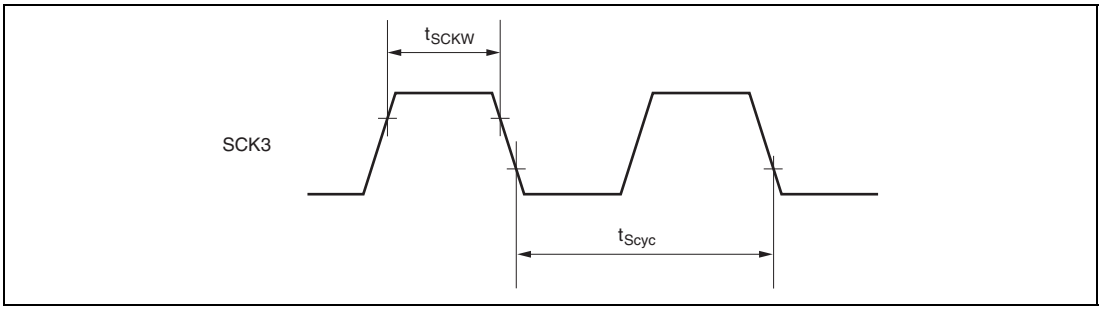
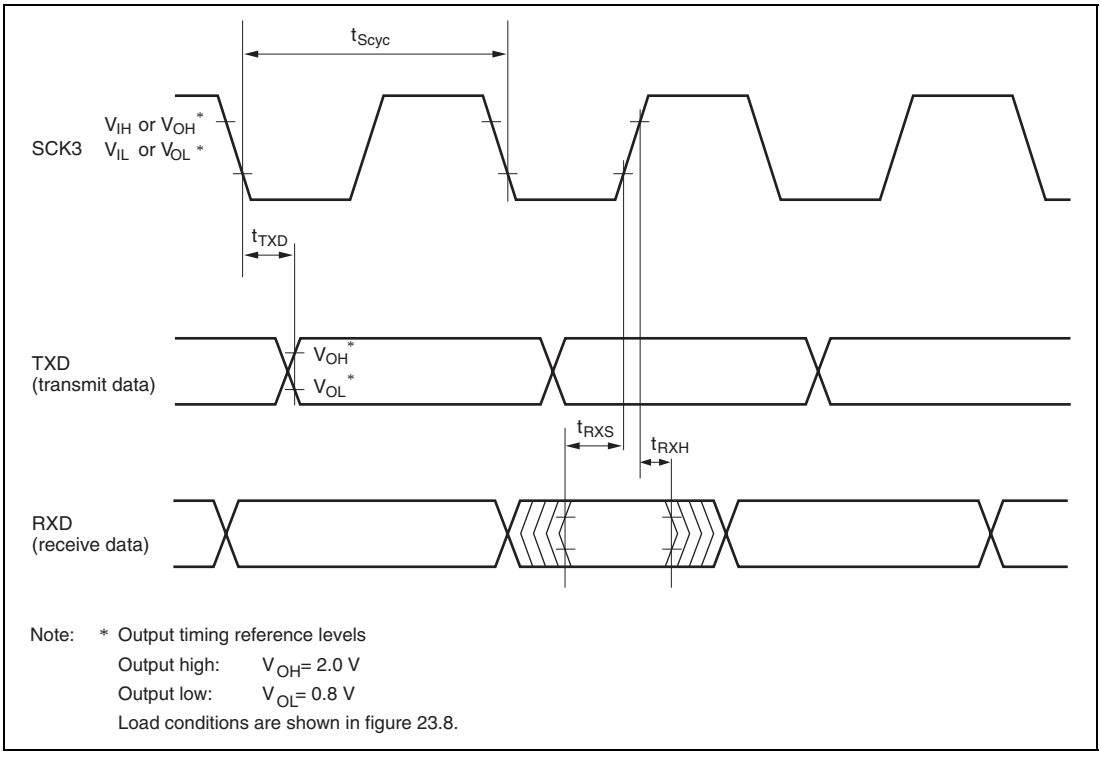


Figure 23.5 SCK3 Input Clock Timing



Note: * Output timing reference levels
Output high: V_{OH}= 2.0 V
Output low: V_{OL}= 0.8 V
Load conditions are shown in figure 23.8.

Figure 23.6 SCI Input/Output Timing in Clocked Synchronous Mode

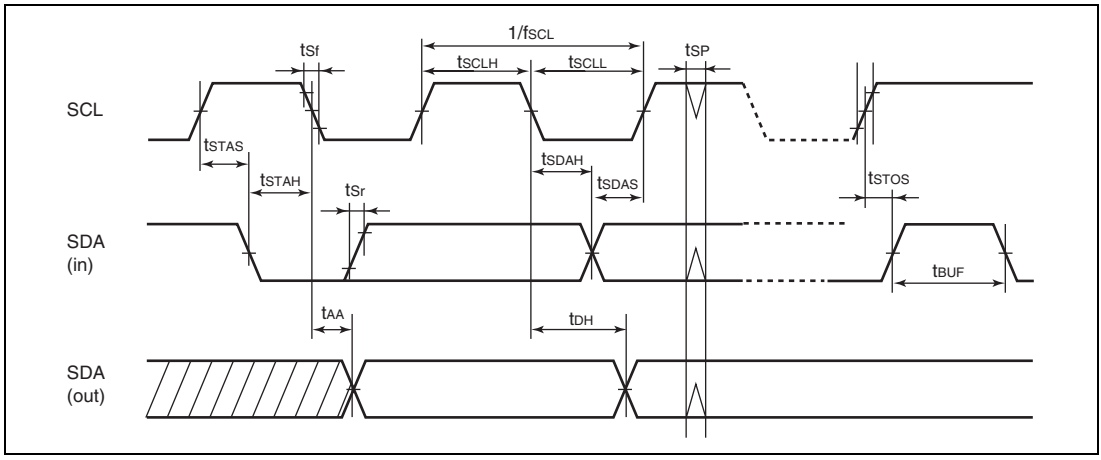


Figure 23.7 EEPROM Bus Timing

23.5 Output Load Condition

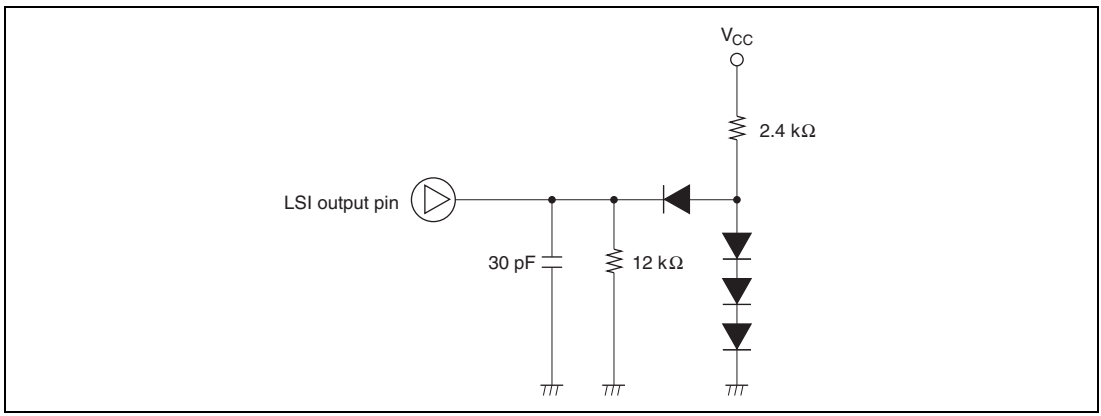


Figure 23.8 Output Load Circuit

Appendix A Instruction Set

A.1 Instruction List

Condition Code

Symbol	Description
Rd	General destination register
Rs	General source register
Rn	General register
ERd	General destination register (address register or 32-bit register)
ERs	General source register (address register or 32-bit register)
ERn	General register (32-bit register)
(EAd)	Destination operand
(EAs)	Source operand
PC	Program counter
SP	Stack pointer
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
disp	Displacement
→	Transfer from the operand on the left to the operand on the right, or transition from the state on the left to the state on the right
+	Addition of the operands on both sides
–	Subtraction of the operand on the right from the operand on the left
×	Multiplication of the operands on both sides
÷	Division of the operand on the left by the operand on the right
∧	Logical AND of the operands on both sides
∨	Logical OR of the operands on both sides
⊕	Logical exclusive OR of the operands on both sides
¬	NOT (logical complement)
( ), < >	Contents of operand

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

Condition Code Notation (cont)

Symbol	Description
↕	Changed according to execution result
*	Undetermined (no guaranteed value)
0	Cleared to 0
1	Set to 1
—	Not affected by execution of the instruction
Δ	Varies depending on conditions, described in notes

Table A.1 Instruction Set

1. Data Transfer Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
MOV	MOV.B #xx:8, Rd	B	2									#xx:8 → Rd8	—	—	↕	↕	0	—	2	
	MOV.B Rs, Rd	B		2								Rs8 → Rd8	—	—	↕	↕	0	—	2	
	MOV.B @ERs, Rd	B			2							@ERs → Rd8	—	—	↕	↕	0	—	4	
	MOV.B @(d:16, ERs), Rd	B				4						@(d:16, ERs) → Rd8	—	—	↕	↕	0	—	6	
	MOV.B @(d:24, ERs), Rd	B				8						@(d:24, ERs) → Rd8	—	—	↕	↕	0	—	10	
	MOV.B @ERs+, Rd	B					2					@ERs → Rd8 ERs32+1 → ERs32	—	—	↕	↕	0	—	6	
	MOV.B @aa:8, Rd	B						2				@aa:8 → Rd8	—	—	↕	↕	0	—	4	
	MOV.B @aa:16, Rd	B						4				@aa:16 → Rd8	—	—	↕	↕	0	—	6	
	MOV.B @aa:24, Rd	B						6				@aa:24 → Rd8	—	—	↕	↕	0	—	8	
	MOV.B Rs, @ERd	B			2							Rs8 → @ERd	—	—	↕	↕	0	—	4	
	MOV.B Rs, @(d:16, ERd)	B				4						Rs8 → @(d:16, ERd)	—	—	↕	↕	0	—	6	
	MOV.B Rs, @(d:24, ERd)	B				8						Rs8 → @(d:24, ERd)	—	—	↕	↕	0	—	10	
	MOV.B Rs, @–ERd	B					2					ERd32–1 → ERd32 Rs8 → @ERd	—	—	↕	↕	0	—	6	
	MOV.B Rs, @aa:8	B						2				Rs8 → @aa:8	—	—	↕	↕	0	—	4	
	MOV.B Rs, @aa:16	B						4				Rs8 → @aa:16	—	—	↕	↕	0	—	6	
	MOV.B Rs, @aa:24	B						6				Rs8 → @aa:24	—	—	↕	↕	0	—	8	
	MOV.W #xx:16, Rd	W	4									#xx:16 → Rd16	—	—	↕	↕	0	—	4	
	MOV.W Rs, Rd	W		2								Rs16 → Rd16	—	—	↕	↕	0	—	2	
	MOV.W @ERs, Rd	W			2							@ERs → Rd16	—	—	↕	↕	0	—	4	
	MOV.W @(d:16, ERs), Rd	W				4						@(d:16, ERs) → Rd16	—	—	↕	↕	0	—	6	
	MOV.W @(d:24, ERs), Rd	W				8						@(d:24, ERs) → Rd16	—	—	↕	↕	0	—	10	
	MOV.W @ERs+, Rd	W					2					@ERs → Rd16 ERs32+2 → @ERd32	—	—	↕	↕	0	—	6	
	MOV.W @aa:16, Rd	W						4				@aa:16 → Rd16	—	—	↕	↕	0	—	6	
	MOV.W @aa:24, Rd	W						6				@aa:24 → Rd16	—	—	↕	↕	0	—	8	
	MOV.W Rs, @ERd	W			2							Rs16 → @ERd	—	—	↕	↕	0	—	4	
	MOV.W Rs, @(d:16, ERd)	W				4						Rs16 → @(d:16, ERd)	—	—	↕	↕	0	—	6	
	MOV.W Rs, @(d:24, ERd)	W				8						Rs16 → @(d:24, ERd)	—	—	↕	↕	0	—	10	

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
MOV	MOV.W Rs, @–ERd	W					2					ERd32–2 → ERd32 Rs16 → @ERd	—	—	↕	↕	0	—	6	
	MOV.W Rs, @aa:16	W						4				Rs16 → @aa:16	—	—	↕	↕	0	—	6	
	MOV.W Rs, @aa:24	W						6				Rs16 → @aa:24	—	—	↕	↕	0	—	8	
	MOV.L #xx:32, Rd	L	6									#xx:32 → Rd32	—	—	↕	↕	0	—	6	
	MOV.L ERs, ERd	L		2								ERs32 → ERd32	—	—	↕	↕	0	—	2	
	MOV.L @ERs, ERd	L			4							@ERs → ERd32	—	—	↕	↕	0	—	8	
	MOV.L @(d:16, ERs), ERd	L				6						@(d:16, ERs) → ERd32	—	—	↕	↕	0	—	10	
	MOV.L @(d:24, ERs), ERd	L				10						@(d:24, ERs) → ERd32	—	—	↕	↕	0	—	14	
	MOV.L @ERs+, ERd	L					4					@ERs → ERd32 ERs32+4 → ERs32	—	—	↕	↕	0	—	10	
	MOV.L @aa:16, ERd	L						6				@aa:16 → ERd32	—	—	↕	↕	0	—	10	
	MOV.L @aa:24, ERd	L						8				@aa:24 → ERd32	—	—	↕	↕	0	—	12	
	MOV.L ERs, @ERd	L			4							ERs32 → @ERd	—	—	↕	↕	0	—	8	
	MOV.L ERs, @(d:16, ERd)	L				6						ERs32 → @(d:16, ERd)	—	—	↕	↕	0	—	10	
	MOV.L ERs, @(d:24, ERd)	L				10						ERs32 → @(d:24, ERd)	—	—	↕	↕	0	—	14	
	MOV.L ERs, @–ERd	L					4					ERd32–4 → ERd32 ERs32 → @ERd	—	—	↕	↕	0	—	10	
	MOV.L ERs, @aa:16	L						6				ERs32 → @aa:16	—	—	↕	↕	0	—	10	
	MOV.L ERs, @aa:24	L						8				ERs32 → @aa:24	—	—	↕	↕	0	—	12	
POP	POP.W Rn	W									2	@SP → Rn16 SP+2 → SP	—	—	↕	↕	0	—	6	
	POP.L ERn	L									4	@SP → ERn32 SP+4 → SP	—	—	↕	↕	0	—	10	
PUSH	PUSH.W Rn	W									2	SP–2 → SP Rn16 → @SP	—	—	↕	↕	0	—	6	
	PUSH.L ERn	L									4	SP–4 → SP ERn32 → @SP	—	—	↕	↕	0	—	10	
MOVFPE	MOVFPE @aa:16, Rd	B						4				Cannot be used in this LSI	Cannot be used in this LSI							
MOVTPE	MOVTPE Rs, @aa:16	B						4				Cannot be used in this LSI	Cannot be used in this LSI							

2. Arithmetic Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
ADD	ADD.B #xx:8, Rd	B	2									Rd8+#xx:8 → Rd8	—	↕	↕	↕	↕	↕	2	
	ADD.B Rs, Rd	B		2								Rd8+Rs8 → Rd8	—	↕	↕	↕	↕	↕	2	
	ADD.W #xx:16, Rd	W	4									Rd16+#xx:16 → Rd16	—	(1)	↕	↕	↕	↕	4	
	ADD.W Rs, Rd	W		2								Rd16+Rs16 → Rd16	—	(1)	↕	↕	↕	↕	2	
	ADD.L #xx:32, ERd	L	6									ERd32+#xx:32 → ERd32	—	(2)	↕	↕	↕	↕	6	
	ADD.L ERs, ERd	L		2								ERd32+ERs32 → ERd32	—	(2)	↕	↕	↕	↕	2	
ADDX	ADDX.B #xx:8, Rd	B	2									Rd8+#xx:8 +C → Rd8	—	↕	↕	(3)	↕	↕	2	
	ADDX.B Rs, Rd	B		2								Rd8+Rs8 +C → Rd8	—	↕	↕	(3)	↕	↕	2	
ADDS	ADDS.L #1, ERd	L		2								ERd32+1 → ERd32	—	—	—	—	—	—	2	
	ADDS.L #2, ERd	L		2								ERd32+2 → ERd32	—	—	—	—	—	—	2	
	ADDS.L #4, ERd	L		2								ERd32+4 → ERd32	—	—	—	—	—	—	2	
INC	INC.B Rd	B		2								Rd8+1 → Rd8	—	—	↕	↕	↕	—	2	
	INC.W #1, Rd	W		2								Rd16+1 → Rd16	—	—	↕	↕	↕	—	2	
	INC.W #2, Rd	W		2								Rd16+2 → Rd16	—	—	↕	↕	↕	—	2	
	INC.L #1, ERd	L		2								ERd32+1 → ERd32	—	—	↕	↕	↕	—	2	
	INC.L #2, ERd	L		2								ERd32+2 → ERd32	—	—	↕	↕	↕	—	2	
DAA	DAA Rd	B		2								Rd8 decimal adjust → Rd8	—	*	↕	↕	*	—	2	
SUB	SUB.B Rs, Rd	B		2								Rd8–Rs8 → Rd8	—	↕	↕	↕	↕	↕	2	
	SUB.W #xx:16, Rd	W	4									Rd16–#xx:16 → Rd16	—	(1)	↕	↕	↕	↕	4	
	SUB.W Rs, Rd	W		2								Rd16–Rs16 → Rd16	—	(1)	↕	↕	↕	↕	2	
	SUB.L #xx:32, ERd	L	6									ERd32–#xx:32 → ERd32	—	(2)	↕	↕	↕	↕	6	
	SUB.L ERs, ERd	L		2								ERd32–ERs32 → ERd32	—	(2)	↕	↕	↕	↕	2	
SUBX	SUBX.B #xx:8, Rd	B	2									Rd8–#xx:8–C → Rd8	—	↕	↕	(3)	↕	↕	2	
	SUBX.B Rs, Rd	B		2								Rd8–Rs8–C → Rd8	—	↕	↕	(3)	↕	↕	2	
SUBS	SUBS.L #1, ERd	L		2								ERd32–1 → ERd32	—	—	—	—	—	—	2	
	SUBS.L #2, ERd	L		2								ERd32–2 → ERd32	—	—	—	—	—	—	2	
	SUBS.L #4, ERd	L		2								ERd32–4 → ERd32	—	—	—	—	—	—	2	
DEC	DEC.B Rd	B		2								Rd8–1 → Rd8	—	—	↕	↕	↕	—	2	
	DEC.W #1, Rd	W		2								Rd16–1 → Rd16	—	—	↕	↕	↕	—	2	
	DEC.W #2, Rd	W		2								Rd16–2 → Rd16	—	—	↕	↕	↕	—	2	

