To all our customers

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

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

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

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

## Cautions

Keep safety first in your circuit designs!

1. Renesas Technology Corporation puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage.
   Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corporation product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corporation or a third party.
2. Renesas Technology Corporation assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors.
   Renesas Technology Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corporation by various means, including the Renesas Technology Corporation Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corporation semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corporation or an authorized Renesas Technology Corporation product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corporation is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corporation for further details on these materials or the products contained therein.

# H8/3834 Series

H8/3837

HD6433837, HD6433837S, HD64473837

H8/3836

HD6433836, HD6433836S

H8/3835

HD6433835, HD6433835S

H8/3834

HD6433834, HD6433834S, HD6473834

H8/3833

HD6433833, HD6433833S

H8/3832

HD6433832S

Hardware Manual

ADE-602-054D
Rev. 5.0
3/5/03
Hitachi, Ltd.
MC-Setsu

# Notice

When using this document, keep the following in mind:

1. This document may, wholly or partially, be subject to change without notice.
2. All rights are reserved: No one is permitted to reproduce or duplicate, in any form, the whole or part of this document without Hitachi's permission.
3. Hitachi will not be held responsible for any damage to the user that may result from accidents or any other reasons during operation of the user's unit according to this document.
4. Circuitry and other examples described herein are meant merely to indicate the characteristics and performance of Hitachi's semiconductor products. Hitachi assumes no responsibility for any intellectual property claims or other problems that may result from applications based on the examples described herein.
5. No license is granted by implication or otherwise under any patents or other rights of any third party or Hitachi, Ltd.
6. MEDICAL APPLICATIONS: Hitachi's products are not authorized for use in MEDICAL APPLICATIONS without the written consent of the appropriate officer of Hitachi's sales company. Such use includes, but is not limited to, use in life support systems. Buyers of Hitachi's products are requested to notify the relevant Hitachi sales offices when planning to use the products in MEDICAL APPLICATIONS.

# Preface

The H8/300L Series of single-chip microcomputers has the high-speed H8/300L CPU at its core, with many necessary peripheral functions on-chip. The H8/300L CPU instruction set is compatible with the H8/300 CPU.

The H8/3834 Series has a system-on-a-chip architecture that includes such peripheral functions as an LCD controller/driver, five types of timers, a 14-bit PWM, a three-channel serial communication interface, and an A/D converter. This makes it ideal for use in systems requiring an LCD display.

This manual describes the hardware of the H8/3834 Series. For details on the H8/3834 Series instruction set, refer to the H8/300L Series Programming Manual.

Note: The terms H8/3834, H8/3834S, and H8/3834 Series used in the text refer to the products shown below.

1. H8/3834: HD6433837, HD6433836, HD6433835, HD6433834, HD6433833, HD6473837, HD6473834

2. H8/3834S: HD6433837S, HD6433836S, HD6433835S, HD6433834S, HD6433833S, HD6433832S

3. H8/3834 Series: All products, including the H8/3834 and H8/3834S

# Contents


Section 1 Overview ..... 1
1.1 Overview ..... 1
1.2 Internal Block Diagram ..... 5
1.3 Pin Arrangement and Functions ..... 6
1.3.1 Pin Arrangement ..... 6
1.3.2 Pin Functions ..... 8

Section 2 CPU ..... 13
2.1 Overview ..... 13
2.1.1 Features ..... 13
2.1.2 Address Space ..... 14
2.1.3 Register Configuration ..... 14
2.2 Register Descriptions ..... 15
2.2.1 General Registers ..... 15
2.2.2 Control Registers ..... 15
2.2.3 Initial Register Values ..... 17
2.3 Data Formats ..... 17
2.3.1 Data Formats in General Registers ..... 18
2.3.2 Memory Data Formats ..... 19
2.4 Addressing Modes ..... 20
2.4.1 Addressing Modes ..... 20
2.4.2 Effective Address Calculation ..... 22
2.5 Instruction Set ..... 26
2.5.1 Data Transfer Instructions ..... 28
2.5.2 Arithmetic Operations ..... 30
2.5.3 Logic Operations ..... 31
2.5.4 Shift Operations ..... 31
2.5.5 Bit Manipulations ..... 33
2.5.6 Branching Instructions ..... 37
2.5.7 System Control Instructions ..... 39
2.5.8 Block Data Transfer Instruction ..... 40
2.6 Basic Operational Timing ..... 41
2.6.1 Access to On-Chip Memory (RAM, ROM) ..... 41
2.6.2 Access to On-Chip Peripheral Modules ..... 42
2.7 CPU States ..... 43
2.7.1 Overview ..... 43
2.7.2 Program Execution State ..... 45
2.7.3 Program Halt State ..... 45
2.7.4 Exception-Handling State ..... 45

2.8 Memory Map ..... 46
2.8.1 Memory Map ..... 46
2.8.2 LCD RAM Address Relocation ..... 52
2.9 Application Notes ..... 53
2.9.1 Notes on Data Access ..... 53
2.9.2 Notes on Bit Manipulation ..... 55
2.9.3 Notes on Use of the EEPMOV Instruction ..... 61

Section 3 Exception Handling ..... 63
3.1 Overview ..... 63
3.2 Reset ..... 63
3.2.1 Overview ..... 63
3.2.2 Reset Sequence ..... 63
3.2.3 Interrupt Immediately after Reset ..... 65
3.3 Interrupts ..... 66
3.3.1 Overview ..... 66
3.3.2 Interrupt Control Registers ..... 67
3.3.3 External Interrupts ..... 75
3.3.4 Internal Interrupts ..... 76
3.3.5 Interrupt Operations ..... 76
3.3.6 Interrupt Response Time ..... 81
3.4 Application Notes ..... 81
3.4.1 Notes on Stack Area Use ..... 81
3.4.2 Notes on Rewriting Port Mode Registers ..... 82

Section 4 Clock Pulse Generators ..... 85
4.1 Overview ..... 85
4.1.1 Block Diagram ..... 85
4.1.2 System Clock and Subclock ..... 85
4.2 System Clock Generator ..... 86
4.3 Subclock Generator ..... 88
4.4 Prescalers ..... 91
4.5 Note on Oscillators ..... 92

Section 5 Power-Down Modes ..... 93
5.1 Overview ..... 93
5.1.1 System Control Registers ..... 96
5.2 Sleep Mode ..... 99
5.2.1 Transition to Sleep Mode ..... 99
5.2.2 Clearing Sleep Mode ..... 99
5.3 Standby Mode ..... 99
5.3.1 Transition to Standby Mode ..... 99
5.3.2 Clearing Standby Mode ..... 100

5.3.3 Oscillator Settling Time after Standby Mode is Cleared ..... 100
5.3.4 Transition to Standby Mode and Port Pin States ..... 101
5.4 Watch Mode ..... 101
5.4.1 Transition to Watch Mode ..... 101
5.4.2 Clearing Watch Mode ..... 102
5.4.3 Oscillator Settling Time after Watch Mode is Cleared ..... 102
5.5 Subsleep Mode ..... 102
5.5.1 Transition to Subsleep Mode ..... 102
5.5.2 Clearing Subsleep Mode ..... 103
5.6 Subactive Mode ..... 103
5.6.1 Transition to Subactive Mode ..... 103
5.6.2 Clearing Subactive Mode ..... 103
5.6.3 Operating Frequency in Subactive Mode ..... 103
5.7 Active (medium-speed) Mode ..... 104
5.7.1 Transition to Active (medium-speed) Mode ..... 104
5.7.2 Clearing Active (medium-speed) Mode ..... 104
5.7.3 Operating Frequency in Active (medium-speed) Mode ..... 104
5.8 Direct Transfer ..... 104
5.8.1 Direct Transfer Overview ..... 104
5.8.2 Calculation of Direct Transfer Time before Transition ..... 106

Section 6 ROM ..... 109
6.1 Overview ..... 109
6.1.1 Block Diagram ..... 109
6.2 H8/3834 PROM Mode ..... 110
6.2.1 Setting to PROM Mode ..... 110
6.2.2 Socket Adapter Pin Arrangement and Memory Map ..... 110
6.3 H8/3834 Programming ..... 113
6.3.1 Writing and Verifying ..... 113
6.3.2 Programming Precautions ..... 117
6.4 H8/3837 PROM Mode ..... 118
6.4.1 Setting to PROM Mode ..... 118
6.4.2 Socket Adapter Pin Arrangement and Memory Map ..... 118
6.5 H8/3837 Programming ..... 121
6.5.1 Writing and Verifying ..... 121
6.5.2 Programming Precautions ..... 126
6.6 Reliability of Programmed Data ..... 127

Section 7 RAM ..... 129
7.1 Overview ..... 129
7.1.1 Block Diagram ..... 129

Section 8 I/O Ports ..... 131

8.1 Overview ..... 131
8.2 Port 1 ..... 133
8.2.1 Overview ..... 133
8.2.2 Register Configuration and Description ..... 133
8.2.3 Pin Functions ..... 137
8.2.4 Pin States ..... 139
8.3 Port 2 ..... 140
8.3.1 Overview ..... 140
8.3.2 Register Configuration and Description ..... 140
8.3.3 Pin Functions ..... 144
8.3.4 Pin States ..... 144
8.4 Port 3 ..... 145
8.4.1 Overview ..... 145
8.4.2 Register Configuration and Description ..... 145
8.4.3 Pin Functions ..... 149
8.4.4 Pin States ..... 151
8.4.5 MOS Input Pull-Up ..... 151
8.5 Port 4 ..... 152
8.5.1 Overview ..... 152
8.5.2 Register Configuration and Description ..... 152
8.5.3 Pin Functions ..... 154
8.5.4 Pin States ..... 155
8.6 Port 5 ..... 155
8.6.1 Overview ..... 155
8.6.2 Register Configuration and Description ..... 156
8.6.3 Pin Functions ..... 158
8.6.4 Pin States ..... 158
8.6.5 MOS Input Pull-Up ..... 159
8.7 Port 6 ..... 159
8.7.1 Overview ..... 159
8.7.2 Register Configuration and Description ..... 160
8.7.3 Pin Functions ..... 161
8.7.4 Pin States ..... 162
8.7.5 MOS Input Pull-Up ..... 162
8.8 Port 7 ..... 163
8.8.1 Overview ..... 163
8.8.2 Register Configuration and Description ..... 163
8.8.3 Pin Functions ..... 164
8.8.4 Pin States ..... 165
8.9 Port 8 ..... 165
8.9.1 Overview ..... 165
8.9.2 Register Configuration and Description ..... 165
8.9.3 Pin Functions ..... 167

8.9.4 Pin States ... 167
8.10 Port 9 ... 168
8.10.1 Overview ... 168
8.10.2 Register Configuration and Description ... 168
8.10.3 Pin Functions ... 169
8.10.4 Pin States ... 171
8.11 Port A ... 171
8.11.1 Overview ... 171
8.11.2 Register Configuration and Description ... 172
8.11.3 Pin Functions ... 173
8.11.4 Pin States ... 174
8.12 Port B ... 175
8.12.1 Overview ... 175
8.12.2 Register Configuration and Description ... 175
8.13 Port C ... 176
8.13.1 Overview ... 176
8.13.2 Register Configuration and Description ... 176

Section 9 Timers ... 177
9.1 Overview ... 177
9.2 Timer A ... 178
9.2.1 Overview ... 178
9.2.2 Register Descriptions ... 180
9.2.3 Timer Operation ... 182
9.2.4 Timer A Operation States ... 183
9.3 Timer B ... 183
9.3.1 Overview ... 183
9.3.2 Register Descriptions ... 185
9.3.3 Timer Operation ... 187
9.3.4 Timer B Operation States ... 188
9.4 Timer C ... 188
9.4.1 Overview ... 188
9.4.2 Register Descriptions ... 190
9.4.3 Timer Operation ... 192
9.4.4 Timer C Operation States ... 194
9.5 Timer F ... 194
9.5.1 Overview ... 194
9.5.2 Register Descriptions ... 197
9.5.3 Interface with the CPU ... 203
9.5.4 Timer Operation ... 206
9.5.5 Application Notes ... 208
9.6 Timer G ... 210
9.6.1 Overview ... 210

9.6.2 Register Descriptions ..... 212
9.6.3 Noise Canceller Circuit ..... 215
9.6.4 Timer Operation ..... 217
9.6.5 Application Notes ..... 221
9.6.6 Sample Timer G Application ..... 224

Section 10 Serial Communication Interface ..... 225
10.1 Overview ..... 225
10.2 SCI1 ..... 225
10.2.1 Overview ..... 225
10.2.2 Register Descriptions ..... 227
10.2.3 Operation ..... 231
10.2.4 Interrupts ..... 234
10.2.5 Application Notes ..... 234
10.3 SCI2 ..... 234
10.3.1 Overview ..... 234
10.3.2 Register Descriptions ..... 236
10.3.3 Operation ..... 240
10.3.4 Interrupts ..... 247
10.3.5 Application Notes ..... 247
10.4 SCI3 ..... 248
10.4.1 Overview ..... 248
10.4.2 Register Descriptions ..... 250
10.4.3 Operation ..... 266
10.4.4 Operation in Asynchronous Mode ..... 270
10.4.5 Operation in Synchronous Mode ..... 278
10.4.6 Multiprocessor Communication Function ..... 285
10.4.7 Interrupts ..... 291
10.4.8 Application Notes ..... 292

Section 11 14-Bit PWM ..... 297
11.1 Overview ..... 297
11.1.1 Features ..... 297
11.1.2 Block Diagram ..... 297
11.1.3 Pin Configuration ..... 298
11.1.4 Register Configuration ..... 298
11.2 Register Descriptions ..... 298
11.2.1 PWM Control Register (PWCR) ..... 298
11.2.2 PWM Data Registers U and L (PWDRU, PWDRL) ..... 299
11.3 Operation ..... 299

Section 12 A/D Converter ..... 301
12.1 Overview ..... 301

12.1.1 Features ..... 301
12.1.2 Block Diagram ..... 301
12.1.3 Pin Configuration ..... 302
12.1.4 Register Configuration ..... 302
12.2 Register Descriptions ..... 303
12.2.1 A/D Result Register (ADRR) ..... 303
12.2.2 A/D Mode Register (AMR) ..... 303
12.2.3 A/D Start Register (ADSR) ..... 305
12.3 Operation ..... 306
12.3.1 A/D Conversion Operation ..... 306
12.3.2 Start of A/D Conversion by External Trigger Input ..... 306
12.4 Interrupts ..... 307
12.5 Typical Use ..... 307
12.6 Application Notes ..... 310

Section 13 LCD Controller/Driver ..... 311
13.1 Overview ..... 311
13.1.1 Features ..... 311
13.1.2 Block Diagram ..... 312
13.1.3 Pin Configuration ..... 313
13.1.4 Register Configuration ..... 313
13.2 Register Descriptions ..... 314
13.2.1 LCD Port Control Register (LPCR) ..... 314
13.2.2 LCD Control Register (LCR) ..... 316
13.3 Operation ..... 318
13.3.1 Settings Prior to LCD Display ..... 318
13.3.2 Relation of LCD RAM to Display ..... 320
13.3.3 Connection to HD66100 ..... 320
13.3.4 Operation in Power-Down Modes ..... 329
13.3.5 Boosting the LCD Driver Power Supply ..... 330

Section 14 Electrical Characteristics ..... 331
14.1 H8/3832S, H8/3833S, H8/3834S, H8/3835S, H8/3836S and H8/3837S Absolute Maximum Ratings (Standard Specifications) ..... 331
14.2 H8/3832S, H8/3833S and H8/3834S Electrical Characteristics (Standard Specifications) ..... 332
14.2.1 Power Supply Voltage and Operating Range ..... 332
14.2.2 DC Characteristics ..... 334
14.2.3 AC Characteristics ..... 339
14.2.4 A/D Converter Characteristics ..... 343
14.2.5 LCD Characteristics ..... 344
14.3 H8/3835S, H8/3836S and H8/3837S Electrical Characteristics (Standard Specifications) ..... 345

14.3.1 Power Supply Voltage and Operating Range ..... 345
14.3.2 DC Characteristics ..... 347
14.3.3 AC Characteristics ..... 352
14.3.4 A/D Converter Characteristics ..... 356
14.3.5 LCD Characteristics ..... 357
14.4 H8/3832S, H8/3833S, H8/3834S, H8/3835S, H8/3836S and H8/3837S Absolute Maximum Ratings (Wide Temperature Range (I-Spec) Version) ..... 358
14.5 H8/3832S, H8/3833S and H8/3834S Electrical Characteristics (Wide Temperature Range (I-Spec) Version) ..... 359
14.5.1 Power Supply Voltage and Operating Range ..... 359
14.5.2 DC Characteristics ..... 361
14.5.3 AC Characteristics ..... 366
14.5.4 A/D Converter Characteristics ..... 370
14.5.5 LCD Characteristics ..... 371
14.6 H8/3835S, H8/3836S and H8/3837S Electrical Characteristics (Wide Temperature Range (I-Spec) Version) ..... 373
14.6.1 Power Supply Voltage and Operating Range ..... 373
14.6.2 DC Characteristics ..... 375
14.6.3 AC Characteristics ..... 380
14.6.4 A/D Converter Characteristics ..... 384
14.6.5 LCD Characteristics ..... 385
14.7 H8/3833, H8/3834, H8/3835, H8/3836, and H8/3837 (Standard Specification) Absolute Maximum Ratings ..... 387
14.8 H8/3833 and H8/3834 Electrical Characteristics (Standard Specifications) ..... 388
14.8.1 Power Supply Voltage and Operating Range ..... 388
14.8.2 DC Characteristics ..... 390
14.8.3 AC Characteristics ..... 395
14.8.4 A/D Converter Characteristics ..... 399
14.8.5 LCD Characteristics ..... 400
14.9 H8/3835 and H8/3836 and H8/3837 (Standard Specifications) Electrical Characteristics ..... 401
14.9.1 Power Supply Voltage and Operating Range ..... 401
14.9.2 DC Characteristics ..... 403
14.9.3 AC Characteristics ..... 408
14.9.4 A/D Converter Characteristics ..... 412
14.9.5 LCD Characteristics ..... 413
14.10 H8/3833, H8/3834, H8/3835, H8/3836, and H8/3837 Absolute Maximum Ratings (Wide Temperature Range (I-Spec) Version) ..... 414
14.11 H8/3833 and H8/3834 Electrical Characteristics (Wide Temperature Range (I-Spec) Version) ..... 415
14.11.1 Power Supply Voltage and Operating Range ..... 415
14.11.2 DC Characteristics ..... 417
14.11.3 AC Characteristics ..... 422

14.11.4 A/D Converter Characteristics ..... 426
14.11.5 LCD Characteristics ..... 427
14.12 H8/3835, H8/3836, and H8/3837 Electrical Characteristics (Wide Temperature Range (I-Spec) Version) ..... 429
14.12.1 Power Supply Voltage and Operating Range ..... 429
14.12.2 DC Characteristics ..... 431
14.12.3 AC Characteristics ..... 436
14.12.4 A/D Converter Characteristics ..... 440
14.12.5 LCD Characteristics ..... 441
14.13 Operation Timing ..... 443
14.14 Output Load Circuit ..... 448

Appendix A CPU Instruction Set ..... 449
A.1 Instructions ..... 449
A.2 Operation Code Map ..... 457
A.3 Number of Execution States ..... 459

Appendix B On-Chip Registers ..... 466
B.1 I/O Registers (1) ..... 466
B.2 I/O Registers (2) ..... 470

Appendix C I/O Port Block Diagrams ..... 513
C.1 Block Diagram of Port 1 ..... 513
C.2 Block Diagram of Port 2 ..... 518
C.3 Block Diagram of Port 3 ..... 521
C.4 Block Diagram of Port 4 ..... 527
C.5 Block Diagram of Port 5 ..... 530
C.6 Block Diagram of Port 6 ..... 531
C.7 Block Diagram of Port 7 ..... 532
C.8 Block Diagram of Port 8 ..... 533
C.9 Block Diagram of Port 9 ..... 534
C.10 Block Diagram of Port A ..... 535
C.11 Block Diagram of Port B ..... 536
C.12 Block Diagram of Port C ..... 536

Appendix D Port States in the Different Processing States ..... 537

Appendix E List of Products Codes ..... 538

Appendix F Package Dimensions ..... 542

# Section 1 Overview

## 1.1 Overview

The H8/300L Series is a series of single-chip microcomputers (MCU: microcomputer unit), built around the high-speed H8/300L CPU and equipped with peripheral system functions on-chip.

Within the H8/300L Series, the H8/3834 Series features an on-chip liquid crystal display (LCD) controller/driver. Other on-chip peripheral functions include five timers, a 14-bit pulse width modulator (PWM), three serial communication interface channels, and an analog-to-digital (A/D) converter. Together these functions make the H8/3834 Series ideally suited for embedded control of systems requiring an LCD display. On-chip memory is 16 kbytes of ROM and 1 kbyte of RAM in the H8/3832S, 24 kbytes of ROM and 1 kbyte of RAM in the H8/3833(S), 32 kbytes of ROM and 1 kbyte of RAM in the H8/3834(S), 40 kbytes of ROM and 2 kbytes of RAM in the H8/3835(S), 48 kbytes of ROM and 2 kbytes of RAM in the H8/3836(S), and 60 kbytes of ROM and 2 kbytes of RAM in the H8/3837(S).

The H8/3834 and H8/3837 both include a ZTAT™ version*, featuring a user-programmable on-chip PROM.

Table 1.1 summarizes the features of the H8/3834 Series.

Note: * ZTAT is a trademark of Hitachi, Ltd.

**Table 1.1 Features**

| Item | Description |
|---|---|
| CPU | High-speed H8/300L CPU<br>• General-register architecture<br>General registers: Sixteen 8-bit registers (can be used as eight 16-bit registers)<br>• Operating speed<br>— Max. operating speed: 5 MHz<br>— Add/subtract: 0.4 µs (operating at 5 MHz)<br>— Multiply/divide: 2.8 µs (operating at 5 MHz)<br>— Can run on 32.768 kHz subclock<br>• Instruction set compatible with H8/300 CPU<br>— Instruction length of 2 bytes or 4 bytes<br>— Basic arithmetic operations between registers<br>— MOV instruction for data transfer between memory and registers |

**Table 1.1 Features (cont)**

| Item | Description |
|---|---|
| CPU | Typical instructions<br>• Multiply (8 bits × 8 bits)<br>• Divide (16 bits ÷ 8 bits)<br>• Bit accumulator<br>• Register-indirect designation of bit position |
| Interrupts | 33 interrupt sources<br>• 13 external interrupt pins: $IRQ_4$ to $IRQ_0$, $WKP_7$ to $WKP_0$<br>• 20 internal interrupt sources |
| Clock pulse generators | Two on-chip clock pulse generators<br>• System clock pulse generator: 1 to 10 MHz<br>• Subclock pulse generator: 32.768 kHz |
| Power-down modes | Six power-down modes<br>• Sleep mode<br>• Standby mode<br>• Watch mode<br>• Subsleep mode<br>• Subactive mode<br>• Active (medium-speed) mode |
| Memory | Large on-chip memory<br>• H8/3832S: 16-kbyte ROM, 1-kbyte RAM<br>• H8/3833(S): 24-kbyte ROM, 1-kbyte RAM<br>• H8/3834(S): 32-kbyte ROM, 1-kbyte RAM<br>• H8/3835(S): 40-kbyte ROM, 2-kbyte RAM<br>• H8/3836(S): 48-kbyte ROM, 2-kbyte RAM<br>• H8/3837(S): 60-kbyte ROM, 2-kbyte RAM |
| I/O ports | 84 I/O port pins<br>• I/O pins: 71<br>• Input pins: 13 |
| Timers | Five on-chip timers<br>• Timer A: 8-bit timer<br>Count-up timer with selection of eight internal clock signals divided from the system clock ($\phi$)* and four clock signals divided from the watch clock ($\phi_w$)* |

Note: * $\phi$ and $\phi_w$ are defined in section 4, Clock Pulse Generators.

**Table 1.1 Features (cont)**

| Item | Description |
|---|---|
| Timers | • Timer B: 8-bit timer<br>— Count-up timer with selection of seven internal clock signals or event input from external pin<br>— Auto-reloading<br>• Timer C: 8-bit timer<br>— Count-up/count-down timer with selection of seven internal clock signals or event input from external pin<br>— Auto-reloading<br>• Timer F: 16-bit timer<br>— Can be used as two independent 8-bit timers.<br>— Count-up timer with selection of four internal clock signals or event input from external pin<br>— Compare-match function with toggle output<br>• Timer G: 8-bit timer<br>— Count-up timer with selection of four internal clock signals<br>— Input capture function with built-in noise canceller circuit |
| Serial communication interface | Three channels on chip<br>• SCI1: synchronous serial interface<br>Choice of 8-bit or 16-bit data transfer<br>• SCI2: 8-bit synchronous serial interface<br>Automatic transfer of 32-byte data segments<br>• SCI3: 8-bit synchronous or asynchronous serial interface<br>Built-in function for multiprocessor communication |
| 14-bit PWM | Pulse-division PWM output for reduced ripple<br>• Can be used as a 14-bit D/A converter by connecting to an external low-pass filter. |
| A/D converter | • Successive approximations using a resistance ladder resolution: 8 bits<br>• 12-channel analog input port<br>• Conversion time: 31/$\phi$ or 62/$\phi$ per channel |
| LCD controller/driver | Up to 40 segment pins and 4 common pins<br>• Choice of four duty cycles (static, 1/2, 1/3, 1/4)<br>• Segments can be expanded externally<br>• Segment pins can be switched to general-purpose ports in groups of four |

**Table 1.1 Features (cont)**

| Item | Description | | | |
|---|---|---|---|---|
| Product lineup | Product Code | | | |
| | Mask ROM Version | ZTAT™ Version | Package | ROM/RAM Size |
| | HD6433832SH | — | 100-pin QFP (FP-100B) | ROM: 16 kbytes |
| | HD6433832SF | — | 100-pin QFP (FP-100A) | RAM: 1 kbyte |
| | HD6433832SX | — | 100-pin TQFP (TFP-100B) | |
| | HD6433833H | — | 100-pin QFP (FP-100B) | ROM: 24 kbytes |
| | HD6433833SH | | | RAM: 1 kbyte |
| | HD6433833F | — | 100-pin QFP (FP-100A) | |
| | HD6433833SF | | | |
| | HD6433833X | — | 100-pin TQFP (TFP-100B) | |
| | HD6433833SX | | | |
| | HD6433834H | HD6473834H | 100-pin QFP (FP-100B) | ROM: 32 kbytes |
| | HD6433834SH | | | RAM: 1 kbyte |
| | HD6433834F | HD6473834F | 100-pin QFP (FP-100A) | |
| | HD6433834SF | | | |
| | HD6433834X | HD6473834X | 100-pin TQFP (TFP-100B) | |
| | HD6433834SX | | | |
| | HD6433835H | — | 100-pin QFP (FP-100B) | ROM: 40 kbytes |
| | HD6433835SH | | | RAM: 2 kbytes |
| | HD6433835F | — | 100-pin QFP (FP-100A) | |
| | HD6433835SF | | | |
| | HD6433835X | — | 100-pin TQFP (TFP-100B) | |
| | HD6433835SX | | | |
| | HD6433836H | — | 100-pin QFP (FP-100B) | ROM: 48 kbytes |
| | HD6433836SH | | | RAM: 2 kbytes |
| | HD6433836F | — | 100-pin QFP (FP-100A) | |
| | HD6433836SF | | | |
| | HD6433836X | — | 100-pin TQFP (TFP-100B) | |
| | HD6433836SX | | | |
| | HD6433837H | HD6473837H | 100-pin QFP (FP-100B) | ROM: 60 kbytes |
| | HD6433837SH | | | RAM: 2 kbytes |
| | HD6433837F | HD6473837F | 100-pin QFP (FP-100A) | |
| | HD6433837SF | | | |
| | HD6433837X | HD6473837X | 100-pin TQFP (TFP-100B) | |
| | HD6433837SX | | | |
| | HD6433832SD | — | 100-pin QFP (FP-100B) | ROM: 16 kbytes |
| | HD6433832SE | — | 100-pin QFP (FP-100A) | RAM: 1 kbyte |
| | HD6433832SL | — | 100-pin TQFP (TFP-100B) | WTR (I-spec) |
| | HD6433833D | — | 100-pin QFP (FP-100B) | ROM: 24 kbytes |
| | HD6433833SD | | | RAM: 1 kbyte |
| | HD6433833E | — | 100-pin QFP (FP-100A) | WTR (I-spec) |
| | HD6433833SE | | | |
| | HD6433833L | — | 100-pin TQFP (TFP-100B) | |
| | HD6433833SL | | | |
| | HD6433834D | HD6473834D | 100-pin QFP (FP-100B) | ROM: 32 kbytes |
| | HD6433834SD | | | RAM: 1 kbyte |
| | HD6433834E | HD6473834E | 100-pin QFP (FP-100A) | WTR (I-spec) |
| | HD6433834SE | | | |
| | HD6433834L | — | 100-pin TQFP (TFP-100B) | |
| | HD6433834SL | | | |
| | HD6433835D | — | 100-pin QFP (FP-100B) | ROM: 40 kbytes |
| | HD6433835SD | | | RAM: 2 kbytes |
| | HD6433835E | — | 100-pin QFP (FP-100A) | WTR (I-spec) |
| | HD6433835SE | | | |
| | HD6433835L | — | 100-pin TQFP (TFP-100B) | |
| | HD6433835SL | | | |
| | HD6433836D | — | 100-pin QFP (FP-100B) | ROM: 48 kbytes |
| | HD6433836SD | | | RAM: 2 kbytes |
| | HD6433836E | — | 100-pin QFP (FP-100A) | WTR (I-spec) |
| | HD6433836SE | | | |
| | HD6433836L | — | 100-pin TQFP (TFP-100B) | |
| | HD6433836SL | | | |
| | HD6433837D | HD6473837D | 100-pin QFP (FP-100B) | ROM: 60 kbytes |
| | HD6433837SD | | | RAM: 2 kbytes |
| | HD6433837E | HD6473837E | 100-pin QFP (FP-100A) | WTR (I-spec) |
| | HD6433837SE | | | |
| | HD6433837L | HD6473837L | 100-pin TQFP (TFP-100B) | |
| | HD6433837SL | | | |

## 1.2 Internal Block Diagram

Figure 1.1 shows a block diagram of the H8/3834 Series.

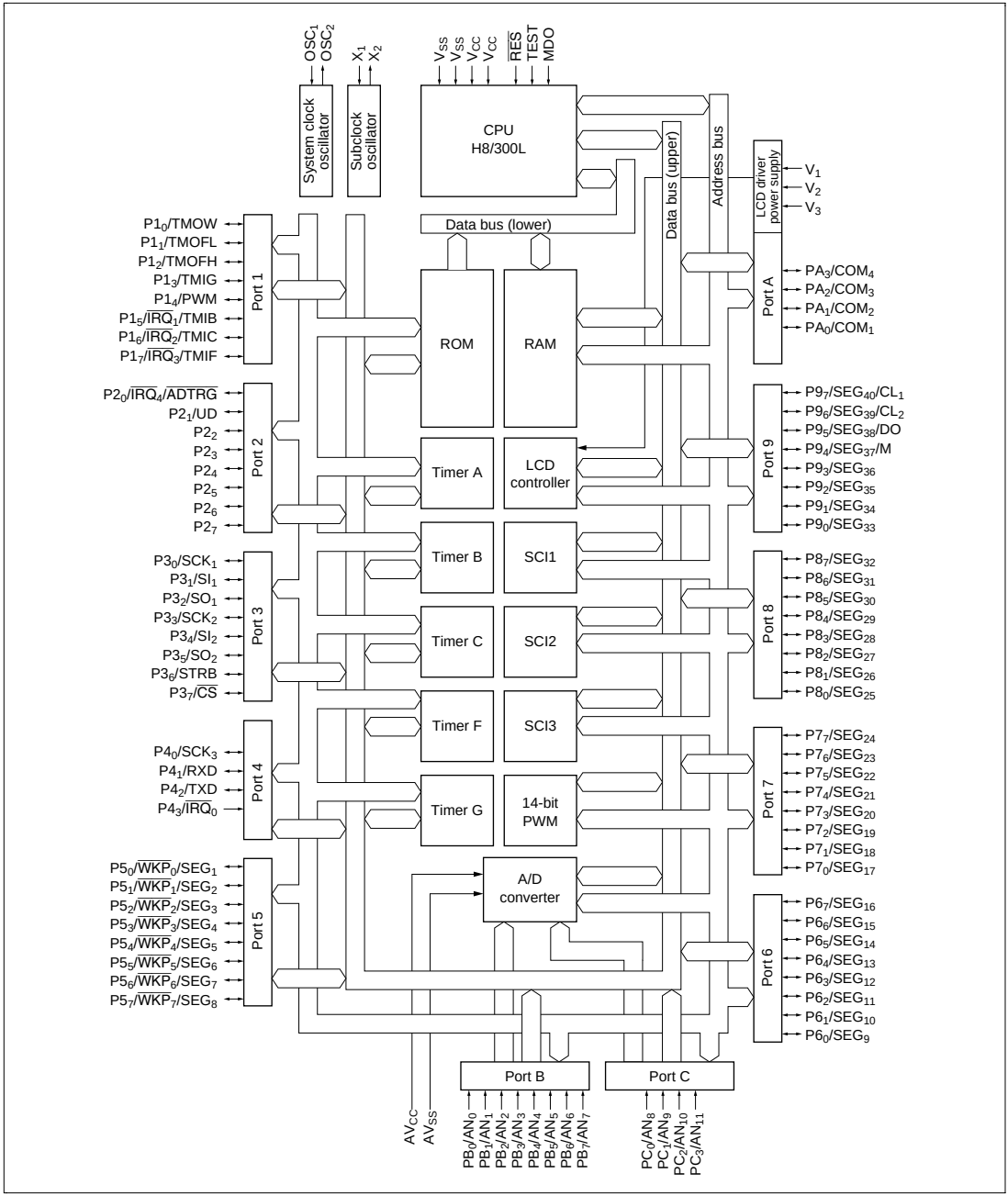



**Figure 1.1 Block Diagram**

# 1.3 Pin Arrangement and Functions

## 1.3.1 Pin Arrangement

The pin arrangement of the H8/3834 Series is shown in figures 1.2 and 1.3.