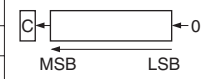
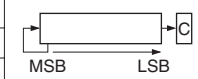
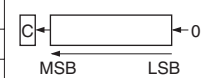
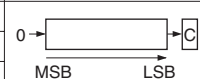
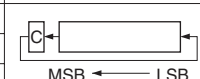
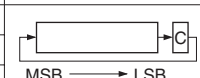
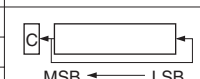
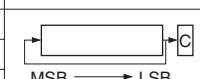
Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
DEC	DEC.L #1, ERd	L		2								ERd32–1 → ERd32	—	—	↕	↕	↕	—	2	
	DEC.L #2, ERd	L		2								ERd32–2 → ERd32	—	—	↕	↕	↕	—	2	
DAS	DAS.Rd	B		2								Rd8 decimal adjust → Rd8	—	*	↕	↕	*	—	2	
MULXU	MULXU. B Rs, Rd	B		2								Rd8 × Rs8 → Rd16 (unsigned multiplication)	—	—	—	—	—	—	14	
	MULXU. W Rs, ERd	W		2								Rd16 × Rs16 → ERd32 (unsigned multiplication)	—	—	—	—	—	—	22	
MULXS	MULXS. B Rs, Rd	B		4								Rd8 × Rs8 → Rd16 (signed multiplication)	—	—	↕	↕	—	—	16	
	MULXS. W Rs, ERd	W		4								Rd16 × Rs16 → ERd32 (signed multiplication)	—	—	↕	↕	—	—	24	
DIVXU	DIVXU. B Rs, Rd	B		2								Rd16 ÷ Rs8 → Rd16 (RdH: remainder, RdL: quotient) (unsigned division)	—	—	(6)	(7)	—	—	14	
	DIVXU. W Rs, ERd	W		2								ERd32 ÷ Rs16 → ERd32 (Ed: remainder, Rd: quotient) (unsigned division)	—	—	(6)	(7)	—	—	22	
DIVXS	DIVXS. B Rs, Rd	B		4								Rd16 ÷ Rs8 → Rd16 (RdH: remainder, RdL: quotient) (signed division)	—	—	(8)	(7)	—	—	16	
	DIVXS. W Rs, ERd	W		4								ERd32 ÷ Rs16 → ERd32 (Ed: remainder, Rd: quotient) (signed division)	—	—	(8)	(7)	—	—	24	
CMP	CMP.B #xx:8, Rd	B	2									Rd8–#xx:8	—	↕	↕	↕	↕	↕	2	
	CMP.B Rs, Rd	B		2								Rd8–Rs8	—	↕	↕	↕	↕	↕	2	
	CMP.W #xx:16, Rd	W	4									Rd16–#xx:16	—	(1)	↕	↕	↕	↕	4	
	CMP.W Rs, Rd	W		2								Rd16–Rs16	—	(1)	↕	↕	↕	↕	2	
	CMP.L #xx:32, ERd	L	6									ERd32–#xx:32	—	(2)	↕	↕	↕	↕	4	
	CMP.L ERs, ERd	L		2								ERd32–ERs32	—	(2)	↕	↕	↕	↕	2	

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
NEG	NEG.B Rd	B		2								0–Rd8 → Rd8	—	↕	↕	↕	↕	↕	2	
	NEG.W Rd	W		2								0–Rd16 → Rd16	—	↕	↕	↕	↕	↕	2	
	NEG.L ERd	L		2								0–ERd32 → ERd32	—	↕	↕	↕	↕	↕	2	
EXTU	EXTU.W Rd	W		2								0 → (<bits 15 to 8> of Rd16)	—	—	0	↕	0	—	2	
	EXTU.L ERd	L		2								0 → (<bits 31 to 16> of ERd32)	—	—	0	↕	0	—	2	
EXTS	EXTS.W Rd	W		2								(<bit 7> of Rd16) → (<bits 15 to 8> of Rd16)	—	—	↕	↕	0	—	2	
	EXTS.L ERd	L		2								(<bit 15> of ERd32) → (<bits 31 to 16> of ERd32)	—	—	↕	↕	0	—	2	

3. Logic Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
AND	AND.B #xx:8, Rd	B	2									Rd8∧#xx:8 → Rd8	—	—	↕	↕	0	—	2	
	AND.B Rs, Rd	B		2								Rd8∧Rs8 → Rd8	—	—	↕	↕	0	—	2	
	AND.W #xx:16, Rd	W	4									Rd16∧#xx:16 → Rd16	—	—	↕	↕	0	—	4	
	AND.W Rs, Rd	W		2								Rd16∧Rs16 → Rd16	—	—	↕	↕	0	—	2	
	AND.L #xx:32, ERd	L	6									ERd32∧#xx:32 → ERd32	—	—	↕	↕	0	—	6	
	AND.L ERs, ERd	L		4								ERd32∧ERs32 → ERd32	—	—	↕	↕	0	—	4	
OR	OR.B #xx:8, Rd	B	2									Rd8∨#xx:8 → Rd8	—	—	↕	↕	0	—	2	
	OR.B Rs, Rd	B		2								Rd8∨Rs8 → Rd8	—	—	↕	↕	0	—	2	
	OR.W #xx:16, Rd	W	4									Rd16∨#xx:16 → Rd16	—	—	↕	↕	0	—	4	
	OR.W Rs, Rd	W		2								Rd16∨Rs16 → Rd16	—	—	↕	↕	0	—	2	
	OR.L #xx:32, ERd	L	6									ERd32∨#xx:32 → ERd32	—	—	↕	↕	0	—	6	
	OR.L ERs, ERd	L		4								ERd32∨ERs32 → ERd32	—	—	↕	↕	0	—	4	
XOR	XOR.B #xx:8, Rd	B	2									Rd8⊕#xx:8 → Rd8	—	—	↕	↕	0	—	2	
	XOR.B Rs, Rd	B		2								Rd8⊕Rs8 → Rd8	—	—	↕	↕	0	—	2	
	XOR.W #xx:16, Rd	W	4									Rd16⊕#xx:16 → Rd16	—	—	↕	↕	0	—	4	
	XOR.W Rs, Rd	W		2								Rd16⊕Rs16 → Rd16	—	—	↕	↕	0	—	2	
	XOR.L #xx:32, ERd	L	6									ERd32⊕#xx:32 → ERd32	—	—	↕	↕	0	—	6	
	XOR.L ERs, ERd	L		4								ERd32⊕ERs32 → ERd32	—	—	↕	↕	0	—	4	
NOT	NOT.B Rd	B		2								¬ Rd8 → Rd8	—	—	↕	↕	0	—	2	
	NOT.W Rd	W		2								¬ Rd16 → Rd16	—	—	↕	↕	0	—	2	
	NOT.L ERd	L		2								¬ Rd32 → Rd32	—	—	↕	↕	0	—	2	

4. Shift Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
SHAL	SHAL.B Rd	B		2									—	—	↕	↕	↕	↕	2	
	SHAL.W Rd	W		2									—	—	↕	↕	↕	↕	2	
	SHAL.L ERd	L		2									—	—	↕	↕	↕	↕	2	
SHAR	SHAR.B Rd	B		2									—	—	↕	↕	0	↕	2	
	SHAR.W Rd	W		2									—	—	↕	↕	0	↕	2	
	SHAR.L ERd	L		2									—	—	↕	↕	0	↕	2	
SHLL	SHLL.B Rd	B		2									—	—	↕	↕	0	↕	2	
	SHLL.W Rd	W		2									—	—	↕	↕	0	↕	2	
	SHLL.L ERd	L		2									—	—	↕	↕	0	↕	2	
SHLR	SHLR.B Rd	B		2									—	—	↕	↕	0	↕	2	
	SHLR.W Rd	W		2									—	—	↕	↕	0	↕	2	
	SHLR.L ERd	L		2									—	—	↕	↕	0	↕	2	
ROTXL	ROTXL.B Rd	B		2									—	—	↕	↕	0	↕	2	
	ROTXL.W Rd	W		2									—	—	↕	↕	0	↕	2	
	ROTXL.L ERd	L		2									—	—	↕	↕	0	↕	2	
ROTXR	ROTXR.B Rd	B		2									—	—	↕	↕	0	↕	2	
	ROTXR.W Rd	W		2									—	—	↕	↕	0	↕	2	
	ROTXR.L ERd	L		2									—	—	↕	↕	0	↕	2	
ROTL	ROTL.B Rd	B		2									—	—	↕	↕	0	↕	2	
	ROTL.W Rd	W		2									—	—	↕	↕	0	↕	2	
	ROTL.L ERd	L		2									—	—	↕	↕	0	↕	2	
ROTR	ROTR.B Rd	B		2									—	—	↕	↕	0	↕	2	
	ROTR.W Rd	W		2									—	—	↕	↕	0	↕	2	
	ROTR.L ERd	L		2									—	—	↕	↕	0	↕	2	

5. Bit-Manipulation Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
BSET	BSET #xx:3, Rd	B		2								(#xx:3 of Rd8) ← 1	—	—	—	—	—	—	2	
	BSET #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← 1	—	—	—	—	—	—	8	
	BSET #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← 1	—	—	—	—	—	—	8	
	BSET Rn, Rd	B		2								(Rn8 of Rd8) ← 1	—	—	—	—	—	—	2	
	BSET Rn, @ERd	B			4							(Rn8 of @ERd) ← 1	—	—	—	—	—	—	8	
	BSET Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 1	—	—	—	—	—	—	8	
BCLR	BCLR #xx:3, Rd	B		2								(#xx:3 of Rd8) ← 0	—	—	—	—	—	—	2	
	BCLR #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← 0	—	—	—	—	—	—	8	
	BCLR #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← 0	—	—	—	—	—	—	8	
	BCLR Rn, Rd	B		2								(Rn8 of Rd8) ← 0	—	—	—	—	—	—	2	
	BCLR Rn, @ERd	B			4							(Rn8 of @ERd) ← 0	—	—	—	—	—	—	8	
	BCLR Rn, @aa:8	B						4				(Rn8 of @aa:8) ← 0	—	—	—	—	—	—	8	
BNOT	BNOT #xx:3, Rd	B		2								(#xx:3 of Rd8) ← ¬ (#xx:3 of Rd8)	—	—	—	—	—	—	2	
	BNOT #xx:3, @ERd	B			4							(#xx:3 of @ERd) ← ¬ (#xx:3 of @ERd)	—	—	—	—	—	—	8	
	BNOT #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) ← ¬ (#xx:3 of @aa:8)	—	—	—	—	—	—	8	
	BNOT Rn, Rd	B		2								(Rn8 of Rd8) ← ¬ (Rn8 of Rd8)	—	—	—	—	—	—	2	
	BNOT Rn, @ERd	B			4							(Rn8 of @ERd) ← ¬ (Rn8 of @ERd)	—	—	—	—	—	—	8	
	BNOT Rn, @aa:8	B						4				(Rn8 of @aa:8) ← ¬ (Rn8 of @aa:8)	—	—	—	—	—	—	8	
BTST	BTST #xx:3, Rd	B		2								¬ (#xx:3 of Rd8) → Z	—	—	—	↕	—	—	2	
	BTST #xx:3, @ERd	B			4							¬ (#xx:3 of @ERd) → Z	—	—	—	↕	—	—	6	
	BTST #xx:3, @aa:8	B						4				¬ (#xx:3 of @aa:8) → Z	—	—	—	↕	—	—	6	
	BTST Rn, Rd	B		2								¬ (Rn8 of @Rd8) → Z	—	—	—	↕	—	—	2	
	BTST Rn, @ERd	B			4							¬ (Rn8 of @ERd) → Z	—	—	—	↕	—	—	6	
	BTST Rn, @aa:8	B						4				¬ (Rn8 of @aa:8) → Z	—	—	—	↕	—	—	6	
BLD	BLD #xx:3, Rd	B		2								(#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
BLD	BLD #xx:3, @ERd	B			4							(#xx:3 of @ERd) → C	—	—	—	—	—	↕	6	
	BLD #xx:3, @aa:8	B						4				(#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BILD	BILD #xx:3, Rd	B		2								¬ (#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BILD #xx:3, @ERd	B			4							¬ (#xx:3 of @ERd) → C	—	—	—	—	—	↕	6	
	BILD #xx:3, @aa:8	B						4				¬ (#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BST	BST #xx:3, Rd	B		2								C → (#xx:3 of Rd8)	—	—	—	—	—	—	2	
	BST #xx:3, @ERd	B			4							C → (#xx:3 of @ERd24)	—	—	—	—	—	—	8	
	BST #xx:3, @aa:8	B						4				C → (#xx:3 of @aa:8)	—	—	—	—	—	—	8	
BIST	BIST #xx:3, Rd	B		2								¬ C → (#xx:3 of Rd8)	—	—	—	—	—	—	2	
	BIST #xx:3, @ERd	B			4							¬ C → (#xx:3 of @ERd24)	—	—	—	—	—	—	8	
	BIST #xx:3, @aa:8	B						4				¬ C → (#xx:3 of @aa:8)	—	—	—	—	—	—	8	
BAND	BAND #xx:3, Rd	B		2								C∧(#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BAND #xx:3, @ERd	B			4							C∧(#xx:3 of @ERd24) → C	—	—	—	—	—	↕	6	
	BAND #xx:3, @aa:8	B						4				C∧(#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BIAND	BIAND #xx:3, Rd	B		2								C∧ ¬ (#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BIAND #xx:3, @ERd	B			4							C∧ ¬ (#xx:3 of @ERd24) → C	—	—	—	—	—	↕	6	
	BIAND #xx:3, @aa:8	B						4				C∧ ¬ (#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BOR	BOR #xx:3, Rd	B		2								C∨(#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BOR #xx:3, @ERd	B			4							C∨(#xx:3 of @ERd24) → C	—	—	—	—	—	↕	6	
	BOR #xx:3, @aa:8	B						4				C∨(#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BIOR	BIOR #xx:3, Rd	B		2								C∨ ¬ (#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BIOR #xx:3, @ERd	B			4							C∨ ¬ (#xx:3 of @ERd24) → C	—	—	—	—	—	↕	6	
	BIOR #xx:3, @aa:8	B						4				C∨ ¬ (#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BXOR	BXOR #xx:3, Rd	B		2								C⊕(#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BXOR #xx:3, @ERd	B			4							C⊕(#xx:3 of @ERd24) → C	—	—	—	—	—	↕	6	
	BXOR #xx:3, @aa:8	B						4				C⊕(#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	
BIXOR	BIXOR #xx:3, Rd	B		2								C⊕ ¬ (#xx:3 of Rd8) → C	—	—	—	—	—	↕	2	
	BIXOR #xx:3, @ERd	B			4							C⊕ ¬ (#xx:3 of @ERd24) → C	—	—	—	—	—	↕	6	
	BIXOR #xx:3, @aa:8	B						4				C⊕ ¬ (#xx:3 of @aa:8) → C	—	—	—	—	—	↕	6	

6. Branching Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		Branch Condition	I	H	N	Z	V	C	Normal	Advanced
Bcc	BRA d:8 (BT d:8)	—							2			If condition is true then PC ← PC+d else next;	Always	—	—	—	—	—	—	4	
	BRA d:16 (BT d:16)	—							4					—	—	—	—	—	—	6	
	BRN d:8 (BF d:8)	—							2				Never	—	—	—	—	—	—	4	
	BRN d:16 (BF d:16)	—							4					—	—	—	—	—	—	6	
	BHI d:8	—							2				C∨ Z = 0	—	—	—	—	—	—	4	
	BHI d:16	—							4					—	—	—	—	—	—	6	
	BLS d:8	—							2				C∨ Z = 1	—	—	—	—	—	—	4	
	BLS d:16	—							4					—	—	—	—	—	—	6	
	BCC d:8 (BHS d:8)	—							2				C = 0	—	—	—	—	—	—	4	
	BCC d:16 (BHS d:16)	—							4					—	—	—	—	—	—	6	
	BCS d:8 (BLO d:8)	—							2				C = 1	—	—	—	—	—	—	4	
	BCS d:16 (BLO d:16)	—							4					—	—	—	—	—	—	6	
	BNE d:8	—							2				Z = 0	—	—	—	—	—	—	4	
	BNE d:16	—							4					—	—	—	—	—	—	6	
	BEQ d:8	—							2				Z = 1	—	—	—	—	—	—	4	
	BEQ d:16	—							4					—	—	—	—	—	—	6	
	BVC d:8	—							2				V = 0	—	—	—	—	—	—	4	
	BVC d:16	—							4					—	—	—	—	—	—	6	
	BVS d:8	—							2				V = 1	—	—	—	—	—	—	4	
	BVS d:16	—							4					—	—	—	—	—	—	6	
	BPL d:8	—							2				N = 0	—	—	—	—	—	—	4	
	BPL d:16	—							4					—	—	—	—	—	—	6	
	BMI d:8	—							2				N = 1	—	—	—	—	—	—	4	
	BMI d:16	—							4					—	—	—	—	—	—	6	
	BGE d:8	—							2				N⊕V = 0	—	—	—	—	—	—	4	
	BGE d:16	—							4					—	—	—	—	—	—	6	
	BLT d:8	—							2				N⊕V = 1	—	—	—	—	—	—	4	
	BLT d:16	—							4					—	—	—	—	—	—	6	
	BGT d:8	—							2				Z∨ (N⊕V) = 0	—	—	—	—	—	—	4	
	BGT d:16	—							4					—	—	—	—	—	—	6	
	BLE d:8	—							2				Z∨ (N⊕V) = 1	—	—	—	—	—	—	4	
	BLE d:16	—							4					—	—	—	—	—	—	6	

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
JMP	JMP @ERn	—			2							PC ← ERn	—	—	—	—	—	—	4	
	JMP @aa:24	—						4				PC ← aa:24	—	—	—	—	—	—	6	
	JMP @@aa:8	—								2		PC ← @aa:8	—	—	—	—	—	—	8	10
BSR	BSR d:8	—							2			PC → @–SP PC ← PC+d:8	—	—	—	—	—	—	6	8
	BSR d:16	—							4			PC → @–SP PC ← PC+d:16	—	—	—	—	—	—	8	10
JSR	JSR @ERn	—			2							PC → @–SP PC ← ERn	—	—	—	—	—	—	6	8
	JSR @aa:24	—						4				PC → @–SP PC ← aa:24	—	—	—	—	—	—	8	10
	JSR @@aa:8	—								2		PC → @–SP PC ← @aa:8	—	—	—	—	—	—	8	12
RTS	RTS	—									2	PC ← @SP+	—	—	—	—	—	—	8	10

7. System Control Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
TRAPA	TRAPA #x:2	—									2	PC → @–SP CCR → @–SP <vector> → PC	1	—	—	—	—	—	14	16
RTE	RTE	—										CCR ← @SP+ PC ← @SP+	↕	↕	↕	↕	↕	↕	10	
SLEEP	SLEEP	—										Transition to power-down state	—	—	—	—	—	—	2	
LDC	LDC #xx:8, CCR	B	2									#xx:8 → CCR	↕	↕	↕	↕	↕	↕	2	
	LDC Rs, CCR	B		2								Rs8 → CCR	↕	↕	↕	↕	↕	↕	2	
	LDC @ERs, CCR	W			4							@ERs → CCR	↕	↕	↕	↕	↕	↕	6	
	LDC @(d:16, ERs), CCR	W				6						@(d:16, ERs) → CCR	↕	↕	↕	↕	↕	↕	8	
	LDC @(d:24, ERs), CCR	W				10						@(d:24, ERs) → CCR	↕	↕	↕	↕	↕	↕	12	
	LDC @ERs+, CCR	W					4					@ERs → CCR ERs32+2 → ERs32	↕	↕	↕	↕	↕	↕	8	
	LDC @aa:16, CCR	W						6				@aa:16 → CCR	↕	↕	↕	↕	↕	↕	8	
	LDC @aa:24, CCR	W						8				@aa:24 → CCR	↕	↕	↕	↕	↕	↕	10	
STC	STC CCR, Rd	B		2								CCR → Rd8	—	—	—	—	—	—	2	
	STC CCR, @ERd	W			4							CCR → @ERd	—	—	—	—	—	—	6	
	STC CCR, @(d:16, ERd)	W				6						CCR → @(d:16, ERd)	—	—	—	—	—	—	8	
	STC CCR, @(d:24, ERd)	W				10						CCR → @(d:24, ERd)	—	—	—	—	—	—	12	
	STC CCR, @–ERd	W					4					ERd32–2 → ERd32 CCR → @ERd	—	—	—	—	—	—	8	
	STC CCR, @aa:16	W						6				CCR → @aa:16	—	—	—	—	—	—	8	
	STC CCR, @aa:24	W						8				CCR → @aa:24	—	—	—	—	—	—	10	
ANDC	ANDC #xx:8, CCR	B	2									CCR∧#xx:8 → CCR	↕	↕	↕	↕	↕	↕	2	
ORC	ORC #xx:8, CCR	B	2									CCR∨#xx:8 → CCR	↕	↕	↕	↕	↕	↕	2	
XORC	XORC #xx:8, CCR	B	2									CCR⊕#xx:8 → CCR	↕	↕	↕	↕	↕	↕	2	
NOP	NOP	—									2	PC ← PC+2	—	—	—	—	—	—	2	

8. Block Transfer Instructions

Mnemonic		Operand Size	Addressing Mode and Instruction Length (bytes)									Operation	Condition Code						No. of States*1	
			#xx	Rn	@ERn	@(d, ERn)	@–ERn/@ERn+	@aa	@(d, PC)	@@aa	—		I	H	N	Z	V	C	Normal	Advanced
EEPMOV	EEPMOV. B	—									4	if R4L ≠ 0 then repeat @R5 → @R6 R5+1 → R5 R6+1 → R6 R4L–1 → R4L until R4L=0 else next	—	—	—	—	—	—	8+ 4n*2	
	EEPMOV. W	—									4	if R4 ≠ 0 then repeat @R5 → @R6 R5+1 → R5 R6+1 → R6 R4–1 → R4 until R4=0 else next	—	—	—	—	—	—	8+ 4n*2	

Notes: 1. The number of states in cases where the instruction code and its operands are located in on-chip memory is shown here. For other cases see appendix A.3, Number of Execution States.

2. n is the value set in register R4L or R4.

(1) Set to 1 when a carry or borrow occurs at bit 11; otherwise cleared to 0.

(2) Set to 1 when a carry or borrow occurs at bit 27; otherwise cleared to 0.

(3) Retains its previous value when the result is zero; otherwise cleared to 0.

(4) Set to 1 when the adjustment produces a carry; otherwise retains its previous value.

(5) The number of states required for execution of an instruction that transfers data in synchronization with the E clock is variable.

(6) Set to 1 when the divisor is negative; otherwise cleared to 0.

(7) Set to 1 when the divisor is zero; otherwise cleared to 0.

(8) Set to 1 when the quotient is negative; otherwise cleared to 0.

A.2 Operation Code Map

Table A.2 Operation Code Map (1)

Instruction code:

1st byte	2nd byte
AH	AL	BH	BL

Instruction when most significant bit of BH is 0.

Instruction when most significant bit of BH is 1.

AH \ AL	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NOP	Table A.2 (2)	STC	LDC	ORC	XORC	ANDC	LDC	ADD		Table A.2 (2)	Table A.2 (2)	MOV		ADDX	Table A.2 (2)
1	Table A.2 (2)	Table A.2 (2)	Table A.2 (2)	Table A.2 (2)	OR.B	XOR.B	AND.B	Table A.2 (2)	SUB		Table A.2 (2)	Table A.2 (2)	CMP		SUBX	Table A.2 (2)
2	MOV.B															
3																
4	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE	BLT	BGT	BLE
5	MULXU	DIVXU	MULXU	DIVXU	RTS	BSR	RTE	TRAPA	Table A.2 (2)	JMP			BSR	JSR		
6	BSET	BNOT	BCLR	BTST	OR	XOR	AND	BST / BIST	MOV				MOV			
7					BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BLD / BILD		Table A.2 (2)	Table A.2 (2)	EEPMOV	Table A.2 (3)			
8	ADD															
9	ADDX															
A	CMP															
B	SUBX															
C	OR															
D	XOR															
E	AND															
F	MOV															

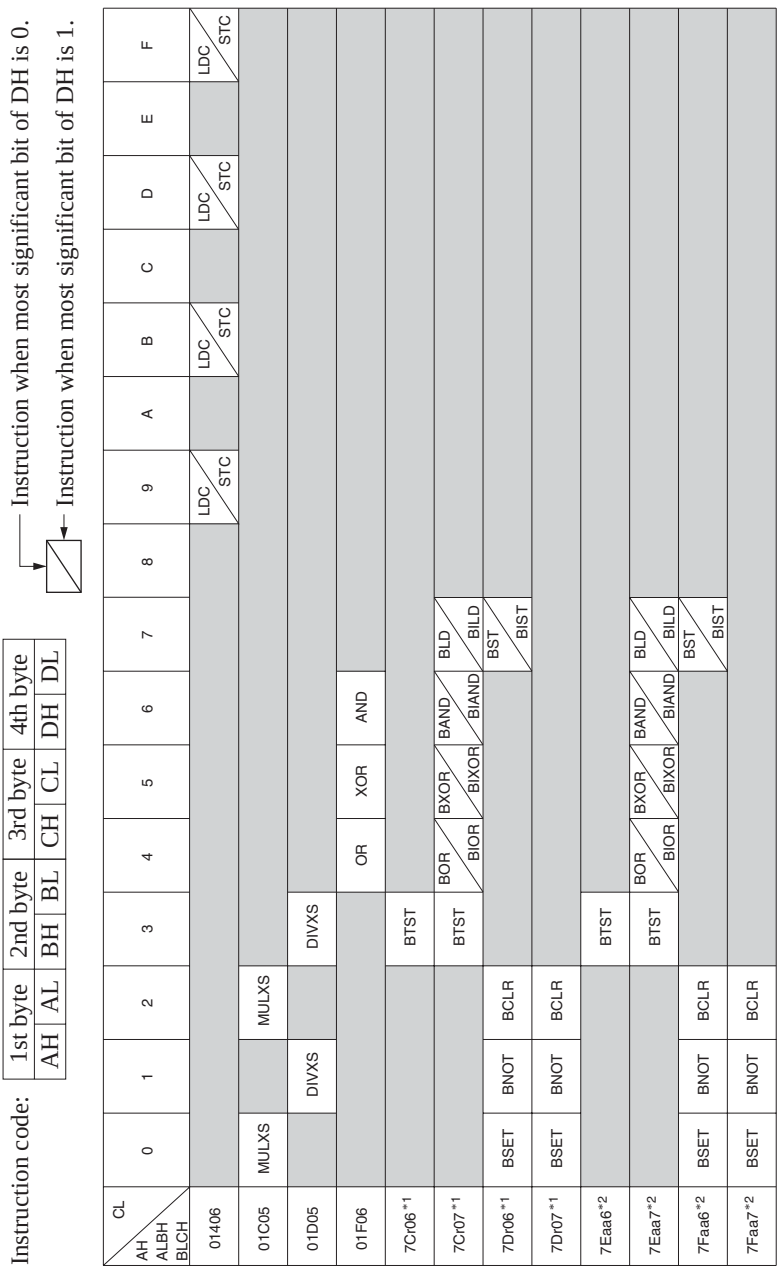
Table A.2 Operation Code Map (2)

Instruction code:

1st byte		2nd byte	
AH	AL	BH	BL

AH AL \ BH	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
01	MOV				LDC/STC				SLEEP				Table A.2 (3)	Table A.2 (3)		Table A.2 (3)
0A	INC												ADD			
0B	ADDS					INC		INC	ADDS					INC		INC
0F	DAA												MOV			
10	SHLL	SHLL		SHLL					SHAL	SHAL		SHAL				
11	SHLR	SHLR		SHLR					SHAR	SHAR		SHAR				
12	ROTXL	ROTXL		ROTXL					ROTL	ROTL		ROTL				
13	ROTXR	ROTXR		ROTXR					ROTR	ROTR		ROTR				
17	NOT	NOT		NOT		EXTU		EXTU	NEG	NEG		NEG		EXTS		EXTS
1A	DEC												SUB			
1B	SUBS					DEC		DEC	SUB	SUB				DEC		DEC
1F	DAS												CMP			
58	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE	BLT	BGT	BLE
79	MOV	ADD	CMP	SUB	OR	XOR	AND									
7A	MOV	ADD	CMP	SUB	OR	XOR	AND									

Table A.2 Operation Code Map (3)



Notes: 1. r is the register designation field.
2. aa is the absolute address field.

A.3 Number of Execution States

The status of execution for each instruction of the H8/300H CPU and the method of calculating the number of states required for instruction execution are shown below. Table A.4 shows the number of cycles of each type occurring in each instruction, such as instruction fetch and data read/write. Table A.3 shows the number of states required for each cycle. The total number of states required for execution of an instruction can be calculated by the following expression:

$$\text{Execution states} = I \times S_I + J \times S_J + K \times S_K + L \times S_L + M \times S_M + N \times S_N$$

Examples: When instruction is fetched from on-chip ROM, and an on-chip RAM is accessed.

BSET #0, @FF00

From table A.4:
I = L = 2, J = K = M = N= 0

From table A.3:
$S_I = 2$, $S_L = 2$

Number of states required for execution = $2 \times 2 + 2 \times 2 = 8$

When instruction is fetched from on-chip ROM, branch address is read from on-chip ROM, and on-chip RAM is used for stack area.

JSR @@ 30

From table A.4:
I = 2, J = K = 1, L = M = N = 0

From table A.3:
$S_I = S_J = S_K = 2$

Number of states required for execution = $2 \times 2 + 1 \times 2 + 1 \times 2 = 8$

Table A.3 Number of Cycles in Each Instruction

Execution Status (Instruction Cycle)		Access Location	
		On-Chip Memory	On-Chip Peripheral Module
Instruction fetch	S_I	2	—
Branch address read	S_J		
Stack operation	S_K		
Byte data access	S_L		2 or 3*
Word data access	S_M		2 or 3*
Internal operation	S_N	1	

Note: * Depends on which on-chip peripheral module is accessed. See section 22.1, Register Addresses (Address Order).

Table A.4 Number of Cycles in Each Instruction

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
ADD	ADD.B #xx:8, Rd	1					
	ADD.B Rs, Rd	1					
	ADD.W #xx:16, Rd	2					
	ADD.W Rs, Rd	1					
	ADD.L #xx:32, ERd	3					
	ADD.L ERs, ERd	1					
ADDS	ADDS #1/2/4, ERd	1					
ADDX	ADDX #xx:8, Rd	1					
	ADDX Rs, Rd	1					
AND	AND.B #xx:8, Rd	1					
	AND.B Rs, Rd	1					
	AND.W #xx:16, Rd	2					
	AND.W Rs, Rd	1					
	AND.L #xx:32, ERd	3					
	AND.L ERs, ERd	2					
ANDC	ANDC #xx:8, CCR	1					
BAND	BAND #xx:3, Rd	1					
	BAND #xx:3, @ERd	2			1		
	BAND #xx:3, @aa:8	2			1		
Bcc	BRA d:8 (BT d:8)	2					
	BRN d:8 (BF d:8)	2					
	BHI d:8	2					
	BLS d:8	2					
	BCC d:8 (BHS d:8)	2					
	BCS d:8 (BLO d:8)	2					
	BNE d:8	2					
	BEQ d:8	2					
	BVC d:8	2					
	BVS d:8	2					
	BPL d:8	2					
	BMI d:8	2					
	BGE d:8	2					

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
Bcc	BLT d:8	2					
	BGT d:8	2					
	BLE d:8	2					
	BRA d:16(BT d:16)	2					2
	BRN d:16(BF d:16)	2					2
	BHI d:16	2					2
	BLS d:16	2					2
	BCC d:16(BHS d:16)	2					2
	BCS d:16(BLO d:16)	2					2
	BNE d:16	2					2
	BEQ d:16	2					2
	BVC d:16	2					2
	BVS d:16	2					2
	BPL d:16	2					2
	BMI d:16	2					2
	BGE d:16	2					2
	BLT d:16	2					2
	BGT d:16	2					2
	BLE d:16	2					2
BCLR	BCLR #xx:3, Rd	1					
	BCLR #xx:3, @ERd	2			2		
	BCLR #xx:3, @aa:8	2			2		
	BCLR Rn, Rd	1					
	BCLR Rn, @ERd	2			2		
	BCLR Rn, @aa:8	2			2		
BIAND	BIAND #xx:3, Rd	1					
	BIAND #xx:3, @ERd	2			1		
	BIAND #xx:3, @aa:8	2			1		
BILD	BILD #xx:3, Rd	1					
	BILD #xx:3, @ERd	2			1		
	BILD #xx:3, @aa:8	2			1		

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
BIOR	BIOR #xx:8, Rd	1					
	BIOR #xx:8, @ERd	2			1		
	BIOR #xx:8, @aa:8	2			1		
BIST	BIST #xx:3, Rd	1					
	BIST #xx:3, @ERd	2			2		
	BIST #xx:3, @aa:8	2			2		
BIXOR	BIXOR #xx:3, Rd	1					
	BIXOR #xx:3, @ERd	2			1		
	BIXOR #xx:3, @aa:8	2			1		
BLD	BLD #xx:3, Rd	1					
	BLD #xx:3, @ERd	2			1		
	BLD #xx:3, @aa:8	2			1		
BNOT	BNOT #xx:3, Rd	1					
	BNOT #xx:3, @ERd	2			2		
	BNOT #xx:3, @aa:8	2			2		
	BNOT Rn, Rd	1					
	BNOT Rn, @ERd	2			2		
	BNOT Rn, @aa:8	2			2		
BOR	BOR #xx:3, Rd	1					
	BOR #xx:3, @ERd	2			1		
	BOR #xx:3, @aa:8	2			1		
BSET	BSET #xx:3, Rd	1					
	BSET #xx:3, @ERd	2			2		
	BSET #xx:3, @aa:8	2			2		
	BSET Rn, Rd	1					
	BSET Rn, @ERd	2			2		
	BSET Rn, @aa:8	2			2		
BSR	BSR d:8	2		1			
	BSR d:16	2		1			2
BST	BST #xx:3, Rd	1					
	BST #xx:3, @ERd	2			2		
	BST #xx:3, @aa:8	2			2		

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
BTST	BTST #xx:3, Rd	1					
	BTST #xx:3, @ERd	2			1		
	BTST #xx:3, @aa:8	2			1		
	BTST Rn, Rd	1					
	BTST Rn, @ERd	2			1		
	BTST Rn, @aa:8	2			1		
BXOR	BXOR #xx:3, Rd	1					
	BXOR #xx:3, @ERd	2			1		
	BXOR #xx:3, @aa:8	2			1		
CMP	CMP.B #xx:8, Rd	1					
	CMP.B Rs, Rd	1					
	CMP.W #xx:16, Rd	2					
	CMP.W Rs, Rd	1					
	CMP.L #xx:32, ERd	3					
	CMP.L ERs, ERd	1					
DAA	DAA Rd	1					
DAS	DAS Rd	1					
DEC	DEC.B Rd	1					
	DEC.W #1/2, Rd	1					
	DEC.L #1/2, ERd	1					
DUVXS	DIVXS.B Rs, Rd	2					12
	DIVXS.W Rs, ERd	2					20
DIVXU	DIVXU.B Rs, Rd	1					12
	DIVXU.W Rs, ERd	1					20
EEPMOV	EEPMOV.B	2			2n+2*[1]		
	EEPMOV.W	2			2n+2*[1]		
EXTS	EXTS.W Rd	1					
	EXTS.L ERd	1					
EXTU	EXTU.W Rd	1					
	EXTU.L ERd	1					

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
INC	INC.B Rd	1					
	INC.W #1/2, Rd	1					
	INC.L #1/2, ERd	1					
JMP	JMP @ERn	2					
	JMP @aa:24	2					2
	JMP @@aa:8	2	1				2
JSR	JSR @ERn	2		1			
	JSR @aa:24	2		1			2
	JSR @@aa:8	2	1	1			
LDC	LDC #xx:8, CCR	1					
	LDC Rs, CCR	1					
	LDC@ERs, CCR	2				1	
	LDC@(d:16, ERs), CCR	3				1	
	LDC@(d:24,ERs), CCR	5				1	
	LDC@ERs+, CCR	2				1	2
	LDC@aa:16, CCR	3				1	
	LDC@aa:24, CCR	4				1	
MOV	MOV.B #xx:8, Rd	1					
	MOV.B Rs, Rd	1					
	MOV.B @ERs, Rd	1			1		
	MOV.B @(d:16, ERs), Rd	2			1		
	MOV.B @(d:24, ERs), Rd	4			1		
	MOV.B @ERs+, Rd	1			1		2
	MOV.B @aa:8, Rd	1			1		
	MOV.B @aa:16, Rd	2			1		
	MOV.B @aa:24, Rd	3			1		
	MOV.B Rs, @Erd	1			1		
	MOV.B Rs, @(d:16, ERd)	2			1		
	MOV.B Rs, @(d:24, ERd)	4			1		
	MOV.B Rs, @-ERd	1			1		2
	MOV.B Rs, @aa:8	1			1		

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
MOV	MOV.B Rs, @aa:16	2			1		
	MOV.B Rs, @aa:24	3			1		
	MOV.W #xx:16, Rd	2					
	MOV.W Rs, Rd	1					
	MOV.W @ERs, Rd	1				1	
	MOV.W @(d:16,ERs), Rd	2				1	
	MOV.W @(d:24,ERs), Rd	4				1	
	MOV.W @ERs+, Rd	1				1	2
	MOV.W @aa:16, Rd	2				1	
	MOV.W @aa:24, Rd	3				1	
	MOV.W Rs, @ERd	1				1	
	MOV.W Rs, @(d:16,ERd)	2				1	
	MOV.W Rs, @(d:24,ERd)	4				1	
MOV	MOV.W Rs, @-ERd	1				1	2
	MOV.W Rs, @aa:16	2				1	
	MOV.W Rs, @aa:24	3				1	
	MOV.L #xx:32, ERd	3					
	MOV.L ERs, ERd	1					
	MOV.L @ERs, ERd	2				2	
	MOV.L @(d:16,ERs), ERd	3				2	
	MOV.L @(d:24,ERs), ERd	5				2	
	MOV.L @ERs+, ERd	2				2	2
	MOV.L @aa:16, ERd	3				2	
	MOV.L @aa:24, ERd	4				2	
	MOV.L ERs,@ERd	2				2	
	MOV.L ERs, @(d:16,ERd)	3				2	
	MOV.L ERs, @(d:24,ERd)	5				2	
	MOV.L ERs, @-ERd	2				2	2
	MOV.L ERs, @aa:16	3				2	
	MOV.L ERs, @aa:24	4				2	
MOVFPE	MOVFPE @aa:16, Rd*[2]	2			1		
MOVTPE	MOVTPE Rs,@aa:16*[2]	2			1		