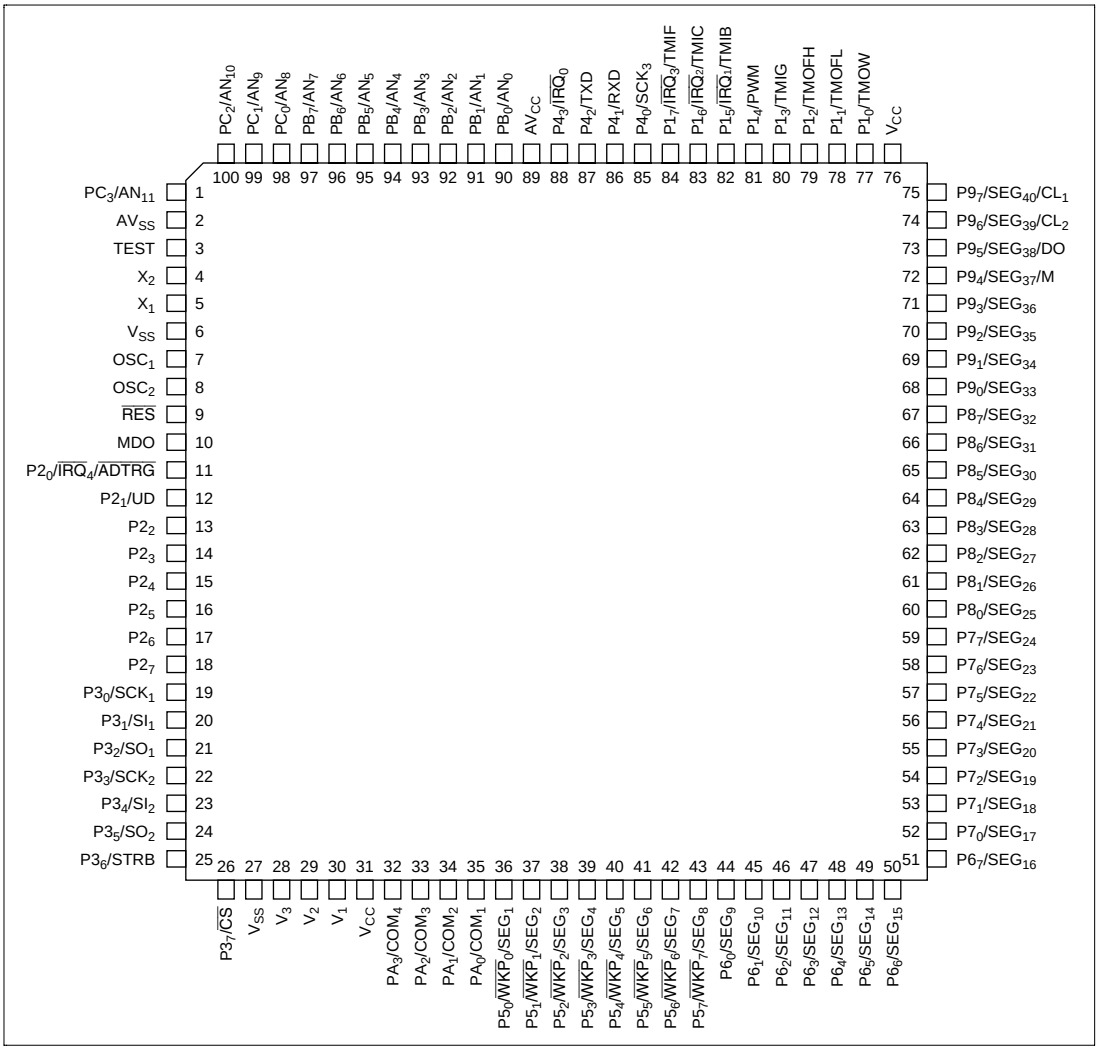



**Figure 1.2 Pin Arrangement (FP-100B, TFP-100B: Top View)**

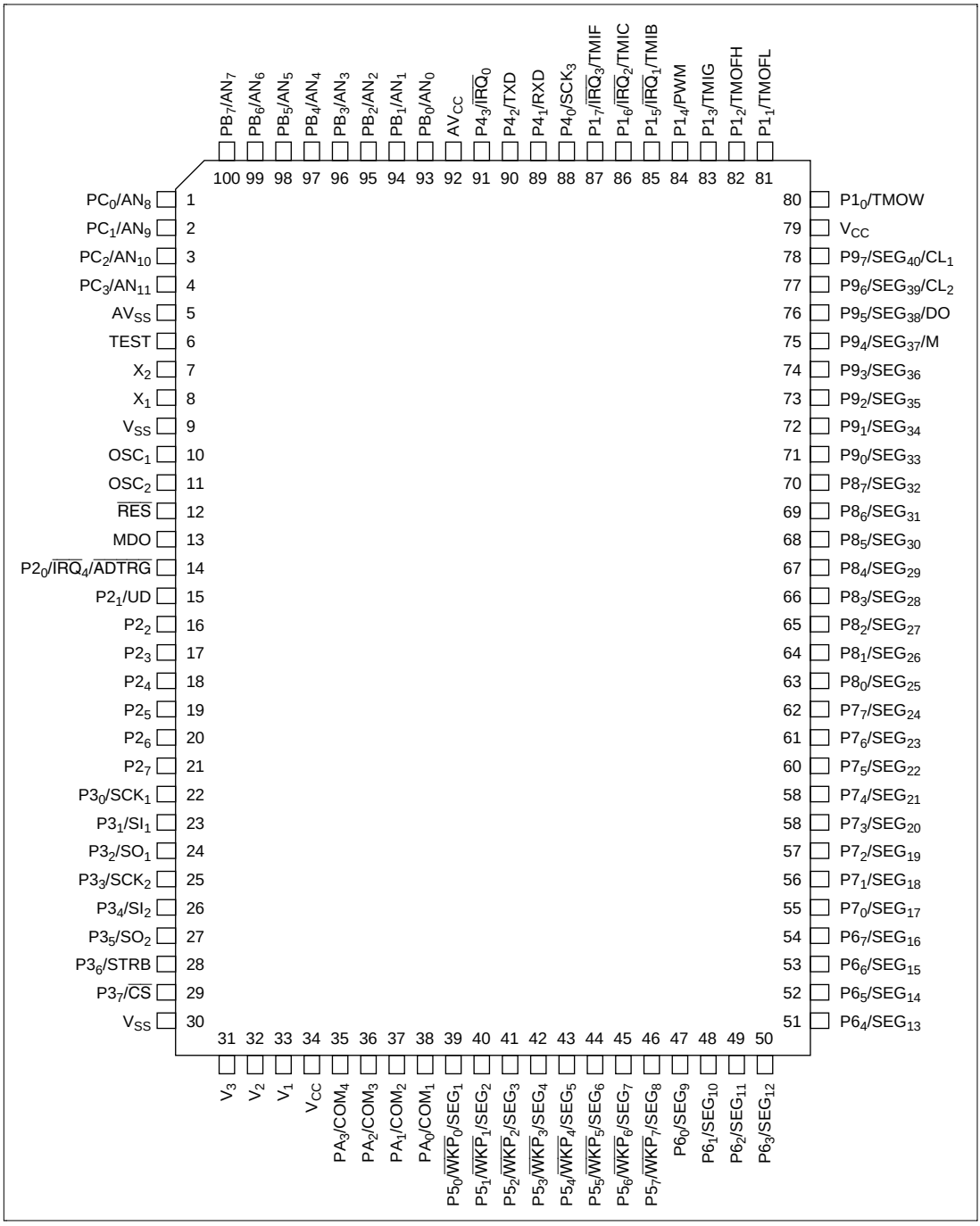



**Figure 1.3 Pin Arrangement (FP-100A: Top View)**

### 1.3.2 Pin Functions

Table 1.2 outlines the pin functions of the H8/3834 Series.

**Table 1.2 Pin Functions**

| | | Pin No. | | | |
|---|---|---|---|---|---|
| **Type** | **Symbol** | **FP-100B TFP-100B** | **FP-100A** | **I/O** | **Name and Functions** |
| Power source pins | $V_{CC}$ | 31, 76 | 34, 79 | Input | **Power supply:** All $V_{CC}$ pins should be connected to the system power supply (+5 V) |
| | $V_{SS}$ | 6, 27 | 9, 30 | Input | **Ground:** All $V_{SS}$ pins should be connected to the system power supply (0 V) |
| | $AV_{CC}$ | 89 | 92 | Input | **Analog power supply:** This is the power supply pin for the A/D converter. When the A/D converter is not used, connect this pin to the system power supply (+5 V). |
| | $AV_{SS}$ | 2 | 5 | Input | **Analog ground:** This is the A/D converter ground pin. It should be connected to the system power supply (0 V). |
| | $V_1$, $V_2$, $V_3$ | 30, 29, 28 | 33, 32, 31 | Input | **LCD power supply:** These are power supply pins for the LCD controller/ driver. A built-in resistor divider is provided for the power supply, so these pins are normally left open.<br>Power supply conditions are $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$. |
| Clock pins | $OSC_1$ | 7 | 10 | Input | This pin connects to a crystal or ceramic oscillator, or can be used to input an external clock. |
| | $OSC_2$ | 8 | 11 | Output | See section 4, Clock Pulse Generators, for a typical connection diagram. |
| | $X_1$ | 5 | 8 | Input | This pin connects to a 32.768-kHz crystal oscillator. |
| | $X_2$ | 4 | 7 | Output | See section 4, Clock Pulse Generators, for a typical connection diagram. |

**Table 1.2 Pin Functions (cont)**

| Type | Symbol | Pin No. FP-100B TFP-100B | Pin No. FP-100A | I/O | Name and Functions |
|---|---|---|---|---|---|
| System control | $\overline{RES}$ | 9 | 12 | Input | **Reset:** When this pin is driven low, the chip is reset |
| | MDO | 10 | 13 | Input | **Mode:** This pin controls system clock oscillation in the reset state |
| | TEST | 3 | 6 | Input | **Test:** This is a test pin, not for use in application systems. It should be connected to $V_{SS}$. |
| Interrupt pins | $\overline{IRQ}_0$<br>$\overline{IRQ}_1$<br>$\overline{IRQ}_2$<br>$\overline{IRQ}_3$<br>$\overline{IRQ}_4$ | 88<br>82<br>83<br>84<br>11 | 91<br>85<br>86<br>87<br>14 | Input | **External interrupt request 0 to 4:** These are input pins for external interrupts for which there is a choice between rising and falling edge sensing |
| | $\overline{WKP}_7$ to $\overline{WKP}_0$ | 43 to 36 | 46 to 39 | Input | **Wakeup interrupt request 0 to 7:** These are input pins for external interrupts that are detected at the falling edge |
| Timer pins | TMOW | 77 | 80 | Output | **Clock output:** This is an output pin for waveforms generated by the timer A output circuit |
| | TMIB | 82 | 85 | Input | **Timer B event counter input:** This is an event input pin for input to the timer B counter |
| | TMIC | 83 | 86 | Input | **Timer C event counter input:** This is an event input pin for input to the timer C counter |
| | UD | 12 | 15 | Input | **Timer C up/down select:** This pin selects whether the timer C counter isused for up- or down-counting. At high level it selects up-counting, and at low level down-counting. |
| | TMIF | 84 | 87 | Input | **Timer F event counter input:** This is an event input pin for input to the timer F counter |
| | TMOFL | 78 | 81 | Output | **Timer FL output:** This is an output pin for waveforms generated by the timer FL output compare function |

**Table 1.2 Pin Functions (cont)**

| Type | Symbol | Pin No. FP-100B TFP-100B | Pin No. FP-100A | I/O | Name and Functions |
|---|---|---|---|---|---|
| Timer pins | TMOFH | 79 | 82 | Output | **Timer FH output:** This is an output pin for waveforms generated by the timer FH output compare function |
| | TMIG | 80 | 83 | Input | **Timer G capture input:** This is an input pin for the timer G input capture function |
| 14-bit PWM pin | PWM | 81 | 84 | Output | **14-bit PWM output:** This is an output pin for waveforms generated by the 14-bit PWM |
| I/O ports | $PB_7$ to $PB_0$ | 97 to 90 | 100 to 93 | Input | **Port B:** This is an 8-bit input port |
| | $PC_3$ to $PC_0$ | 1, 100 to 98 | 4 to 1 | Input | **Port C:** This is a 4-bit input port |
| | $P4_3$ | 88 | 91 | Input | **Port 4 (bit 3):** This is a 1-bit input port |
| | $P4_2$ to $P4_0$ | 87 to 85 | 90 to 88 | I/O | **Port 4 (bits 2 to 0):** This is a 3-bit I/O port. Input or output can be designated for each bit by means of port control register 4 (PCR4). |
| | $PA_3$ to $PA_0$ | 32 to 35 | 35 to 38 | I/O | **Port A:** This is a 4-bit I/O port. Input or output can be designated for each bit by means of port control register A (PCRA). |
| | $P1_7$ to $P1_0$ | 84 to 77 | 87 to 80 | I/O | **Port 1:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 1 (PCR1). |
| | $P2_7$ to $P2_0$ | 18 to 11 | 21 to 14 | I/O | **Port 2:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 2 (PCR2). |
| | $P3_7$ to $P3_0$ | 26 to 19 | 29 to 22 | I/O | **Port 3:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 3 (PCR3). |
| | $P5_7$ to $P5_0$ | 43 to 36 | 46 to 39 | I/O | **Port 5:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 5 (PCR5). |

**Table 1.2 Pin Functions (cont)**

| Type | Symbol | Pin No. FP-100B TFP-100B | Pin No. FP-100A | I/O | Name and Functions |
|---|---|---|---|---|---|
| I/O ports | $P6_7$ to $P6_0$ | 51 to 44 | 54 to 47 | I/O | **Port 6:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 6 (PCR6). |
| | $P7_7$ to $P7_0$ | 59 to 52 | 62 to 55 | I/O | **Port 7:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 7 (PCR7). |
| | $P8_7$ to $P8_0$ | 67 to 60 | 70 to 63 | I/O | **Port 8:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 8 (PCR8). |
| | $P9_7$ to $P9_0$ | 75 to 68 | 78 to 71 | I/O | **Port 9:** This is an 8-bit I/O port. Input or output can be designated for each bit by means of port control register 9 (PCR9). |
| Serial communication interface (SCI) | $SI_1$ | 20 | 23 | Input | **SCI1 receive data input:** This is the SCI1 data input pin |
| | $SO_1$ | 21 | 24 | Output | **SCI1 send data output:** This is the SCI1 data output pin |
| | $SCK_1$ | 19 | 22 | I/O | **SCI1 clock I/O :** This is the SCI1 clock I/O pin |
| | $SI_2$ | 23 | 26 | Input | **SCI2 receive data input:** This is the SCI2 data input pin |
| | $SO_2$ | 24 | 27 | Output | **SCI2 send data output:** This is the SCI2 data output pin |
| | $SCK_2$ | 22 | 25 | I/O | **SCI2 clock I/O :** This is the SCI2 clock I/O pin |
| | $\overline{CS}$ | 26 | 29 | Input | **SCI2 chip select input:** This pin controls the start of SCI2 transfers |
| | STRB | 25 | 28 | Output | **SCI2 strobe output:** This pin outputs a strobe pulse each time a byte of data is transferred |
| | RXD | 86 | 89 | Input | **SCI3 receive data input:** This is the SCI3 data input pin |

**Table 1.2 Pin Functions (cont)**

| Type | Symbol | Pin No. FP-100B TFP-100B | Pin No. FP-100A | I/O | Name and Functions |
|---|---|---|---|---|---|
| Serial communication interface (SCI) | TXD | 87 | 90 | Output | **SCI3 send data output:** This is the SCI3 data output pin |
| | $SCK_3$ | 85 | 88 | I/O | **SCI3 clock I/O :** This is the SCI3 clock I/O pin |
| A/D converter | $AN_{11}$ to $AN_0$ | 1, 100 to 90 | 4 to 1<br>100 to 93 | Input | **Analog input channels 0 to 11:** These are analog data input channels to the A/D converter |
| | $\overline{ADTRG}$ | 11 | 14 | Input | **A/D converter trigger input:** This is the external trigger input pin to the A/D converter |
| LCD controller/ driver | $COM_4$ to $COM_1$ | 32 to 35 | 35 to 38 | Output | **LCD common output:** These are LCD common output pins |
| | $SEG_{40}$ to $SEG_1$ | 75 to 36 | 78 to 39 | Output | **LCD segment output:** These are LCD segment output pins |
| | $CL_1$ | 75 | 78 | Output | **LCD latch clock:** This is the display data latch clock output pin for external segment expansion |
| | $CL_2$ | 74 | 77 | Output | **LCD shift clock:** This is the display data shift clock output pin for external segment expansion |
| | DO | 73 | 76 | Output | **LCD serial data output:** This is the serial display data output pin for external segment expansion |
| | M | 72 | 75 | Output | **LCD alternating signal output:** This is the LCD alternating signal output pin for external segment expansion |

# Section 2 CPU

## 2.1 Overview

The H8/300L CPU has sixteen 8-bit general registers, which can also be paired as eight 16-bit registers. Its concise, optimized instruction set is designed for high-speed operation.

### 2.1.1 Features

Features of the H8/300L CPU are listed below.

- General-register architecture
  Sixteen 8-bit general registers, also usable as eight 16-bit general registers
- Instruction set with 55 basic instructions, including:
  — Multiply and divide instructions
  — Powerful bit-manipulation instructions
- Eight addressing modes
  — Register direct
  — Register indirect
  — Register indirect with displacement
  — Register indirect with post-increment or pre-decrement
  — Absolute address
  — Immediate
  — Program-counter relative
  — Memory indirect
- 64-kbyte address space
- High-speed operation
  — All frequently used instructions are executed in two to four states
  — High-speed arithmetic and logic operations
    8- or 16-bit register-register add or subtract: 0.4 µs*
    8 × 8-bit multiply: 2.8 µs*
    16 ÷ 8-bit divide: 2.8 µs*

Note: * These values are at $\phi$ = 5 MHz.

- Low-power operation modes
  SLEEP instruction for transfer to low-power operation

### 2.1.2 Address Space

The H8/300L CPU supports an address space of up to 64 kbytes for storing program code and data.

See 2.8, Memory Map, for details of the memory map.

### 2.1.3 Register Configuration

Figure 2.1 shows the register structure of the H8/300L CPU. There are two groups of registers: the general registers and control registers.

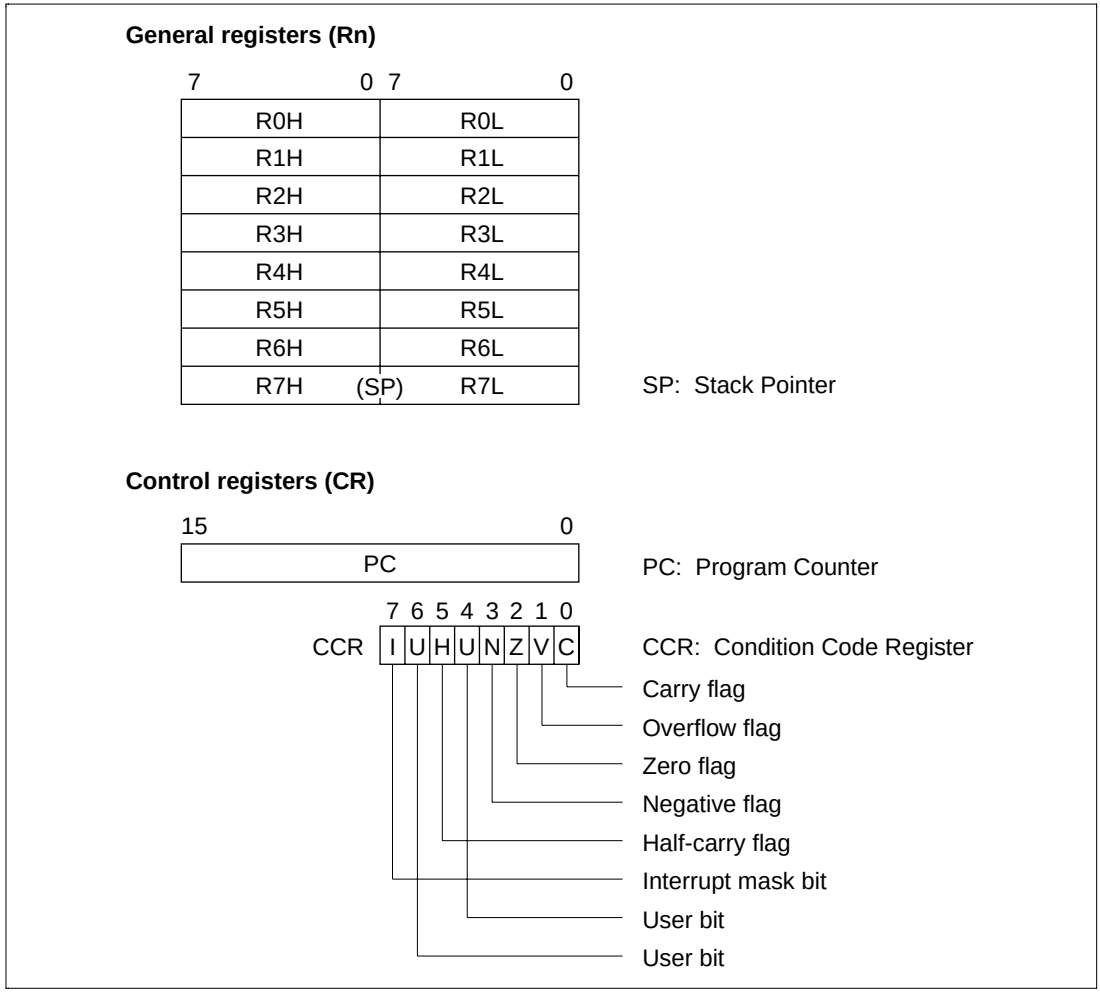



**Figure 2.1 CPU Registers**

## 2.2 Register Descriptions

### 2.2.1 General Registers

All the general registers can be used as both data registers and address registers.

When used as data registers, they can be accessed as 16-bit registers (R0 to R7), or the high bytes (R0H to R7H) and low bytes (R0L to R7L) can be accessed separately as 8-bit registers.

When used as address registers, the general registers are accessed as 16-bit registers (R0 to R7).

R7 also functions as the stack pointer (SP), used implicitly by hardware in exception processing and subroutine calls. When it functions as the stack pointer, as indicated in figure 2.2, SP (R7) points to the top of the stack.

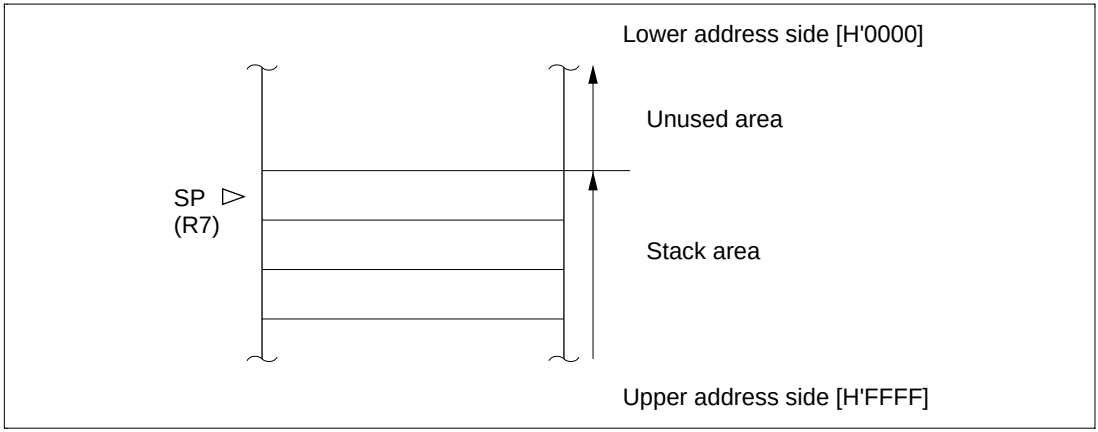



**Figure 2.2 Stack Pointer**

### 2.2.2 Control Registers

The CPU control registers include a 16-bit program counter (PC) and an 8-bit condition code register (CCR).

**Program Counter (PC):** This 16-bit register indicates the address of the next instruction the CPU will execute. All instructions are fetched 16 bits (1 word) at a time, so the least significant bit of the PC is ignored (always regarded as 0).

**Condition Code Register (CCR):** This 8-bit register contains internal status information, including the interrupt mask bit (I) and half-carry (H), negative (N), zero (Z), overflow (V), and carry (C) flags. These bits can be read and written by software (using the LDC, STC, ANDC, ORC, and XORC instructions). The N, Z, V, and C flags are used as branching conditions for conditional branching (Bcc) instructions.

**Bit 7—Interrupt Mask Bit (I):** When this bit is set to 1, interrupts are masked. This bit is set to 1 automatically at the start of exception handling. The interrupt mask bit may be read and written by software. For further details, see section 3.3, Interrupts.

**Bit 6—User Bit (U):** Can be used freely by the user.

**Bit 5—Half-Carry Flag (H):** 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 is cleared to 0 otherwise.

The H flag is used implicitly by the DAA and DAS instructions.

When the ADD.W, SUB.W, or CMP.W instruction is executed, the H flag is set to 1 if there is a carry or borrow at bit 11, and is cleared to 0 otherwise.

**Bit 4—User Bit (U):** Can be used freely by the user.

**Bit 3—Negative Flag (N):** Indicates the most significant bit (sign bit) of the result of an instruction.

**Bit 2—Zero Flag (Z):** Set to 1 to indicate a zero result, and cleared to 0 to indicate a non-zero result.

**Bit 1—Overflow Flag (V):** Set to 1 when an arithmetic overflow occurs, and cleared to 0 at other times.

**Bit 0—Carry Flag (C):** Set to 1 when operation execution generates a carry, and cleared to 0 otherwise. Used by:

- Add instructions, to indicate a carry
- Subtract instructions, to indicate a borrow
- Shift and rotate instructions

The carry flag is also used as a bit accumulator by bit manipulation instructions.

Some instructions leave some or all of the flag bits unchanged.

Refer to the *H8/300L Series Programming Manual* for the action of each instruction on the flag bits.

### 2.2.3 Initial Register Values

When the CPU is reset, the program counter (PC) is initialized to the value stored at address H'0000 in the vector table, and the I bit in the CCR is set to 1. The other CCR bits and the general registers are not initialized. In particular, the stack pointer (R7) is not initialized. To prevent program crashes the stack pointer should be initialized by software, by the first instruction executed after a reset.

## 2.3 Data Formats

The H8/300L CPU can process 1-bit data, 4-bit (BCD) data, 8-bit (byte) data, and 16-bit (word) data.

- Bit manipulation instructions operate on 1-bit data specified as bit n in a byte operand (n = 0, 1, 2, ..., 7).
- All arithmetic and logic instructions except ADDS and SUBS can operate on byte data.
- The MOV.W, ADD.W, SUB.W, CMP.W, ADDS, SUBS, MULXU (8 bits × 8 bits), and DIVXU (16 bits ÷ 8 bits) instructions operate on word data.
- The DAA and DAS instructions perform decimal arithmetic adjustments on byte data in packed BCD form. Each nibble of the byte is treated as a decimal digit.

### 2.3.1 Data Formats in General Registers

The general register data formats are shown in figure 2.3.

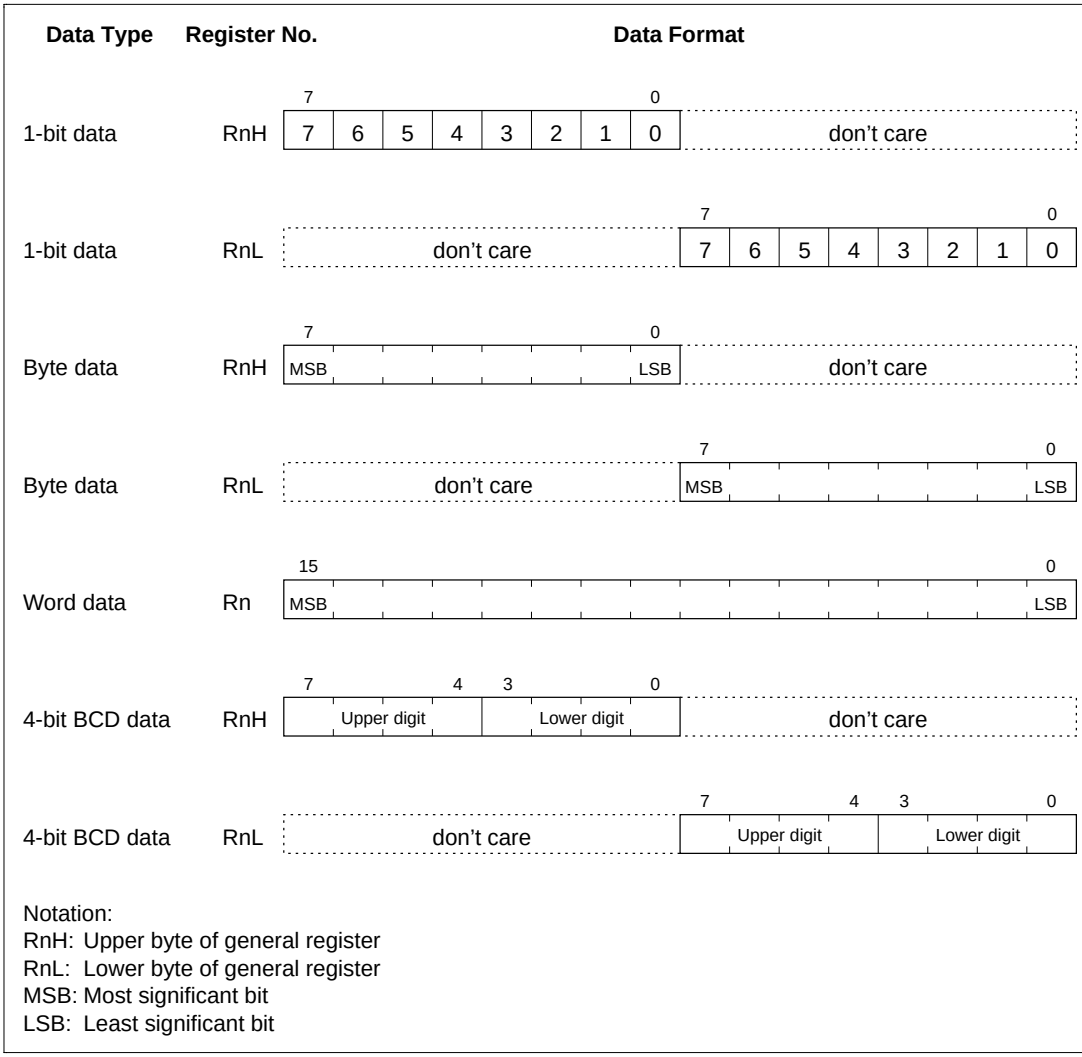



**Figure 2.3 Register Data Formats**

### 2.3.2 Memory Data Formats

Figure 2.4 indicates the data formats in memory. For access by the H8/300L CPU, word data stored in memory must always begin at an even address. In word access the least significant bit of the address is regarded as 0. If an odd address is specified, the access is performed at the preceding even address. This rule affects the MOV.W instruction, and also applies to instruction fetching.

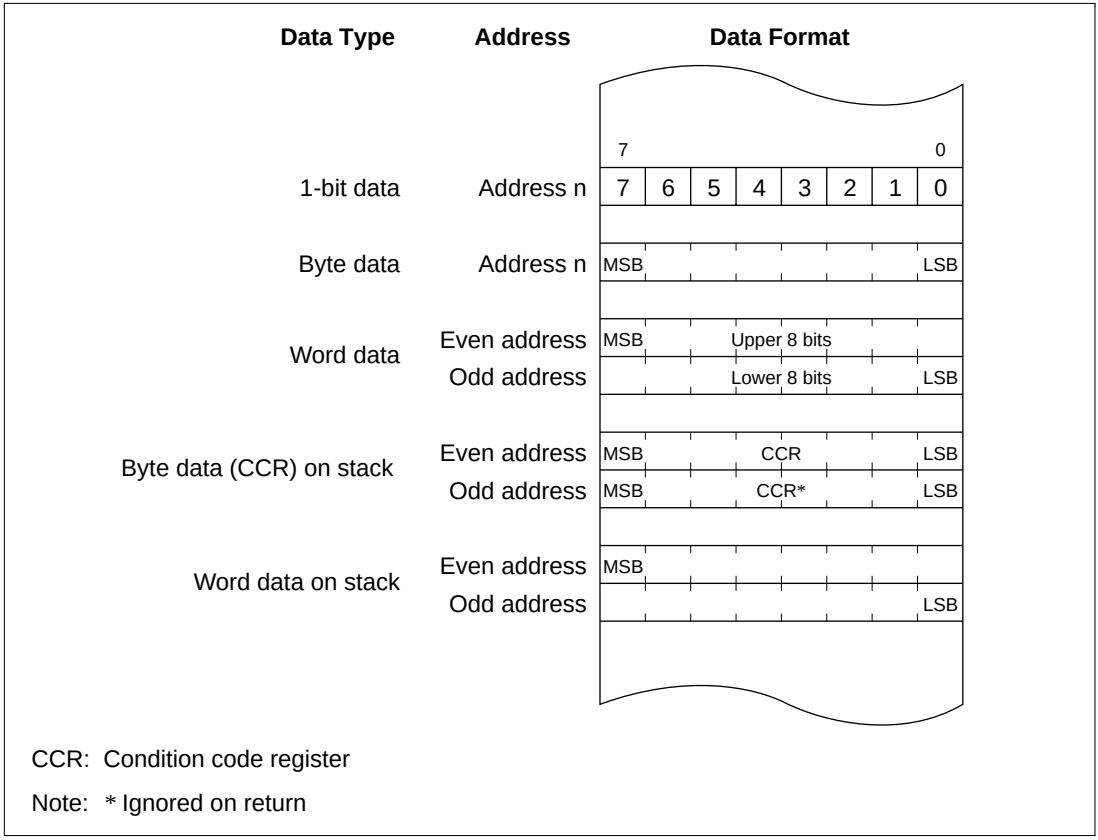



**Figure 2.4 Memory Data Formats**

When the stack is accessed using R7 as an address register, word access should always be performed. For the CCR, the same value is stored in the upper 8 bits and lower 8 bits as word data. On return, the lower 8 bits are ignored.

## 2.4 Addressing Modes

### 2.4.1 Addressing Modes

The H8/300L CPU supports the eight addressing modes listed in table 2.1. Each instruction uses a subset of these addressing modes.