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
MULXS	MULXS.B Rs, Rd	2					12
	MULXS.W Rs, ERd	2					20
MULXU	MULXU.B Rs, Rd	1					12
	MULXU.W Rs, ERd	1					20
NEG	NEG.B Rd	1					
	NEG.W Rd	1					
	NEG.L ERd	1					
NOP	NOP	1					
NOT	NOT.B Rd	1					
	NOT.W Rd	1					
	NOT.L ERd	1					
OR	OR.B #xx:8, Rd	1					
	OR.B Rs, Rd	1					
	OR.W #xx:16, Rd	2					
	OR.W Rs, Rd	1					
	OR.L #xx:32, ERd	3					
	OR.L ERs, ERd	2					
ORC	ORC #xx:8, CCR	1					
POP	POP.W Rn	1				1	2
	POP.L ERn	2				2	2
PUSH	PUSH.W Rn	1				1	2
	PUSH.L ERn	2				2	2
ROTL	ROTL.B Rd	1					
	ROTL.W Rd	1					
	ROTL.L ERd	1					
ROTR	ROTR.B Rd	1					
	ROTR.W Rd	1					
	ROTR.L ERd	1					
ROTXL	ROTXL.B Rd	1					
	ROTXL.W Rd	1					
	ROTXL.L ERd	1					

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
ROTXR	ROTXR.B Rd	1					
	ROTXR.W Rd	1					
	ROTXR.L ERd	1					
RTE	RTE	2		2			2
RTS	RTS	2		1			2
SHAL	SHAL.B Rd	1					
	SHAL.W Rd	1					
	SHAL.L ERd	1					
SHAR	SHAR.B Rd	1					
	SHAR.W Rd	1					
	SHAR.L ERd	1					
SHLL	SHLL.B Rd	1					
	SHLL.W Rd	1					
	SHLL.L ERd	1					
SHLR	SHLR.B Rd	1					
	SHLR.W Rd	1					
	SHLR.L ERd	1					
SLEEP	SLEEP	1					
STC	STC CCR, Rd	1					
	STC CCR, @ERd	2				1	
	STC CCR, @(d:16,ERd)	3				1	
	STC CCR, @(d:24,ERd)	5				1	
	STC CCR,@-ERd	2				1	2
	STC CCR, @aa:16	3				1	
	STC CCR, @aa:24	4				1	
SUB	SUB.B Rs, Rd	1					
	SUB.W #xx:16, Rd	2					
	SUB.W Rs, Rd	1					
	SUB.L #xx:32, ERd	3					
	SUB.L ERs, ERd	1					
SUBS	SUBS #1/2/4, ERd	1					

Instruction	Mnemonic	Instruction Fetch I	Branch Addr. Read J	Stack Operation K	Byte Data Access L	Word Data Access M	Internal Operation N
SUBX	SUBX #xx:8, Rd	1					
	SUBX. Rs, Rd	1					
TRAPA	TRAPA #xx:2	2	1	2			4
XOR	XOR.B #xx:8, Rd	1					
	XOR.B Rs, Rd	1					
	XOR.W #xx:16, Rd	2					
	XOR.W Rs, Rd	1					
	XOR.L #xx:32, ERd	3					
	XOR.L ERs, ERd	2					
XORC	XORC #xx:8, CCR	1					

Notes: 1. n: Specified value in R4L and R4. The source and destination operands are accessed n+1 times respectively.
2. Cannot be used in this LSI.

A.4 Combinations of Instructions and Addressing Modes

Table A.5 Combinations of Instructions and Addressing Modes

Functions	Instructions	Addressing Mode												
		#xx	Rn	@ERn	@(d:16.ERn)	@(d:24.ERn)	@ERn+/@ERn	@aa:8	@aa:16	@aa:24	@(d:8.PC)	@(d:16.PC)	@@aa:8	|
Data transfer instructions	MOV	BWL	BWL	BWL	BWL	BWL	BWL	B	BWL	BWL	—	—	—	—
	POP, PUSH	—	—	—	—	—	—	—	—	—	—	—	—	WL
	MOVFPE, MOVTPE	—	—	—	—	—	—	—	—	—	—	—	—	—
Arithmetic operations	ADD, CMP	BWL	BWL	—	—	—	—	—	—	—	—	—	—	—
	SUB	WL	BWL	—	—	—	—	—	—	—	—	—	—	—
	ADDX, SUBX	B	B	—	—	—	—	—	—	—	—	—	—	—
	ADDS, SUBS	—	L	—	—	—	—	—	—	—	—	—	—	—
	INC, DEC	—	BWL	—	—	—	—	—	—	—	—	—	—	—
	DAA, DAS	—	B	—	—	—	—	—	—	—	—	—	—	—
	MULXU, MULXS, DIVXU, DIVXS	—	BW	—	—	—	—	—	—	—	—	—	—	—
	NEG	—	BWL	—	—	—	—	—	—	—	—	—	—	—
	EXTU, EXTS	—	WL	—	—	—	—	—	—	—	—	—	—	—
Logical operations	AND, OR, XOR	—	BWL	—	—	—	—	—	—	—	—	—	—	—
	NOT	—	BWL	—	—	—	—	—	—	—	—	—	—	—
Shift operations		—	BWL	—	—	—	—	—	—	—	—	—	—	—
Bit manipulations		—	B	B	—	—	—	B	—	—	—	—	—	—
Branching instructions	BCC, BSR	—	—	—	—	—	—	—	—	—	—	—	—	—
	JMP, JSR	—	—	◯	—	—	—	—	—	—	◯	◯	—	—
	RTS	—	—	—	—	—	—	—	—	◯	—	—	◯	—
System control instructions	TRAPA	—	—	—	—	—	—	—	—	—	—	—	—	◯
	RTE	—	—	—	—	—	—	—	—	—	—	—	—	◯
	SLEEP	—	—	—	—	—	—	—	—	—	—	—	—	◯
	LDC	B	B	W	W	W	W	—	W	W	—	—	—	◯
	STC	—	B	W	W	W	W	—	W	W	—	—	—	—
	ANDC, ORC, XORC	B	—	—	—	—	—	—	—	—	—	—	—	—
	NOP	—	—	—	—	—	—	—	—	—	—	—	—	◯
Block data transfer instructions		—	—	—	—	—	—	—	—	—	—	—	—	BW

Appendix B I/O Port Block Diagrams

B.1 I/O Port Block Diagrams

$\overline{\text{RES}}$ goes low in a reset, and $\overline{\text{SBY}}$ goes low at reset and in standby mode.

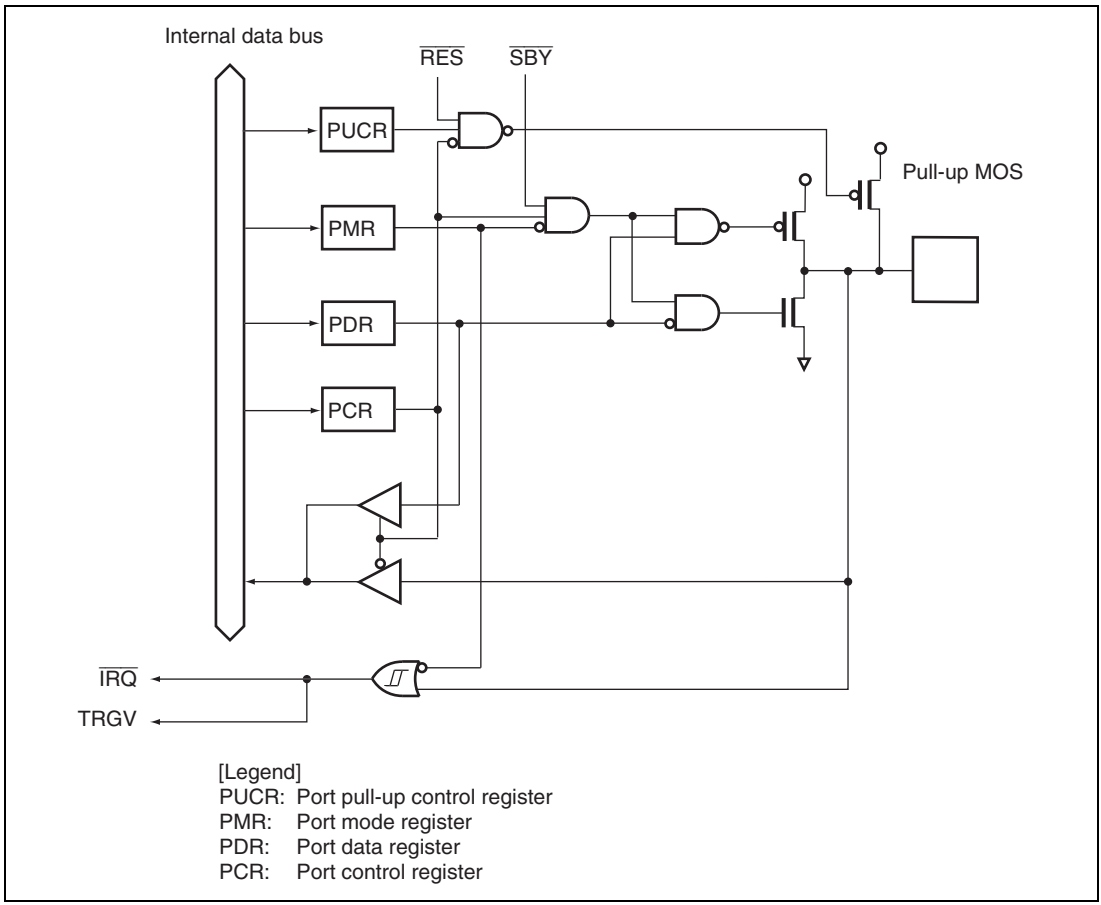


Figure B.1 Port 1 Block Diagram (P17)

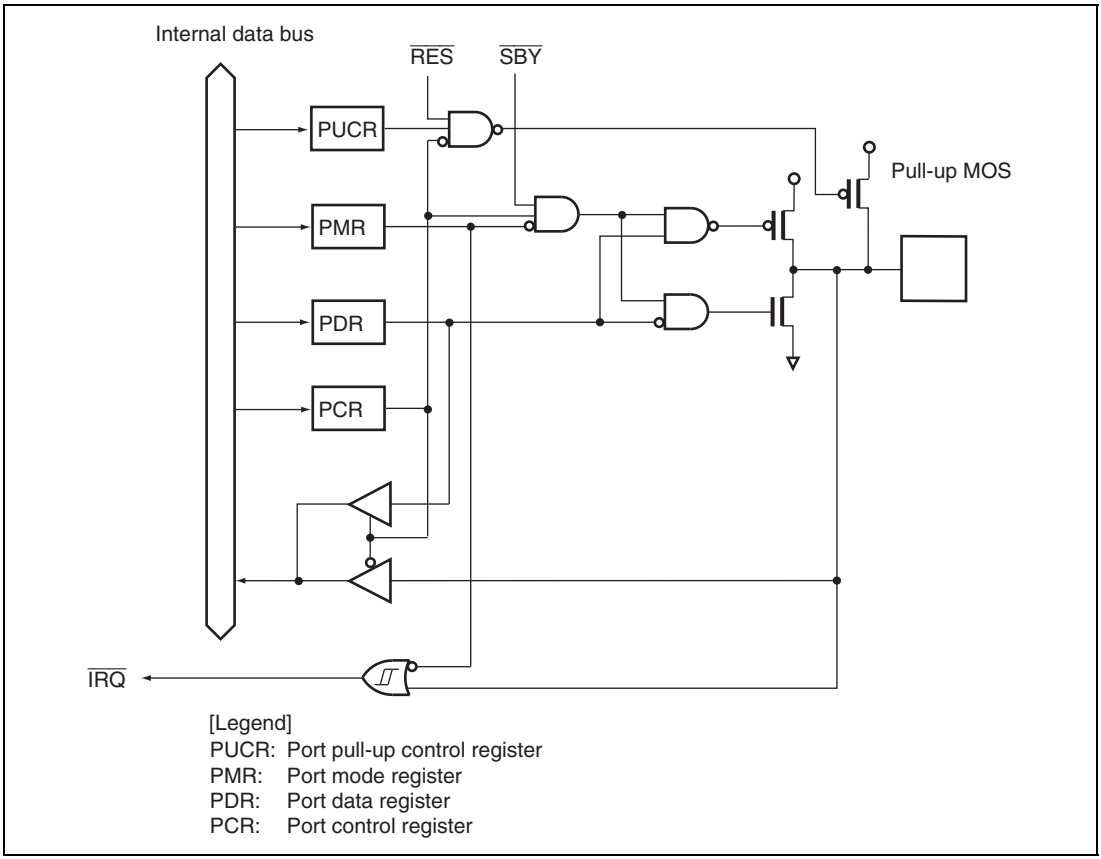


Figure B.2 Port 1 Block Diagram (P14, P16)

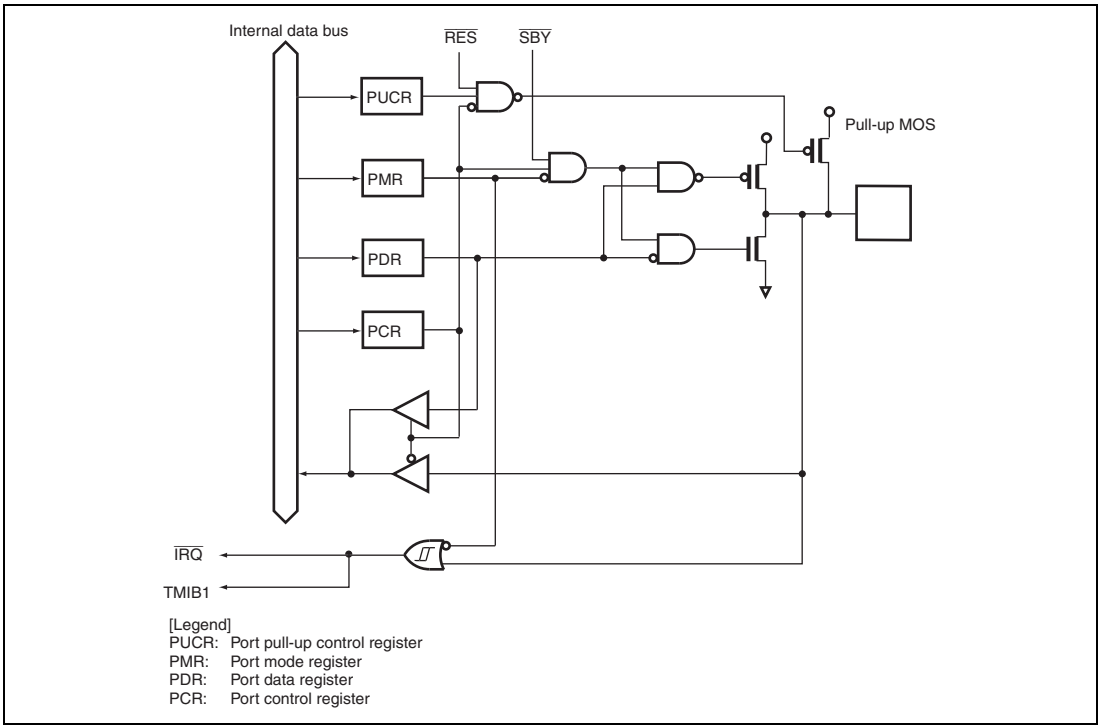


Figure B.3 Port 1 Block Diagram (P15)

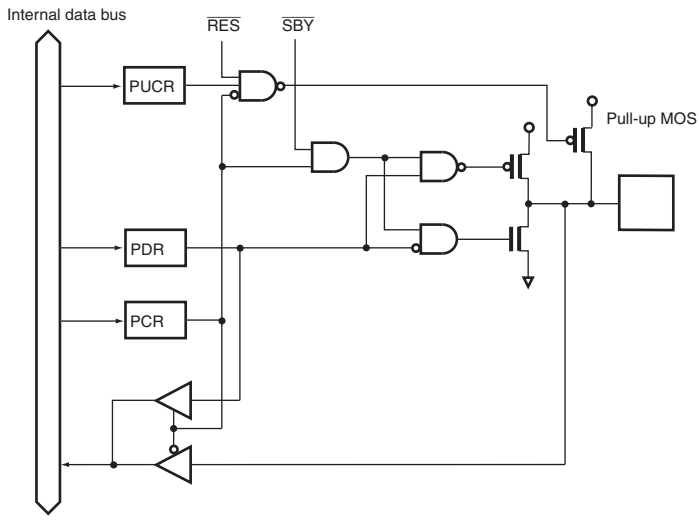


Figure B.4 Port 1 Block Diagram (P12)

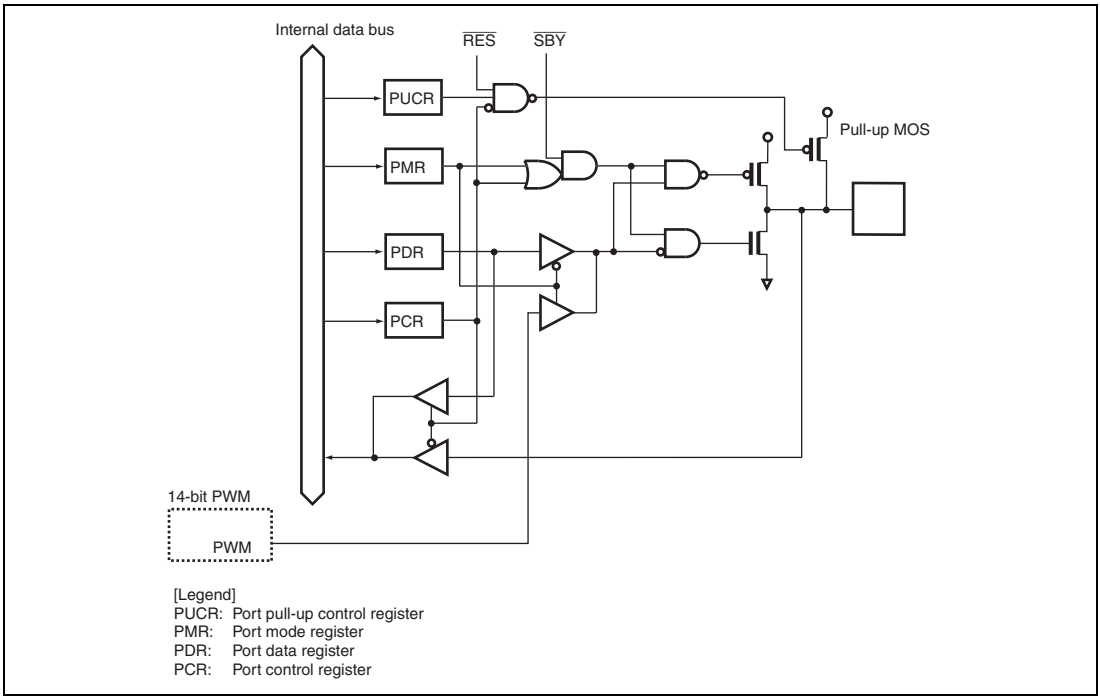


Figure B.5 Port 2 Block Diagram (P11)

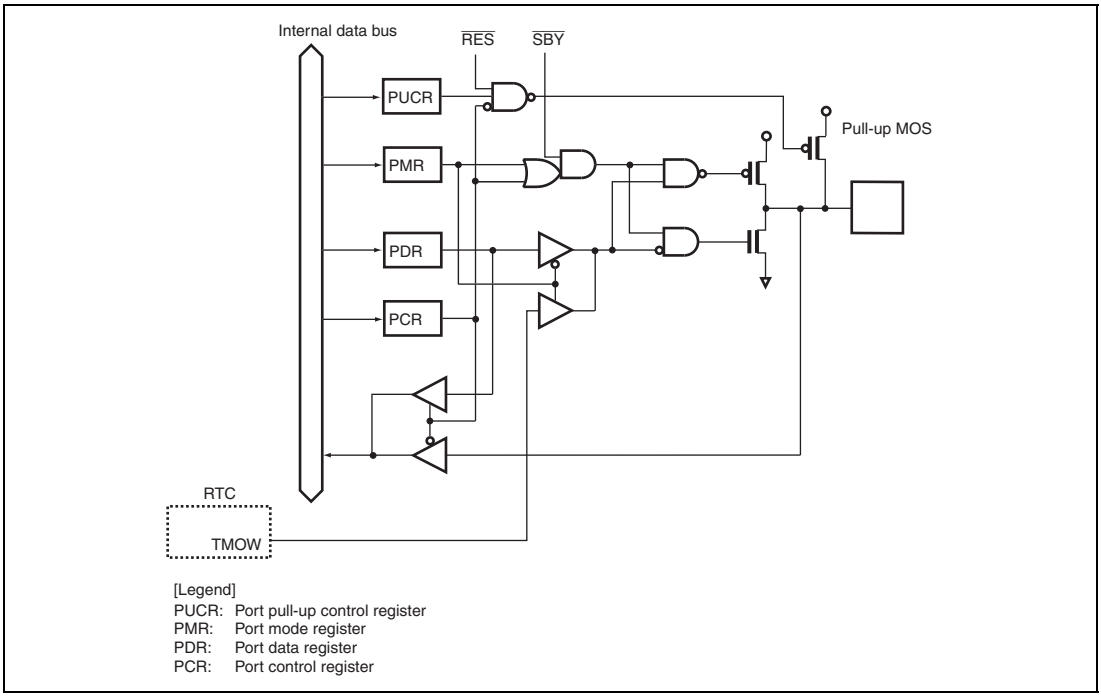


Figure B.6 Port 1 Block Diagram (P10)

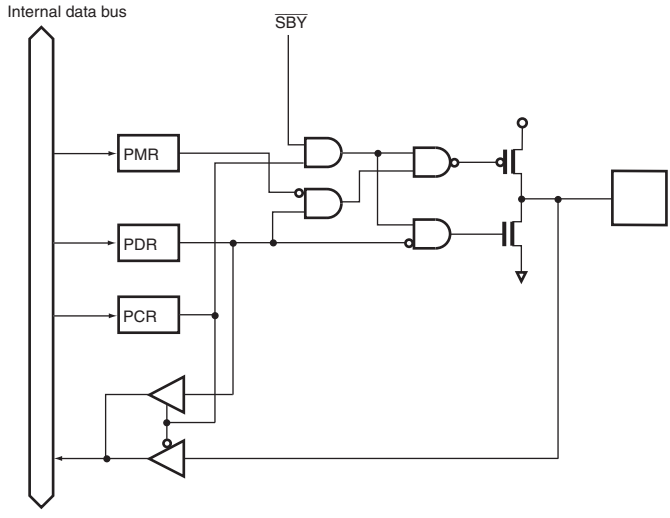


Figure B.7 Port 2 Block Diagram (P24, P23)

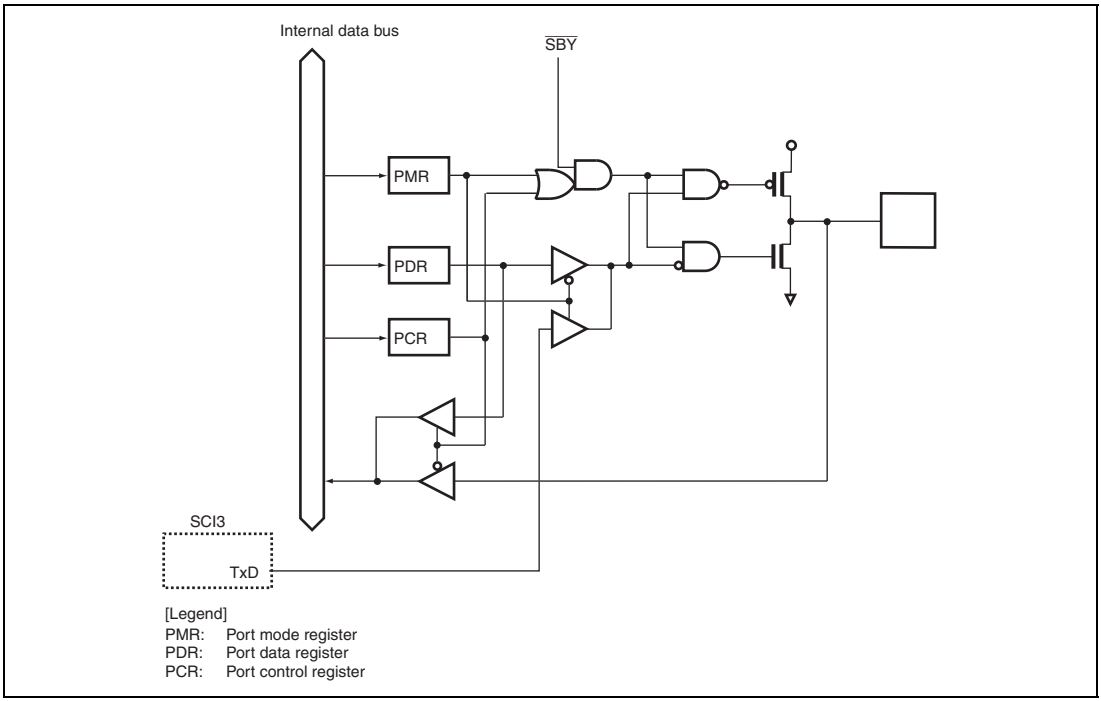


Figure B.8 Port 2 Block Diagram (P22)

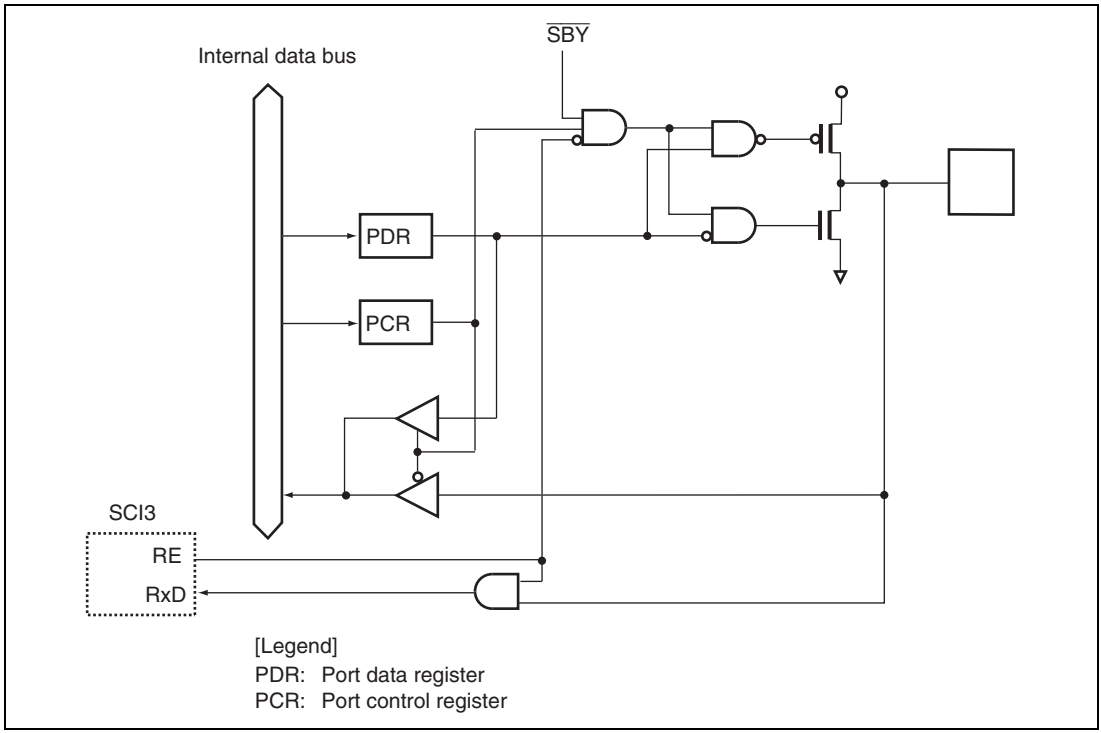


Figure B.9 Port 2 Block Diagram (P21)

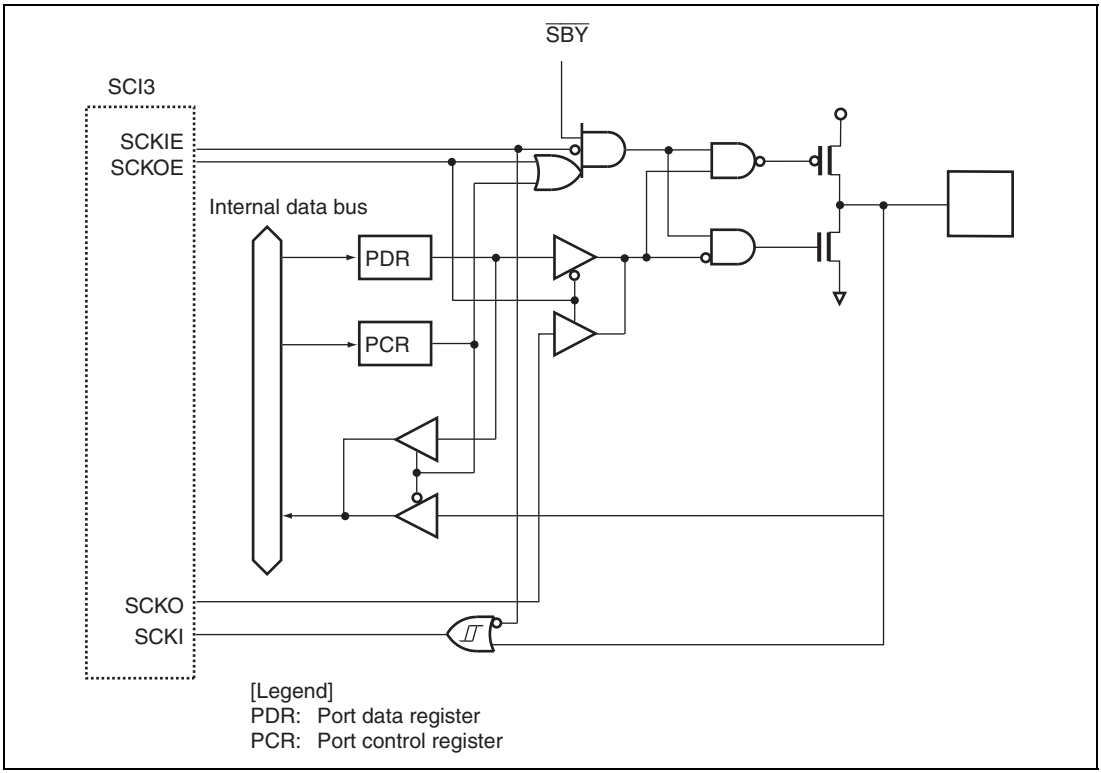


Figure B.10 Port 2 Block Diagram (P20)

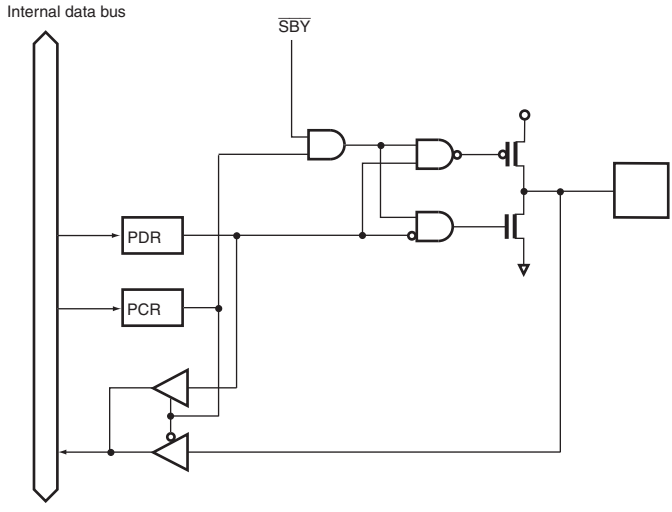


Figure B.11 Port 3 Block Diagram (P37 to P30)

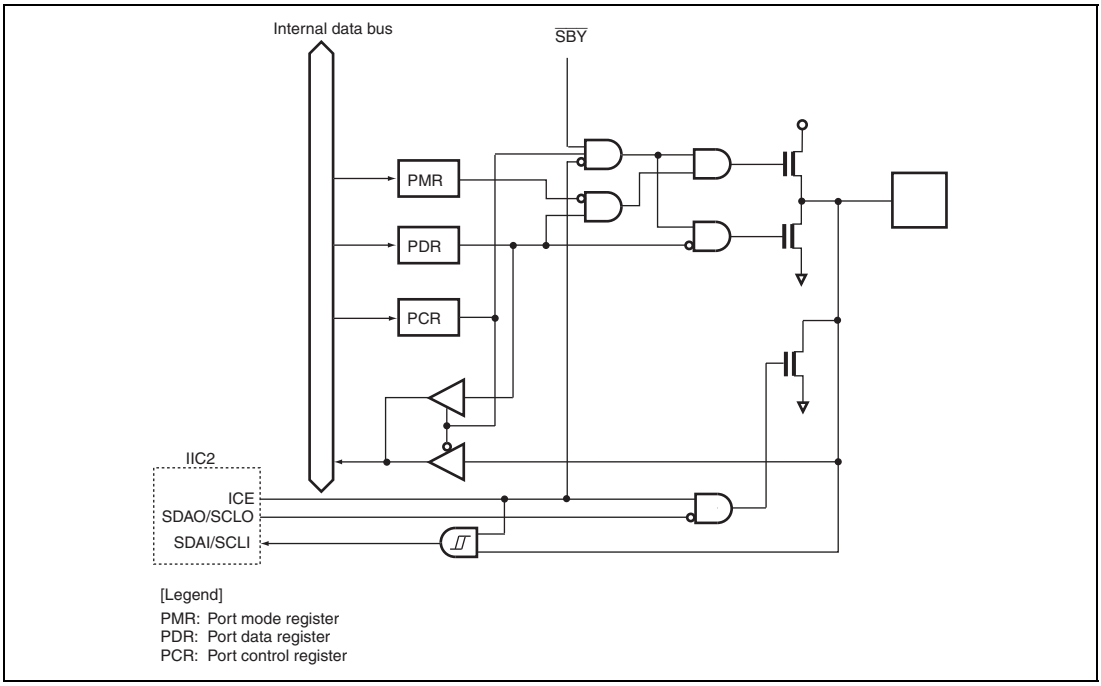


Figure B.12 Port 5 Block Diagram (P57, P56)*

Note: * This diagram is applied to the SCL and SDA pins in the H8/3687N.

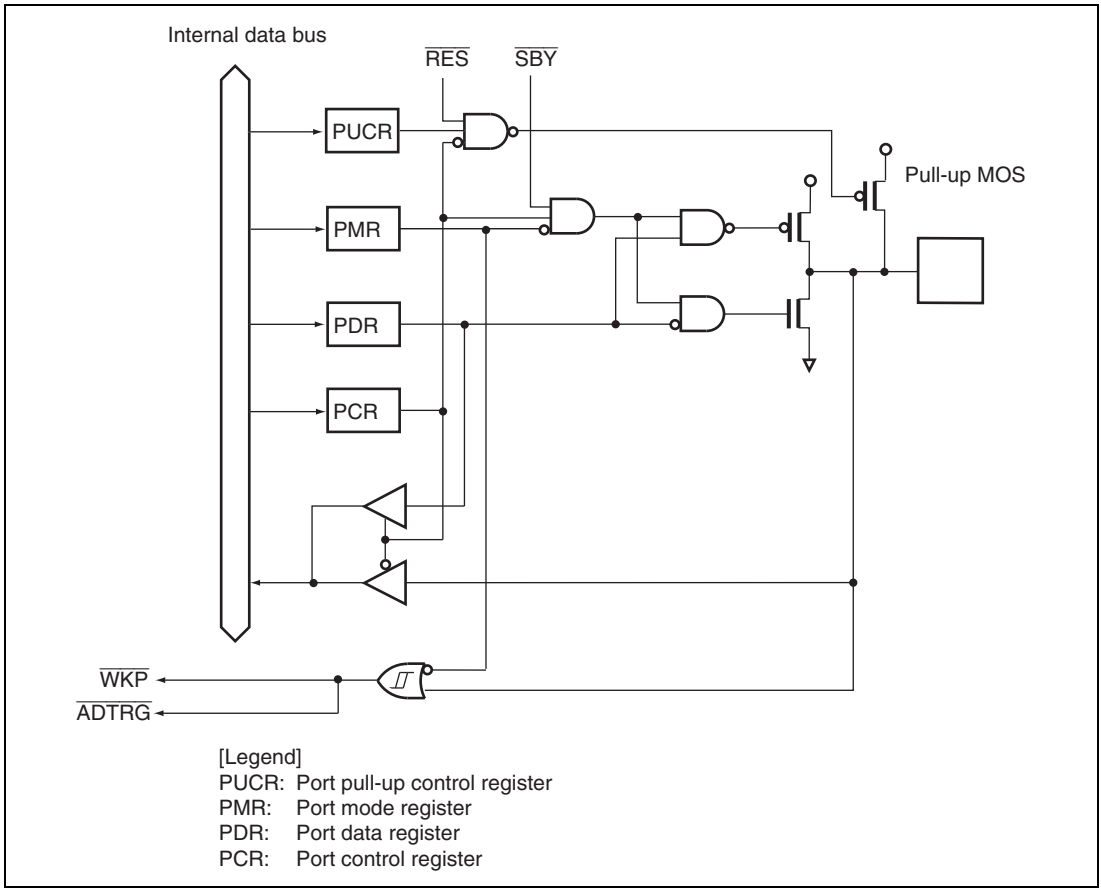


Figure B.13 Port 5 Block Diagram (P55)