**Table 2.1 Addressing Modes**

| No. | Address Modes | Symbol |
|---|---|---|
| 1 | Register direct | Rn |
| 2 | Register indirect | @Rn |
| 3 | Register indirect with displacement | @(d:16, Rn) |
| 4 | Register indirect with post-increment<br>Register indirect with pre-decrement | @Rn+<br>@–Rn |
| 5 | Absolute address | @aa:8 or @aa:16 |
| 6 | Immediate | #xx:8 or #xx:16 |
| 7 | Program-counter relative | @(d:8, PC) |
| 8 | Memory indirect | @@aa:8 |

**1. Register Direct—Rn:** The register field of the instruction specifies an 8- or 16-bit general register containing the operand.

Only the MOV.W, ADD.W, SUB.W, CMP.W, ADDS, SUBS, MULXU (8 bits × 8 bits), and DIVXU (16 bits ÷ 8 bits) instructions have 16-bit operands.

**2. Register Indirect—@Rn:** The register field of the instruction specifies a 16-bit general register containing the address of the operand in memory.

**3. Register Indirect with Displacement—@(d:16, Rn):** The instruction has a second word (bytes 3 and 4) containing a displacement which is added to the contents of the specified general register to obtain the operand address in memory.

This mode is used only in MOV instructions. For the MOV.W instruction, the resulting address must be even.

**4. Register Indirect with Post-Increment or Pre-Decrement—@Rn+ or @–Rn:**

- Register indirect with post-increment—@Rn+

  The @Rn+ mode is used with MOV instructions that load registers from memory.

  The register field of the instruction specifies a 16-bit general register containing the address of the operand. After the operand is accessed, the register is incremented by 1 for MOV.B or 2 for MOV.W. For MOV.W, the original contents of the 16-bit general register must be even.

- Register indirect with pre-decrement—@–Rn

  The @–Rn mode is used with MOV instructions that store register contents to memory.

  The register field of the instruction specifies a 16-bit general register which is decremented by 1 or 2 to obtain the address of the operand in memory. The register retains the decremented value. The size of the decrement is 1 for MOV.B or 2 for MOV.W. For MOV.W, the original contents of the register must be even.

**5. Absolute Address—@aa:8 or @aa:16:** The instruction specifies the absolute address of the operand in memory.

The absolute address may be 8 bits long (@aa:8) or 16 bits long (@aa:16). The MOV.B and bit manipulation instructions can use 8-bit absolute addresses. The MOV.B, MOV.W, JMP, and JSR instructions can use 16-bit absolute addresses.

For an 8-bit absolute address, the upper 8 bits are assumed to be 1 (H'FF). The address range is H'FF00 to H'FFFF (65280 to 65535).

**6. Immediate—#xx:8 or #xx:16:** The instruction contains an 8-bit operand (#xx:8) in its second byte, or a 16-bit operand (#xx:16) in its third and fourth bytes. Only MOV.W instructions can contain 16-bit immediate values.

The ADDS and SUBS instructions implicitly contain the value 1 or 2 as immediate data. Some bit manipulation instructions contain 3-bit immediate data in the second or fourth byte of the instruction, specifying a bit number.

**7. Program-Counter Relative—@(d:8, PC):** This mode is used in the Bcc and BSR instructions. An 8-bit displacement in byte 2 of the instruction code is sign-extended to 16 bits and added to the program counter contents to generate a branch destination address. The possible branching range is –126 to +128 bytes (–63 to +64 words) from the current address. The displacement should be an even number.

**8. Memory Indirect—@@aa:8:** This mode can be used by the JMP and JSR instructions. The second byte of the instruction code specifies an 8-bit absolute address. The word located at this address contains the branch destination address.

The upper 8 bits of the absolute address are assumed to be 0 (H'00), so the address range is from H'0000 to H'00FF (0 to 255). Note that with the H8/300L Series, the lower end of the address area is also used as a vector area. See 3.3, Interrupts, for details on the vector area.

If an odd address is specified as a branch destination or as the operand address of a MOV.W instruction, the least significant bit is regarded as 0, causing word access to be performed at the address preceding the specified address. See 2.3.2, Memory Data Formats, for further information.

### 2.4.2 Effective Address Calculation

Table 2.2 shows how effective addresses are calculated in each of the addressing modes.

Arithmetic and logic instructions use register direct addressing (1). The ADD.B, ADDX, SUBX, CMP.B, AND, OR, and XOR instructions can also use immediate addressing (6).

Data transfer instructions can use all addressing modes except program-counter relative (7) and memory indirect (8).

Bit manipulation instructions use register direct (1), register indirect (2), or absolute addressing (8-bit) (5) to specify a byte operand, and 3-bit immediate addressing (6) to specify a bit position in that byte. The BSET, BCLR, BNOT, and BTST instructions can also use register direct addressing (1) to specify the bit position.

**Table 2.2 Effective Address Calculation**

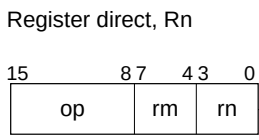
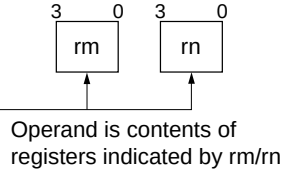
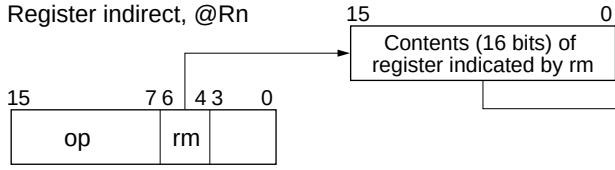
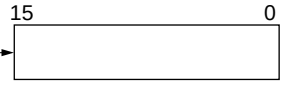
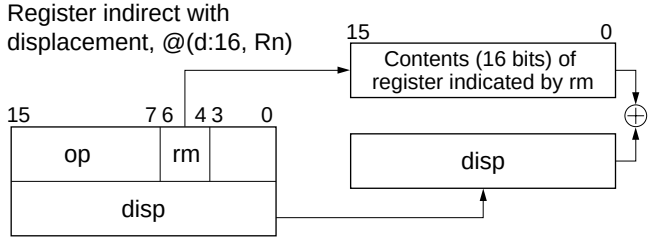
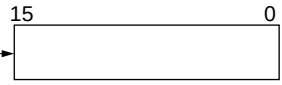
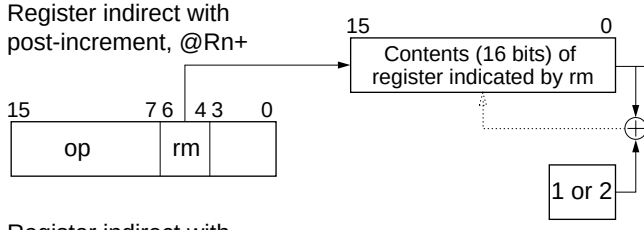
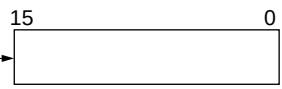
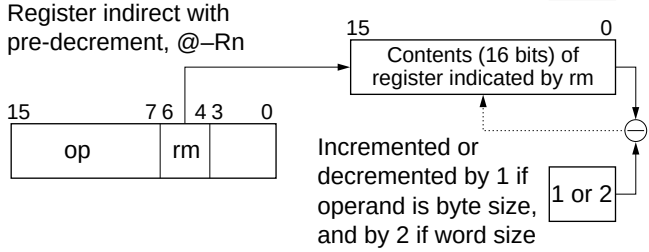
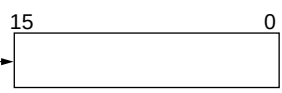
| No. | Addressing Mode and Instruction Format | Effective Address Calculation Method | Effective Address (EA) |
|---|---|---|---|
| 1 | Register direct, Rn |  |  |
| 2 | Register indirect, @Rn |  |  |
| 3 | Register indirect with displacement, @(d:16, Rn) |  |  |
| 4 | Register indirect with post-increment, @Rn+ |  |  |
| | Register indirect with pre-decrement, @–Rn |  |  |

**Table 2.2 Effective Address Calculation (cont)**

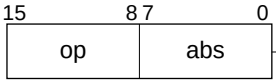
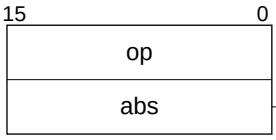
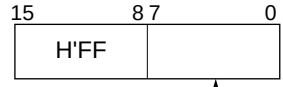
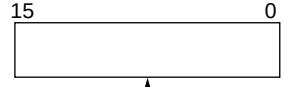
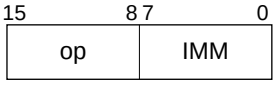
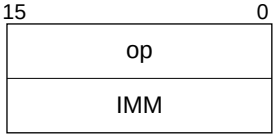
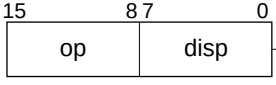
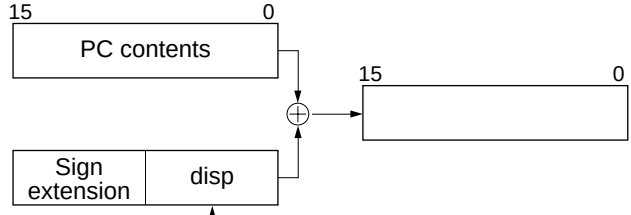
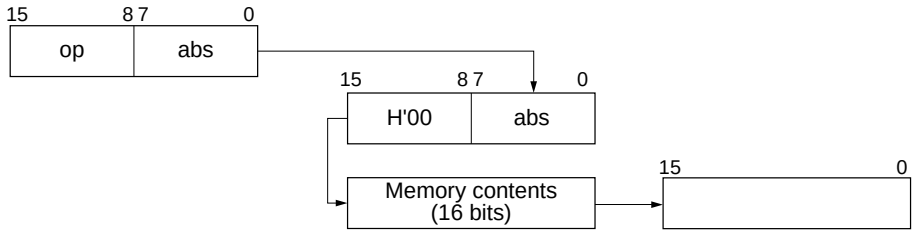
| No. | Addressing Mode and Instruction Format | Effective Address Calculation Method | Effective Address (EA) |
|---|---|---|---|
| 5 | Absolute address<br>@aa:8<br>  <br><br>@aa:16<br>  <br> |  <br><br>  <br> | |
| 6 | Immediate<br>#xx:8<br>  <br><br>#xx:16<br>  <br> | | Operand is 1- or 2-byte immediate data |
| 7 | Program-counter relative<br>@(d:8, PC)<br>  <br> |  <br> | |

**Table 2.2 Effective Address Calculation (cont)**

| No. | Addressing Mode and Instruction Format | Effective Address Calculation Method | Effective Address (EA) |
|---|---|---|---|
| 8 | Memory indirect, @@aa:8 |  | |

Notation:

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

## 2.5 Instruction Set

The H8/300L Series can use a total of 55 instructions, which are grouped by function in table 2.3.

**Table 2.3 Instruction Set**

| Function | Instructions | Number |
|---|---|---|
| Data transfer | MOV, PUSH*[1], POP*[1] | 1 |
| Arithmetic operations | ADD, SUB, ADDX, SUBX, INC, DEC, ADDS, SUBS, DAA, DAS, MULXU, DIVXU, CMP, NEG | 14 |
| Logic operations | AND, OR, XOR, NOT | 4 |
| Shift | SHAL, SHAR, SHLL, SHLR, ROTL, ROTR, ROTXL, ROTXR | 8 |
| Bit manipulation | BSET, BCLR, BNOT, BTST, BAND, BIAND, BOR, BIOR, BXOR, BIXOR, BLD, BILD, BST, BIST | 14 |
| Branch | Bcc*[2], JMP, BSR, JSR, RTS | 5 |
| System control | RTE, SLEEP, LDC, STC, ANDC, ORC, XORC, NOP | 8 |
| Block data transfer | EEPMOV | 1 |
| | | Total: 55 |

Notes: 1. PUSH Rn is equivalent to MOV.W Rn, @–SP.
POP Rn is equivalent to MOV.W @SP+, Rn. The machine language is also the same.
2. Bcc is the generic term for conditional branch instructions.

The functions of the instructions are shown in tables 2.4 to 2.11. The meaning of the operation symbols used in the tables is as follows.

**Notation**

| | |
|---|---|
| Rd | General register (destination) |
| Rs | General register (source) |
| Rn | General register |
| (EAd), <EAd> | Destination operand |
| (EAs), <EAs> | Source operand |
| CCR | Condition code register |
| N | N (negative) flag of CCR |
| Z | Z (zero) flag of CCR |
| V | V (overflow) flag of CCR |
| C | C (carry) flag of CCR |
| PC | Program counter |
| SP | Stack pointer |
| #IMM | Immediate data |
| disp | Displacement |
| + | Addition |
| – | Subtraction |
| × | Multiplication |
| ÷ | Division |
| ∧ | AND logical |
| ∨ | OR logical |
| ⊕ | Exclusive OR logical |
| → | Move |
| ~ | Logical negation (logical complement) |
| :3 | 3-bit length |
| :8 | 8-bit length |
| :16 | 16-bit length |
| ( ), < > | Contents of operand indicated by effective address |

### 2.5.1 Data Transfer Instructions

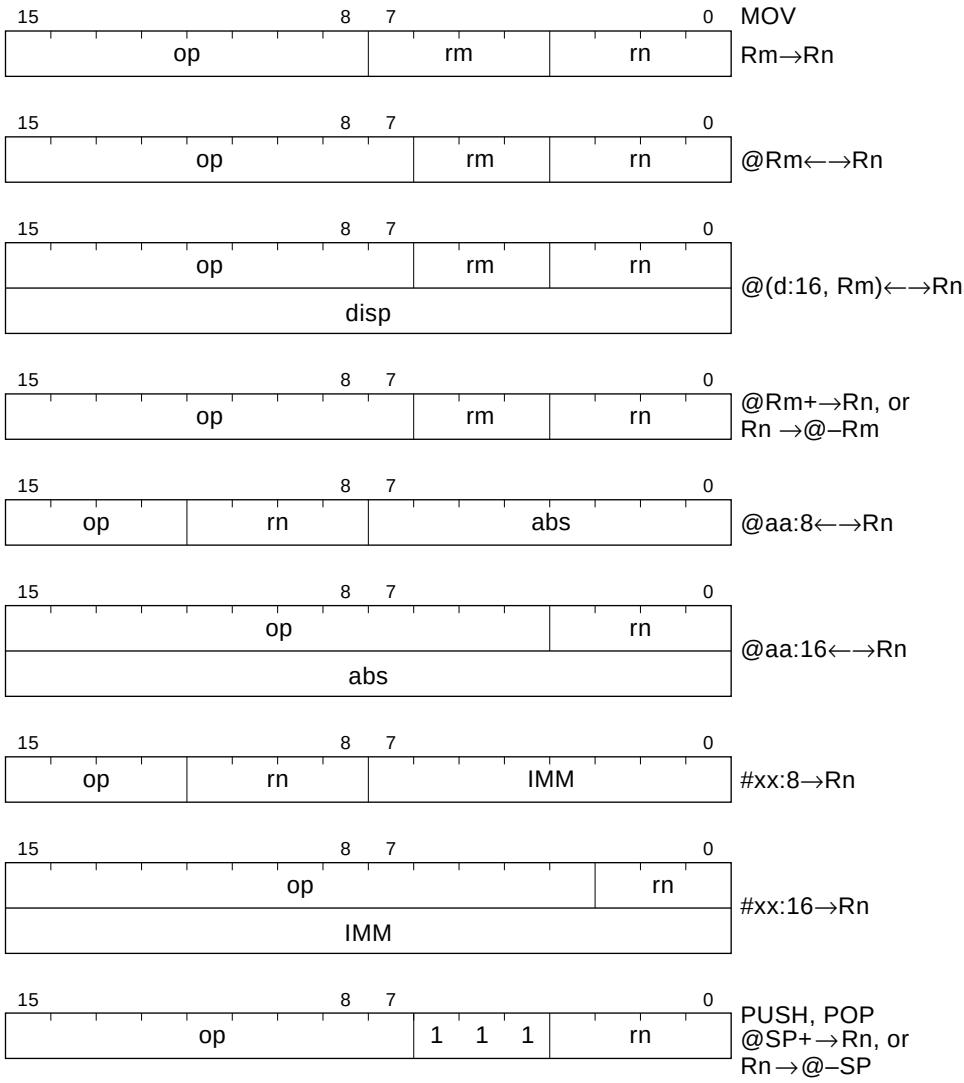
Table 2.4 describes the data transfer instructions. Figure 2.5 shows their object code formats.

**Table 2.4 Data Transfer Instructions**

| Instruction | Size* | Function |
|---|---|---|
| MOV | B/W | (EAs) → Rd, Rs → (EAd)<br>Moves data between two general registers or between a general register and memory, or moves immediate data to a general register.<br>The Rn, @Rn, @(d:16, Rn), @aa:16, #xx:16, @–Rn, and @Rn+ addressing modes are available for byte or word data. The @aa:8 addressing mode is available for byte data only.<br>The @–R7 and @R7+ modes require word operands. Do not specify byte size for these two modes. |
| POP | W | @SP+ → Rn<br>Pops a 16-bit general register from the stack. Equivalent to MOV.W @SP+, Rn. |
| PUSH | W | Rn → @–SP<br>Pushes a 16-bit general register onto the stack. Equivalent to MOV.W Rn, @–SP. |

Note: * Size: Operand size
B: Byte
W: Word

Certain precautions are required in data access. See 2.9.1, Notes on Data Access, for details.




Notation:
op: Operation field
rm, rn: Register field
disp: Displacement
abs: Absolute address
IMM: Immediate data

**Figure 2.5 Data Transfer Instruction Codes**

### 2.5.2 Arithmetic Operations

Table 2.5 describes the arithmetic instructions.

**Table 2.5 Arithmetic Instructions**

| Instruction | Size* | Function |
|---|---|---|
| ADD | B/W | Rd ± Rs → Rd, Rd + #IMM → Rd |
| SUB | | Performs addition or subtraction on data in two general registers, or addition on immediate data and data in a general register. Immediate data cannot be subtracted from data in a general register. Word data can be added or subtracted only when both words are in general registers. |
| ADDX | B | Rd ± Rs ± C → Rd, Rd ± #IMM ± C → Rd |
| SUBX | | Performs addition or subtraction with carry or borrow on byte data in two general registers, or addition or subtraction on immediate data and data in a general register. |
| INC | B | Rd ± 1 → Rd |
| DEC | | Increments or decrements a general register |
| ADDS | W | Rd ± 1 → Rd, Rd ± 2 → Rd |
| SUBS | | Adds or subtracts immediate data to or from data in a general register. The immediate data must be 1 or 2. |
| DAA | B | Rd decimal adjust → Rd |
| DAS | | Decimal-adjusts (adjusts to packed 4-bit BCD) an addition or subtraction result in a general register by referring to the CCR |
| MULXU | B | Rd × Rs → Rd |
| | | Performs 8-bit × 8-bit unsigned multiplication on data in two general registers, providing a 16-bit result |
| DIVXU | B | Rd ÷ Rs → Rd |
| | | Performs 16-bit ÷ 8-bit unsigned division on data in two general registers, providing an 8-bit quotient and 8-bit remainder |
| CMP | B/W | Rd – Rs, Rd – #IMM |
| | | Compares data in a general register with data in another general register or with immediate data, and the result is stored in the CCR. Word data can be compared only between two general registers. |
| NEG | B | 0 – Rd → Rd |
| | | Obtains the two's complement (arithmetic complement) of data in a general register |

Note: * Size: Operand size
B: Byte
W: Word

### 2.5.3 Logic Operations

Table 2.6 describes the four instructions that perform logic operations.

**Table 2.6 Logic Operation Instructions**

| Instruction | Size* | Function |
|---|---|---|
| AND | B | Rd ∧ Rs → Rd, Rd ∧ #IMM → Rd<br>Performs a logical AND operation on a general register and another general register or immediate data |
| OR | B | Rd ∨ Rs → Rd, Rd ∨ #IMM → Rd<br>Performs a logical OR operation on a general register and another general register or immediate data |
| XOR | B | Rd ⊕ Rs → Rd, Rd ⊕ #IMM → Rd<br>Performs a logical exclusive OR operation on a general register and another general register or immediate data |
| NOT | B | ~ Rd → Rd<br>Obtains the one's complement (logical complement) of general register contents |

Note: * Size: Operand size
B: Byte

### 2.5.4 Shift Operations

Table 2.7 describes the eight shift instructions.

**Table 2.7 Shift Instructions**

| Instruction | Size* | Function |
|---|---|---|
| SHAL | B | Rd shift → Rd |
| SHAR | | Performs an arithmetic shift operation on general register contents |
| SHLL | B | Rd shift → Rd |
| SHLR | | Performs a logical shift operation on general register contents |
| ROTL | B | Rd rotate → Rd |
| ROTR | | Rotates general register contents |
| ROTXL | B | Rd rotate through carry → Rd |
| ROTXR | | Rotates general register contents through the C (carry) bit |

Notes: * Size: Operand size
B: Byte

Figure 2.6 shows the instruction code format of arithmetic, logic, and shift instructions.

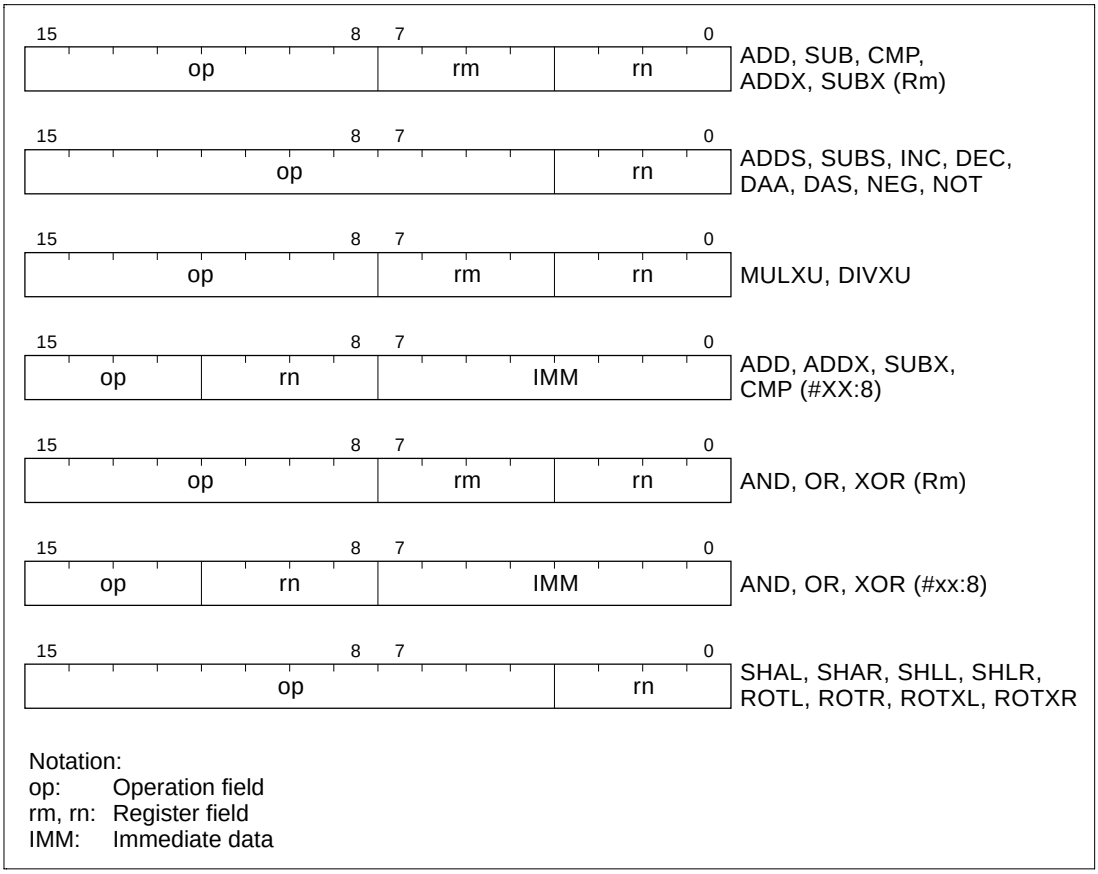



**Figure 2.6 Arithmetic, Logic, and Shift Instruction Codes**

### 2.5.5 Bit Manipulations

Table 2.8 describes the bit-manipulation instructions. Figure 2.7 shows their object code formats.

**Table 2.8 Bit-Manipulation Instructions**

| Instruction | Size* | Function |
|---|---|---|
| BSET | B | 1 → (<bit-No.> of <EAd>)<br>Sets a specified bit to 1 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. |
| BCLR | B | 0 → (<bit-No.> of <EAd>)<br>Clears a specified bit to 0 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. |
| BNOT | B | ~ (<bit-No.> of <EAd>) → (<bit-No.> of <EAd>)<br>Inverts a specified bit in a general register or memory operand. The bit number is specified by 3-bit immediate data or the lower three bits of a general register. |
| BTST | B | ~ (<bit-No.> of <EAd>) → Z<br>Tests a specified bit in a general register or memory operand and sets or clears the Z flag accordingly. The bit number is specified by 3-bit immediate data or the lower three bits of a general register. |
| BAND | B | C ∧ (<bit-No.> of <EAd>) → C<br>ANDs the C flag with a specified bit in a general register or memory operand, and stores the result in the C flag. |
| BIAND | B | C ∧ [~ (<bit-No.> of <EAd>)] → C<br>ANDs the C flag with the inverse of a specified bit in a general register or memory operand, and stores the result in the C flag.<br>The bit number is specified by 3-bit immediate data. |
| BOR | B | C ∨ (<bit-No.> of <EAd>) → C<br>ORs the C flag with a specified bit in a general register or memory operand, and stores the result in the C flag. |
| BIOR | B | C ∨ [~ (<bit-No.> of <EAd>)] → C<br>ORs the C flag with the inverse of a specified bit in a general register or memory operand, and stores the result in the C flag.<br>The bit number is specified by 3-bit immediate data. |

Note: * Size: Operand size
B: Byte

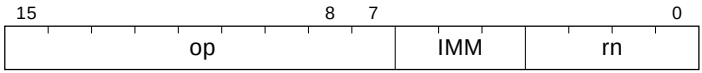
**Table 2.8 Bit-Manipulation Instructions (cont)**

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

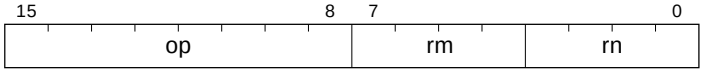
Note: * Size: Operand size
B: Byte

Certain precautions are required in bit manipulation. See 2.9.2, Notes on Bit Manipulation, for details.

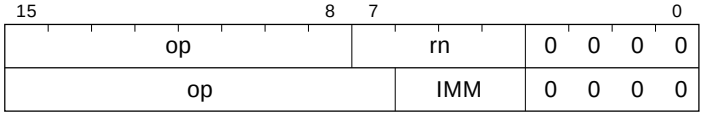
**BSET, BCLR, BNOT, BTST**



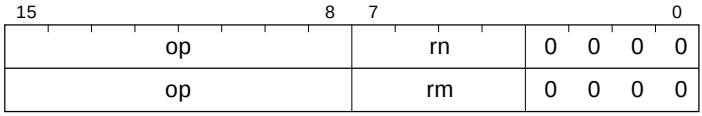

Operand: register direct (Rn)
Bit No.: immediate (#xx:3)



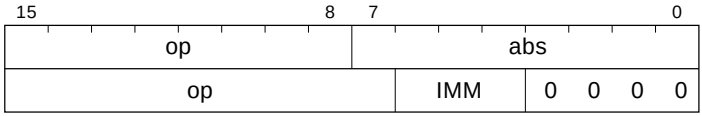

Operand: register direct (Rn)
Bit No.: register direct (Rm)



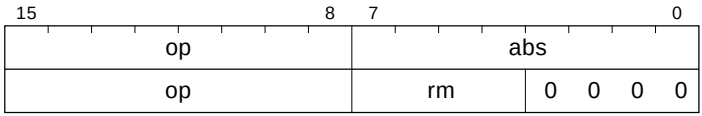

Operand: register indirect (@Rn)
Bit No.: immediate (#xx:3)




Operand: register indirect (@Rn)
Bit No.: register direct (Rm)

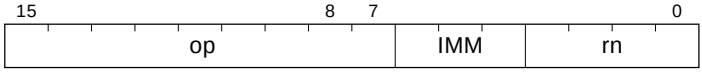



Operand: absolute (@aa:8)
Bit No.: immediate (#xx:3)

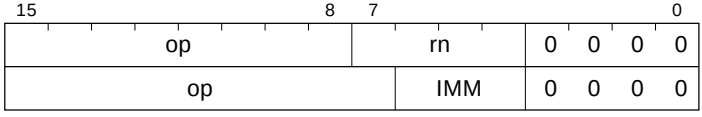



Operand: absolute (@aa:8)
Bit No.: register direct (Rm)

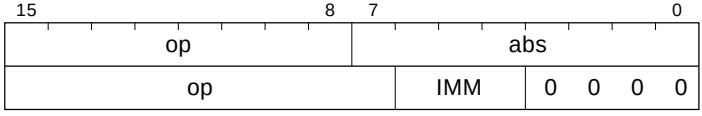
**BAND, BOR, BXOR, BLD, BST**




Operand: register direct (Rn)
Bit No.: immediate (#xx:3)




Operand: register indirect (@Rn)
Bit No.: immediate (#xx:3)

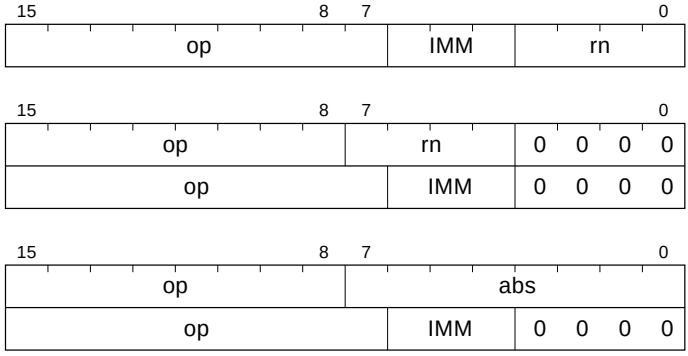



Operand: absolute (@aa:8)
Bit No.: immediate (#xx:3)

Notation:
op: Operation field
rm, rn: Register field
abs: Absolute address
IMM: Immediate data

**Figure 2.7 Bit Manipulation Instruction Codes**

**BIAND, BIOR, BIXOR, BILD, BIST**




Notation:
op: Operation field
rm, rn: Register field
abs: Absolute address
IMM: Immediate data

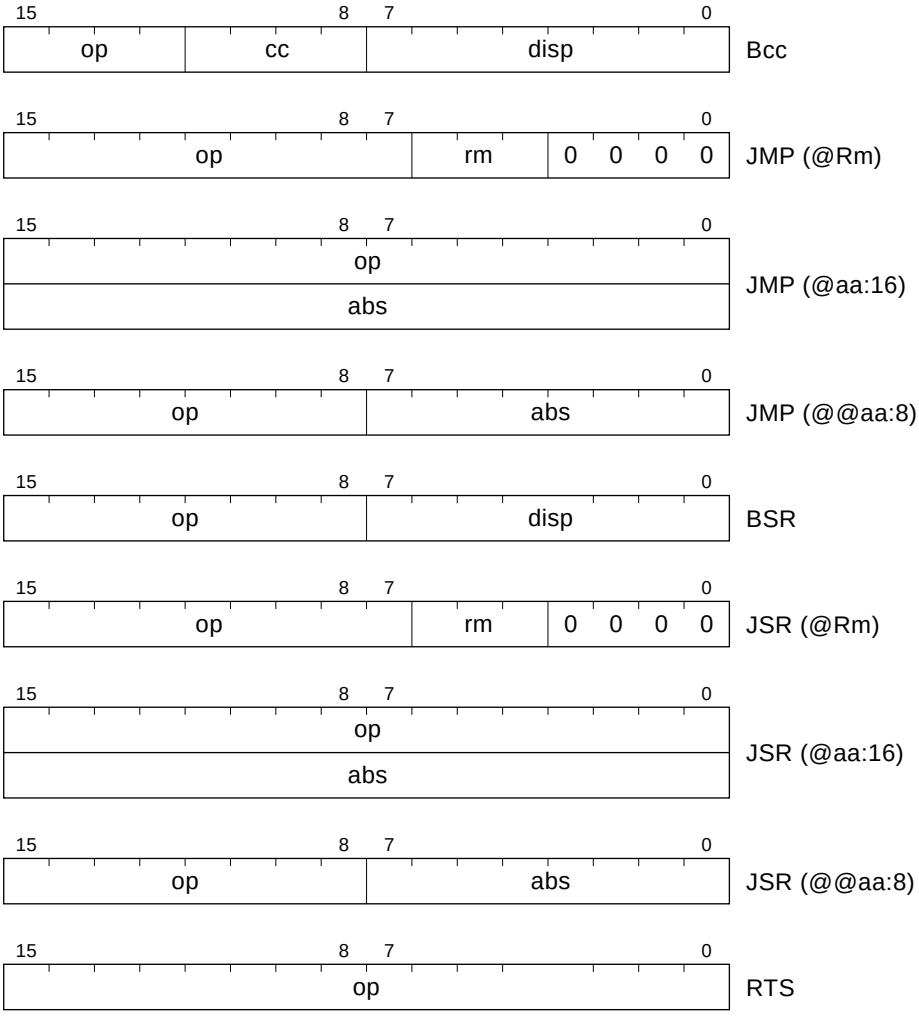
**Figure 2.7 Bit Manipulation Instruction Codes (cont)**

### 2.5.6 Branching Instructions

Table 2.9 describes the branching instructions. Figure 2.8 shows their object code formats.

**Table 2.9 Branching Instructions**

| Instruction | Size | Function | | |
|---|---|---|---|---|
| Bcc | — | Branches to the designated address if the specified condition is true. The branching conditions are given 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 | | |




Notation:
op: Operation field
cc: Condition field
rm: Register field
disp: Displacement
abs: Absolute address

**Figure 2.8 Branching Instruction Codes**

### 2.5.7 System Control Instructions

Table 2.10 describes the system control instructions. Figure 2.9 shows their object code formats.