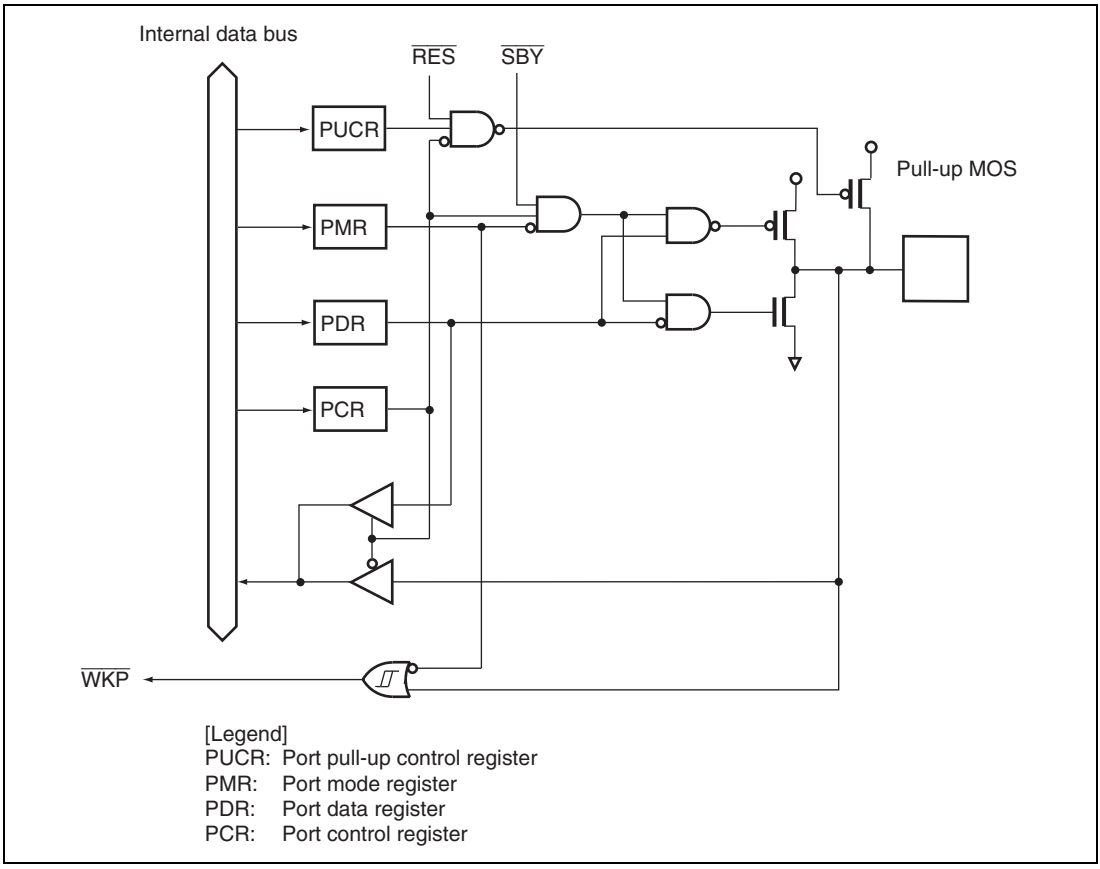


Figure B.14 Port 5 Block Diagram (P54 to P50)

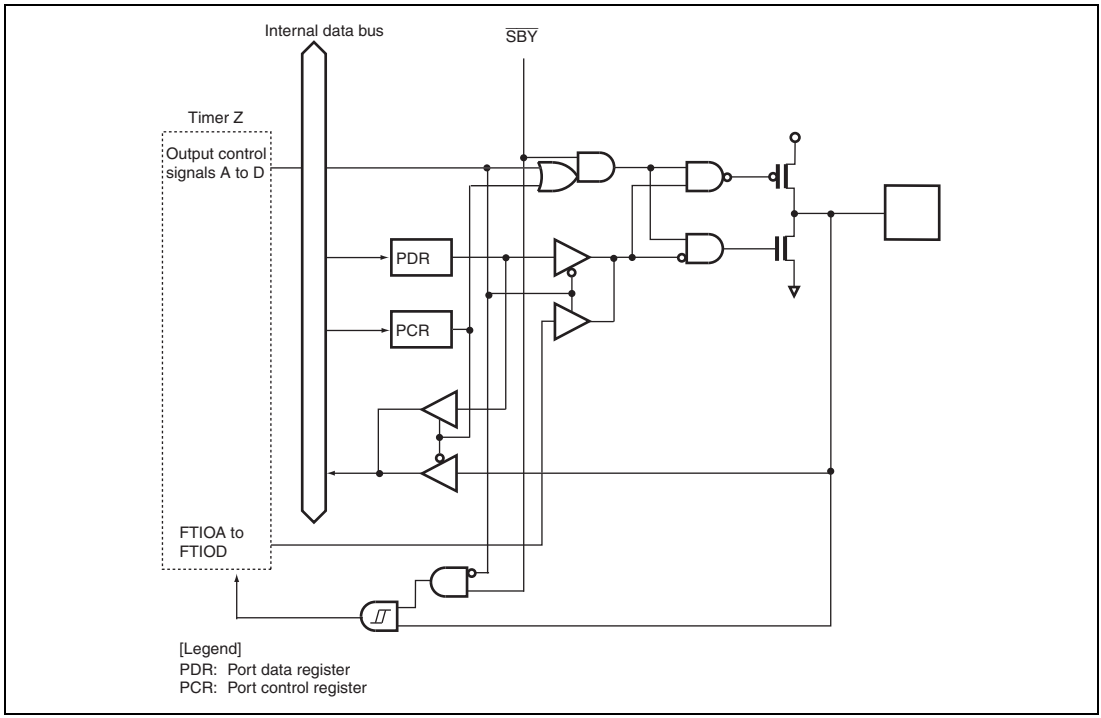


Figure B.15 Port 6 Block Diagram (P67 to P60)

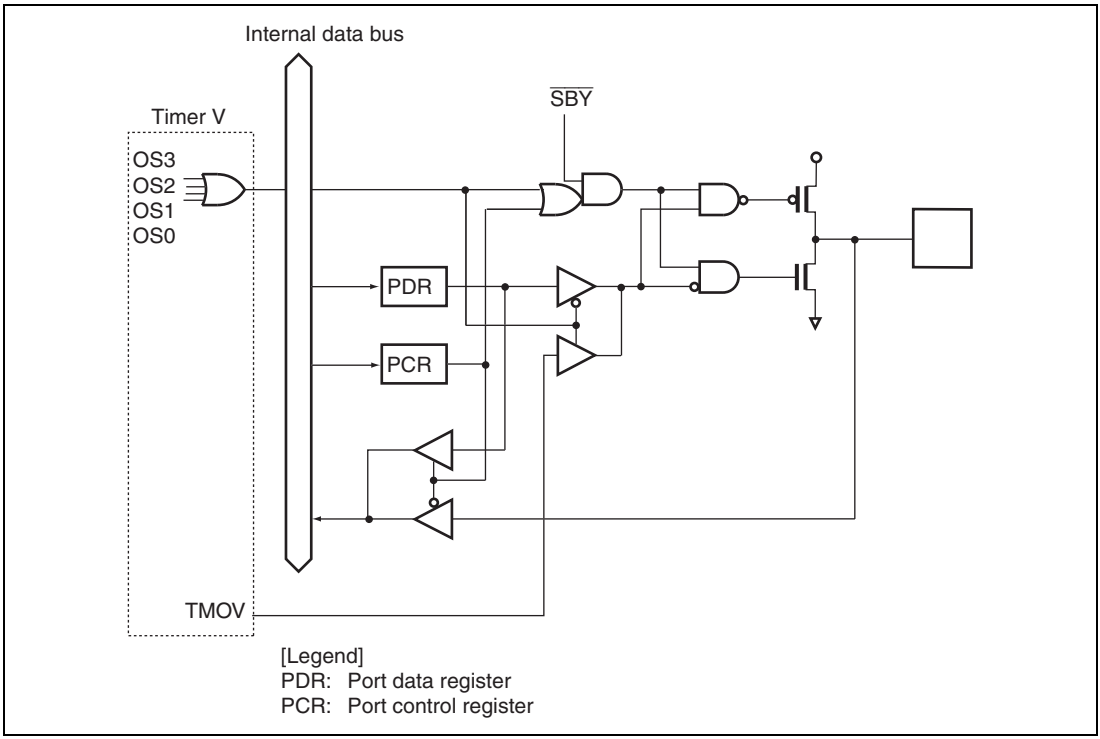


Figure B.16 Port 7 Block Diagram (P76)

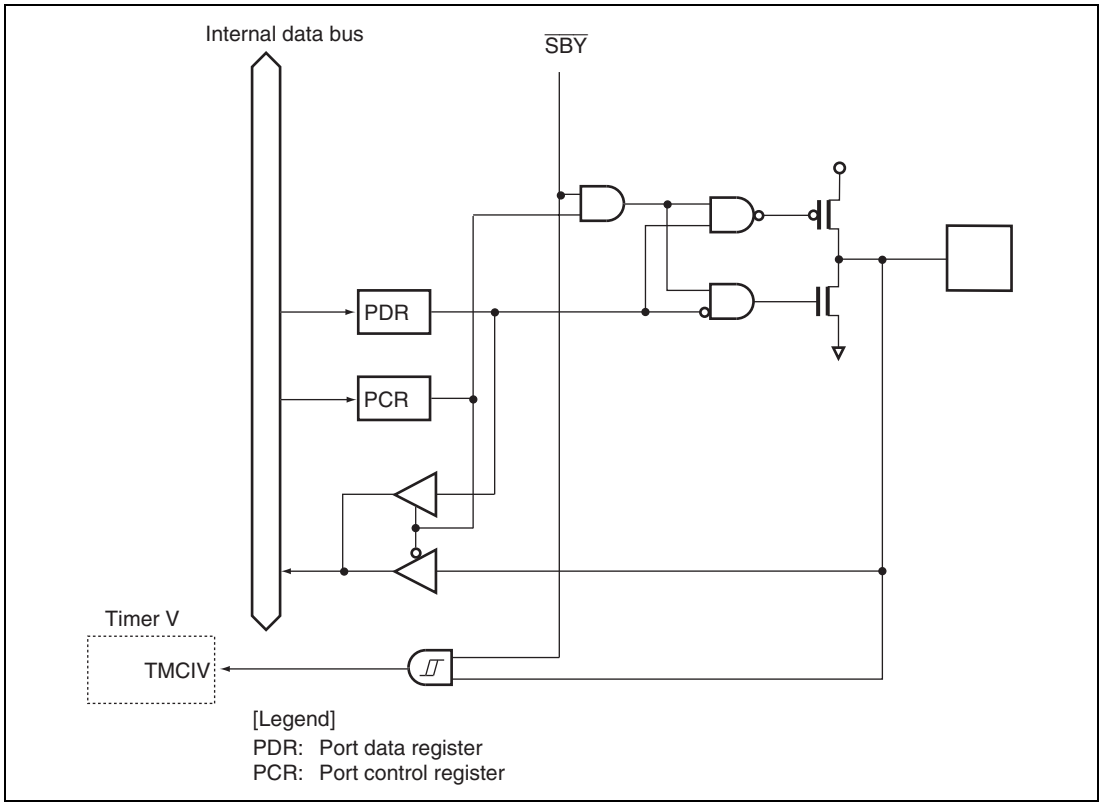


Figure B.17 Port 7 Block Diagram (P75)

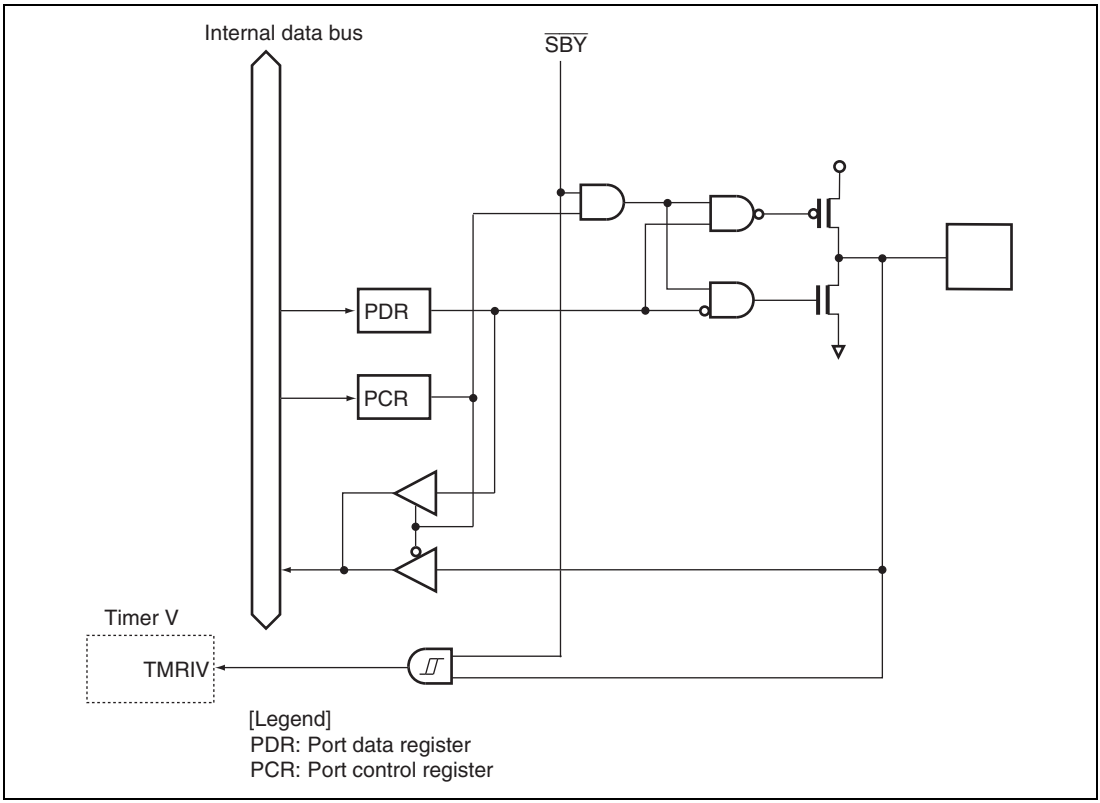


Figure B.18 Port 7 Block Diagram (P74)

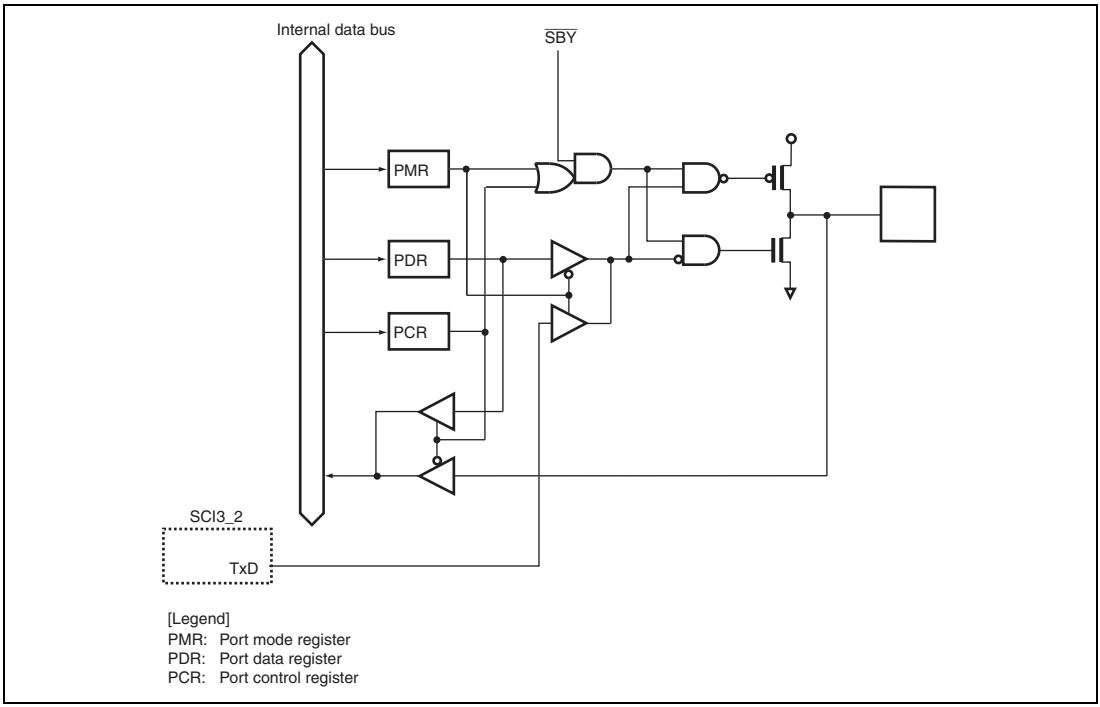


Figure B.19 Port 7 Block Diagram (P72)

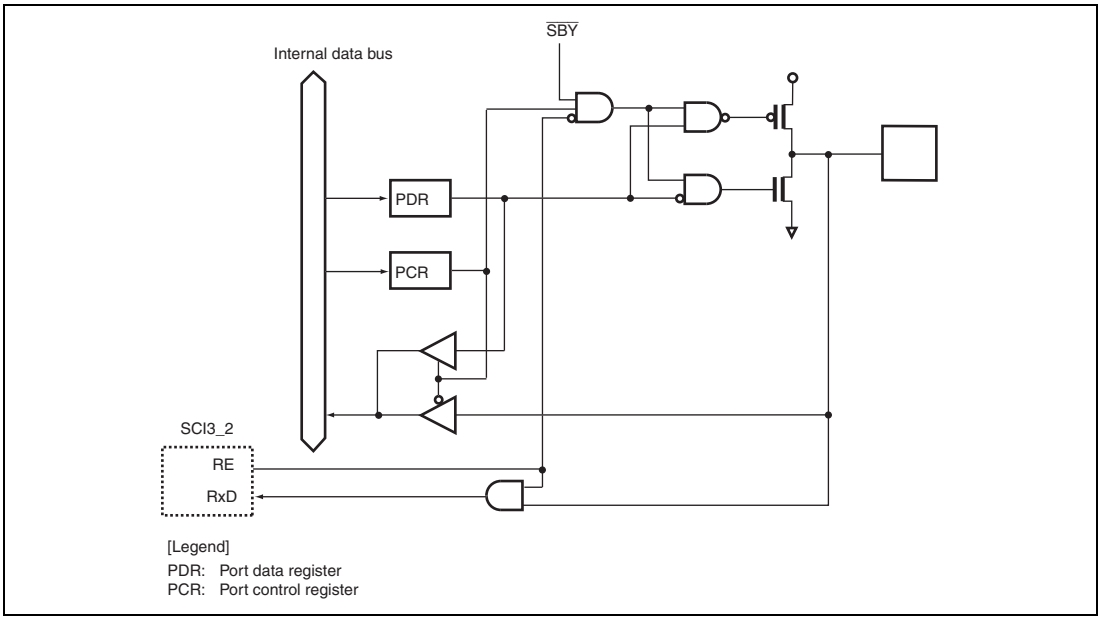


Figure B.20 Port 7 Block Diagram (P71)

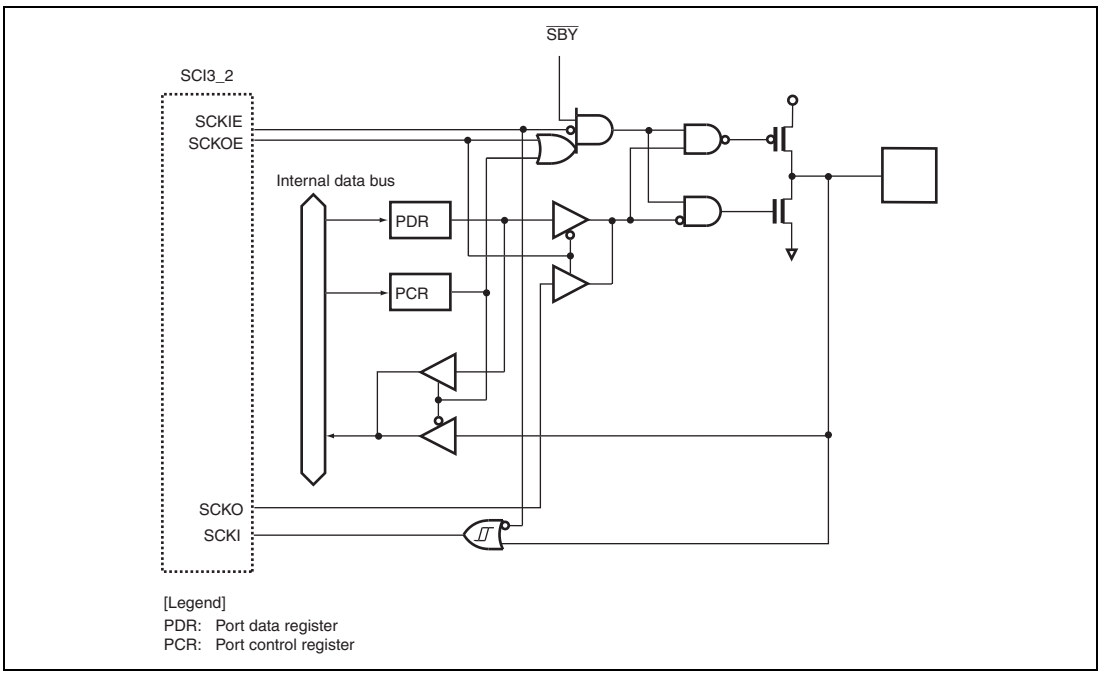


Figure B.21 Port 7 Block Diagram (P70)