**Table 2.10 System Control Instructions**

| Instruction | Size* | Function |
|---|---|---|
| RTE | — | Returns from an exception-handling routine |
| SLEEP | — | Causes a transition from active mode to a power-down mode. See section 5, Power-Down Modes, for details |
| LDC | B | Rs → CCR, #IMM → CCR<br>Moves immediate data or general register contents to the condition code register |
| STC | B | CCR → Rd<br>Copies the condition code register to a specified general register |
| ANDC | B | CCR ∧ #IMM → CCR<br>Logically ANDs the condition code register with immediate data |
| ORC | B | CCR ∨ #IMM → CCR<br>Logically ORs the condition code register with immediate data |
| XORC | B | CCR ⊕ #IMM → CCR<br>Logically exclusive-ORs the condition code register with immediate data |
| NOP | — | PC + 2 → PC<br>Only increments the program counter |

Note: * Size: Operand size
B: Byte

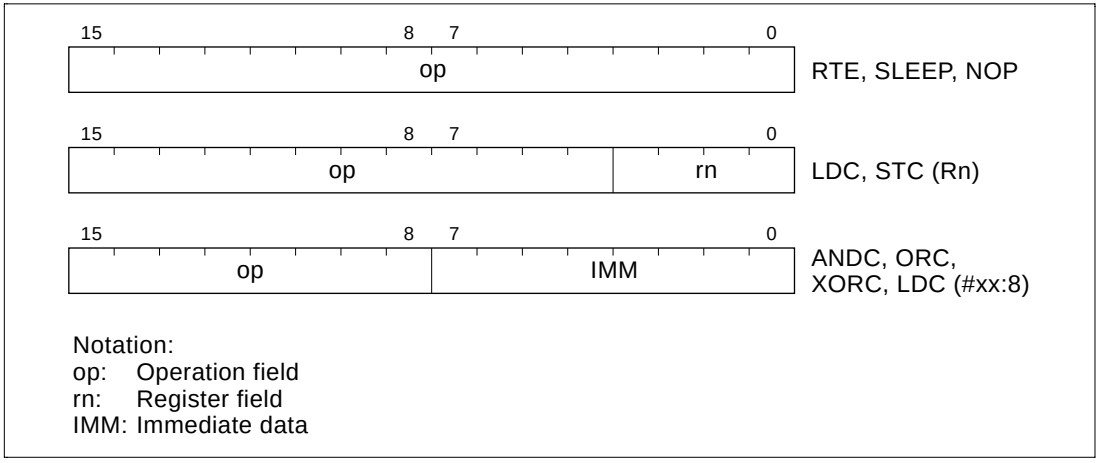



**Figure 2.9 System Control Instruction Codes**

### 2.5.8 Block Data Transfer Instruction

Table 2.11 describes the block data transfer instruction. Figure 2.10 shows its object code format.

**Table 2.11 Block Data Transfer Instruction**

| Instruction | Size | Function |
|---|---|---|
| EEPMOV | — | If R4L ≠ 0 then<br>repeat @R5+ → @R6+<br>R4L – 1 → R4L<br>until R4L = 0<br>else next;<br>Block transfer instruction. Transfers the number of bytes specified by R4L, from locations starting at the address specified by R5, to locations starting at the address specified by R6. On completion of the transfer, the next instruction is executed. |

Certain precautions are required in using the EEPMOV instruction. See 2.9.3, Notes on Use of the EEPMOV Instruction, for details.

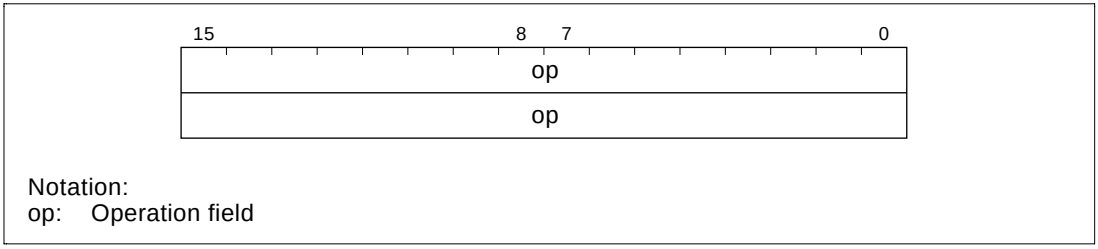



**Figure 2.10 Block Data Transfer Instruction Code**

## 2.6 Basic Operational Timing

CPU operation is synchronized by a system clock ($\phi$) or a subclock ($\phi_{SUB}$). For details on these clock signals see section 4, Clock Pulse Generators. The period from a rising edge of $\phi$ or $\phi_{SUB}$ 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 2.11 shows the on-chip memory access cycle.

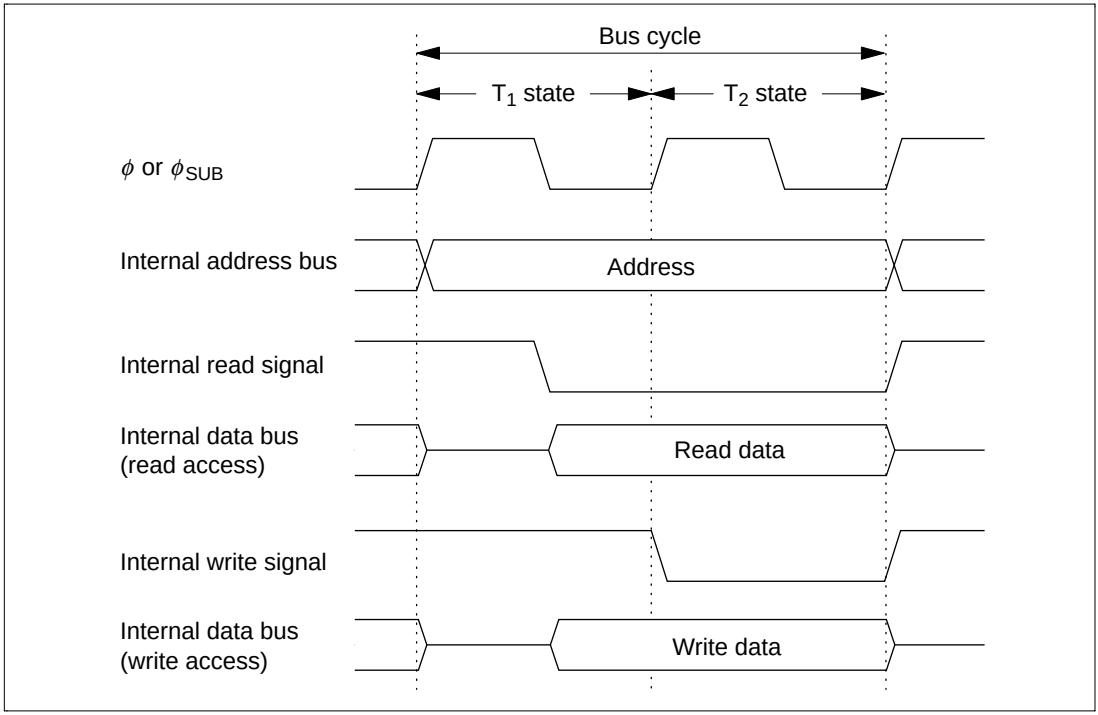



**Figure 2.11 On-Chip Memory Access Cycle**

### 2.6.2 Access to On-Chip Peripheral Modules

On-chip peripheral modules are accessed in two states or three states. The data bus width is 8 bits, so access is by byte size only. This means that for accessing word data, two instructions must be used. Figures 2.12 and 2.13 show the on-chip peripheral module access cycle.

**Two-State Access to On-Chip Peripheral Modules**

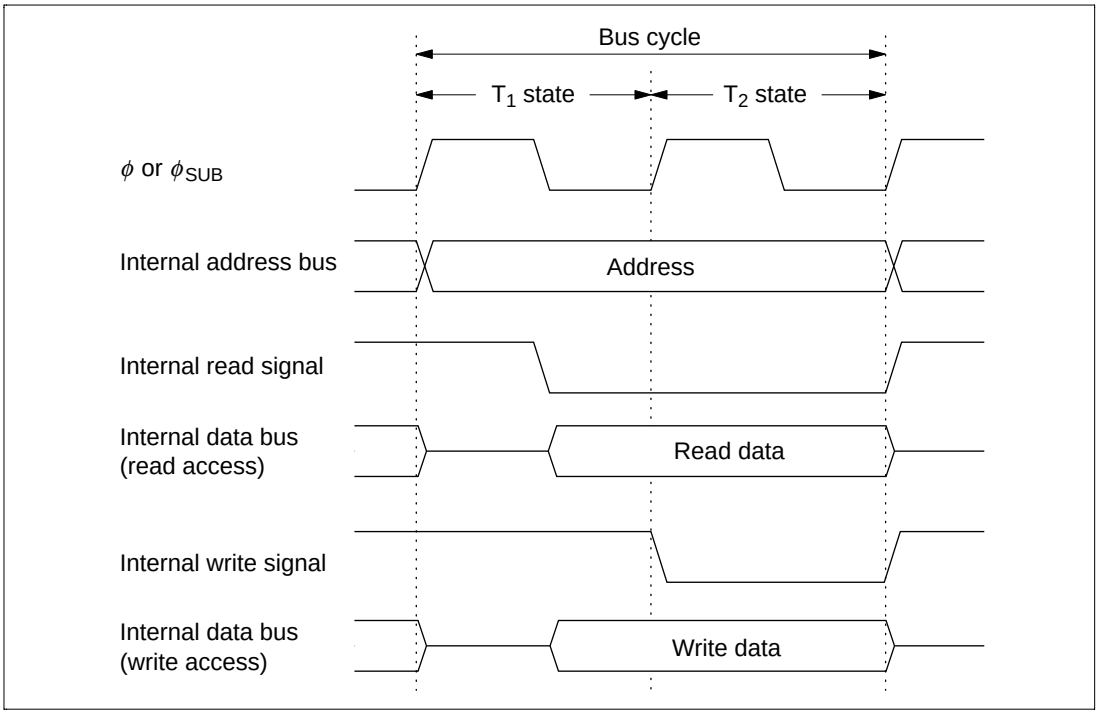



**Figure 2.12 On-Chip Peripheral Module Access Cycle (2-State Access)**

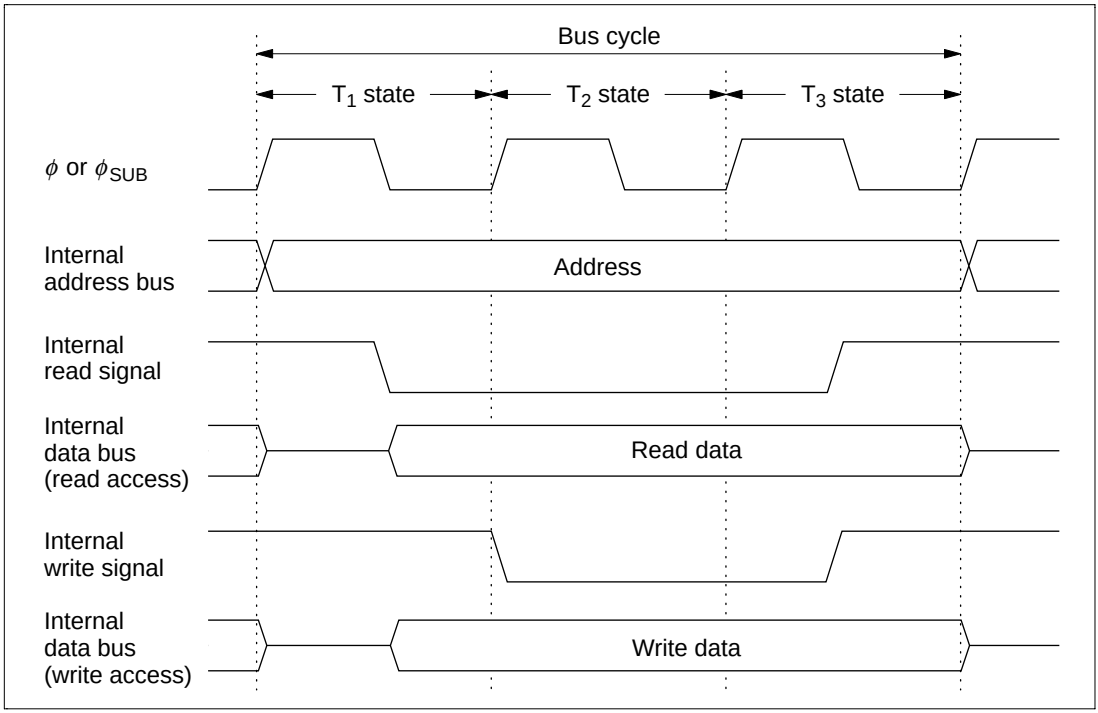



**Figure 2.13 On-Chip Peripheral Module Access Cycle (3-State Access)**

## 2.7 CPU States

### 2.7.1 Overview

There are four CPU states: the reset state, program execution state, program halt state, and exception-handling state. The program execution state includes active (high-speed or medium-speed) mode and subactive mode. In the program halt state there are a sleep mode, standby mode, watch mode, and sub-sleep mode. These states are shown in figure 2.14.
Figure 2.15 shows the state transitions.

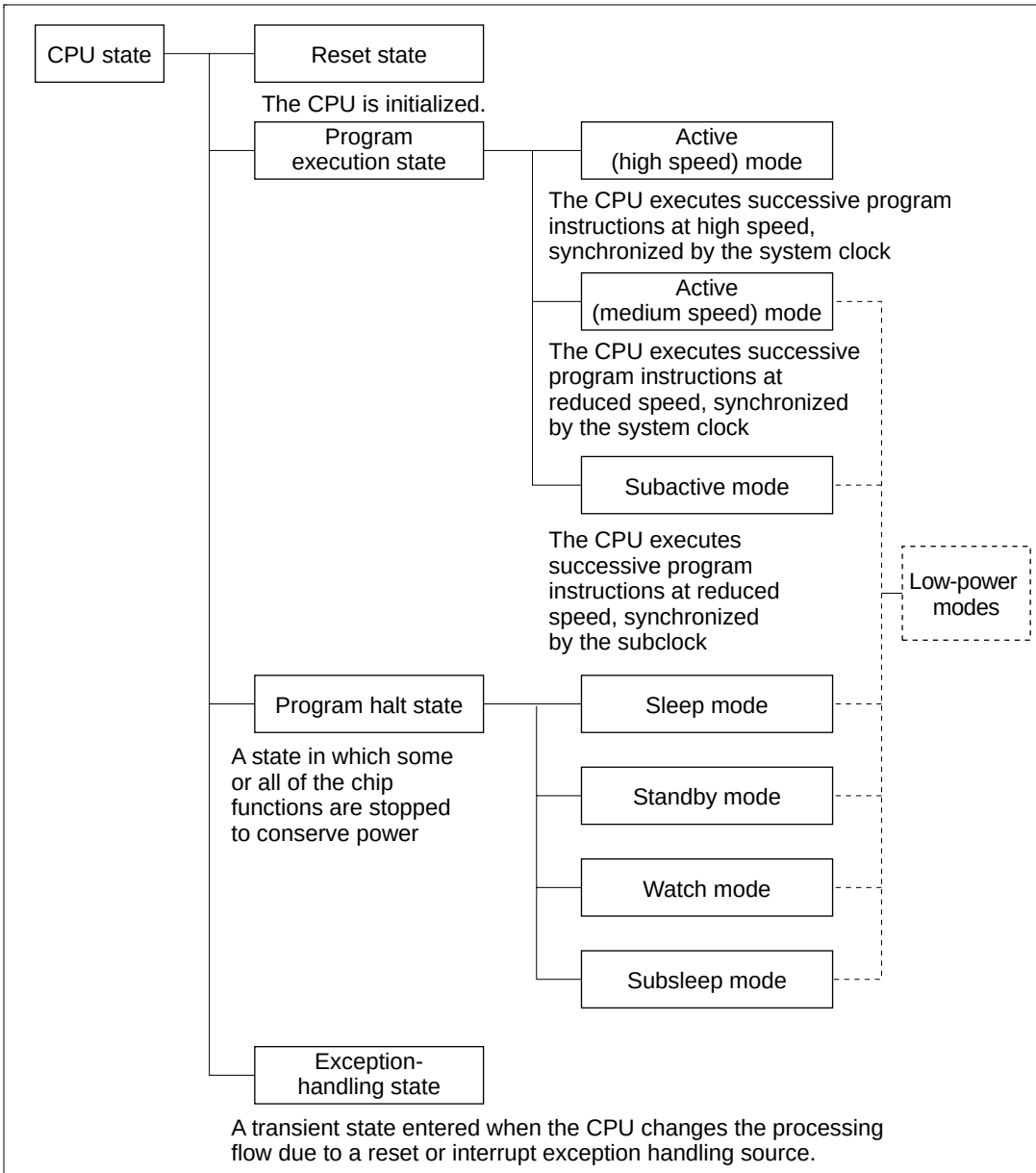



**Figure 2.14 CPU Operation States**

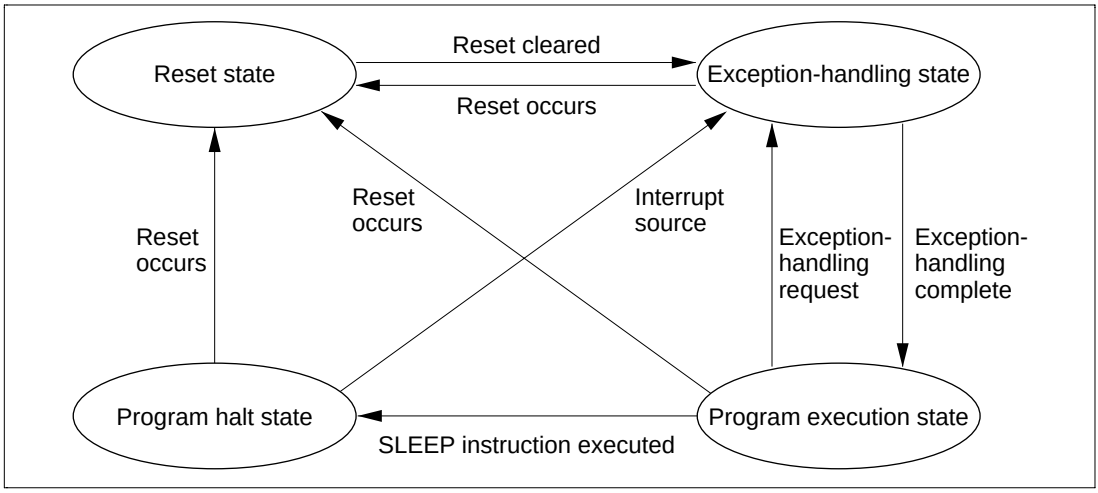



**Figure 2.15 State Transitions**

### 2.7.2 Program Execution State

In the program execution state the CPU executes program instructions in sequence.

There are three modes in this state, two active modes (high speed and medium speed) and one subactive mode. Operation is synchronized with the system clock in active mode (high speed and medium speed), and with the subclock in subactive mode. See section 5, Power-Down Modes for details on these modes.

### 2.7.3 Program Halt State

In the program halt state there are four modes: sleep mode, standby mode, watch mode, and subsleep mode. See section 5, Power-Down Modes for details on these modes.

### 2.7.4 Exception-Handling State

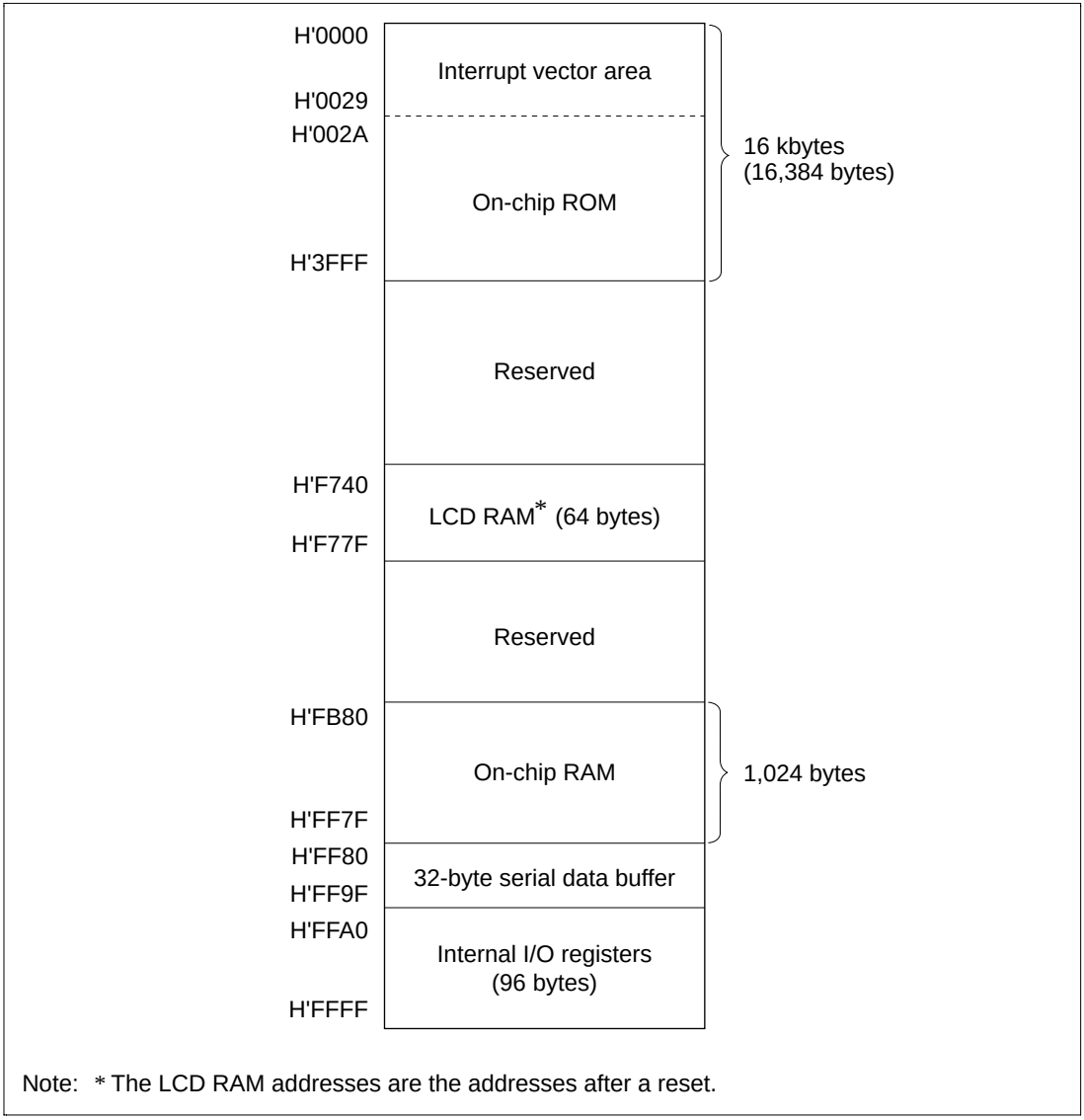
The exception-handling state is a transient state occurring when exception handling is started by a reset or interrupt and the CPU changes its normal processing flow. In exception handling caused by an interrupt, SP (R7) is referenced and the PC and CCR values are saved on the stack.

For details on interrupt handling, see section 3, Exception Handling.

## 2.8 Memory Map

### 2.8.1 Memory Map

Figure 2.16 shows the H8/3832 memory map. Figure 2.17 shows the H8/3833 memory map. Figure 2.18 shows the H8/3834 memory map. Figure 2.19 shows the H8/3835 memory map. Figure 2.20 shows the H8/3836 memory map. Figure 2.21 shows the H8/3837 memory map.




Note: * The LCD RAM addresses are the addresses after a reset.