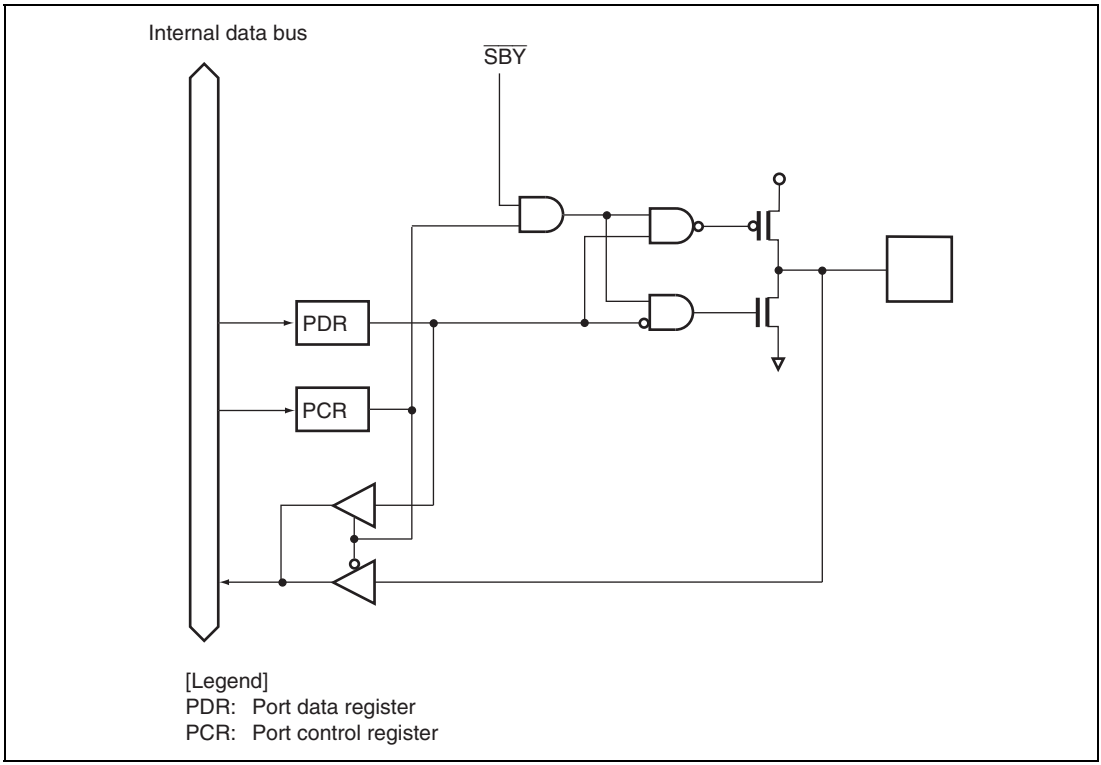


Figure B.22 Port 8 Block Diagram (P87 to P85)

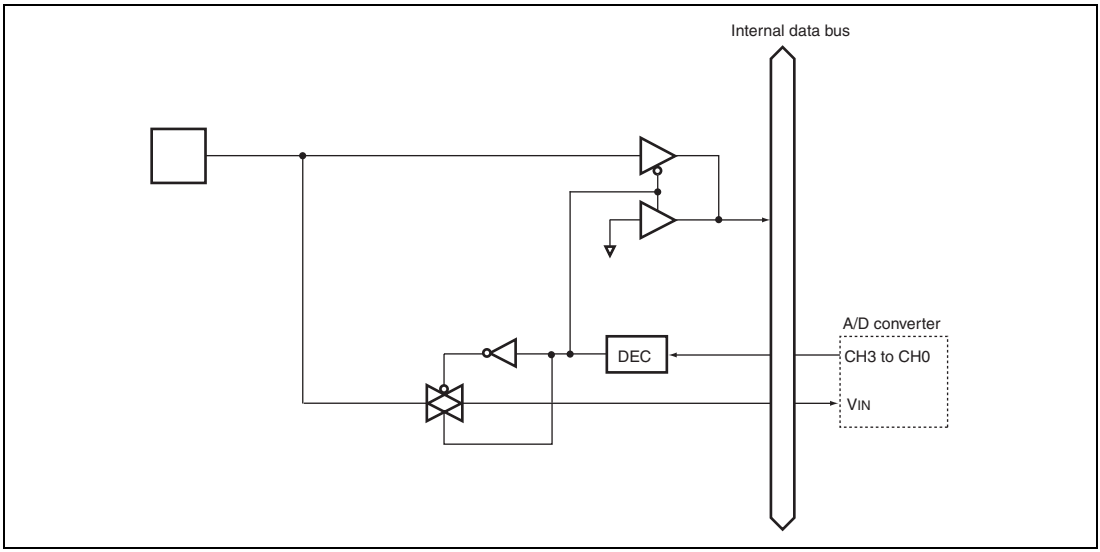


Figure B.23 Port B Block Diagram (PB7 to PB0)

B.2 Port States in Each Operating State

Port	Reset	Sleep	Subsleep	Standby	Subactive	Active
P17 to P14, P12 to P10	High impedance	Retained	Retained	High impedance*[1]	Functioning	Functioning
P24 to P20	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P37 to P30	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P57 to P50*[2]	High impedance	Retained	Retained	High impedance*[1]	Functioning	Functioning
P67 to P60	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P76 to P74, P72 to P70	High impedance	Retained	Retained	High impedance	Functioning	Functioning
P87 to P85	High impedance	Retained	Retained	High impedance	Functioning	Functioning
PB7 to PB0	High impedance	High impedance	High impedance	High impedance	High impedance	High impedance

Notes: 1. High level output when the pull-up MOS is in on state.
2. The P55 to P50 pins are applied to the H8/3687N.

Appendix C Product Code Lineup

Product Classification			Product Code	Model Marking	Package Code
H8/3687	Flash memory version	Standard product	HD64F3687H	HD64F3687H	QFP-64 (FP-64A)
			HD64F3687FP	HD64F3687FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD64F3687GH	HD64F3687GH	QFP-64 (FP-64A)
			HD64F3687GFP	HD64F3687GFP	LQFP-64 (FP-64E)
	Mask ROM version	Standard product	HD6433687H	HD6433687(***)H	QFP-64 (FP-64A)
			HD6433687FP	HD6433687(***)FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD6433687GH	HD6433687G(***)H	QFP-64 (FP-64A)
			HD6433687GFP	HD6433687G(***)FP	LQFP-64 (FP-64E)
H8/3686	Mask ROM version	Standard product	HD6433686H	HD6433686(***)H	QFP-64 (FP-64A)
			HD6433686FP	HD6433686(***)FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD6433686GH	HD6433686G(***)H	QFP-64 (FP-64A)
			HD6433686GFP	HD6433686G(***)FP	LQFP-64 (FP-64E)
H8/3685	Mask ROM version	Standard product	HD6433685H	HD6433685(***)H	QFP-64 (FP-64A)
			HD6433685FP	HD6433685(***)FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD6433685GH	HD6433685G(***)H	QFP-64 (FP-64A)
			HD6433685GFP	HD6433685G(***)FP	LQFP-64 (FP-64E)
H8/3684	Flash memory version	Standard product	HD64F3684H	HD64F3684H	QFP-64 (FP-64A)
			HD64F3684FP	HD64F3684FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD64F3684GH	HD64F3684GH	QFP-64 (FP-64A)
			HD64F3684GFP	HD64F3684GFP	LQFP-64 (FP-64E)
	Mask ROM version	Standard product	HD6433684H	HD6433684(***)H	QFP-64 (FP-64A)
			HD6433684FP	HD6433684(***)FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD6433684GH	HD6433684G(***)H	QFP-64 (FP-64A)
			HD6433684GFP	HD6433684G(***)FP	LQFP-64 (FP-64E)
H8/3683	Mask ROM version	Standard product	HD6433683H	HD6433683(***)H	QFP-64 (FP-64A)
			HD6433683FP	HD6433683(***)FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD6433683GH	HD6433683G(***)H	QFP-64 (FP-64A)
			HD6433683GFP	HD6433683G(***)FP	LQFP-64 (FP-64E)
H8/3682	Mask ROM version	Standard product	HD6433682H	HD6433682(***)H	QFP-64 (FP-64A)
			HD6433682FP	HD6433682(***)FP	LQFP-64 (FP-64E)
		Product with POR & LVDC	HD6433682GH	HD6433682G(***)H	QFP-64 (FP-64A)
			HD6433682GFP	HD6433682G(***)FP	LQFP-64 (FP-64E)

Product Classification				Product Code	Model Marking	Package Code
H8/3687	EEPROM stacked version	Flash memory version	Product with POR & LVDC	HD64N3687GFP	HD64N3687GFP	LQFP-64 (FP-64E)
		Mask ROM version		HD6483687GFP	HD6483687G(***)FP	LQFP-64 (FP-64E)

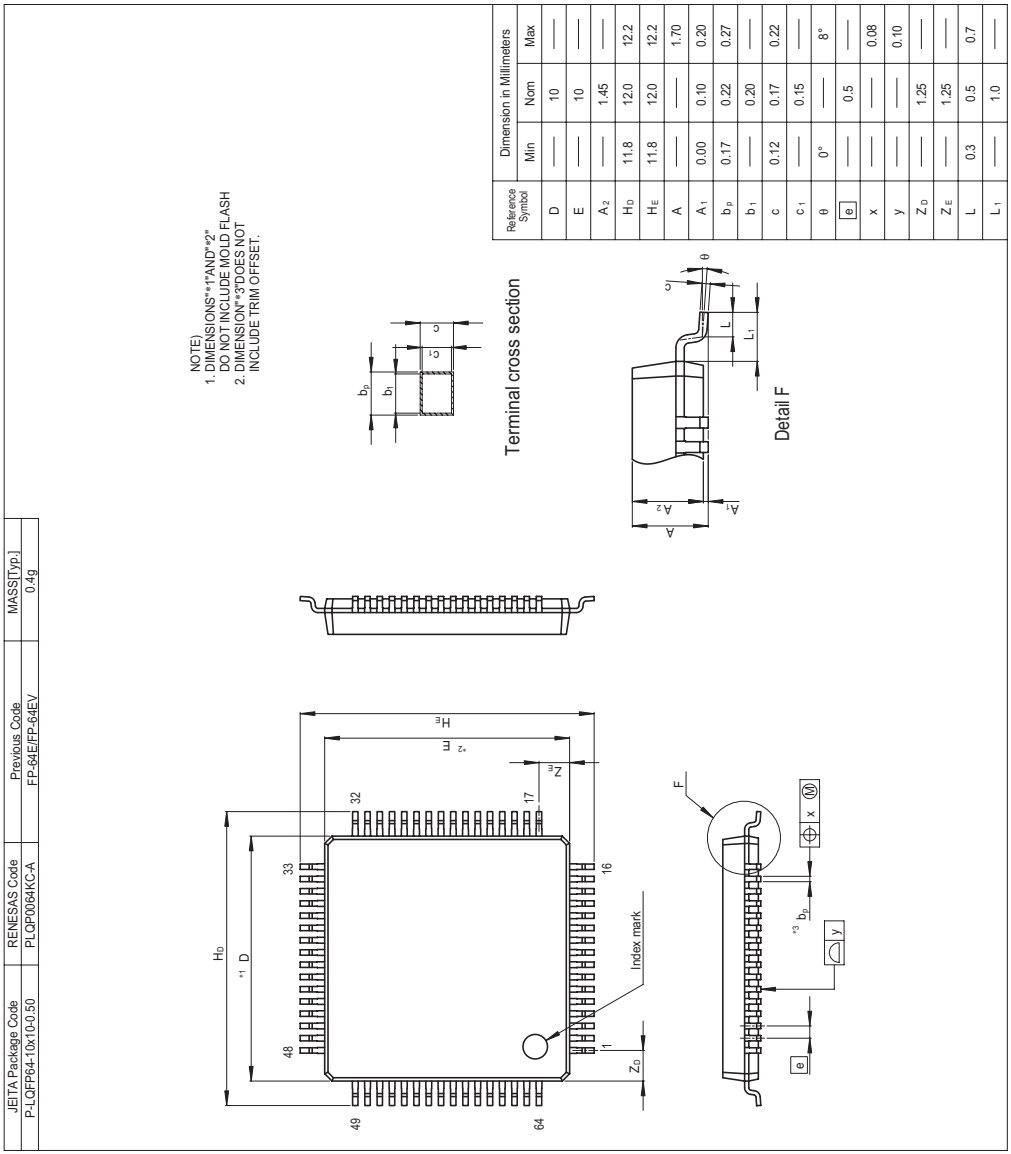
Legend:

(***): ROM code.

POR & LVDC: Power-on reset and low-voltage detection circuits.

Appendix D Package Dimensions

The package dimensions that are shown in the Renesas Semiconductor Packages Data Book have priority.



Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	—	10	—
E	—	10	—
A_2	—	1.45	—
H_D	11.8	12.0	12.2
H_E	11.8	12.0	12.2
A	—	—	1.70
A_1	0.00	0.10	0.20
b_p	0.17	0.22	0.27
b_1	—	0.20	—
c	0.12	0.17	0.22
c_1	—	0.15	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.10
Z_D	—	1.25	—
Z_E	—	1.25	—
L	0.3	0.5	0.7
L_1	—	1.0	—

Figure D.1 FP-64E Package Dimensions

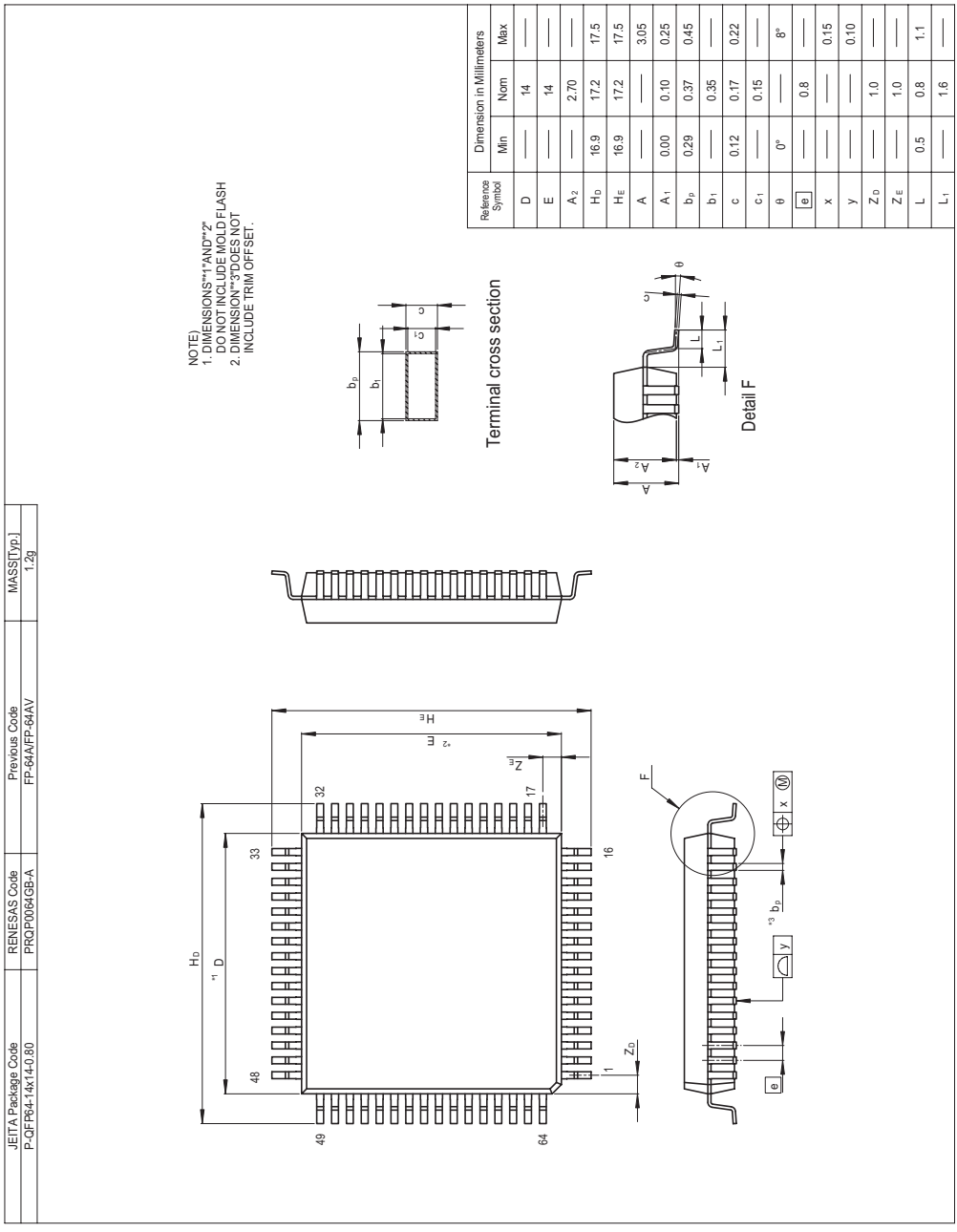


Figure D.2 FP-64A Package Dimensions

Appendix E EEPROM Stacked-Structure Cross-Sectional View

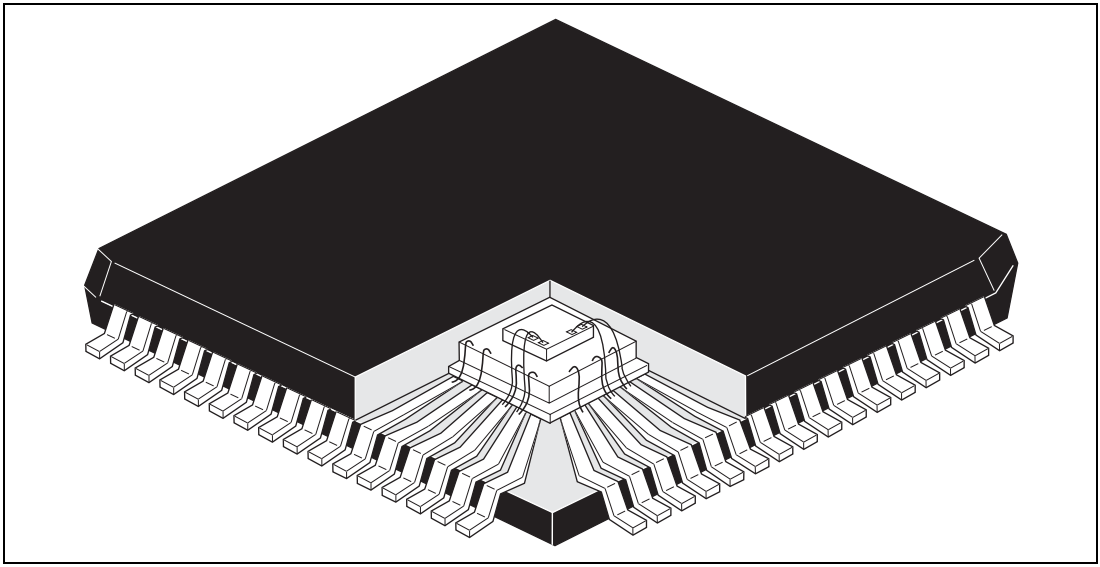
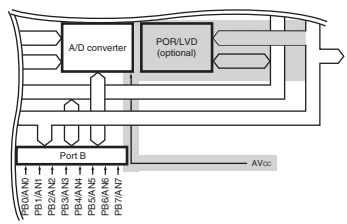
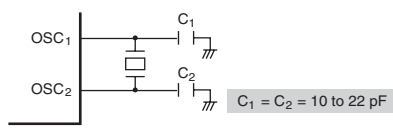
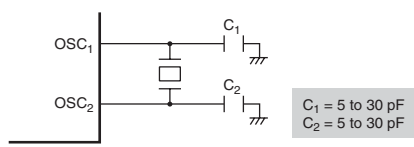
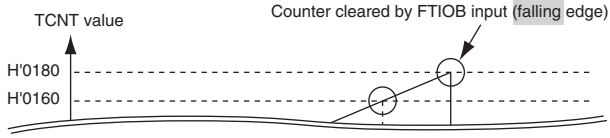
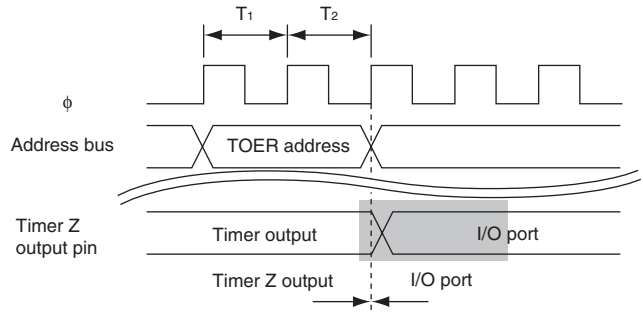
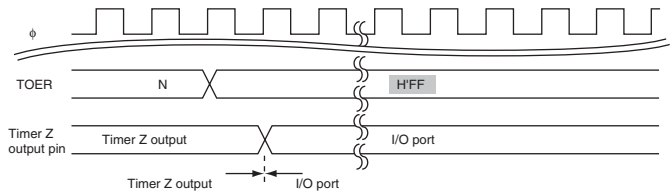


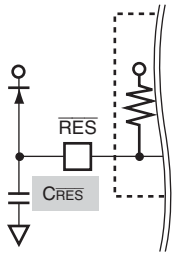
Figure E.1 EEPROM Stacked-Structure Cross-Sectional View

Main Revisions and Additions in this Edition

Item	Page	Revision (See Manual for Details)
Preface	vi, vii	Notes: When using the on-chip emulator (E7, E8) for H8/3687 program development and debugging, the following restrictions must be noted. 1. The $\overline{\text{NMI}}$ pin is reserved for the E7 or E8, and cannot be used. 3. Area H'D000 to H'DFFF is used by the E7 or E8, and is not available to the user. 5. When the E7 or E8 is used, address breaks can be set as either available to the user or for use by the E7 or E8. If address breaks are set as being used by the E7 or E8, the address break control registers must not be accessed. 6. When the E7 or E8 is used, $\overline{\text{NMI}}$ is an input/output pin (open-drain in output mode), P85 and P87 are input pins, and P86 is an output pin. 7. Use channel 1 of the SCI3 (P21/RXD, P22/TXD) in on-board programming mode by boot mode. Note has been deleted.
Section 1 Overview Figure 1.2 Internal Block Diagram of H8/3687N (EEPROM Stacked Version)	4	
Section 5 Clock Pulse Generators Figure 5.3 Typical Connection to Crystal Resonator	70	
Figure 5.5 Typical Connection to Ceramic Resonator	71	

Item	Page	Revision (See Manual for Details)
Section 6 Power-Down Modes 6.1.1 System Control Register 1 (SYSCR1)	76	**Bit** / **Bit Name** / **Description** 3 / NESEL / Noise Elimination Sampling Frequency Select The subclock pulse generator generates the watch clock signal (ϕ_W) and the system clock pulse generator generates the oscillator clock (ϕ_{OSC}). This bit selects the sampling frequency of the oscillator clock when the watch clock signal (ϕ_W) is sampled. When ϕ_{OSC} = 4 to 20 MHz, clear NESEL to 0.
Section 8 RAM	107	Note: * When the E7 or E8 is used, area H'F780 to H'FB7F must not be accessed.
Section 13 Timer Z Figure 13.17 Example of Input Capture Operation	208	
13.4.4 Synchronous Operation	211	Figure 13.20 shows an example of synchronous operation. In this example, synchronous operation has been selected, FTIOB0 and FTIOB1 have been designated for PWM mode, GRA_0 compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 counter clearing source. In addition, the same input clock has been set as the counter input clock for channel 0 and channel 1. Two-phase PWM waveforms are output from pins FTIOB0 and FTIOB1.
13.4.5 PWM Mode	213	Figure 13.22 shows an example of operation in PWM mode. The output signals go to 1 and TCNT is reset at compare match A, and the output signals go to 0 at compare match B, C, and D (TOB, TOC, and TOD = 1, POLB, POLC, and POLD = 0).
	214	Figures 13.24 (when TOB, TOC, and TOD = 1, POLB, POLC, and POLD = 0) and 13.25 (when TOB, TOC, and TOD = 0, POLB, POLC, and POLD = 1) show examples of the output of PWM waveforms with duty cycles of 0% and 100% in PWM mode.

Item	Page	Revision (See Manual for Details)
13.4.9 Timer Z Output Timing Figure 13.44 Example of Output Disable Timing of Timer Z by Writing to TOER	238	
Figure 13.45 Example of Output Disable Timing of Timer Z by External Trigger	238	
Section 14 Watchdog Timer 14.2.1 Timer Control/Status Register WD (TCSRWD)	252	**Bit** 4 — **Bit Name** TCSRWE — **Description** Timer Control/Status Register WD Write Enable
Section 17 I²C Bus Interface 2 (IIC2) 17.3.5 I²C Bus Status Register (ICSR)	314	**Bit** 3 — **Bit Name** STOP — **Description** Stop Condition Detection Flag [Setting conditions] • In master mode, when a stop condition is detected after frame transfer • In slave mode, when a stop condition is detected after the general call address or the first byte slave address, next to detection of start condition, accords with the address set in SAR
17.7 Usage Notes	336	Added
Section 18 A/D Converter 18.3.1 A/D Data Registers A to D (ADDRA to ADDRD)	340	Therefore byte access to ADDR should be done by reading the upper byte first then the lower one. Word access is also possible. ADDR is initialized to H'0000.

Item	Page	Revision (See Manual for Details)
Section 20 Power-On Reset and Low-Voltage Detection Circuits (Optional) Figure 20.1 Block Diagram of Power-On Reset Circuit and Low-Voltage Detection Circuit	362	

Section 23 Electrical Characteristics
Table 23.2 DC Characteristics (1)

394, 395

Item	Symbol	Applicable Pins	Test Condition	Values Min
Input high voltage	V_{IH}	PB0 to PB7	V_{CC} = 4.0 to 5.5 V	$V_{CC} \times 0.7$
				$V_{CC} \times 0.8$
Input low voltage	V_{IL}	RXD, RXD2, SCL, SDA, P10 to P12, : P85 to P87, PB0 to PB7	V_{CC} = 4.0 to 5.5 V	–0.3
				–0.3

398

Mode	$\overline{RES}$ Pin	Internal State
Active mode 1	V_{CC}	Operates
Active mode 2		Operates (φOSC/64)
Sleep mode 1	V_{CC}	Only timers operate
Sleep mode 2		Only timers operate (φOSC/64)

Item	Page	Revision (See Manual for Details)

Table 23.12 DC Characteristics (1) — page 413, 414

Item	Symbol	Applicable Pins	Test Condition	Values Min
Input high voltage	V_{IH}	PB0 to PB7	V_{CC} = 4.0 to 5.5 V	V_{CC} × 0.7
				V_{CC} × 0.8
Input low voltage	V_{IL}	RXD, RXD_2, SCL, SDA, P10 to P12, : P85 to P87, PB0 to PB7	V_{CC} = 4.0 to 5.5 V	–0.3
				–0.3

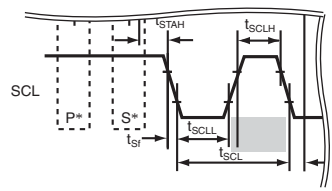
page 417

Mode	RES Pin	Internal State
Active mode 1	V_{CC}	Operates
Active mode 2		Operates (φOSC/64)
Sleep mode 1	V_{CC}	Only timers operate
Sleep mode 2		Only timers operate (φOSC/64)

Table 23.16 A/D Converter Characteristics — page 424

Item	Test Condition	Values Min
Conversion time (single mode)	AV_{CC} = 3.3 to 5.5 V	134

Figure 23.4 I²C Bus Interface Input/Output Timing — page 429



Appendix D Package Dimensions — page 488, 489

Swapped with new ones.

Index

Numerics

14-bit PWM 255
- Register settings 257
- Waveform output 258

A

A/D converter 337
- Sample-and-hold circuit 344
- Scan mode 343
- Single mode 343

Address break 63
Addressing modes 33
- Absolute address 34
- Immediate 35
- Memory indirect 35
- Program-counter relative 35
- Register direct 33
- Register indirect 34
- Register indirect with displacement 34
- Register indirect with post-increment 34
- Register indirect with pre-decrement 34

C

Clock pulse generators 69
- Prescaler S 73
- Prescaler W 73
- Subclock generator 72
- System clock generator 70

Condition field 32
Condition-code register (CCR) 17
CPU 11

E

EEPROM 349
- Acknowledge 353
- Acknowledge polling 356
- Byte write 354
- Current address read 356
- EEPROM interface 352
- Page write 355
- Random address read 357
- Sequential read 358
- Slave address reference register (ESAR) 353
- Slave addressing 353
- Start condition 352
- Stop condition 353

Effective address 36
Effective address extension 32
Exception handling 47
- Reset exception handling 57
- Stack status 60
- Trap instruction 47

F

Flash memory 89
- Boot mode 95
- Boot program 95
- Erase/erase-verify 102
- Erasing units 89
- Error protection 105
- Hardware protection 105
- Power-down states 106
- Program/program-verify 100
- Programmer mode 106
- Programming units 89
- Programming/erasing in user program mode 98
- Software protection 105

G

General registers 16

I

I/O ports 109
  I/O port block diagrams 463
I²C bus format 318
I²C bus interface 2 (IIC2) 303
  Acknowledge 319
  Bit synchronous circuit 335
  Clocked synchronous serial format 327
  Noise canceler 329
  Slave address 318
  Start condition 318
  Stop condition 319
  Transfer rate 307
Instruction set 22
  Arithmetic operations instructions 24
  Bit manipulation instructions 27
  Block data transfer instructions 31
  Branch instructions 29
  Data transfer instructions 23
  Logic operations instructions 26
  Shift instructions 26
  System control instructions 30
Internal power supply
step-down circuit 371
Interrupt
  Internal interrupts 59
  Interrupt response time 60
  IRQ3 to IRQ0 interrupts 57
  NMI interrupt 57
  WKP5 to WKP0 interrupts 58
Interrupt mask bit 18

L

Large current ports 2
Low-voltage detection circuit 361
LVDI 367
LVDI (interrupt by low
voltage detect) circuit 367
LVDR 366
LVDR (reset by low
voltage detect) circuit 366

M

Memory map 12
Module standby function 87

O

On-board programming modes 95
Operation field 32

P

Package 2
Package dimensions 488
Pin arrangement 5
Power-down modes 75
  Sleep mode 84
  Standby mode 84
  Subactive mode 85
  Subsleep mode 84
Power-on reset 361
Power-on reset circuit 365
Product code lineup 486
Program counter (PC) 17

R

Realtime clock (RTC) 141
  Data reading procedure 151
  Initial setting procedure 150
Register
  ABRKCR 64, 378, 384, 388
  ABRKSR 65, 378, 384, 388

ADCR 342, 377, 384, 388
ADCSR 341, 377, 384, 388
ADDRA 340, 377, 383, 388
ADDRB 340, 377, 384, 388
ADDRC 340, 377, 384, 388
ADDRD 340, 377, 384, 388
BARH 65, 378, 384, 388
BARL 65, 378, 384, 388
BDRH 66, 378, 384, 388
BDRL 66, 378, 384, 388
BRR 269, 377, 383, 388
EBR1 ..... 93, 376, 383, 387
EKR 351, 380, 385, 390
FENR 94, 376, 383, 387
FLMCR1 ..... 91, 376, 383, 387
FLMCR2 ..... 92, 376, 383, 387
FLPWCR 94, 376, 383, 387
GRA 190, 374, 381, 386
GRB 190, 374, 381, 386
GRC 190, 374, 381, 386
GRD 190, 374, 381, 386
ICCR1 ..... 306, 376, 382, 387
ICCR2 ..... 308, 376, 382, 387
ICDRR 317, 376, 383, 387
ICDRS 317
ICDRT 317, 376, 383, 387
ICIER 311, 376, 383, 387
ICMR 309, 376, 382, 387
ICSR 313, 376, 383, 387
IEGR1 ..... 50, 379, 385, 389
IEGR2 ..... 51, 379, 385, 389
IENR1 ..... 52, 379, 385, 389
IENR2 ..... 53, 379, 385, 389
IRR1 ..... 53, 379, 385, 389
IRR2 ..... 55, 379, 385, 389
IWPR 55, 379, 385, 389
LVDCR 362, 375, 382, 387
LVDSR 364, 375, 382, 387
MSTCR1 ..... 79, 379, 385, 389
MSTCR2 ..... 80, 379, 385, 389
PCR1 ..... 111, 379, 385, 389
PCR2 ..... 115, 379, 385, 389
PCR3 ..... 119, 379, 385, 389
PCR5 ..... 124, 379, 385, 389
PCR6 ..... 128, 379, 385, 389
PCR7 ..... 134, 379, 385, 389
PCR8 ..... 137, 379, 385, 389
PDR1 ..... 111, 378, 384, 388
PDR2 ..... 116, 378, 384, 389
PDR3 ..... 120, 378, 384, 389
PDR5 ..... 124, 378, 384, 389
PDR6 ..... 129, 378, 384, 389
PDR7 ..... 135, 378, 384, 389
PDR8 ..... 137, 378, 385, 389
PDRB 139, 379, 385, 389
PMR1 ..... 110, 379, 385, 389
PMR3 ..... 116, 379, 385, 389
PMR5 ..... 123, 379, 385, 389
POCR 197, 374, 381, 386
PUCR1 ..... 112, 378, 384, 388
PUCR5 ..... 125, 378, 384, 388
PWCR 256, 377, 384, 388
PWDRL 257, 377, 384, 388
PWDRU 257, 377, 384, 388
RDR 263, 377, 383, 388
RHRDR 145, 375, 382, 387
RMINDR 144, 375, 382, 387
RSECDR 143, 375, 382, 386
RSR 263
RTCCR1 ..... 147, 375, 382, 387
RTCCR2 ..... 148, 375, 382, 387
RTCCSR 149, 375, 382, 387
RWKDR 146, 375, 382, 387
SAR 316, 376, 383, 387
SCR3 ..... 265, 377, 383, 388
SMR 264, 377, 383, 388
SSR 267, 377, 383, 388
SYSCR1 ..... 76, 379, 385, 389
SYSCR2 ..... 78, 379, 385, 389
TCB1 ..... 155, 376, 383, 387

TCNT.....189, 374, 381, 386
TCNTV.....161, 377, 383, 388
TCORA 161, 377, 383, 388
TCORB.....161, 377, 383, 388
TCR 191, 374, 381, 386
TCRV0 .....162, 376, 383, 387
TCRV1 .....165, 377, 383, 388
TCSRV 164, 376, 383, 387
TCSRWD 252, 378, 384, 388
TCWD 253, 378, 384, 388
TDR 263, 377, 383, 388
TFCR 185, 375, 382, 386
TIER 196, 374, 381, 386
TIORA.....192, 374, 381, 386
TIORC 193, 374, 381, 386
TLB1 156
TMB1 .....155, 376, 383, 387
TMDR 183, 375, 382, 386
TMWD 253, 378, 384, 388
TOCR 188, 375, 382, 386
TOER.....187, 375, 382, 386
TPMR 184, 375, 382, 386
TSR.....194, 374, 381, 386
TSTR 182, 375, 382, 386
Register field 32

S

Serial communication interface 3 (SCI3) 259
Asynchronous mode 276
Bit rate 269
Break 300
Clocked synchronous mode..... 284
Framing error 280
Multiprocessor communication function 292
Overrun error 280
Parity error 280
Stacked-structure cross-sectional view of H8/3687N 491
Stack pointer (SP) 17

T

Timer B1 153
Auto-reload timer operation..... 156
Event counter operation 157
Interval timer operation..... 156
Timer V..... 159
Timer Z 175
Buffer operation..... 230
Complementary PWM mode 221
Input capture function 207
PWM mode 211
Reset synchronous PWM mode 217
Synchronous operation..... 210
Waveform output by compare match 203

V

Vector address..... 48

W

Watchdog timer..... 251

Renesas 16-Bit Single-Chip Microcomputer
Hardware Manual
H8/3687 Group

Publication Date: 1st Edition, Jul, 2001
Rev.5.00, Nov. 02, 2005
Published by: Sales Strategic Planning Div.
Renesas Technology Corp.
Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

©2005. Renesas Technology Corp. All rights reserved. Printed in Japan.

Renesas Technology Corp. Sales Strategic Planning Div. Nippon Bldg., 2-6-2, Ohte-machi, Chiyoda-ku, Tokyo 100-0004, Japan

RENESAS SALES OFFICES

http://www.renesas.com

Refer to "**http://www.renesas.com/en/network**" for the latest and detailed information.

Renesas Technology America, Inc.
450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, United Kingdom
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.
Unit2607 Ruijing Building, No.205 Maoming Road (S), Shanghai 200020, China
Tel: <86> (21) 6472-1001, Fax: <86> (21) 6415-2952

Renesas Technology Hong Kong Ltd.
7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.
10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.
1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.
Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> 2-796-3115, Fax: <82> 2-796-2145

Renesas Technology Malaysia Sdn. Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: <603> 7955-9390, Fax: <603> 7955-9510

Colophon 4.0

H8/3687 Group
Hardware Manual

Renesas Electronics Corporation
1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ09B0027-0500

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

Renesas Electronics:

HD64F3687GHV HD64F3687FPIV