**Figure 2.16 H8/3832 Memory Map**

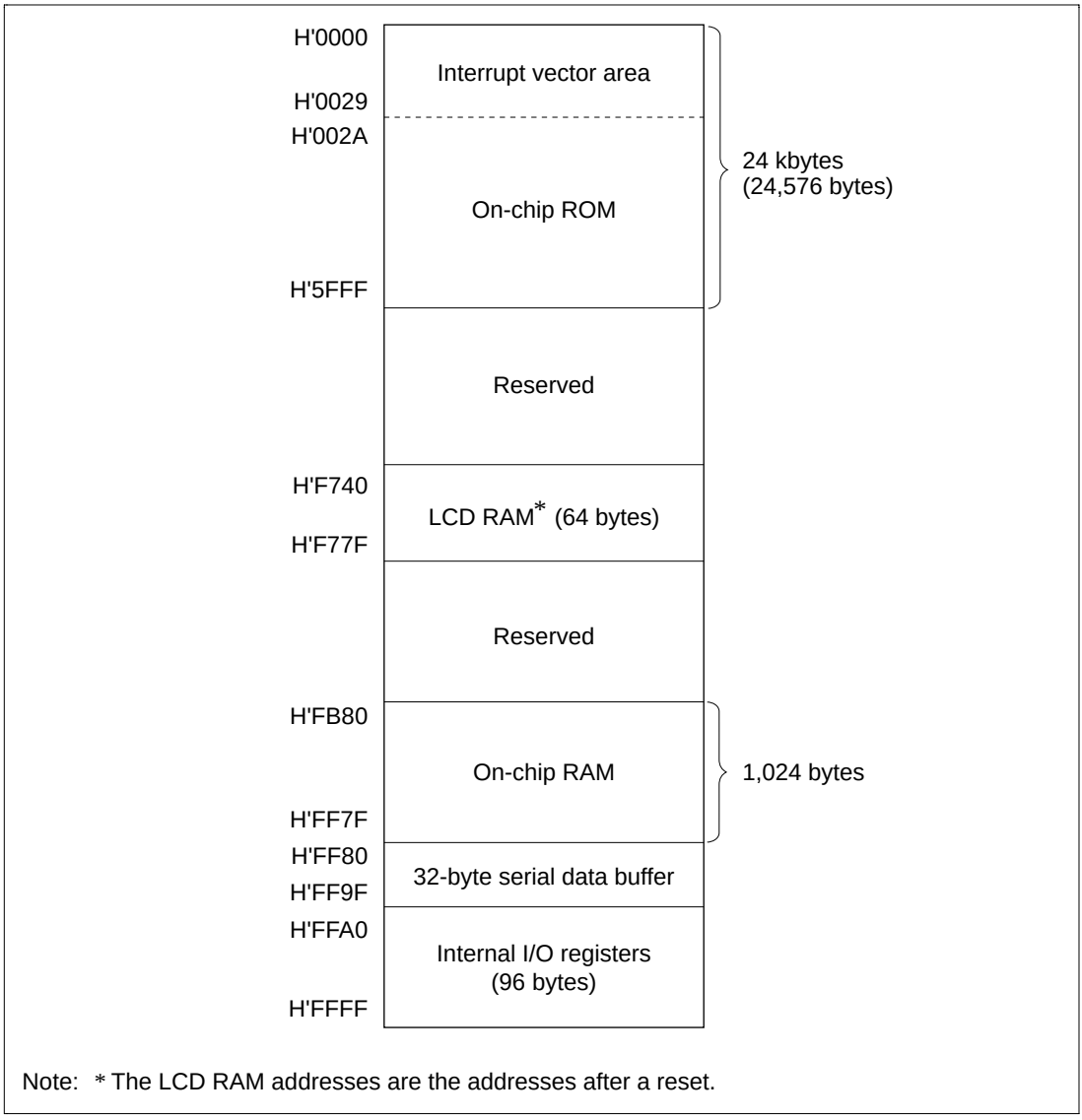



**Figure 2.17 H8/3833 Memory Map**

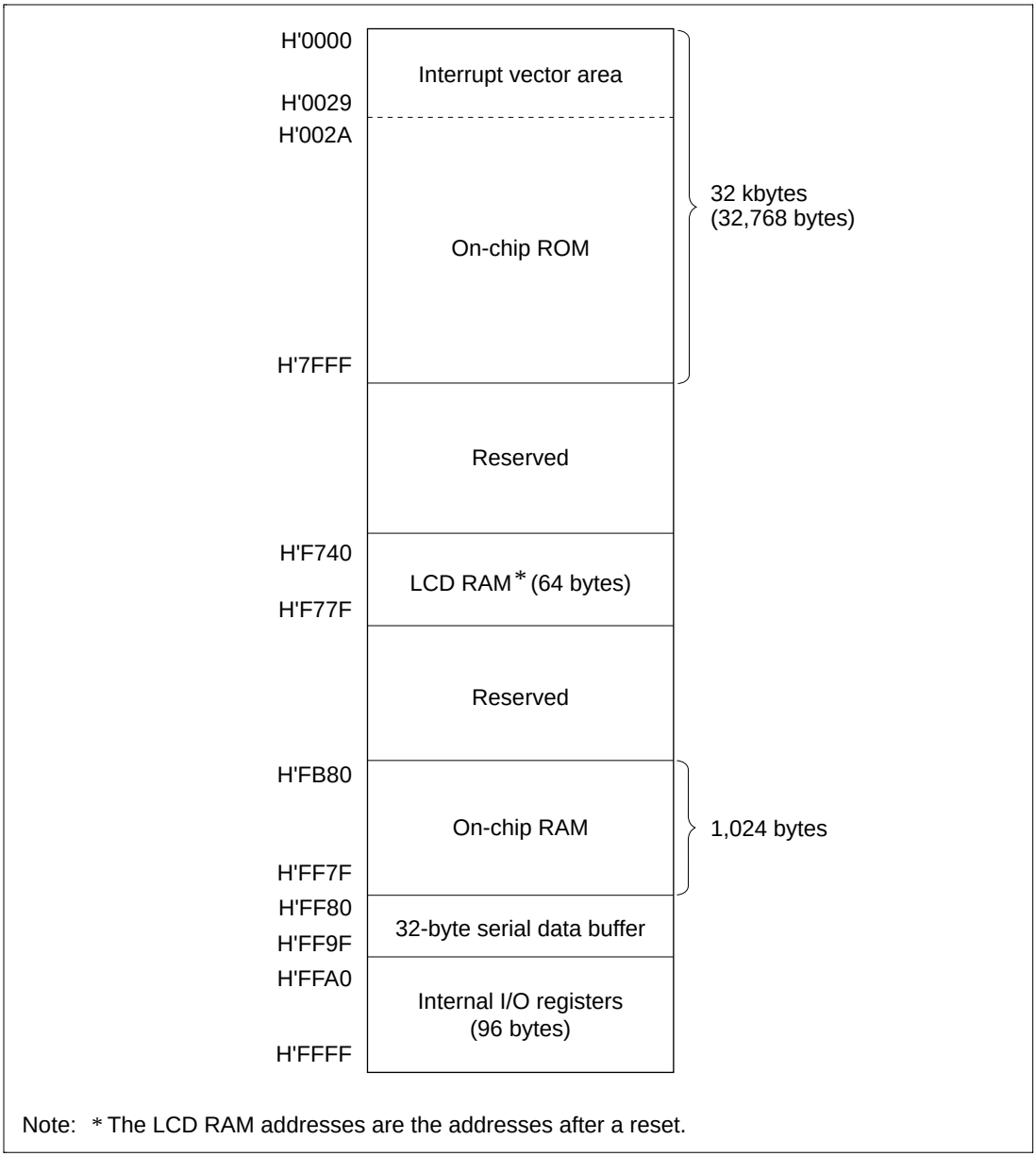



**Figure 2.18 H8/3834 Memory Map**

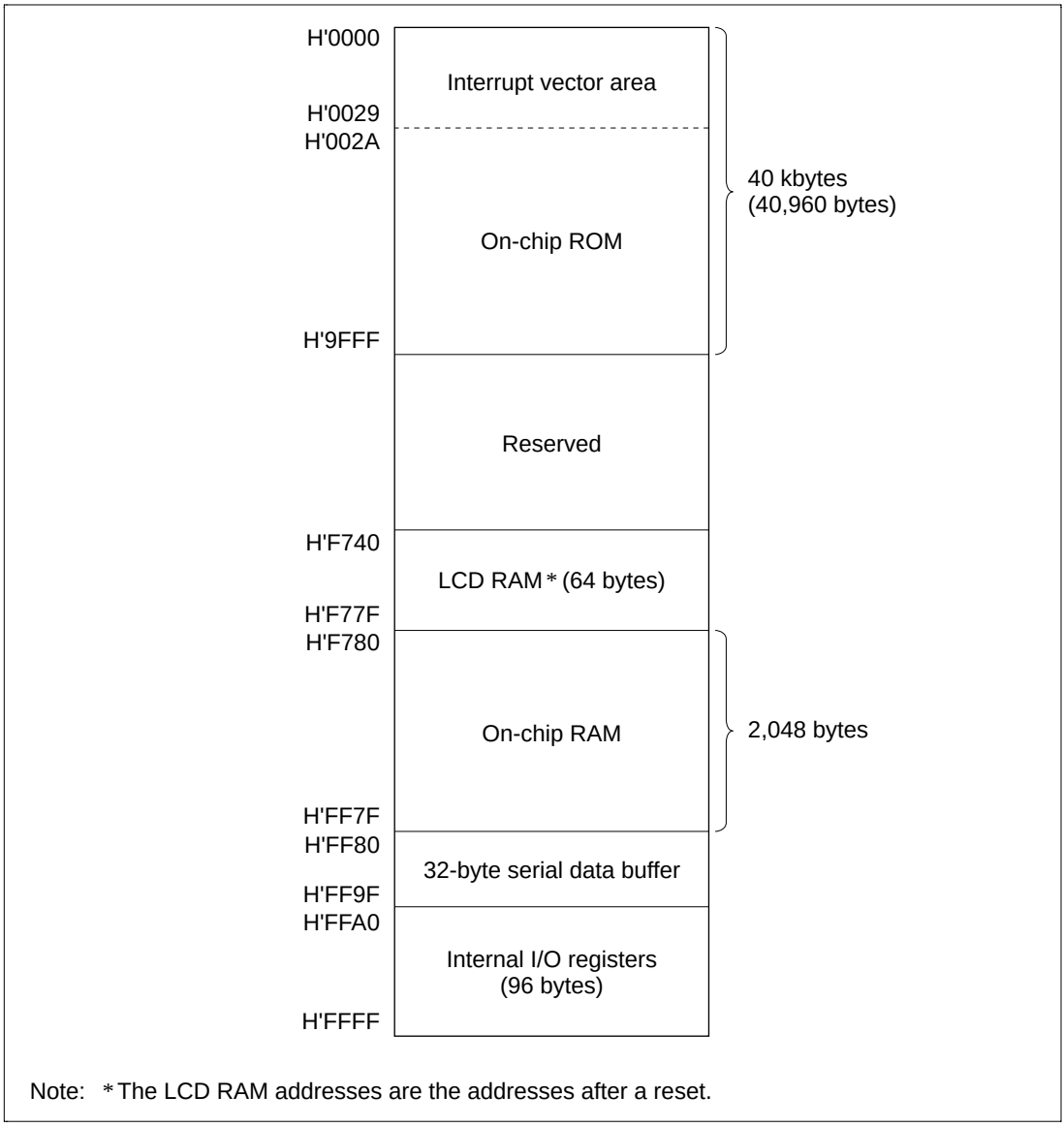



**Figure 2.19 H8/3835 Memory Map**

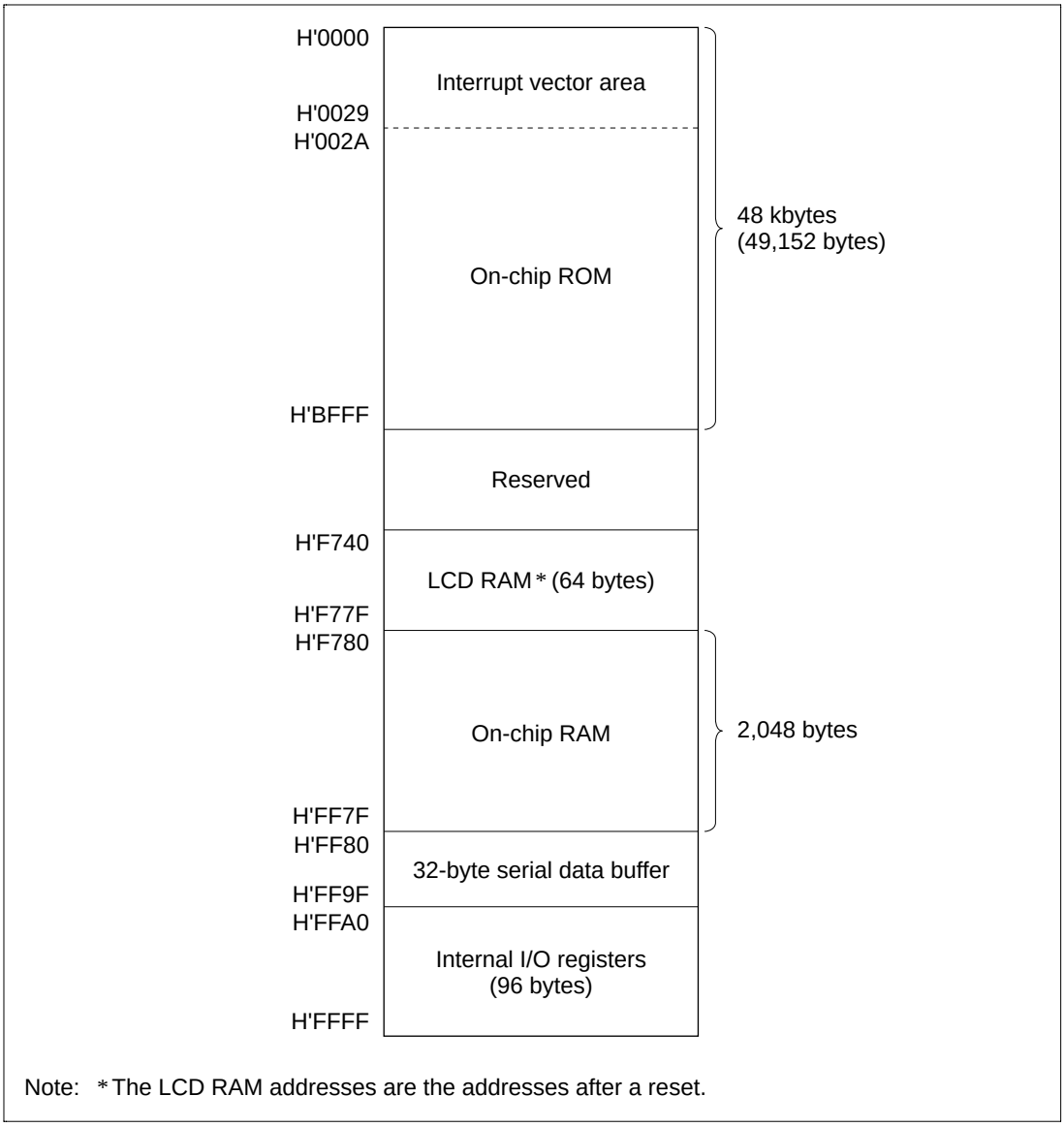



**Figure 2.20 H8/3836 Memory Map**

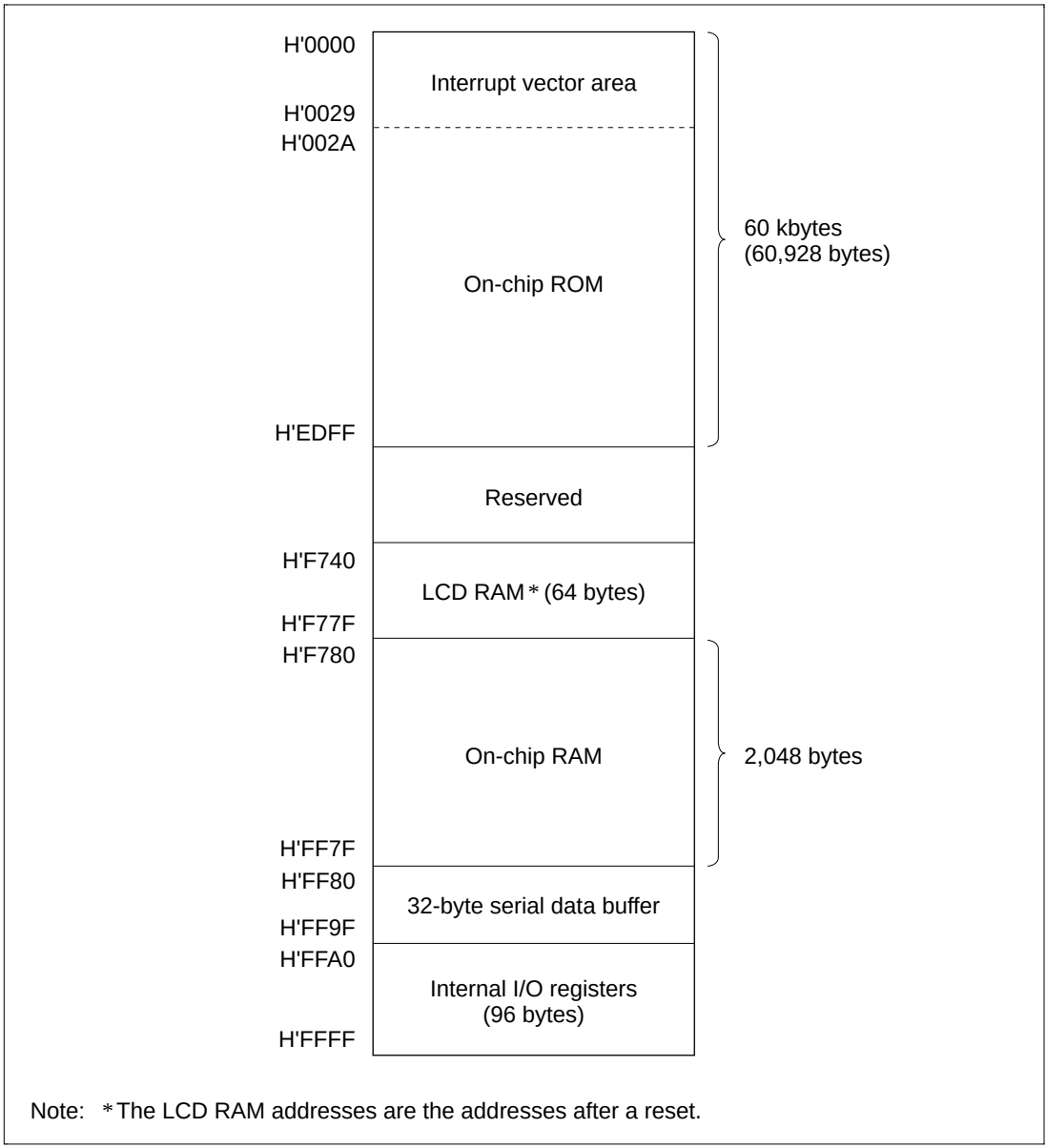



**Figure 2.21 H8/3837 Memory Map**

### 2.8.2 LCD RAM Address Relocation

After a reset, the LCD RAM area is located at addresses H'F740 to H'F77F. However, this area can be relocated by setting the LCD RAM relocation register (RLCTR) bits. The LCD RAM relocation register is explained below.

LCD RAM relocation register (RLCTR: H'FFCF)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | — | — | RLCT1 | RLCT0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| Read/Write | — | — | — | — | — | — | R/W | R/W |

RLCTR is an 8-bit read/write register that selects the LCD RAM address space. Upon reset, RLCTR is initialized to H'00.

**Bits 7 to 2:** Reserved Bits

Bits 7 to 2 are reserved; they are always read as 1, and cannot be modified.

**Bits 1 and 0:** LCD RAM relocation select (RLCT1, RLCT0)

Bits 1 and 0 select the LCD RAM address space.

| Bit 1: RLCT1 | Bit 0: RLCT0 | Description | |
|---|---|---|---|
| 0 | 0 | H'F740 toH'F77F | (initial value) |
| | 1 | H'F940 to H'F97F*[2] | |
| 1 | 0 | H'FB40 to H'FB7F*[2] | |
| | 1 | H'FD40 to H'FD7F*[1, 2] | |

Notes: 1. In devices with 1,024-byte RAM, if RLCT1 to 0 are set to 11, on-chip RAM addresses H'FB80 to H'FD7F become inaccessible.

2. In devices with 2,048-byte RAM, if RLCT1 to 0 are set to any value except 00, these on-chip RAM addresses become inaccessible.

## 2.9 Application Notes

### 2.9.1 Notes on Data Access

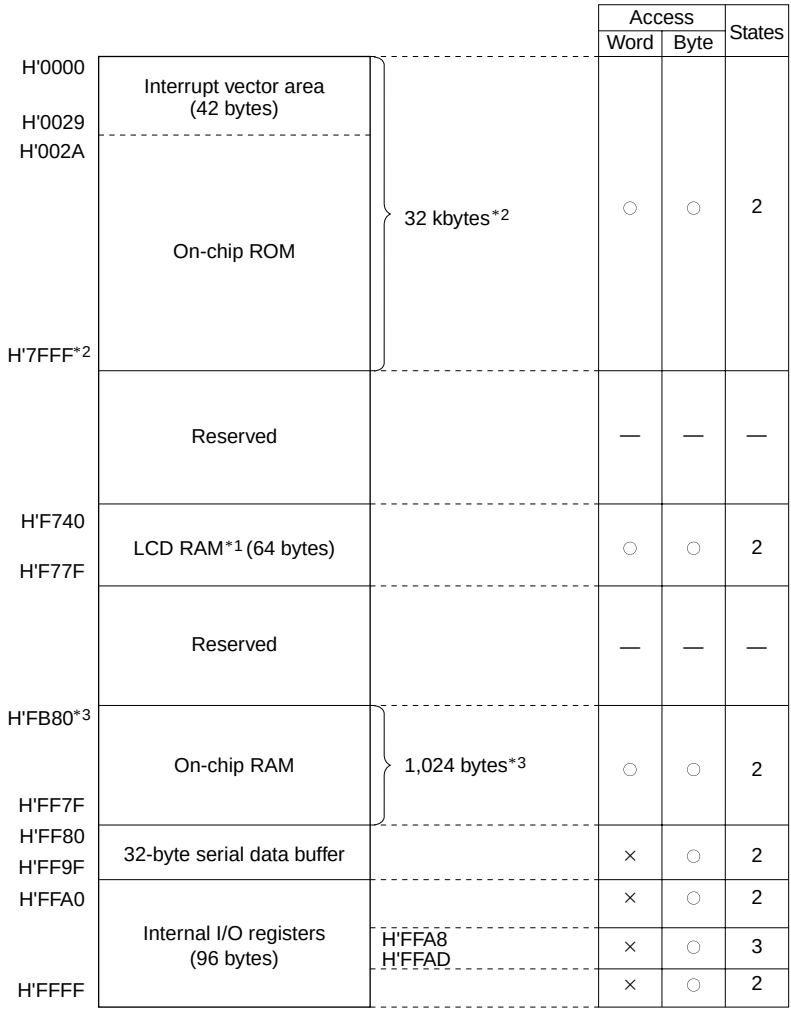
**Access to Empty Areas:** The address space of the H8/300L CPU includes empty areas in addition to the RAM, registers, and ROM areas available to the user. If these empty areas are mistakenly accessed by an application program, the following results will occur.

- Data transfer from CPU to empty area
  The transferred data will be lost. This action may also cause the CPU to misoperate.
- Data transfer from empty area to CPU
  Unpredictable data is transferred.

**Access to Internal I/O Registers:** Internal data transfer to or from on-chip modules other than the ROM and RAM areas makes use of an 8-bit data width. If word access is attempted to these areas, the following results will occur.

- Word access from CPU to I/O register area
  Upper byte: Will be written to I/O register.
  Lower byte: Transferred data will be lost.
- Word access from I/O register to CPU
  Upper byte: Will be written to upper part of CPU register.
  Lower byte: Unpredictable data will be written to lower part of CPU register.

Byte size instructions should therefore be used when transferring data to or from I/O registers other than the on-chip ROM and RAM areas. Figure 2.22 shows the data size and number of states in which on-chip peripheral modules can be accessed.




Notes: The above example is a description of the H8/3834.

1. The indicated addresses for the LCD RAM area are initial values after system reset.
2. The H8/3832 has 16 kbytes of on-chip ROM, and its ending address is H'3FFF.
   The H8/3833 has 24 kbytes of on-chip ROM, and its ending address is H'5FFF.
   The H8/3835 has 40 kbytes of on-chip ROM, and its ending address is H'9FFF.
   The H8/3836 has 48 kbytes of on-chip ROM, and its ending address is H'BFFF.
   The H8/3837 has 60 kbytes of on-chip ROM, and its ending address is H'EDFF.
3. The H8/3832 and H8/3833 have 1,024 bytes of on-chip RAM and its starting address is H8/3834.
   The H8/3835, H8/3836, and H8/3837 each have 2,048 bytes of on-chip RAM, and their starting address is H'F780.

**Figure 2.22 Data Size and Number of States for Access to and from On-Chip Peripheral Modules**

### 2.9.2 Notes on Bit Manipulation

The BSET, BCLR, BNOT, BST, and BIST instructions read one byte of data, modify the data, then write the data byte again. Special care is required when using these instructions in cases where two registers are assigned to the same address, in the case of registers that include write-only bits, and when the instruction accesses an I/O.

| Order of Operation | | Operation |
|---|---|---|
| 1 | Read | Read byte data at the designated address |
| 2 | Modify | Modify a designated bit in the read data |
| 3 | Write | Write the altered byte data to the designated address |

**Bit Manipulation in Two Registers Assigned to the Aame Address**

**Example 1:** Timer load register and timer count bit manipulation

Figure 2.23 shows an example in which two timer registers share 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 take place.

| Order of Operation | | Operation |
|---|---|---|
| 1 | Read | Timer counter data is read (one byte) |
| 2 | Modify | The CPU modifies (sets or resets) the bit designated in the instruction |
| 3 | Write | The altered byte data is written to the timer load register |

The timer counter 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 load register may be modified to the timer counter value.

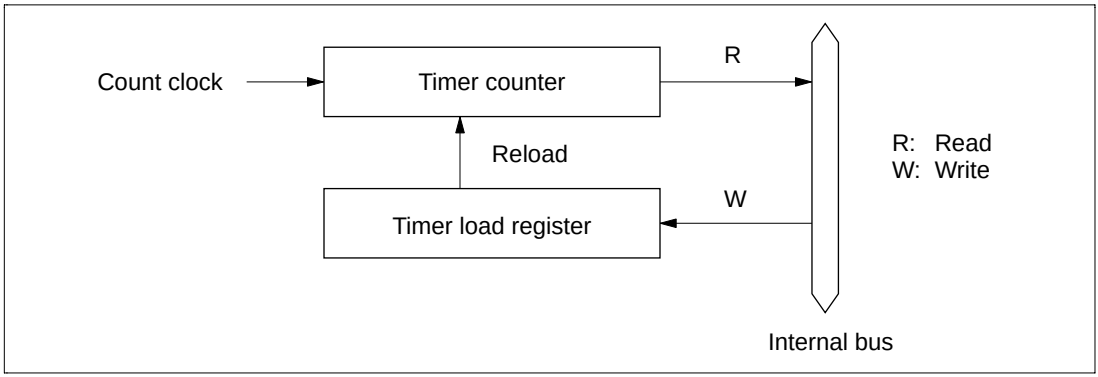



**Figure 2.23 Timer Configuration Example**

**Example 2:** When a BSET instruction is executed on port 3

Here a BSET instruction is executed designating port 3.

$P3_7$ and $P3_6$ are designated as input pins, with a low-level signal input at $P3_7$ and a high-level signal at $P3_6$. The remaining pins, $P3_5$ to $P3_0$, are output pins and output low-level signals. In this example, the BSET instruction is used to change pin $P3_0$ to high-level output.

[A: Prior to executing BSET]

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| PDR3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

[B: BSET instruction executed]

```
BSET    #0  ,  @PDR3
```

The BSET instruction is executed designating port 3.

[C: After executing BSET]

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| PDR3 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |

[D: Explanation of how BSET operates]

When the BSET instruction is executed, first the CPU reads port 3.

Since $P3_7$ and $P3_6$ are input pins, the CPU reads the pin states (low-level and high-level input). $P3_5$ to $P3_0$ are output pins, so the CPU reads the value in PDR3. In this example PDR3 has a value of H'80, but the value read by the CPU is H'40.

Next, the CPU sets bit 0 of the read data to 1, changing the PDR3 data to H'41. Finally, the CPU writes this value (H'41) to PDR3, completing execution of BSET.

As a result of this operation, bit 0 in PDR3 becomes 1, and $P3_0$ outputs a high-level signal. However, bits 7 and 6 of PDR3 end up with different values.

To avoid this problem, store a copy of the PDR3 data in a work area in memory. Perform the bit manipulation on the data in the work area, then write this data to PDR3.

[A: Prior to executing BSET]

```
MOV. B  #80,  R0L
MOV. B  R0L,  @RAM0
MOV. B  R0L,  @PDR3
```

The PDR3 value (H'80) is written to a work area in memory (RAM0) as well as to PDR3.

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| PDR3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RAM0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

[B: BSET instruction executed]

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

The BSET instruction is executed designating the PDR3 work area (RAM0).

[C: After executing BSET]

```
MOV. B  @RAM0, R0L
MOV. B  R0L,  @PDR3
```

The work area (RAM0) value is written to PDR3.

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| PDR3 | 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:** When a BCLR instruction is executed on PCR3 of port 3

In this example, the port 3 control register PCR3 is accessed by a BCLR instruction.

As in the examples above, $P3_7$ and $P3_6$ are input pins, with a low-level signal input at $P3_7$ and a high-level signal at $P3_6$. The remaining pins, $P3_5$ to $P3_0$, are output pins that output low-level signals. In this example, the BCLR instruction is used to change pin $P3_0$ to an input port. It is assumed that a high-level signal will be input to this input pin.

[A: Prior to executing BCLR]

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| PDR3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

[B: BCLR instruction executed]

```
BCLR   #0  ,  @PCR3
```

The BCLR instruction is executed designating PCR3.

[C: After executing BCLR]

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| PDR3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

[D: Explanation of how BCLR operates]

When the BCLR instruction is executed, first the CPU reads PCR3. Since PCR3 is a write-only register, the CPU reads a value of H'FF, even though the PCR3 value is actually H'3F.

Next, the CPU clears bit 0 in the read data to 0, changing the data to H'FE. Finally, this value (H'FE) is written to PCR3 and BCLR instruction execution ends.

As a result of this operation, bit 0 in PCR3 becomes 0, making $P3_0$ an input port. However, bits 7 and 6 in PCR3 change to 1, so that $P3_7$ and $P3_6$ change from input pins to output pins.

To avoid this problem, store a copy of the PCR3 data in a work area in memory. Perform the bit manipulation on the data in the work area, then write this data to PCR3.

[A: Prior to executing BCLR]

```
MOV. B  #3F,  R0L
MOV. B  R0L,  @RAM0
MOV. B  R0L,  @PCR3
```

The PCR3 value (H'3F) is written to a work area in memory (RAM0) as well as to PCR3.

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| PDR3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RAM0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |

[B: BCLR instruction executed]

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

The BCLR instruction is executed designating the PCR3 work area (RAM0).

[C: After executing BCLR]

```
MOV. B  @RAM0, R0L
MOV. B  R0L,  @PCR3
```

The work area (RAM0) value is written to PCR3.

| | **$P3_7$** | **$P3_6$** | **$P3_5$** | **$P3_4$** | **$P3_3$** | **$P3_2$** | **$P3_1$** | **$P3_0$** |
|---|---|---|---|---|---|---|---|---|
| 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 |
| PCR3 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |
| PDR3 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| RAM0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |

Table 2.12 lists registers that share the same address, and table 2.13 lists registers that contain write-only bits.

**Table 2.12 Registers with shared addresses**

| Register Name | Abbreviation | Address |
|---|---|---|
| Timer counter B and timer load register B | TCB/TLB | H'FFB3 |
| Timer counter C and timer load register C | TCC/TLC | H'FFB5 |
| Port data register 1* | PDR1 | H'FFD4 |
| Port data register 2* | PDR2 | H'FFD5 |
| Port data register 3* | PDR3 | H'FFD6 |
| Port data register 4* | PDR4 | H'FFD7 |
| Port data register 5* | PDR5 | H'FFD8 |
| Port data register 6* | PDR6 | H'FFD9 |
| Port data register 7* | PDR7 | H'FFDA |
| Port data register 8* | PDR8 | H'FFDB |
| Port data register 9* | PDR9 | H'FFDC |
| Port data register A* | PDRA | H'FFDD |

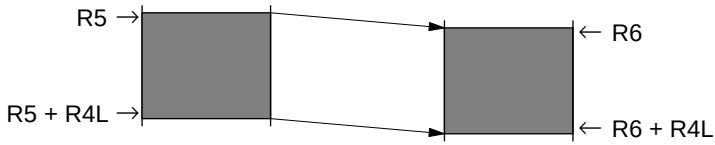
Note: * These port registers are used also for pin input.

**Table 2.13 Registers with write-only bits**

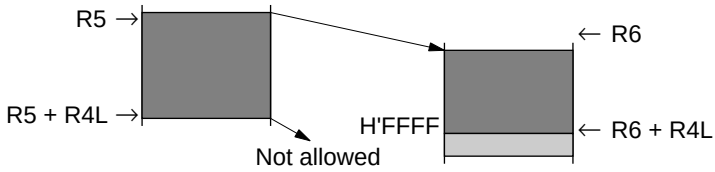
| Register Name | Abbreviation | Address |
|---|---|---|
| Port control register 1 | PCR1 | H'FFE4 |
| Port control register 2 | PCR2 | H'FFE5 |
| Port control register 3 | PCR3 | H'FFE6 |
| Port control register 4 | PCR4 | H'FFE7 |
| Port control register 5 | PCR5 | H'FFE8 |
| Port control register 6 | PCR6 | H'FFE9 |
| Port control register 7 | PCR7 | H'FFEA |
| Port control register 8 | PCR8 | H'FFEB |
| Port control register 9 | PCR9 | H'FFEC |
| Port control register A | PCRA | H'FFED |
| Timer control register F | TCRF | H'FFB6 |
| PWM control register | PWCR | H'FFD0 |
| PWM data register U | PWDRU | H'FFD1 |
| PWM data register L | PWDRL | H'FFD2 |

### 2.9.3 Notes on Use of the EEPMOV Instruction

- The EEPMOV instruction is a block data transfer instruction. It moves the number of bytes specified by R4L from the address specified by R5 to the address specified by R6.




- When setting R4L and R6, make sure that the final destination address (R6 + R4L) does not exceed H'FFFF. The value in R6 must not change from H'FFFF to H'0000 during execution of the instruction.



# Section 3 Exception Handling

## 3.1 Overview

Exception handling is performed in the H8/3834 Series when a reset or interrupt occurs. Table 3.1 shows the priorities of these two types of exception handling.

**Table 3.1 Exception Handling Types and Priorities**

| Priority | Exception Source | Time of Start of Exception Handling |
|---|---|---|
| High | Reset | Exception handling starts as soon as the reset state is cleared |
| ↑ | Interrupt | When an interrupt is requested, exception handling starts after execution of the present instruction or the exception handling in progress is completed |
| Low | | |

## 3.2 Reset

### 3.2.1 Overview

A reset is the highest-priority exception. The internal state of the CPU and the registers of the on-chip peripheral modules are initialized.

### 3.2.2 Reset Sequence

As soon as the $\overline{\text{RES}}$ pin goes low, all processing is stopped and the H8/3834 enters the reset state.

To make sure the chip is reset properly, observe the following precautions.

- At power on: Hold the $\overline{\text{RES}}$ pin low until the clock pulse generator output stabilizes.
- Resetting during operation: Hold the $\overline{\text{RES}}$ pin low for at least 10 system clock cycles.

If the MD0 pin is at the high level, reset exception handling begins when the $\overline{\text{RES}}$ pin is held low for a given period, then returned to the high level. If the MD0 pin is low, however, when the $\overline{\text{RES}}$ pin is held low for a given period and then returned to high level, the reset is not cleared immediately. First the MD0 pin must go from low to high, then after 8,192 clock cycles the reset is cleared and reset exception handling begins.

Reset exception handling takes place as follows.

- The CPU internal state and the registers of on-chip peripheral modules are initialized, with the I bit of the condition code register (CCR) set to 1.
- The PC is loaded from the reset exception handling vector address (H'0000 to H'0001), after which the program starts executing from the address indicated in PC.

When system power is turned on or off, the $\overline{\text{RES}}$ pin should be held low.

Figures 3.1 and 3.2 show the reset sequence.

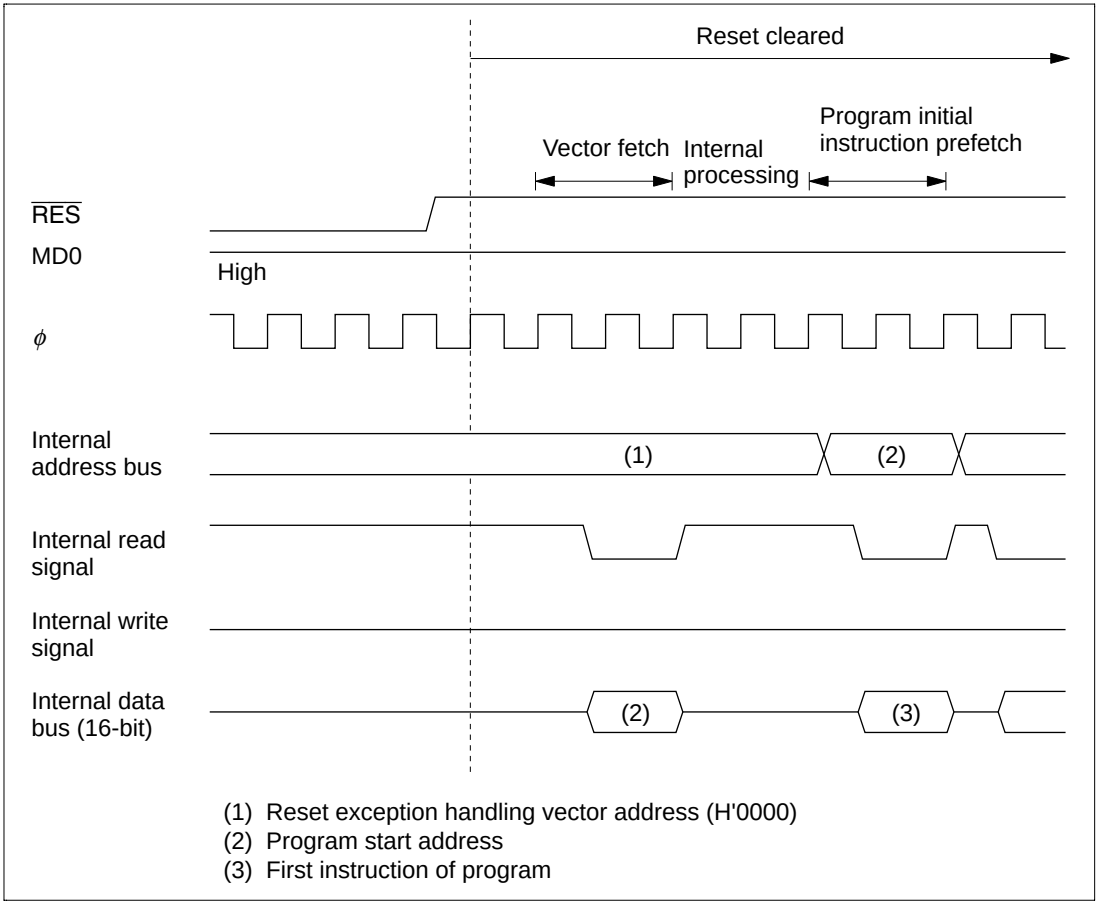



**Figure 3.1 Reset Sequence (when MD0 Pin is High)**

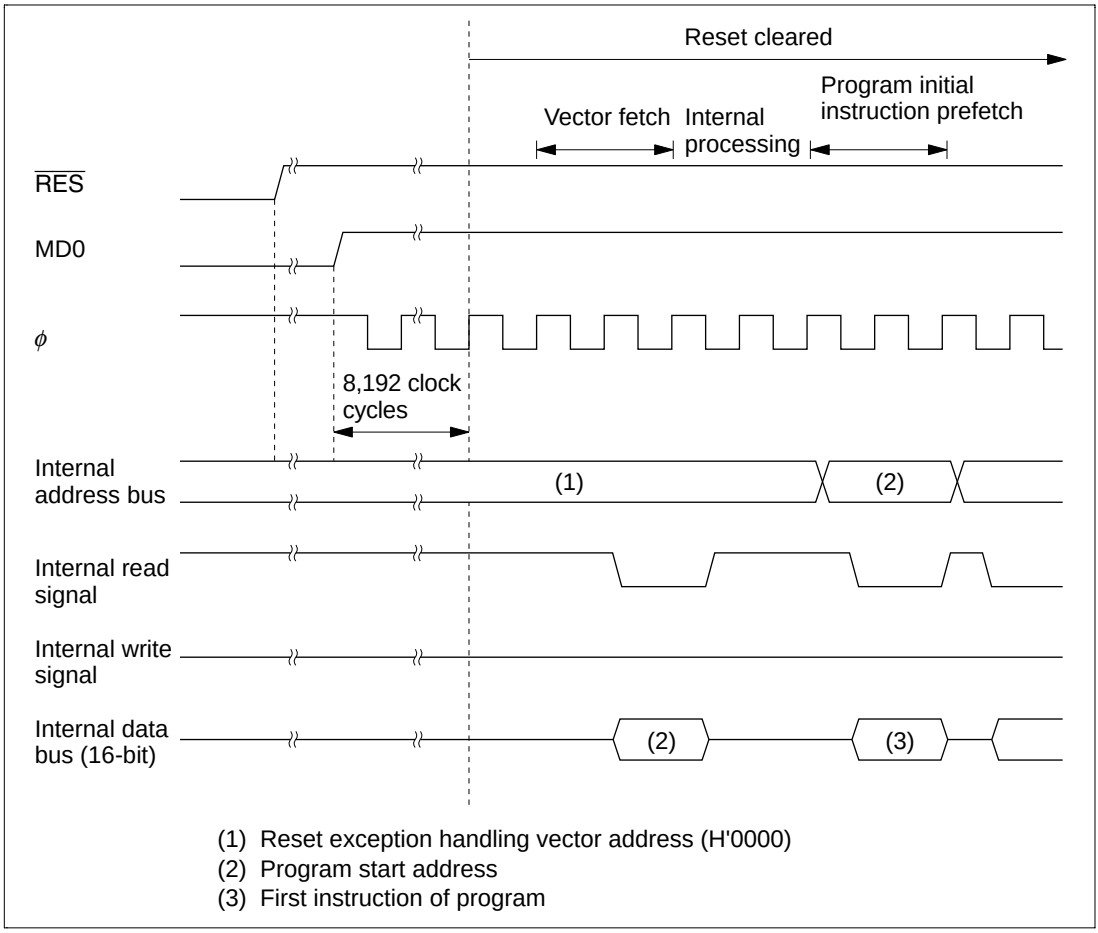



**Figure 3.2 Reset Sequence (when MD0 Pin is Low)**

### 3.2.3 Interrupt Immediately after Reset

After a reset, if an interrupt were to be accepted before the stack pointer (SP: R7) was initialized, PC and CCR would not be pushed onto the stack correctly, resulting in program runaway. To prevent this, immediately after reset exception handling all interrupts are masked. For this reason, the initial program instruction is always executed immediately after a reset. This instruction should initialize the stack pointer (e.g. MOV.W #xx: 16, SP).

## 3.3 Interrupts

### 3.3.1 Overview

The interrupt sources include 13 external interrupts ($WKP_0$ to $WKP_7$, $IRQ_0$ to $IRQ_4$), and 20 internal interrupts from on-chip peripheral modules. Table 3.2 shows the interrupt sources, their priorities, and their vector addresses. When more than one interrupt is requested, the interrupt with the highest priority is processed.

The interrupts have the following features:

- Both internal and external interrupts can be masked by the I bit of CCR. When this bit is set to 1, interrupt request flags are set but interrupts are not accepted.
- The external interrupt pins $IRQ_0$ to $IRQ_4$ can each be set independently to either rising edge sensing or falling edge sensing.

**Table 3.2 Interrupt Sources and Priorities**

| Priority | Interrupt Source | Interrupt | Vector Number | Vector Address |
|---|---|---|---|---|
| High | $\overline{RES}$ | Reset | 0 | H'0000 to H'0001 |
| | $\overline{IRQ_0}$ | $IRQ_0$ | 4 | H'0008 to H'0009 |
| | $\overline{IRQ_1}$ | $IRQ_1$ | 5 | H'000A to H'000B |
| | $\overline{IRQ_2}$ | $IRQ_2$ | 6 | H'000C to H'000D |
| | $\overline{IRQ_3}$ | $IRQ_3$ | 7 | H'000E to H'000F |
| | $\overline{IRQ_4}$ | $IRQ_4$ | 8 | H'0010 to H'0011 |
| | $\overline{WKP_0}$ | $WKP_0$ | 9 | H'0012 to H'0013 |
| | $\overline{WKP_1}$ | $WKP_1$ | | |
| | $\overline{WKP_2}$ | $WKP_2$ | | |
| | $\overline{WKP_3}$ | $WKP_3$ | | |
| | $\overline{WKP_4}$ | $WKP_4$ | | |
| | $\overline{WKP_5}$ | $WKP_5$ | | |
| | $\overline{WKP_6}$ | $WKP_6$ | | |
| | $\overline{WKP_7}$ | $WKP_7$ | | |
| | SCI1 | SCI1 transfer complete | 10 | H'0014 to H'0015 |
| | Timer A | Timer A overflow | 11 | H'0016 to H'0017 |
| | Timer B | Timer B overflow | 12 | H'0018 to H'0019 |
| Low | Timer C | Timer C overflow or underflow | 13 | H'001A to H'001B |

**Table 3.2 Interrupt Sources and Priorities (cont)**

| Priority | Interrupt Source | Interrupt | Vector Number | Vector Address |
|---|---|---|---|---|
| High | Timer FL | Timer FL compare match | 14 | H'001C to H'001D |
| | | Timer FL overflow | | |
| | Timer FH | Timer FH compare match | 15 | H'001E to H'001F |
| | | Timer FH overflow | | |
| | Timer G | Timer G input capture | 16 | H'0020 to H'0021 |
| | | Timer G overflow | | |
| | SCI2 | SCI2 transfer complete | 17 | H'0022 to H'0023 |
| | | SCI2 transfer abort | | |
| | SCI3 | SCI3 transmit end | 18 | H'0024 to H'0025 |
| | | SCI3 transmit data empty | | |
| | | SCI3 receive data full | | |
| | | SCI3 overrun error | | |
| | | SCI3 framing error | | |
| | | SCI3 parity error | | |
| | A/D converter | A/D conversion end | 19 | H'0026 to H'0027 |
| Low | (SLEEP instruction executed) | Direct transfer | 20 | H'0028 to H'0029 |

Note: Vector addresses H'0002 to H'0007 are reserved and cannot be used.

### 3.3.2 Interrupt Control Registers

Table 3.3 lists the registers that control interrupts.

**Table 3.3 Interrupt Control Registers**

| Register Name | Abbreviation | R/W | Initial Value | Address |
|---|---|---|---|---|
| IRQ edge select register | IEGR | R/W | H'E0 | H'FFF2 |
| Interrupt enable register 1 | IENR1 | R/W | H'00 | H'FFF3 |
| Interrupt enable register 2 | IENR2 | R/W | H'00 | H'FFF4 |
| Interrupt request register 1 | IRR1 | R/W* | H'20 | H'FFF6 |
| Interrupt request register 2 | IRR2 | R/W* | H'00 | H'FFF7 |
| Wakeup interrupt request register | IWPR | R/W* | H'00 | H'FFF9 |

Note: * Write is enabled only for writing of 0 to clear a flag.

**IRQ Edge Select Register (IEGR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | IEG4 | IEG3 | IEG2 | IEG1 | IEG0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

IEGR is an 8-bit read/write register, used to designate whether pins $\overline{IRQ_0}$ to $\overline{IRQ_4}$ are set to rising edge sensing or falling edge sensing.

**Bits 7 to 5—Reserved Bits:** Bits 7 to 5 are reserved; they are always read as 1, and cannot be modified.

**Bit 4—$\overline{IRQ_4}$ Edge Select (IEG4):** Bit 4 selects the input sensing of pin $\overline{IRQ_4}$/$\overline{ADTRG}$.

| Bit 4: IEG4 | Description | |
|---|---|---|
| 0 | Falling edge of $\overline{IRQ_4}$/$\overline{ADTRG}$ pin input is detected | (initial value) |
| 1 | Rising edge of $\overline{IRQ_4}$/$\overline{ADTRG}$ pin input is detected | |

**Bit 3—$\overline{IRQ_3}$ Edge Select (IEG3):** Bit 3 selects the input sensing of pin $\overline{IRQ_3}$/TMIF.

| Bit 3: IEG3 | Description | |
|---|---|---|
| 0 | Falling edge of $\overline{IRQ_3}$/TMIF pin input is detected | (initial value) |
| 1 | Rising edge of $\overline{IRQ_3}$/TMIF pin input is detected | |

**Bit 2—$\overline{IRQ_2}$ Edge Select (IEG2):** Bit 2 selects the input sensing of pin $\overline{IRQ_2}$/TMIC.

| Bit 2: IEG2 | Description | |
|---|---|---|
| 0 | Falling edge of $\overline{IRQ_2}$/TMIC pin input is detected | (initial value) |
| 1 | Rising edge of $\overline{IRQ_2}$/TMIC pin input is detected | |

**Bit 1—$\overline{IRQ_1}$ Edge Select (IEG1):** Bit 1 selects the input sensing of pin $\overline{IRQ_1}$/TMIB.

| Bit 1: IEG1 | Description | |
|---|---|---|
| 0 | Falling edge of $\overline{IRQ_1}$/TMIB pin input is detected | (initial value) |
| 1 | Rising edge of $\overline{IRQ_1}$/TMIB pin input is detected | |

**Bit 0—$\overline{IRQ_0}$ Edge Select (IEG0):** Bit 0 selects the input sensing of pin $\overline{IRQ_0}$.

| Bit 0: IEG0 | Description | |
|---|---|---|
| 0 | Falling edge of $\overline{IRQ_0}$ pin input is detected | (initial value) |
| 1 | Rising edge of $\overline{IRQ_0}$ pin input is detected | |

**Interrupt Enable Register 1 (IENR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IENTA | IENS1 | IENWP | IEN4 | IEN3 | IEN2 | IEN1 | IEN0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

IENR1 is an 8-bit read/write register that enables or disables interrupt requests.

**Bit 7—Timer A Interrupt Enable (IENTA):** Bit 7 enables or disables timer A overflow interrupt requests.

| Bit 7: IENTA | Description | |
|---|---|---|
| 0 | Disables timer A interrupts | (initial value) |
| 1 | Enables timer A interrupts | |

**Bit 6—SCI1 Interrupt Enable (IENS1):** Bit 6 enables or disables SCI1 transfer complete interrupt requests.

| Bit 6: IENS1 | Description | |
|---|---|---|
| 0 | Disables SCI1 interrupts | (initial value) |
| 1 | Enables SCI1 interrupts | |

**Bit 5—Wakeup Interrupt Enable (IENWP):** Bit 5 enables or disables $WKP_7$ to $WKP_0$ interrupt requests.

| Bit 5: IENWP | Description | |
|---|---|---|
| 0 | Disables interrupt requests from $\overline{WKP_7}$ to $\overline{WKP_0}$ | (initial value) |
| 1 | Enables interrupt requests from $\overline{WKP_7}$ to $\overline{WKP_0}$ | |

**Bits 4 to 0—$IRQ_4$ to $IRQ_0$ Interrupt Enable (IEN4 to IEN0):** Bits 4 to 0 enable or disable $IRQ_4$ to $IRQ_0$ interrupt requests.

| Bit n: IENn | Description | |
|---|---|---|
| 0 | Disables interrupt request $\overline{IRQn}$ | (initial value) |
| 1 | Enables interrupt request $\overline{IRQn}$ | |

(n = 4 to 0)

**Interrupt Enable Register 2 (IENR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IENDT | IENAD | IENS2 | IENTG | IENTFH | IENTFL | IENTC | IENTB |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

IENR2 is an 8-bit read/write register that enables or disables interrupt requests.

**Bit 7—Direct Transfer Interrupt Enable (IENDT):** Bit 7 enables or disables direct transfer interrupt requests.

| Bit 7: IENDT | Description | |
|---|---|---|
| 0 | Disables direct transfer interrupt requests | (initial value) |
| 1 | Enables direct transfer interrupt requests | |

**Bit 6—A/D Converter Interrupt Enable (IENAD):** Bit 6 enables or disables A/D converter interrupt requests.

| Bit 6: IENAD | Description | |
|---|---|---|
| 0 | Disables A/D converter interrupt requests | (initial value) |
| 1 | Enables A/D converter interrupt requests | |

**Bit 5—SCI2 Interrupt Enable (IENS2):** Bit 5 enables or disables SCI2 transfer complete and transfer abort interrupt requests.

| Bit 5: IENS2 | Description | |
|---|---|---|
| 0 | Disables SCI2 interrupts | (initial value) |
| 1 | Enables SCI2 interrupts | |

**Bit 4—Timer G Interrupt Enable (IENTG):** Bit 4 enables or disables timer G input capture and overflow interrupt requests.

| Bit 4: IENTG | Description | |
|---|---|---|
| 0 | Disables timer G interrupts | (initial value) |
| 1 | Enables timer G interrupts | |

**Bit 3—Timer FH Interrupt Enable (IENTFH):** Bit 3 enables or disables timer FH compare match and overflow interrupt requests.

| Bit 3: IENTFH | Description | |
|---|---|---|
| 0 | Disables timer FH interrupts | (initial value) |
| 1 | Enables timer FH interrupts | |

**Bit 2—Timer FL Interrupt Enable (IENTFL):** Bit 2 enables or disables timer FL compare match and overflow interrupt requests.

| Bit 2: IENTFL | Description | |
|---|---|---|
| 0 | Disables timer FL interrupts | (initial value) |
| 1 | Enables timer FL interrupts | |

**Bit 1—Timer C Interrupt Enable (IENTC):** Bit 1 enables or disables timer C overflow or underflow interrupt requests.

| Bit 1: IENTC | Description | |
|---|---|---|
| 0 | Disables timer C interrupts | (initial value) |
| 1 | Enables timer C interrupts | |

**Bit 0—Timer B Interrupt Enable (IENTB):** Bit 0 enables or disables timer B overflow or underflow interrupt requests.

| Bit 0: IENTB | Description | |
|---|---|---|
| 0 | Disables timer B interrupts | (initial value) |
| 1 | Enables timer B interrupts | |

SCI3 interrupt control is covered in 10.4.2, in the description of serial control register 3.

**Interrupt request register 1 (IRR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IRRTA | IRRS1 | — | IRRI4 | IRRI3 | IRRI2 | IRRI1 | IRRI0 |
| Initial value | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | — | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * Only a write of 0 for flag clearing is possible.

IRR1 is an 8-bit read/write register, in which the corresponding bit is set to 1 when a timer A, SCI1, or $IRQ_4$ to $IRQ_0$ interrupt is requested. The flags are not cleared automatically when an interrupt is accepted. It is necessary to write 0 to clear each flag.

**Bit 7—Timer A Interrupt Request Flag (IRRTA)**

| Bit 7: IRRTA | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRTA = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When the timer A counter value overflows (goes from H'FF to H'00) |

**Bit 6—SCI1 Interrupt Request Flag (IRRS1)**

| Bit 6: IRRS1 | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRS1 = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When an SCI1 transfer is completed |

**Bit 5—Reserved Bit:** Bit 5 is reserved; it is always read as 1, and cannot be modified.

**Bits 4 to 0—$IRQ_4$ to $IRQ_0$ Interrupt Request Flags (IRRI4 to IRRI0)**

| Bit n: IRRIn | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRIn = 1, it is cleared by writing 0 to IRRIn (initial value) |
| 1 | Setting conditions:<br>IRRIn is set when pin $\overline{IRQ}_n$ is set to interrupt input, and the designated signal edge is detected |

(n = 4 to 0)

**Interrupt Request Register 2 (IRR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IRRDT | IRRAD | IRRS2 | IRRTG | IRRTFH | IRRTFL | IRRTC | IRRTB |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * Only a write of 0 for flag clearing is possible.

IRR2 is an 8-bit read/write register, in which the corresponding bit is set to 1 when a direct transfer, A/D converter, SCI2, timer G, timer FH, timer FL, timer C, or timer B interrupt is requested. The flags are not cleared automatically when an interrupt is accepted. It is necessary to write 0 to clear each flag.

**Bit 7—Direct Transfer Interrupt Request Flag (IRRDT)**

| Bit 7: IRRDT | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRDT = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When DTON = 1 and a direct transfer is made immediately after a SLEEP instruction is executed |

**Bit 6—A/D Converter Interrupt Request Flag (IRRAD)**

| Bit 6: IRRAD | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRAD = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When A/D conversion is completed and ADSF is reset |

**Bit 5—SCI2 Interrupt Request Flag (IRRS2)**

| Bit 5: IRRS2 | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRS2 = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When an SCI2 transfer is completed or aborted |

**Bit 4—Timer G Interrupt Request Flag (IRRTG)**

| Bit 4: IRRTG | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRTG = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When pin TMIG is set to TMIG input and the designated signal edge is detected |

**Bit 3—Timer FH Interrupt Request Flag (IRRTFH)**

| Bit 3: IRRTFH | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRTFH = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When counter FH matches output compare register FH in 8-bit timer mode, or when 16-bit counter F (TCFL, TCFH) matches output compare register F (OCRFL, OCRFH) in 16-bit timer mode |

**Bit 2—Timer FL Interrupt Request Flag (IRRTFL)**

| Bit 2: IRRTFL | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRTFL = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When counter FL matches output compare register FL in 8-bit timer mode |

**Bit 1—Timer C Interrupt Request Flag (IRRTC)**

| Bit 1: IRRTC | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRTC = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When the timer C counter value overflows (goes from H'FF to H'00) or underflows (goes from H'00 to H'FF) |

**Bit 0—Timer B Interrupt Request Flag (IRRTB)**

| Bit 0: IRRTB | Description |
|---|---|
| 0 | Clearing conditions:<br>When IRRTB = 1, it is cleared by writing 0 (initial value) |
| 1 | Setting conditions:<br>When the timer B counter value overflows (goes from H'FF to H'00) |

**Wakeup Interrupt Request Register (IWPR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IWPF7 | IWPF6 | IWPF5 | IWPF4 | IWPF3 | IWPF2 | IWPF1 | IWPF0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

Note: * Only a write of 0 for flag clearing is possible.

IWPR is an 8-bit read/write register, in which the corresponding bit is set to 1 when pins $\overline{WKP_7}$ to $\overline{WKP_0}$ are set to wakeup input and a pin receives a falling edge input. The flags are not cleared automatically when an interrupt is accepted. It is necessary to write 0 to clear each flag.

**Bits 7 to 0—Wakeup Interrupt Request Flags (WKPF7 to WKPF0)**

| Bit n: IWPFn | Description |
|---|---|
| 0 | Clearing conditions:<br>When IWPFn = 1, it is cleared by writing 0 to IWPFn |
| 1 | Setting conditions:<br>IWPFn is set when pin $\overline{WKP_n}$ is set to wakeup interrupt input, and a falling edge input is detected at the pin |

(n = 7 to 0)

### 3.3.3 External Interrupts

There are 13 external interrupts, $WKP_0$ to $WKP_7$ and $IRQ_0$ to $IRQ_4$.

**Interrupts $WKP_0$ to $WKP_7$:** Interrupts $WKP_0$ to $WKP_7$ are requested by falling edge inputs at pins $\overline{WKP_0}$ to $\overline{WKP_7}$. When these pins are designated as $\overline{WKP_0}$ to $\overline{WKP_7}$ pins in port mode register 5 (PMR5) and falling edge input is detected, the corresponding bit in the wakeup interrupt request register (IWPR) is set to 1, requesting an interrupt. Wakeup interrupt requests can be disabled by clearing the IENWP bit in IENR1 to 0. It is also possible to mask all interrupts by setting the CCR I bit to 1.

When an interrupt exception handling request is received for interrupts $WKP_0$ to $WKP_7$, the CCR I bit is set to 1. The vector number for interrupts $WKP_0$ to $WKP_7$ is 9. Since all eight interrupts are assigned the same vector number, the interrupt source must be determined by the exception handling routine.

**Interrupts $IRQ_0$ to $IRQ_4$:** Interrupts $IRQ_0$ to $IRQ_4$ are requested by into pins inputs to $\overline{IRQ_0}$ to $\overline{IRQ_4}$. These interrupts are detected by either rising edge sensing or falling edge sensing, depending on the settings of bits IEG0 to IEG4 in the edge select register (IEGR).

When these pins are designated as pins $\overline{IRQ_0}$ to $\overline{IRQ_4}$ in port mode registers 1 and 2 (PMR1 and PMR2) and the designated edge is input, the corresponding bit in IRR1 is set to 1, requesting an interrupt. Interrupts $IRQ_0$ to $IRQ_4$ can be disabled by clearing bits IEN0 to IEN4 in IENR1 to 0. All interrupts can be masked by setting the I bit in CCR to 1.

When $IRQ_0$ to $IRQ_4$ interrupt exception handling is initiated, the I bit is set to 1. Vector numbers 4 to 8 are assigned to interrupts $IRQ_0$ to $IRQ_4$. The order of priority is from $IRQ_0$ (high) to $IRQ_4$ (low). Table 3.2 gives details.

### 3.3.4 Internal Interrupts

There are 20 internal interrupts that can be requested by the on-chip peripheral modules. When a peripheral module requests an interrupt, the corresponding bit in IRR1 or IRR2 is set to 1. Individual interrupt requests can be disabled by clearing the corresponding bit in IENR1 or IENR2 to 0. All interrupts can be masked by setting the I bit in CCR to 1. When an internal interrupt request is accepted, the I bit is set to 1. Vector numbers 10 to 20 are assigned to these interrupts. Table 3.2 shows the order of priority of interrupts from on-chip peripheral modules.

### 3.3.5 Interrupt Operations

Interrupts are controlled by an interrupt controller. Figure 3.3 shows a block diagram of the interrupt controller. Figure 3.4 shows the flow up to interrupt acceptance.

Interrupt operation is described as follows.

- When an interrupt condition is met while the interrupt enable register bit is set to 1, an interrupt request signal is sent to the interrupt controller.
- When the interrupt controller receives an interrupt request, it sets the interrupt request flag.
- From among the interrupts with interrupt request flags set to 1, the interrupt controller selects the interrupt request with the highest priority and holds the others pending. (Refer to table 3.2 for a list of interrupt priorities.)
- The interrupt controller checks the I bit of CCR. If the I bit is 0, the selected interrupt request is accepted; if the I bit is 1, the interrupt request is held pending.

- If the interrupt is accepted, after processing of the current instruction is completed, both PC and CCR are pushed onto the stack. The state of the stack at this time is shown in figure 3.5. The PC value pushed onto the stack is the address of the first instruction to be executed upon return from interrupt handling.
- The I bit of CCR is set to 1, masking all further interrupts.
- The vector address corresponding to the accepted interrupt is generated, and the interrupt handling routine located at the address indicated by the contents of the vector address is executed.

Notes: 1. When disabling interrupts by clearing bits in an interrupt enable register, or when clearing bits in an interrupt request register, always do so while interrupts are masked (I = 1).

2. 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 processing for the interrupt will be executed after the clear instruction has been executed.

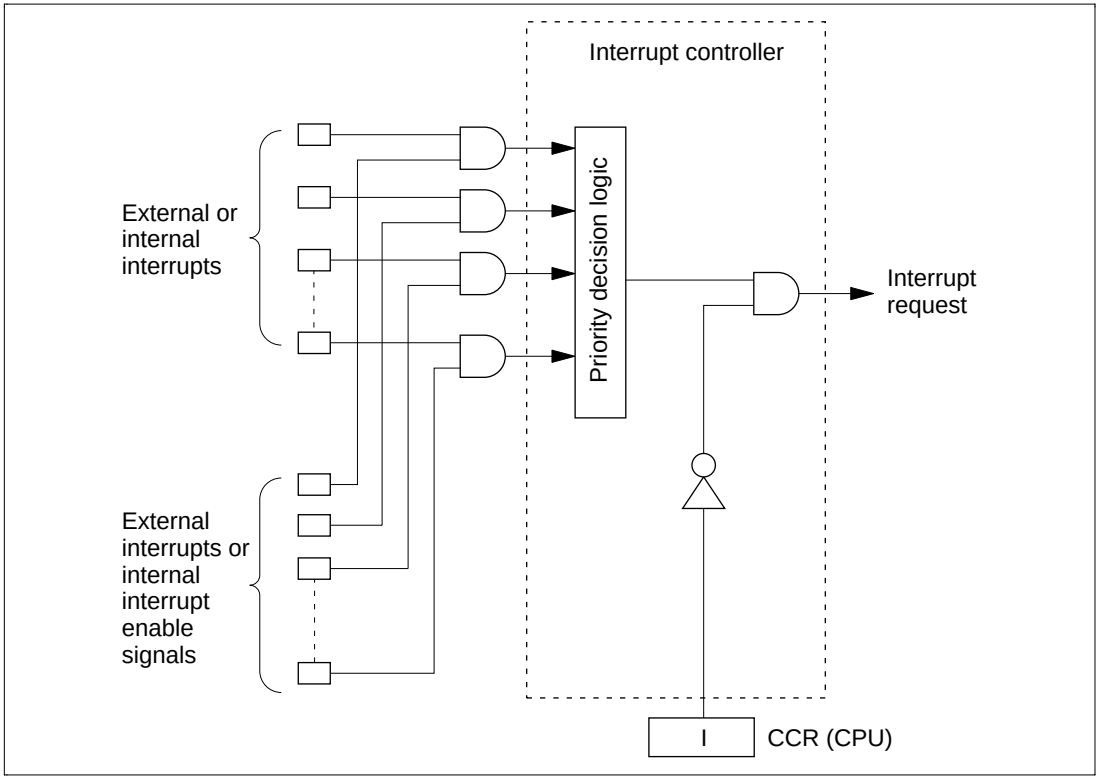
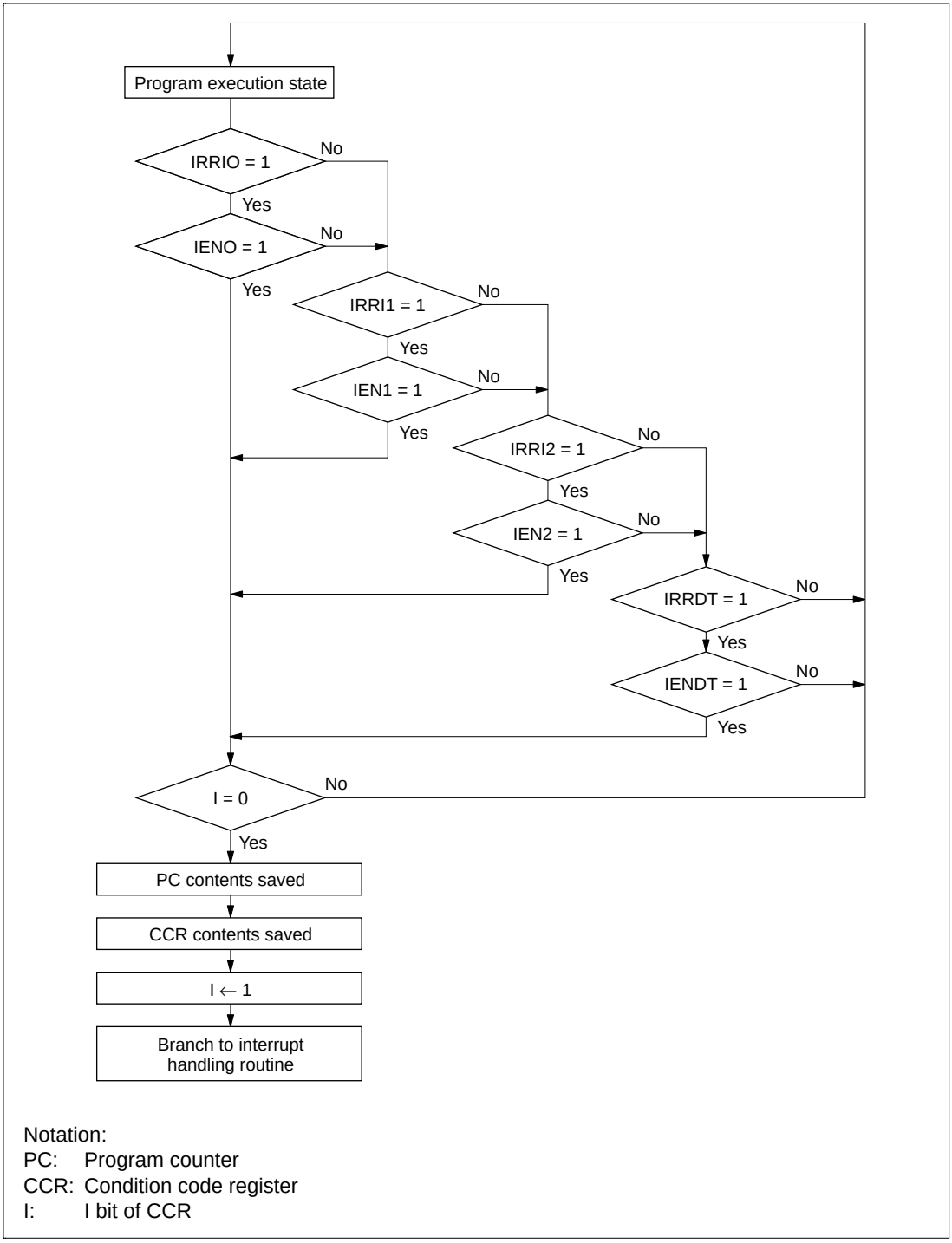
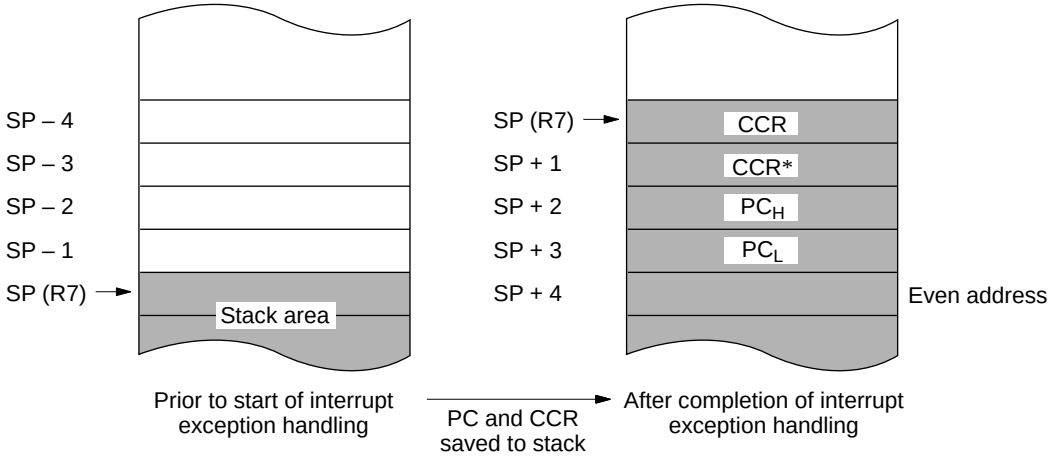



**Figure 3.3 Block Diagram of Interrupt Controller**




Notation:
PC: Program counter
CCR: Condition code register
I: I bit of CCR

**Figure 3.4 Flow up to Interrupt Acceptance**



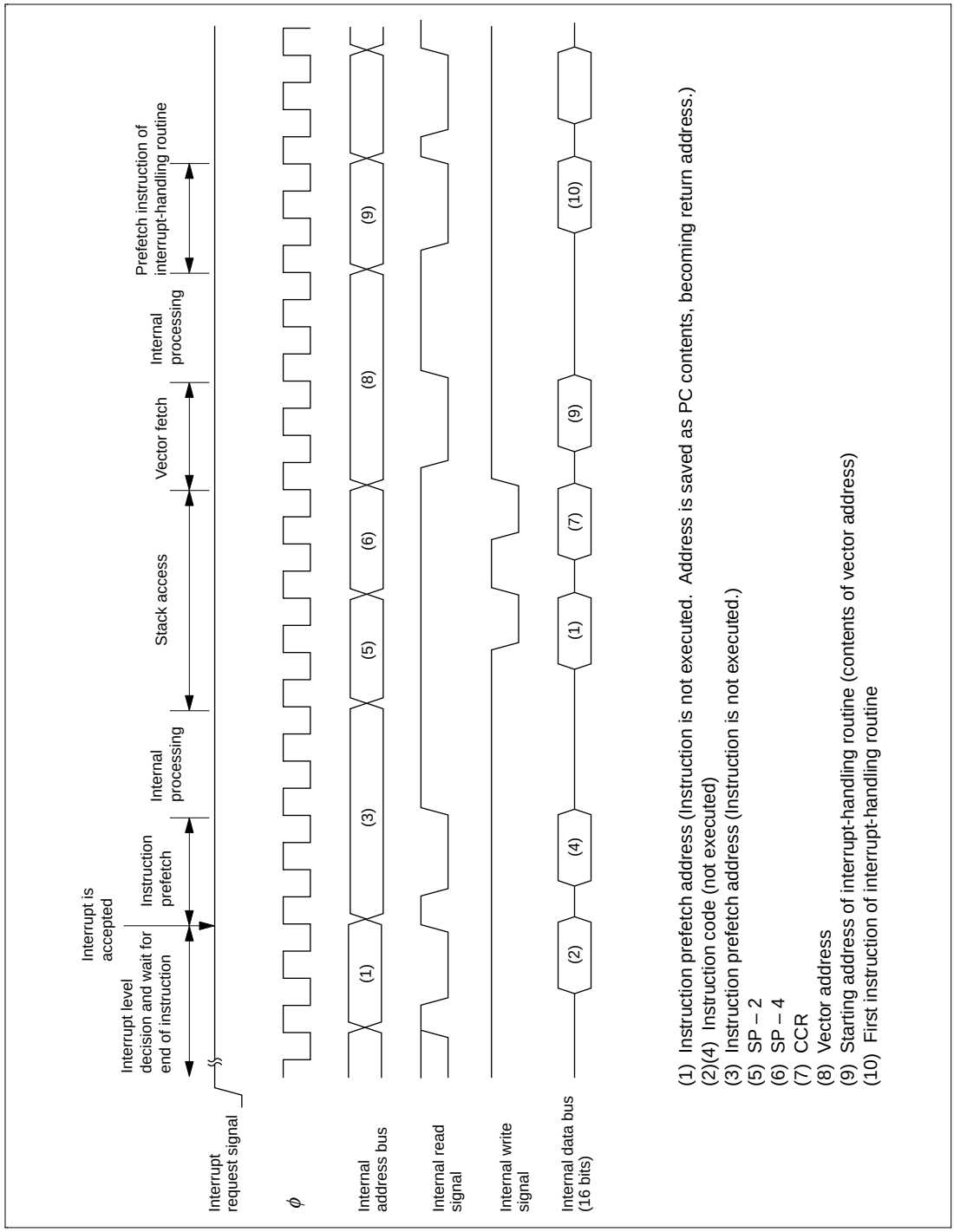

Notation:
$PC_H$: Upper 8 bits of program counter (PC)
$PC_L$: Lower 8 bits of program counter (PC)
CCR: Condition code register
SP: Stack pointer

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 access, starting from an even-numbered address.

* Ignored on return from interrupt.

**Figure 3.5 Stack State after Completion of Interrupt Exception Handling**

Figure 3.6 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.




(1) Instruction prefetch address (Instruction is not executed. Address is saved as PC contents, becoming return address.)
(2)(4) Instruction code (not executed)
(3) Instruction prefetch address (Instruction is not executed.)
(5) SP – 2
(6) SP – 4
(7) CCR
(8) Vector address
(9) Starting address of interrupt-handling routine (contents of vector address)
(10) First instruction of interrupt-handling routine

**Figure 3.6 Interrupt Sequence**

### 3.3.6 Interrupt Response Time

Table 3.4 shows the number of wait states after an interrupt request flag is set until the first instruction of the interrupt handler is executed.

**Table 3.4 Interrupt Wait States**

| Item | States |
|---|---|
| Waiting time for completion of executing instruction* | 1 to 13 |
| Saving of PC and CCR to stack | 4 |
| Vector fetch | 2 |
| Instruction fetch | 4 |
| Internal processing | 4 |
| Total | 15 to 27 |

Note: * Not including EEPMOV instruction.

## 3.4 Application Notes

### 3.4.1 Notes on Stack Area Use

When word data is accessed in the H8/3834 Series, 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.

Setting an odd address in SP may cause a program to crash. An example is shown in figure 3.7.

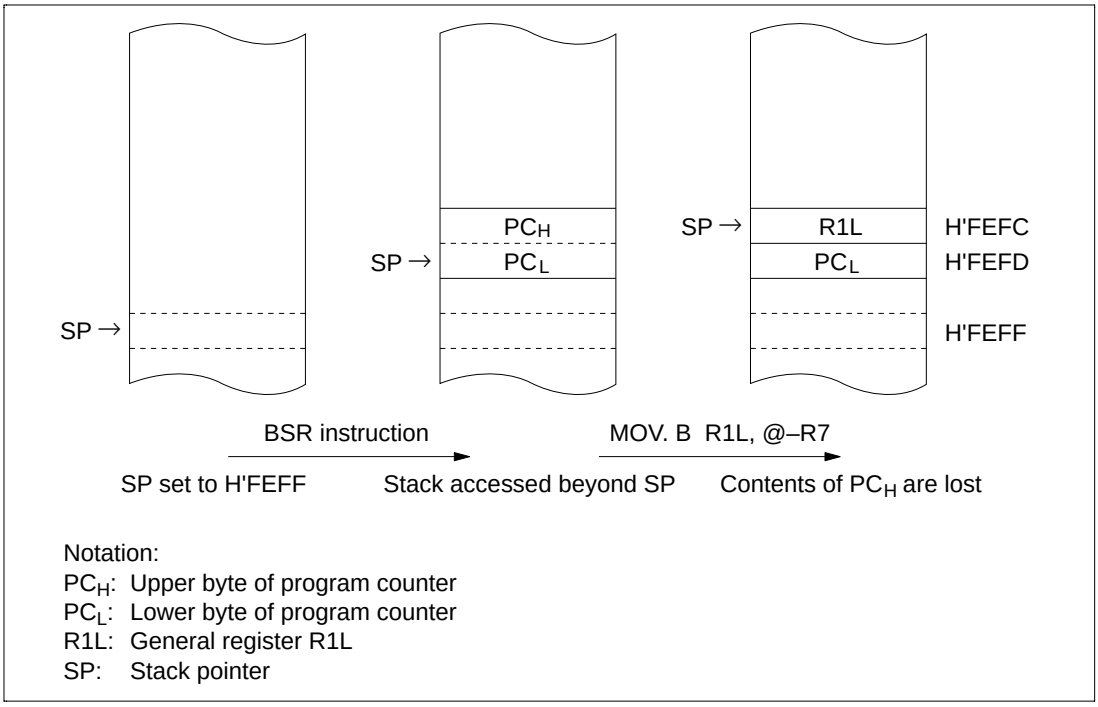



**Figure 3.7 Operation when Odd Address is Set in SP**

When CCR contents are saved to the stack during interrupt exception handling or restored when RTE is executed, this also takes place in word size. Both the upper and lower bytes of word data are saved to the stack; on return, the even address contents are restored to CCR while the odd address contents are ignored.

### 3.4.2 Notes on Rewriting Port Mode Registers

When a port mode register is rewritten to switch the functions of external interrupt pins, the following points should be observed.

When an external interrupt pin function is switched by rewriting the port mode register that controls these pins ($\overline{IRQ}_4$ to $\overline{IRQ}_0$, and $\overline{WKP}_7$ to $\overline{WKP}_0$), the interrupt request flag may be set to 1 at the time the pin function is switched, even if no valid interrupt is input at the pin. Be sure to clear the interrupt request flag to 0 after switching pin functions. Table 3.5 shows the conditions under which interrupt request flags are set to 1 in this way.

**Table 3.5 Conditions under which Interrupt Request Flag is Set to 1**

| Interrupt Request Flags Set to 1 | | Conditions |
|---|---|---|
| IRR1 | IRRI4 | • When PMR2 bit IRQ4 is changed from 0 to 1 while pin $\overline{IRQ}_4$ is low and IEGR bit IEG4 = 0.<br>• When PMR2 bit IRQ4 is changed from 1 to 0 while pin $\overline{IRQ}_4$ is low and IEGR bit IEG4 = 1. |
| | IRRI3 | • When PMR1 bit IRQ3 is changed from 0 to 1 while pin $\overline{IRQ}_3$ is low and IEGR bit IEG3 = 0.<br>• When PMR1 bit IRQ3 is changed from 1 to 0 while pin $\overline{IRQ}_3$ is low and IEGR bit IEG3 = 1. |
| | IRRI2 | • When PMR1 bit IRQ2 is changed from 0 to 1 while pin $\overline{IRQ}_2$ is low and IEGR bit IEG2 = 0.<br>• When PMR1 bit IRQ2 is changed from 1 to 0 while pin $\overline{IRQ}_2$ is low and IEGR bit IEG2 = 1. |
| | IRRI1 | • When PMR1 bit IRQ1 is changed from 0 to 1 while pin $\overline{IRQ}_1$ is low and IEGR bit IEG1 = 0.<br>• When PMR1 bit IRQ1 is changed from 1 to 0 while pin $\overline{IRQ}_1$ is low and IEGR bit IEG1 = 1. |
| | IRRI0 | • When PMR2 bit IRQ0 is changed from 0 to 1 while pin $\overline{IRQ}_0$ is low and IEGR bit IEG0 = 0.<br>• When PMR2 bit IRQ0 is changed from 1 to 0 while pin $\overline{IRQ}_0$ is low and IEGR bit IEG0 = 1. |
| IWPR | IWPF7 | When PMR5 bit WKP7 is changed from 0 to 1 while pin $\overline{WKP}_7$ is low |
| | IWPF6 | When PMR5 bit WKP6 is changed from 0 to 1 while pin $\overline{WKP}_6$ is low |
| | IWPF5 | When PMR5 bit WKP5 is changed from 0 to 1 while pin $\overline{WKP}_5$ is low |
| | IWPF4 | When PMR5 bit WKP4 is changed from 0 to 1 while pin $\overline{WKP}_4$ is low |
| | IWPF3 | When PMR5 bit WKP3 is changed from 0 to 1 while pin $\overline{WKP}_3$ is low |
| | IWPF2 | When PMR5 bit WKP2 is changed from 0 to 1 while pin $\overline{WKP}_2$ is low |
| | IWPF1 | When PMR5 bit WKP1 is changed from 0 to 1 while pin $\overline{WKP}_1$ is low |
| | IWPF0 | When PMR5 bit WKP0 is changed from 0 to 1 while pin $\overline{WKP}_0$ is low |

Figure 3.8 shows the procedure for setting a bit in a port mode register and clearing the interrupt request flag.

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. If the instruction to clear the flag is executed immediately after the port mode register access without executing an intervening instruction, the flag will not be cleared.

An alternative method is to avoid the setting of interrupt request flags when pin functions are switched by keeping the pins at the high level so that the conditions in table 3.5 do not occur.

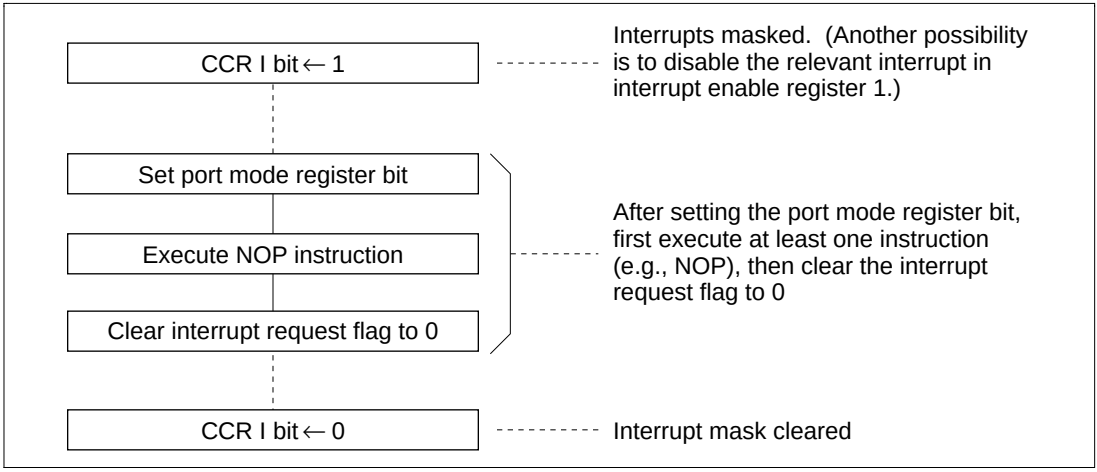



**Figure 3.8 Port Mode Register Setting and Interrupt Request Flag Clearing Procedure**

# Section 4 Clock Pulse Generators

## 4.1 Overview

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 and system clock dividers. The subclock pulse generator consists of a subclock oscillator circuit and a subclock divider.

### 4.1.1 Block Diagram

Figure 4.1 shows a block diagram of the clock pulse generators.

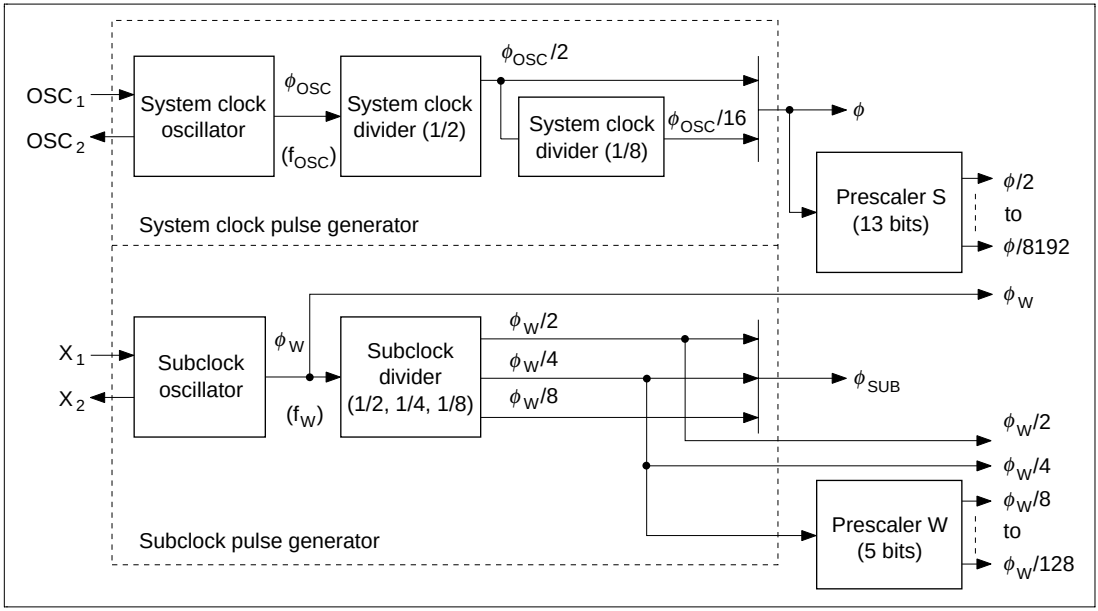



**Figure 4.1 Block Diagram of Clock Pulse Generators**

### 4.1.2 System Clock and Subclock

The basic clock signals that drive the CPU and on-chip peripheral modules are $\phi$ and $\phi_{SUB}$. Four of the clock signals have names: $\phi$ is the system clock, $\phi_{SUB}$ is the subclock, $\phi_{OSC}$ is the oscillator clock, and $\phi_W$ is the watch clock.

The clock signals available for use by peripheral modules are $\phi/2$, $\phi/4$, $\phi/8$, $\phi/16$, $\phi/32$, $\phi/64$, $\phi/128$, $\phi/256$, $\phi/512$, $\phi/1024$, $\phi/2048$, $\phi/4096$, $\phi/8192$, $\phi_W$, $\phi_W/2$, $\phi_W/4$, $\phi_W/8$, $\phi_W/16$, $\phi_W/32$, $\phi_W/64$, and $\phi_W/128$. The clock requirements differ from one module to another.

## 4.2 System Clock Generator

Clock pulse can be supplied to the system clock divider either by connecting a crystal or ceramic oscillator, or by providing external clock input.

**Connecting a Crystal Oscillator:** Figure 4.2 shows a typical method of connecting a crystal oscillator.

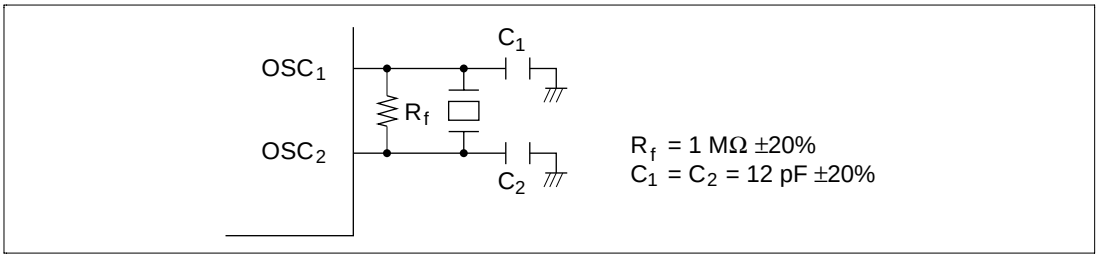



**Figure 4.2 Typical Connection to Crystal Oscillator**

Figure 4.3 shows the equivalent circuit of a crystal oscillator. An oscillator having the characteristics given in table 4.1 should be used.

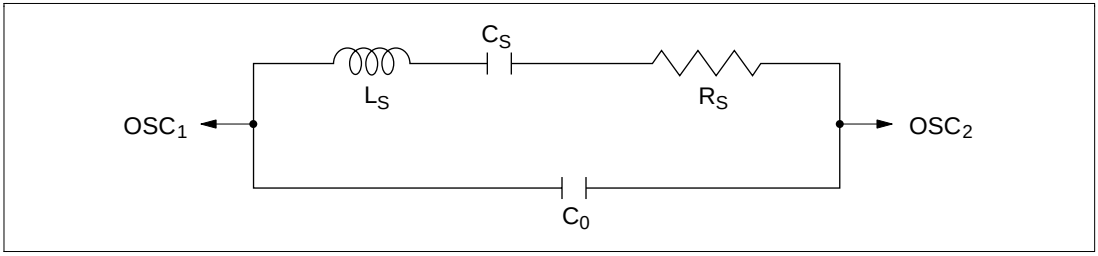



**Figure 4.3 Equivalent Circuit of Crystal Oscillator**

**Table 4.1 Crystal Oscillator Parameters**

| Frequency (MHz) | 2 | 4 | 8 | 10 |
|---|---|---|---|---|
| $R_s$ (max) | 500 Ω | 100 Ω | 50 Ω | 30 Ω |
| $C_o$ (max) | 7 pF | 7 pF | 7 pF | 7 pF |

**Connecting a Ceramic Oscillator:** Figure 4.4 shows a typical method of connecting a ceramic oscillator.

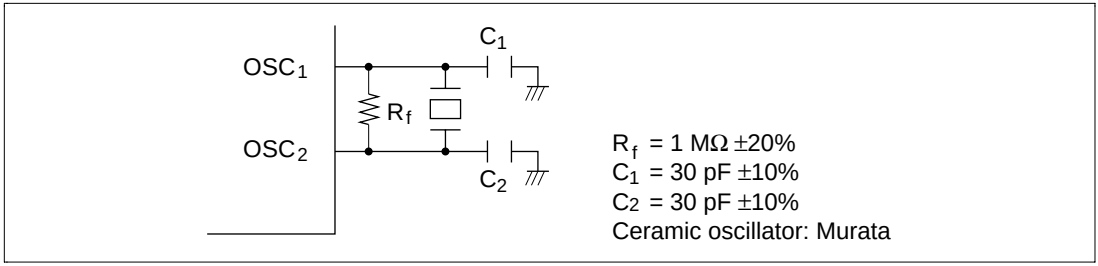



**Figure 4.4 Typical Connection to Ceramic Oscillator**

**Notes on Board Design:** When generating clock pulses by connecting a crystal or ceramic oscillator, pay careful attention to the following points.

Avoid running signal lines close to the oscillator circuit, since the oscillator may be adversely affected by induction currents. (See figure 4.5.)

The board should be designed so that the oscillator and load capacitors are located as close as possible to pins $OSC_1$ and $OSC_2$.

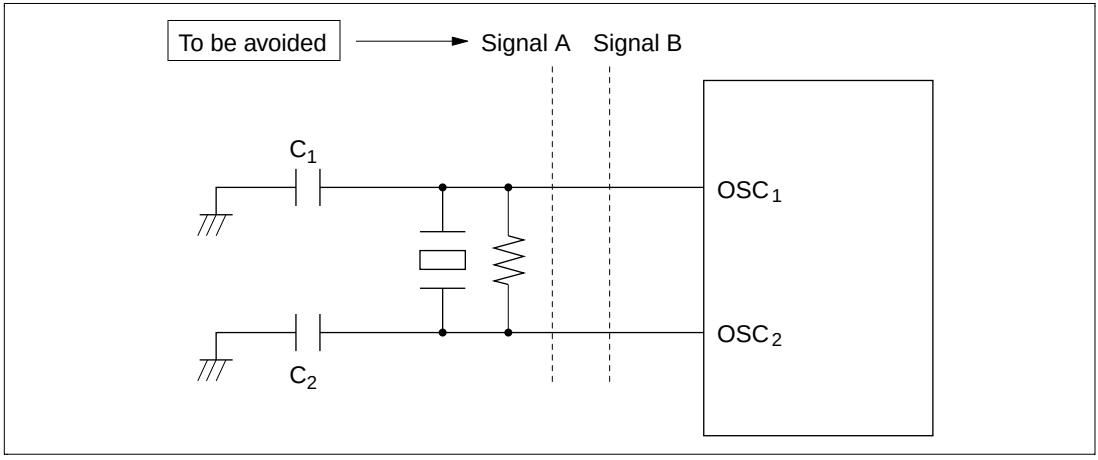



**Figure 4.5 Board Design of Oscillator Circuit**

**External Clock Input Method:** Connect an external clock signal to pin $OSC_1$, and leave pin $OSC_2$ open. Figure 4.6 shows a typical connection.

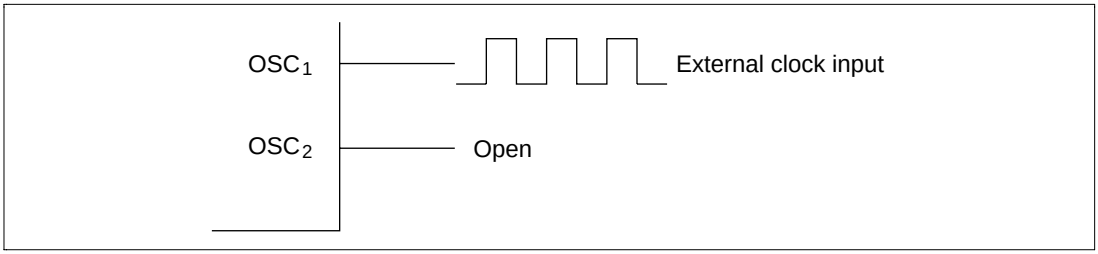



**Figure 4.6 External Clock Input (Example)**

| Frequency | Oscillator Clock ($\phi_{osc}$) |
|---|---|
| Duty cycle | 45% to 55% |

## 4.3 Subclock Generator

**Connecting a 32.768-kHz Crystal Oscillator:** Clock pulses can be supplied to the subclock divider by connecting a 32.768-kHz crystal oscillator, as shown in figure 4.7. Following the same connection precautions as mentioned in section 4.2.3, Notes on Board Design.

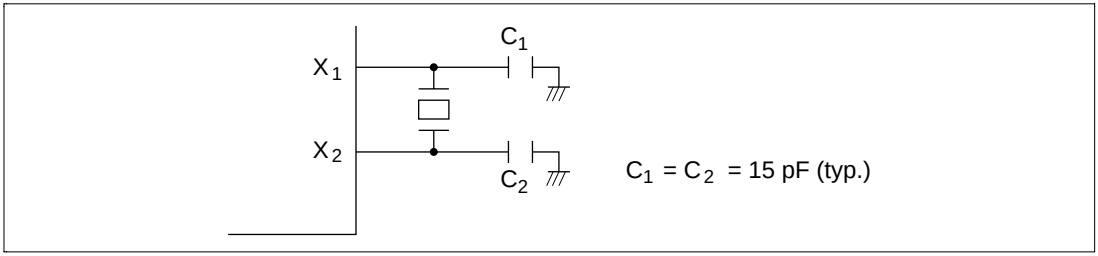



**Figure 4.7 Typical Connection to 32.768-kHz Crystal Oscillator**

Figure 4.8 shows the equivalent circuit of the 32.768-kHz crystal oscillator.

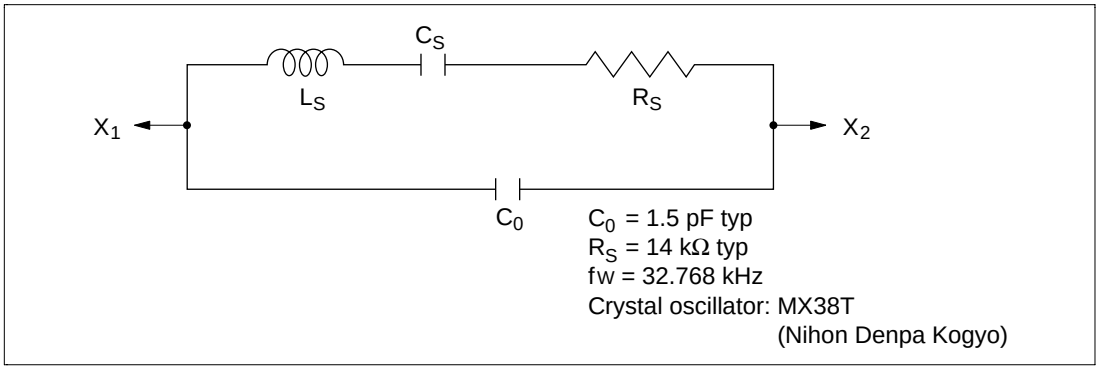



**Figure 4.8 Equivalent Circuit of 32.768-kHz Crystal Oscillator**

2. Inputting an external clock (H8/3832S, H8/3833S, H8/3834S, H8/3835S, H8/3836S, H8/3837S only)

   (1) Circuit configuration

   An external clock is input to the $X_1$ pin. The $X_2$ pin should be left open.

   An example of the connection in this case is shown in figure 4.9.

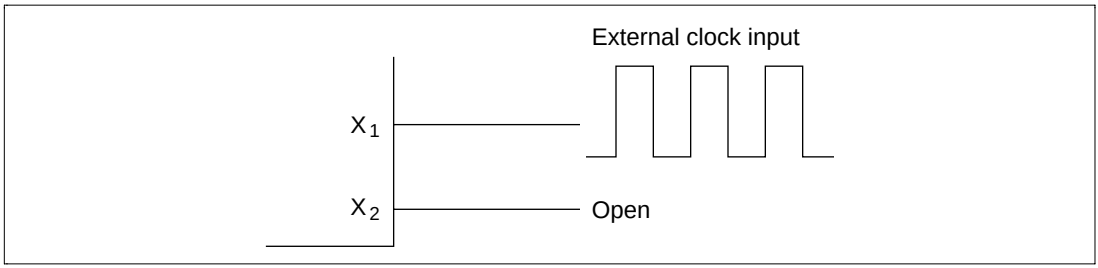



**Figure 4.9 Example of Connection when Inputting an External Clock**

(2) External clock

Input a square waveform to the $X_1$ pin. When using the CPU, timer A, timer C, timer G, or an LCD, with a subclock (øw) clock selected, do not stop the clock supply to the $X_1$ pin.

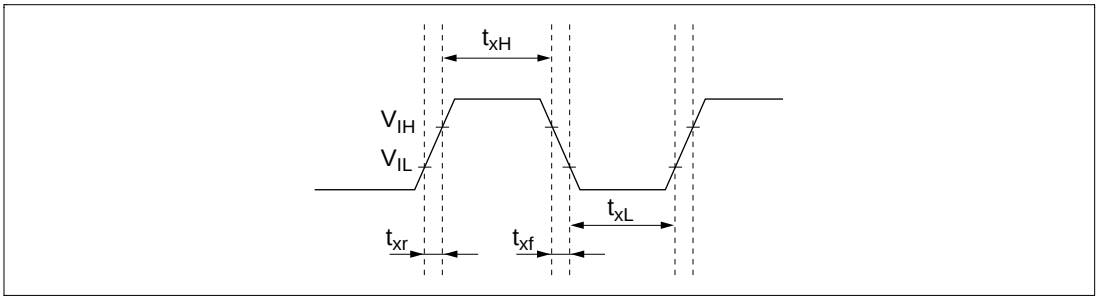



**Figure 4.10 External Subclock Timing**

The DC characteristics and timing of an external clock input to the $X_1$ pin are shown in table 4.2.

**Table 4.2 DC Characteristics and Timing**

**($V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to + 75°C, unless otherwise specified, including subactive mode)**

| Item | Symbol | Applicable Pin | Test Conditions | Values | | | Unit | Notes |
|---|---|---|---|---|---|---|---|---|
| | | | | Min | Typ | Max | | |
| Input high voltage | $V_{IH}$ | $X_1$ | | $V_{CC}$ –0.3 | — | $V_{CC}$ +0.3 | V | Figure 4.10 |
| Input low voltage | $V_{IL}$ | | | –0.3 | — | 0.3 | | |
| External subclock rise time | $t_{xr}$ | | | — | — | 100 | ns | Figure 4.10 |
| External subclock fall time | $t_{xf}$ | | | — | — | 100 | | |
| External subclock oscillation frequency | $f_x$ | | | — | 32.768 | — | kHz | |
| External subclock high width | $t_{xH}$ | | | 12.0 | — | — | µs | Figure 4.10 |
| External subclock low width | $t_{xL}$ | | | 12.0 | — | — | µs | |

**Pin Connection when Not Using Subclock:** When the subclock is not used, connect pin $X_1$ to $V_{CC}$ and leave pin $X_2$ open, as shown in figure 4.9.

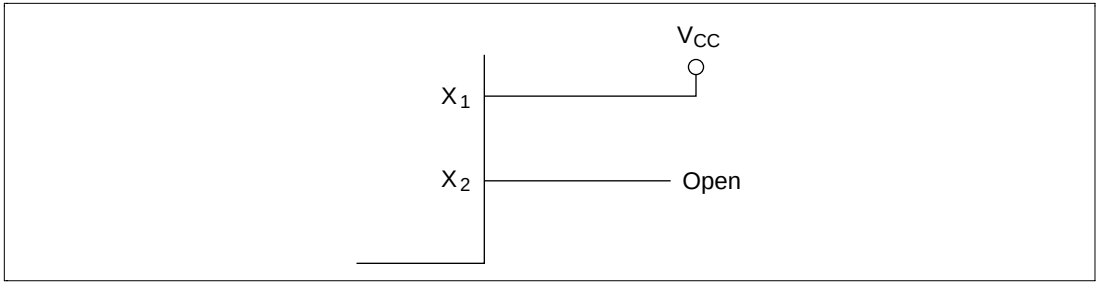



**Figure 4.9 Pin Connection when Not Using Subclock**

## 4.4 Prescalers

The H8/3834 Series is equipped with two on-chip prescalers having different input clocks (prescaler S and prescaler W). Prescaler S is a 13-bit counter using the system clock ($\phi$) as its input clock. Its prescaled outputs provide internal clock signals for on-chip peripheral modules. Prescaler W is a 5-bit counter using a 32.768-kHz signal divided by 4 ($\phi_W/4$) as its input clock. Its prescaled outputs are used by timer A as a time base for timekeeping.

**Prescaler S (PSS):** Prescaler S is a 13-bit counter using the system clock ($\phi$) 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, watch 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 timer A, timer B, timer C, timer F, timer G, SCI1, SCI2, SCI3, the A/D converter, LCD controller, and 14-bit PWM. The divider ratio can be set separately for each on-chip peripheral function.

In active (medium-speed) mode the clock input to prescaler S is $\phi_{OSC}/16$.

**Prescaler W (PSW):** Prescaler W is a 5-bit counter using a 32.768 kHz signal divided by 4 ($\phi_W/4$) as its input clock.

Prescaler W is initialized to H'00 by a reset, and starts counting on exit from the reset state.

Even in standby mode, watch mode, subactive mode, or subsleep mode, prescaler W continues functioning so long as clock signals are supplied to pins $X_1$ and $X_2$.

Prescaler W can be reset by setting 1s in bits TMA3 and TMA2 of timer mode register A (TMA).

Output from prescaler W can be used to drive timer A, in which case timer A functions as a time base for timekeeping.

## 4.5 Note on Oscillators

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

# Section 5 Power-Down Modes

## 5.1 Overview

The H8/3834 Series has seven modes of operation after a reset. These include six power-down modes, in which power dissipation is significantly reduced.

Table 5.1 gives a summary of the seven operation modes.

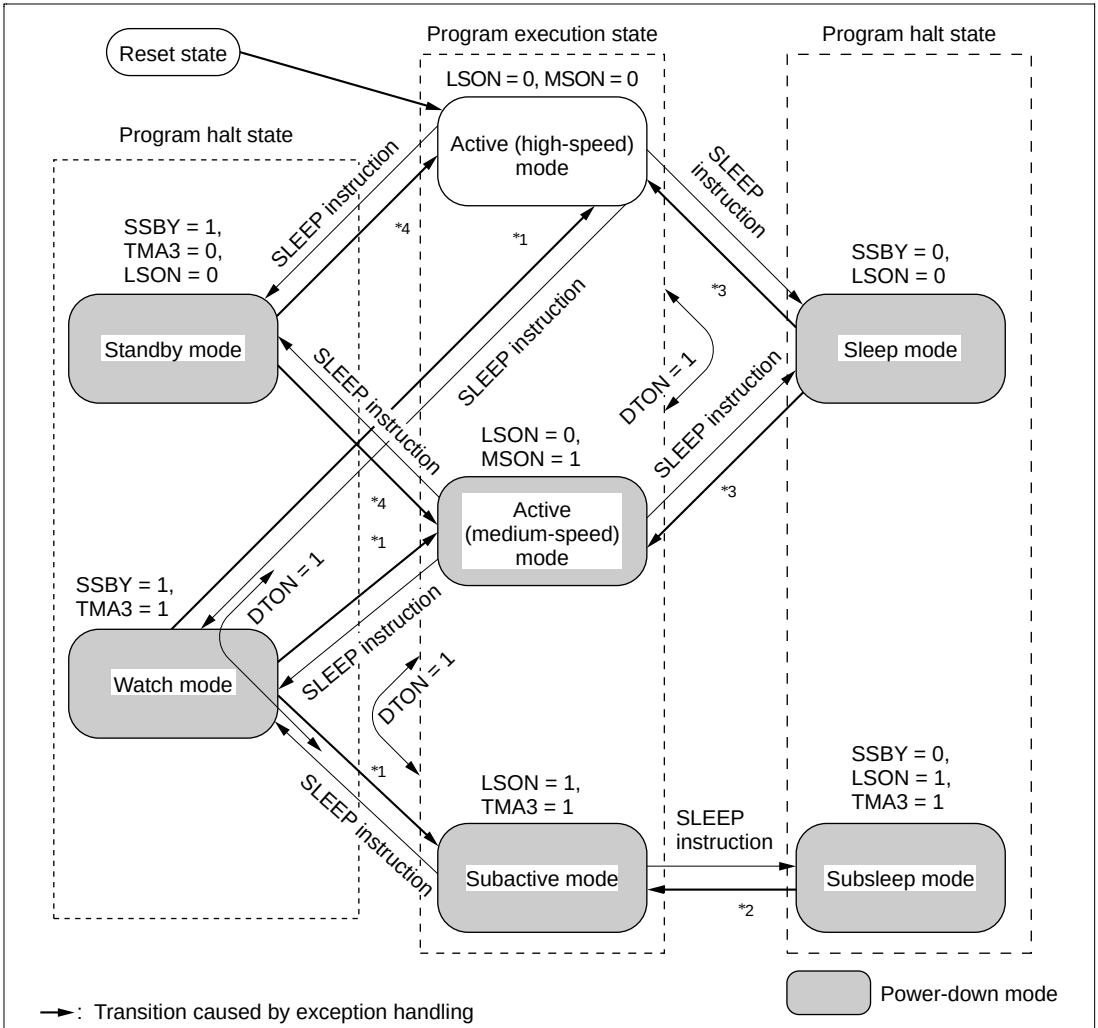
**Table 5.1 Operation Modes**

| Operating Mode | Description |
|---|---|
| Active (high-speed) mode | The CPU runs on the system clock, executing program instructions at high speed |
| Active (medium-speed) mode | The CPU runs on the system clock, executing program instructions at reduced speed |
| Subactive mode | The CPU runs on the subclock, executing program instructions at reduced speed |
| Sleep mode | The CPU halts. On-chip peripheral modules continue to operate on the system clock. |
| Subsleep mode | The CPU halts. Timer A, timer C, timer G, and the LCD controller/driver continue to operate on the subclock. |
| Watch mode | The CPU halts. The time-base function of timer A and the LCD controller/driver continue to operate on the subclock. |
| Standby mode | The CPU and all on-chip peripheral modules stop operating |

All the above operating modes except active (high-speed) mode are referred to as power-down modes.

In this section the two active modes (high-speed and medium-speed) are referred to collectively as active mode.

Figure 5.1 shows the transitions among these operation modes. Table 5.2 indicates the internal states in each mode.




A transition between different modes cannot be made to occur simply because an interrupt request is generated. Make sure that the interrupt is accepted and interrupt handling is performed.
Details on the mode transition conditions are given in the explanations of each mode, in sections 5.2 through 5.8.

Notes: 1. Timer A interrupt, $IRQ_0$ interrupt, $WKP_0$ to $WKP_7$ interrupts
2. Timer A interrupt, timer C interrupt, timer G interrupt, $IRQ_0$ to $IRQ_4$ interrupts, $WKP_0$ to $WKP_7$ interrupts
3. All interrupts
4. $IRQ_0$ interrupt, $IRQ_1$ interrupt, $WKP_0$ to $WKP_7$ interrupts

**Figure 5.1 Operation Mode Transition Diagram**

**Table 5.2 Internal State in Each Operation Mode**

| Function | | Active Mode: High Speed | Active Mode: Medium Speed | Sleep Mode | Watch Mode | Subactive Mode | Subsleep Mode | Standby Mode |
|---|---|---|---|---|---|---|---|---|
| System clock oscillator | | Functions | Functions | Functions | Halted | Halted | Halted | Halted |
| Subclock oscillator | | Functions | Functions | Functions | Functions | Functions | Functions | Functions |
| CPU operation | Instructions | Functions | Functions | Halted | Halted | Functions | Halted | Halted |
| | RAM | | | Retained | Retained | | Retained | Retained |
| | Registers | | | | | | | |
| | I/O | | | | | | | Retained*1 |
| External interrupts | $IRQ_0$ | Functions | Functions | Functions | Functions | Functions | Functions | Functions |
| | $IRQ_1$ | | | | Retained*6 | | | |
| | $IRQ_2$ | | | | | | | Retained*6 |
| | $IRQ_3$ | | | | | | | |
| | $IRQ_4$ | | | | | | | |
| | $WKP_0$ | Functions | Functions | Functions | Functions | Functions | Functions | Functions |
| | $WKP_1$ | | | | | | | |
| | $WKP_2$ | | | | | | | |
| | $WKP_3$ | | | | | | | |
| | $WKP_4$ | | | | | | | |
| | $WKP_5$ | | | | | | | |
| | $WKP_6$ | | | | | | | |
| | $WKP_7$ | | | | | | | |
| Peripheral module functions | Timer A | Functions | Functions | Functions | Functions*5 | Functions*5 | Functions*5 | Retained |
| | Timer B | | | | Retained | Retained | Retained | |
| | Timer C | | | | | Functions/ Retained*2 | Functions/ Retained*2 | |
| | Timer F | | | | | Retained | Retained | |
| | Timer G | | | | | Functions/ Retained*3 | Functions/ Retained*3 | |
| | SCI1 | Functions | Functions | Functions | Retained | Retained | Retained | Retained |
| | SCI2 | | | | | | | |
| | SCI3 | | | | Reset | Reset | Reset | Reset |
| | PWM | Functions | Functions | Retained | Retained | Retained | Retained | Retained |
| | A/D | Functions | Functions | Functions | Retained | Retained | Retained | Retained |
| | LCD | Functions | Functions | Functions | Functions/ Retained*4 | Functions/ Retained*4 | Functions/ Retained*4 | Retained |

Notes:
1. Register contents held; high-impedance output.
2. Functions only if external clock or $\phi_W/4$ internal clock is selected; otherwise halted and retained.
3. Functions only if $\phi_W/2$ internal clock is selected; otherwise halted and retained.
4. Functions only if $\phi_W$ or $\phi_W/2$ internal clock is selected; otherwise halted and retained.
5. Functions when timekeeping time-base function is selected.
6. External interrupt requests are ignored. The interrupt request register contents are not affected.

### 5.1.1 System Control Registers

The operation mode is selected using the system control registers described in table 5.3.

**Table 5.3 System Control Register**

| Name | Abbreviation | R/W | Initial Value | Address |
|---|---|---|---|---|
| System control register 1 | SYSCR1 | R/W | H'07 | H'FFF0 |
| System control register 2 | SYSCR2 | R/W | H'E0 | H'FFF1 |

**System Control Register 1 (SYSCR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SSBY | STS2 | STS1 | STS0 | LSON | — | — | — |
| Initial value | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | — | — | — |

SYSCR1 is an 8-bit read/write register for control of the power-down modes.

**Bit 7—Software Standby (SSBY):** This bit designates transition to standby mode or watch mode.

| Bit 7: SSBY | Description |
|---|---|
| 0 | • When a SLEEP instruction is executed in active mode, a transition is made to sleep mode (initial value)<br>• When a SLEEP instruction is executed in subactive mode, a transition is made to subsleep mode. |
| 1 | • When a SLEEP instruction is executed in active mode, a transition is made to standby mode or watch mode.<br>• When a SLEEP instruction is executed in subactive mode, a transition is made to watch mode. |

**Bits 6 to 4—Standby Timer Select 2 to 0 (STS2 to STS0):** These bits designate the time the CPU and peripheral modules wait for stable clock operation after exiting from standby mode or watch mode to active mode due to an interrupt. The designation should be made according to the clock frequency so that the waiting time is at least 10 ms.

| Bit 6: STS2 | Bit 5: STS1 | Bit 4: STS0 | Description | |
|---|---|---|---|---|
| 0 | 0 | 0 | Wait time = 8,192 states | (initial value) |
| | | 1 | Wait time = 16,384 states | |
| | 1 | 0 | Wait time = 32,768 states | |
| | | 1 | Wait time = 65,536 states | |
| 1 | * | * | Wait time = 131,072 states | |

Note: * Don't care

**Bit 3—Low Speed on Flag (LSON):** This bit chooses the system clock ($\phi$) or subclock ($\phi_{SUB}$) as the CPU operating clock when watch mode is cleared. The resulting operation mode depends on the combination of other control bits and interrupt input.

| Bit 3: LSON | Description | |
|---|---|---|
| 0 | The CPU operates on the system clock ($\phi$) | (initial value) |
| 1 | The CPU operates on the subclock ($\phi_{SUB}$) | |

**Bits 2 to 0—Reserved Bits:** These bits are reserved; they are always read as 1, and cannot be modified.

**System Control Register 2 (SYSCR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | NESEL | DTON | MSON | SA1 | SA0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

SYSCR2 is an 8-bit read/write register for power-down mode control.

**Bits 7 to 5—Reserved Bits:** These bits are reserved; they are always read as 1, and cannot be modified.

**Bit 4—Noise Elimination Sampling Frequency Select (NESEL):** This bit selects the frequency at which the watch clock signal ($\phi_W$) generated by the subclock pulse generator is sampled, in relation to the oscillator clock ($\phi_{OSC}$) generated by the system clock pulse generator. When $\phi_{OSC}$ = 2 to 10 MHz, clear NESEL to 0.

| Bit 4: NESEL | Description | |
|---|---|---|
| 0 | Sampling rate is $\phi_{OSC}$/16 | (initial value) |
| 1 | Sampling rate is $\phi_{OSC}$/4 | |

**Bit 3—Direct Transfer on Flag (DTON):** This bit designates whether or not to make direct transitions among active (high-speed), active (medium-speed) and subactive mode when a SLEEP instruction is executed. The mode to which the transition is made after the SLEEP instruction is executed depends on a combination of this and other control bits.

| Bit 3: DTON | Description |
|---|---|
| 0 | When a SLEEP instruction is executed in active mode, a transition is made to standby mode, watch mode, or sleep mode. (initial value)<br>When a SLEEP instruction is executed in subactive mode, a transition is made to watch mode or subsleep mode. |
| 1 | When a SLEEP instruction is executed in active (high-speed) mode, a direct transition is made to active (medium-speed) mode if SSBY = 0, MSON = 1, and LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = 1.<br>When a SLEEP instruction is executed in active (medium-speed) mode, a direct transition is made to active (high-speed) mode if SSBY = 0, MSON = 0, and LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = 1.<br>When a SLEEP instruction is executed in subactive mode, a direct transition is made to active (high-speed) mode if SSBY = 1, TMA3 = 1, LSON = 0, and MSON = 0, or to active (medium-speed) mode if SSBY = 1, TMA3 = 1, LSON = 0, and MSON = 1. |

**Bit 2—Medium Speed on Flag (MSON):** After standby, watch, or sleep mode is cleared, this bit selects active (high-speed) or active (medium-speed) mode.

| Bit 2: MSON | Description |
|---|---|
| 0 | Operation is in active (high-speed) mode (initial value) |
| 1 | Operation is in active (medium-speed) mode |

**Bits 1 and 0—Subactive Mode Clock Select (SA1 and SA0):** These bits select the CPU clock rate ($\phi_W/2$, $\phi_W/4$, or $\phi_W/8$) in subactive mode. SA1 and SA0 cannot be modified in subactive mode.

| Bit 1: SA1 | Bit 0: SA0 | Description |
|---|---|---|
| 0 | 0 | $\phi_W/8$ (initial value) |
| | 1 | $\phi_W/4$ |
| 1 | * | $\phi_W/2$ |

Note: * Don't care

## 5.2 Sleep Mode

### 5.2.1 Transition to Sleep Mode

The system goes from active mode to sleep mode when a SLEEP instruction is executed while the SSBY and LSON bits in system control register 1 (SYSCR1) are cleared to 0. In sleep mode CPU operation is halted but the on-chip peripheral functions other than PWM are operational. The CPU register contents are retained.

### 5.2.2 Clearing Sleep Mode

Sleep mode is cleared by an interrupt (timer A, timer B, timer C, timer F, timer G, $IRQ_0$ to $IRQ_4$, $WKP_0$ to $WKP_7$, SCI1, SCI2, SCI3, A/D converter) or by input at the RES pin.

**Clearing by Interrupt:** When an interrupt is requested, sleep mode is cleared and interrupt exception handling starts. Operation resumes in active (high-speed) mode if MSON = 0 in SYSCR2, or active (medium-speed) mode if MSON = 1. Sleep mode is not cleared if the I bit of the condition code register (CCR) is set to 1 or the particular interrupt is disabled in the interrupt enable register.

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

## 5.3 Standby Mode

### 5.3.1 Transition to Standby Mode

The system goes from active mode to standby mode when a SLEEP instruction is executed while the SSBY bit in SYSCR1 is set to 1, the LSON bit is cleared to 0, and bit TMA3 in timer register A (TMA) is cleared to 0. In standby mode the clock pulse generator stops, so the CPU and on-chip peripheral modules stop functioning. As long as a minimum required voltage is applied, the contents of CPU registers and some on-chip peripheral registers, and data in the on-chip RAM, are retained. Data in the on-chip RAM will be retained as long as the specified RAM data retention voltage is supplied. The I/O ports go to the high-impedance state.

### 5.3.2 Clearing Standby Mode

Standby mode is cleared by an interrupt ($IRQ_0$, $IRQ_1$, $WKP_0$ to $WKP_7$) or by input at the $\overline{RES}$ pin.

**Clearing by Interrupt:** When an interrupt is requested, the system clock pulse generator starts. After the time set in bits STS2–STS0 in SYSCR1 has elapsed, a stable system clock signal is supplied to the entire chip, standby mode is cleared, and interrupt exception handling starts. Operation resumes in active (high-speed) mode if MSON = 0 in SYSCR2, or active (medium-speed) mode if MSON = 1. Standby mode is not cleared if the I bit of CCR is set to 1 or the particular interrupt is disabled in the interrupt enable register.

**Clearing by $\overline{RES}$ Input:** When the $\overline{RES}$ pin goes low, the system clock pulse generator starts. After the pulse generator output has stabilized, if the $\overline{RES}$ pin is driven high, the CPU starts reset exception handling.

Since system clock signals are supplied to the entire chip as soon as the system clock pulse generator starts functioning, the $\overline{RES}$ pin should be kept at the low level until the pulse generator output stabilizes.

### 5.3.3 Oscillator Settling Time after Standby Mode is Cleared

Bits STS2 to STS0 in SYSCR1 should be set as follows.

- When a Crystal Oscillator is Used

  The table below gives settings for various operating frequencies. Set bits STS2 to STS0 for a waiting time of at least 10 ms.

**Table 5.3 Clock Frequency and Settling Time (Times are in ms)**

| STS2 | STS1 | STS0 | Waiting Time | 5 MHz | 4 MHz | 2 MHz | 1 MHz | 0.5 MHz |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 8,192 states | 1.6 | 2.0 | 4.1 | 8.2 | **16.4** |
| | | 1 | 16,384 states | 3.2 | 4.1 | 8.2 | **16.4** | 32.8 |
| | 1 | 0 | 32,768 states | 6.6 | 8.2 | **16.4** | 32.8 | 65.5 |
| | | 1 | 65,536 states | **13.1** | **16.4** | 32.8 | 65.5 | 131.1 |
| 1 | * | * | 131,072 states | 26.2 | 32.8 | 65.5 | 131.1 | 262.1 |

Note: * Don't care

- When an External Clock is Used

  Any values may be set. Normally the minimum time (STS2 = STS1 = STS0 = 0) should be set.

### 5.3.4 Transition to Standby Mode and Port Pin States

The system goes from active (high-speed or medium-speed) mode to standby mode when a SLEEP instruction is executed while the SSBY bit in SYSCR1 is set to 1, the LSON bit is cleared to 0, and bit TMA3 in TMA is cleared to 0. Port pins (except those with their MOS pull-up turned on) enter high-impedance state when the transition to standby mode is made. This timing is shown in figure 5.2.

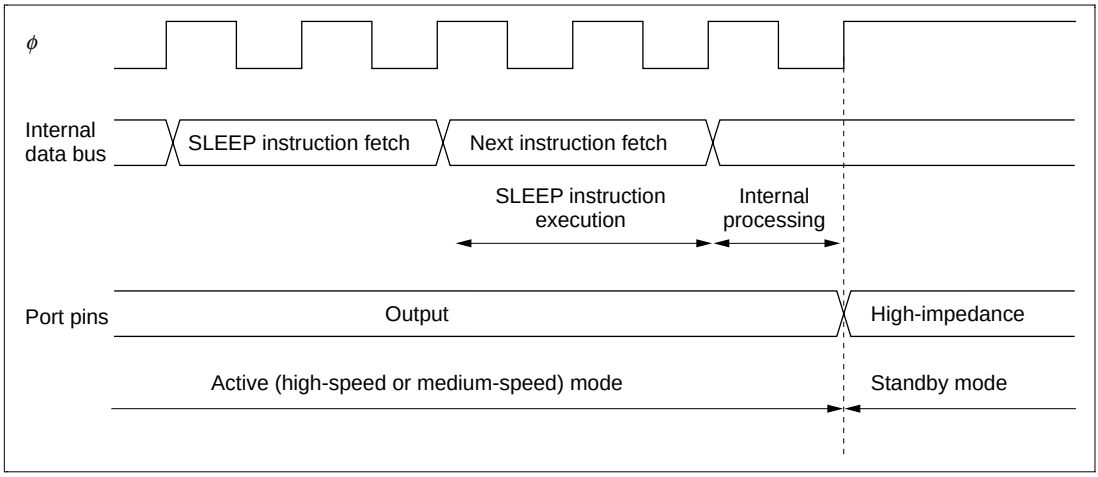



**Figure 5.2 Transition to Standby Mode and Port Pin States**

## 5.4 Watch Mode

### 5.4.1 Transition to Watch Mode

The system goes from active or subactive mode to watch mode when a SLEEP instruction is executed while the SSBY bit in SYSCR1 is set to 1 and bit TMA3 in TMA is set to 1.

In watch mode, operation of on-chip peripheral modules other than timer A and the LCD controller is halted. The LCD controller can be selected to operate or to halt. As long as a minimum required voltage is applied, the contents of CPU registers and some registers of the on-chip peripheral modules, and the on-chip RAM contents, are retained. I/O ports keep the same states as before the transition.

### 5.4.2 Clearing Watch Mode

Watch mode is cleared by an interrupt (timer A, $IRQ_0$, $WKP_0$ to $WKP_7$) or by a input at the $\overline{RES}$ pin.

**Clearing by Interrupt:** Watch mode is cleared when an interrupt is requested. The mode to which a transition is made depends on the settings of LSON in SYSCR1 and MSON in SYSCR2. If both LSON and MSON are cleared to 0, transition is to active (high-speed) mode; if LSON = 0 and MSON = 1, transition is to active (medium-speed) mode; if LSON = 1, transition is to subactive mode. When the transition is to active mode, after the time set in SYSCR1 bits STS2–STS0 has elapsed, a stable clock signal is supplied to the entire chip, and interrupt exception handling starts. Watch mode is not cleared if the I bit of CCR is set to 1 or the particular interrupt is disabled in the interrupt enable register.

**Clearing by $\overline{RES}$ Input:** Clearing by $\overline{RES}$ pin is the same as for standby mode; see 5.3.2, Clearing Standby Mode.

### 5.4.3 Oscillator Settling Time after Watch Mode is Cleared

The waiting time is the same as for standby mode; see 5.3.3, Oscillator Settling Time after Standby Mode is Cleared.

## 5.5 Subsleep Mode

### 5.5.1 Transition to Subsleep Mode

The system goes from subactive mode to subsleep mode when a SLEEP instruction is executed while the SSBY bit in SYSCR1 is cleared to 0, LSON bit in SYSCR1 is set to 1, and TMA3 bit in TMA is set to 1.

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

### 5.5.2 Clearing Subsleep Mode

Subsleep mode is cleared by an interrupt (timer A, timer C, timer G, $IRQ_0$ to $IRQ_4$, $WKP_0$ to $WKP_7$) or by a low input at the $\overline{RES}$ pin.

**Clearing by 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 particular interrupt is disabled in the interrupt enable register.

**Clearing by $\overline{RES}$ Input:** Clearing by $\overline{RES}$ pin is the same as for standby mode; see 5.3.2, Clearing Standby Mode.

## 5.6 Subactive Mode

### 5.6.1 Transition to Subactive Mode

Subactive mode is entered from watch mode if a timer A, $IRQ_0$, or $WKP_0$ to $WKP_7$ interrupt is requested while the LSON bit in SYSCR1 is set to 1. From subsleep mode, subactive mode is entered if a timer A, timer C, timer G, $IRQ_0$ to $IRQ_4$, or $WKP_0$ to $WKP_7$ interrupt is requested. A transition to subactive mode does not take place if the I bit of CCR is set to 1 or the particular interrupt is disabled in the interrupt enable register.

### 5.6.2 Clearing Subactive Mode

Subactive mode is cleared by a SLEEP instruction or by a input at the $\overline{RES}$ pin.

**Clearing by SLEEP Instruction:** If a SLEEP instruction is executed while the SSBY bit in SYSCR1 is set to 1 and TMA3 bit in TMA is set to 1, subactive mode is cleared and watch mode is entered. If a SLEEP instruction is executed while SSBY = 0 and LSON = 1 in SYSCR1 and TMA3 = 1 in TMA, subsleep mode is entered. Direct transfer to active mode is also possible; see 5.8, Direct Transfer, below.

**Clearing by $\overline{RES}$ Pin:** Clearing by $\overline{RES}$ pin is the same as for standby mode; see Clearing by $\overline{RES}$ pin in section 5.3.2, Clearing Standby Mode.

### 5.6.3 Operating Frequency in Subactive Mode

The operating frequency in subactive mode is set in bits SA1 and SA0 in SYSCR2. The choices are $\phi_W/2$, $\phi_W/4$, and $\phi_W/8$.

## 5.7 Active (medium-speed) Mode

### 5.7.1 Transition to Active (medium-speed) Mode

If the MSON bit in SYSCR2 is set to 1 while the LSON bit in SYSCR1 is cleared to 0, a transition to active (medium-speed) mode results from $IRQ_0$, $IRQ_1$, or $WKP_0$ to $WKP_7$ interrupts in standby mode, timer A, $IRQ_0$, or $WKP_0$ to $WKP_7$ interrupts in watch mode, or any interrupt in sleep mode. A transition to active (medium-speed) mode does not take place if the I bit of CCR is set to 1 or the particular interrupt is disabled in the interrupt enable register.

### 5.7.2 Clearing Active (medium-speed) Mode

Active (medium-speed) mode is cleared by a SLEEP instruction or by a input at the $\overline{\text{RES}}$ pin.

**Clearing by SLEEP Instruction:** A transition to standby mode takes place if a SLEEP instruction is executed while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, and TMA3 bit in TMA is cleared to 0. The system goes to watch mode if the SSBY bit in SYSCR1 is set to 1 and TMA3 bit in TMA is set to 1 when a SLEEP instruction is executed. Sleep mode is entered if both SSBY and LSON are cleared to 0 when a SLEEP instruction is executed. Direct transfer to active (high-speed) mode or to subactive mode is also possible. See 5.8, Direct Transfer, below for details.

**Clearing by $\overline{\text{RES}}$ Pin:** When the $\overline{\text{RES}}$ pin goes low, the CPU enters the reset state and active (medium-speed) mode is cleared.

### 5.7.3 Operating Frequency in Active (medium-speed) Mode

In active (medium-speed) mode, the CPU is clocked at 1/8 the frequency in active (high-speed) mode.

## 5.8 Direct Transfer

### 5.8.1 Direct Transfer Overview

The CPU can execute programs in three modes: active (high-speed) mode, active (medium-speed) mode, and subactive mode. A direct transfer is a transition among these three modes without the stopping of program execution. A direct transfer can be made by executing a SLEEP instruction while the DTON bit in SYSCR2 is set to 1. After the mode transition, direct transfer interrupt exception handling starts.

If the direct transfer interrupt is disabled in interrupt enable register 2 (IENR2), a transition is made instead to sleep mode or watch mode. Note that if a direct transition is attempted while the I bit in CCR is set to 1, sleep mode or watch mode will be entered, and it will be impossible to clear the resulting mode by means of an interrupt.

**Direct Transfer from Active (High-Speed) Mode to Active (Medium-Speed) Mode:** When a SLEEP instruction is executed in active (high-speed) mode while the SSBY and LSON bits in SYSCR1 are cleared to 0, the MSON bit in SYSCR2 is set to 1, and the DTON bit in SYSCR2 is set to 1, a transition is made to active (medium-speed) mode via sleep mode.

**Direct Transfer from Active (Medium-Speed) Mode to Active (High-Speed) Mode:** When a SLEEP instruction is executed in active (medium-speed) mode while the SSBY and LSON bits in SYSCR1 are cleared to 0, the MSON bit in SYSCR2 is cleared to 0, and the DTON bit in SYSCR2 is set to 1, a transition is made to active (high-speed) mode via sleep mode.

**Direct Transfer from Active (High-Speed) Mode to Subactive Mode:** When a SLEEP instruction is executed in active (high-speed) mode while the SSBY and LSON bits in SYSCR1 are set to 1, the DTON bit in SYSCR2 is set to 1, and TMA3 bit in TMA is set to 1, a transition is made to subactive mode via watch mode.

**Direct Transfer from Subactive Mode to Active (High-Speed) Mode:** When a SLEEP instruction is executed in subactive mode while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON bit in SYSCR2 is cleared to 0, the DTON bit in SYSCR2 is set to 1, and TMA3 bit in TMA is set to 1, a transition is made directly to active (high-speed) mode via watch mode after the waiting time set in SYSCR1 bits STS2 to STS0 has elapsed.

**Direct Transfer from Active (Medium-Speed) Mode to Subactive Mode:** When a SLEEP instruction is executed in active (medium-speed) while the SSBY and LSON bits in SYSCR1 are set to 1, the DTON bit in SYSCR2 is set to 1, and TMA3 bit in TMA is set to 1, a transition is made to subactive mode via watch mode.

**Direct Transfer from Subactive Mode to Active (Medium-Speed) Mode:** When a SLEEP instruction is executed in subactive mode while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON bit in SYSCR2 is set to 1, the DTON bit in SYSCR2 is set to 1, and TMA3 bit in TMA is set to 1, a transition is made directly to active (medium-speed) mode via watch mode after the waiting time set in SYSCR1 bits STS2 to STS0 has elapsed.

### 5.8.2 Calculation of Direct Transfer Time before Transition

**Time Required before Direct Transfer from Active (High-speed) Mode to Active (Medium-Speed) Mode:** A direct transfer is made from active (high-speed) mode to active (medium-speed) mode when a SLEEP instruction is executed in active (high-speed) mode while the SSBY and LSON bits in SYSCR1 are cleared to 0, the MSON bit in SYSCR2 is set to 1, and the DTON bit in SYSCR2 is set to 1. A direct transfer time, that is, the time from SLEEP instruction execution to interrupt exception handling completion is calculated by expression (1) below.

Direct transfer time = (number of states for SLEEP instruction execution + number of states for internal processing) × tcyc before transition + number of states for interrupt exception handling execution × tcyc after transition ..... (1)

Example: Direct transfer time for the H8/3834 Series
= (2 + 1) × 2tosc + 14 × 16tosc = 230 tosc

Notation:
tosc: OSC clock cycle time
tcyc: System clock ($\phi$) cycle time

**Time Required before Direct Transfer from Active (Medium-Speed) Mode to Active (High-Speed) Mode:** A direct transfer is made from active (medium-speed) mode to active (high-speed) mode when a SLEEP instruction is executed in active (medium-speed) mode while the SSBY and LSON bits in SYSCR1 are cleared to 0, the MSON bit in SYSCR2 is cleared to 0, and the DTON bit in SYSCR2 is set to 1. A direct transfer time, that is, the time from SLEEP instruction execution to interrupt exception handling completion is calculated by expression (2) below.

Direct transfer time = (number of states for SLEEP instruction execution + number of states for internal processing) × tcyc before transition + number of states for interrupt exception handling execution × tcyc after transition ..... (2)

Example: Direct transfer time for the H8/3834 Series
= (2 + 1) × 16tosc + 14 × 2tosc = 76 tosc

Notation:
tosc: OSC clock cycle time
tcyc: System clock ($\phi$) cycle time

**Time Required before Direct Transfer from Subactive Mode to Active (High-Speed) Mode:** A direct transfer is made from subactive mode to active (high-speed) mode when a SLEEP instruction is executed in subactive mode while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON bit in SYSCR2 is cleared to 0, the DTON bit in SYSCR2 is set to 1, and the TMA3 bit in TMA is set to 1. A direct transfer time, that is, the time from SLEEP

instruction execution to interrupt exception handling completion is calculated by expression (3) below.

Direct transfer time = (number of states for SLEEP instruction execution + number of states for internal processing) × tsubcyc before transition + (wait time designated by STS2 to STS0 bits in SCR + number of states for interrupt exception handling execution) × tcyc after transition ..... (3)

Example: Direct transfer time for the H8/3834 Series
(when CPU clock frequency is $\phi$w/8 and wait time is 8192 states)
= (2 + 1) × 8tw + (8192 + 14) × 2tosc = 24tw + 16412tosc

Notation:
tosc: OSC clock cycle time
tw: Watch clock cycle time
tcyc: System clock ($\phi$) cycle time
tsubcyc: Subclock ($\phi_{SUB}$) cycle time

**Time Required before Direct Transfer from Subactive Mode to Active (Medium-Speed) Mode:** A direct transfer is made from subactive mode to active (medium-speed) mode when a SLEEP instruction is executed in subactive mode while the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON and DTON bits in SYSCR2 are set to 1, and the TMA3 bit in TMA is set to 1. A direct transfer time, that is, the time from SLEEP instruction execution to interrupt exception handling completion is calculated by expression (4) below.

Direct transfer time = (number of states for SLEEP instruction execution + number of states for internal processing) × tsubcyc before transition (wait time designated by STS2 to STS0 bits in SCR + number of states for interrupt exception handling execution) × tcyc after transition ..... (4)

Example: Direct transfer time for the H8/3834 Series
(when CPU clock frequency is $\phi$w/8 and wait time is 8192 states)
= (2 + 1) × 8tw + (8192 + 14) × 16tosc = 24tw + 131296tosc

Notation:
tosc: OSC clock cycle time
tw: Watch clock cycle time
tcyc: System clock ($\phi$) cycle time
tsubcyc: Subclock ($\phi_{SUB}$) cycle time

# Section 6 ROM

## 6.1 Overview

The H8/3832 has 16 kbytes of on-chip ROM, while the H8/3833 has 24 kbytes, the H8/3834 has 32 kbytes, the H8/3835 has 40 kbytes, the H8/3836 has 48 kbytes, and the H8/3837 has 60 kbytes. The ROM is connected to the CPU by a 16-bit data bus, allowing high-speed 2-state access for both byte data and word data. The ZTAT™ versions of the H8/3834 and H8/3837 each have 32 kbytes and 60 kbytes of PROM.

With regard to ZTAT versions of the H8/3832S, H8/3833S, H8/3834S, H8/3835S, H8/3836S, and H8/3837S, the ZTAT versions of the H8/3834 and H8/3837 should be used.

### 6.1.1 Block Diagram

Figure 6.1 shows a block diagram of the on-chip ROM.

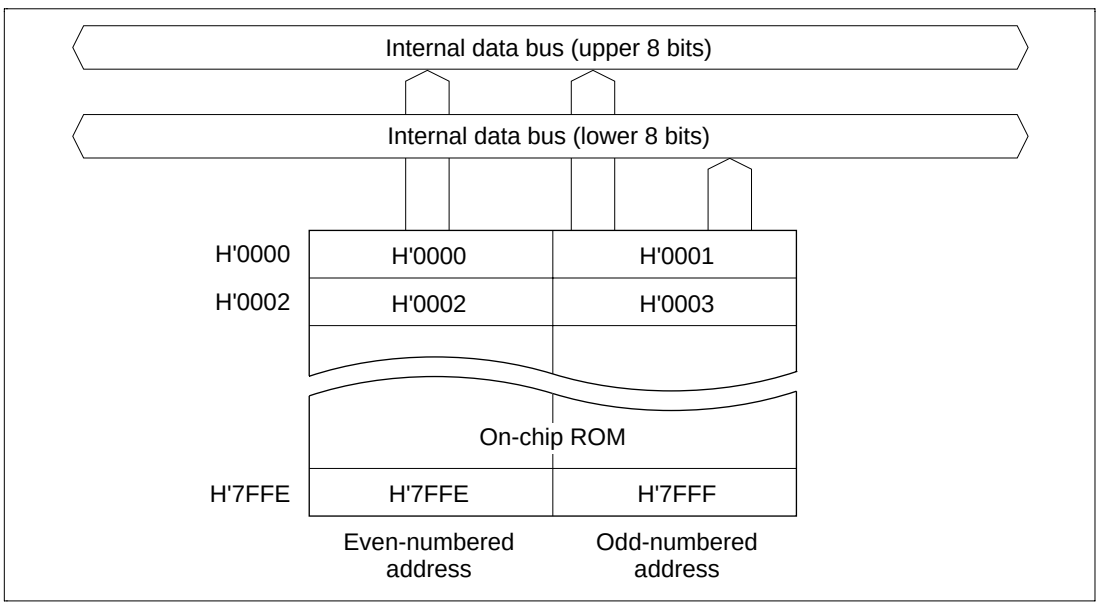



**Figure 6.1 ROM Block Diagram (H8/3834)**

## 6.2 H8/3834 PROM Mode

### 6.2.1 Setting to PROM Mode

If the on-chip ROM is PROM, setting the chip to PROM mode stops operation as a microcontroller and allows the PROM to be programmed in the same way as the standard HN27C256. Table 6.1 shows how to set the chip to PROM mode.

**Table 6.1 Setting to PROM Mode**

| Pin Name | Setting |
|---|---|
| TEST | High level |
| $PB_4/AN_4$ | Low level |
| $PB_5/AN_5$ | |
| $PB_6/AN_6$ | High level |

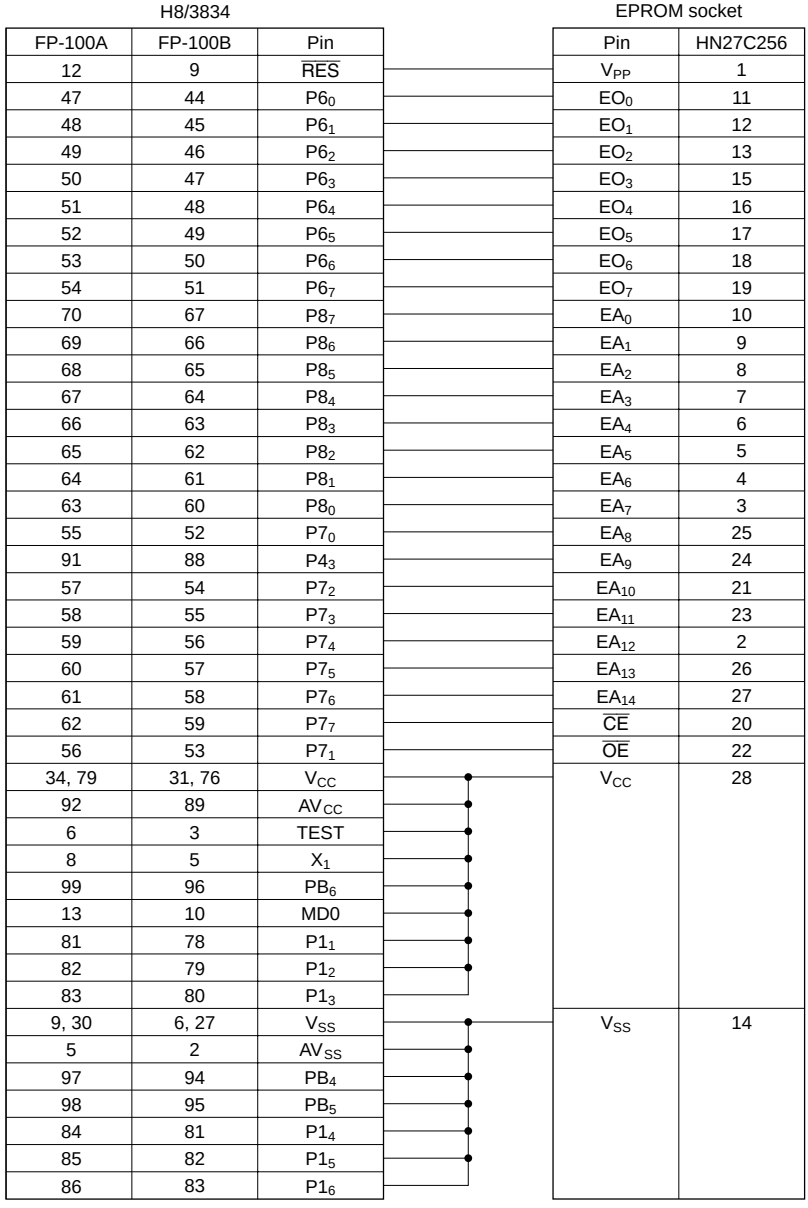
### 6.2.2 Socket Adapter Pin Arrangement and Memory Map

A standard PROM programmer can be used to program the PROM. A socket adapter is required for conversion to 28 pins, as listed in table 6.2.

Figure 6.2 shows the pin-to-pin wiring of the socket adapter. Figure 6.3 shows a memory map.

**Table 6.2 Socket Adapter**

| Package | Socket Adapter |
|---|---|
| 100-pin (FP-100B) | HS3834ESH01H |
| 100-pin (FP-100A) | HS3834ESF01H |
| 100-pin (TFP-100B) | HS3834ESN01H |




| FP-100A | FP-100B | Pin |
|---|---|---|
| 12 | 9 | $\overline{RES}$ |
| 47 | 44 | $P6_0$ |
| 48 | 45 | $P6_1$ |
| 49 | 46 | $P6_2$ |
| 50 | 47 | $P6_3$ |
| 51 | 48 | $P6_4$ |
| 52 | 49 | $P6_5$ |
| 53 | 50 | $P6_6$ |
| 54 | 51 | $P6_7$ |
| 70 | 67 | $P8_7$ |
| 69 | 66 | $P8_6$ |
| 68 | 65 | $P8_5$ |
| 67 | 64 | $P8_4$ |
| 66 | 63 | $P8_3$ |
| 65 | 62 | $P8_2$ |
| 64 | 61 | $P8_1$ |
| 63 | 60 | $P8_0$ |
| 55 | 52 | $P7_0$ |
| 91 | 88 | $P4_3$ |
| 57 | 54 | $P7_2$ |
| 58 | 55 | $P7_3$ |
| 59 | 56 | $P7_4$ |
| 60 | 57 | $P7_5$ |
| 61 | 58 | $P7_6$ |
| 62 | 59 | $P7_7$ |
| 56 | 53 | $P7_1$ |
| 34, 79 | 31, 76 | $V_{CC}$ |
| 92 | 89 | $AV_{CC}$ |
| 6 | 3 | TEST |
| 8 | 5 | $X_1$ |
| 99 | 96 | $PB_6$ |
| 13 | 10 | MD0 |
| 81 | 78 | $P1_1$ |
| 82 | 79 | $P1_2$ |
| 83 | 80 | $P1_3$ |
| 9, 30 | 6, 27 | $V_{SS}$ |
| 5 | 2 | $AV_{SS}$ |
| 97 | 94 | $PB_4$ |
| 98 | 95 | $PB_5$ |
| 84 | 81 | $P1_4$ |
| 85 | 82 | $P1_5$ |
| 86 | 83 | $P1_6$ |

| Pin | HN27C256 |
|---|---|
| $V_{PP}$ | 1 |
| $EO_0$ | 11 |
| $EO_1$ | 12 |
| $EO_2$ | 13 |
| $EO_3$ | 15 |
| $EO_4$ | 16 |
| $EO_5$ | 17 |
| $EO_6$ | 18 |
| $EO_7$ | 19 |
| $EA_0$ | 10 |
| $EA_1$ | 9 |
| $EA_2$ | 8 |
| $EA_3$ | 7 |
| $EA_4$ | 6 |
| $EA_5$ | 5 |
| $EA_6$ | 4 |
| $EA_7$ | 3 |
| $EA_8$ | 25 |
| $EA_9$ | 24 |
| $EA_{10}$ | 21 |
| $EA_{11}$ | 23 |
| $EA_{12}$ | 2 |
| $EA_{13}$ | 26 |
| $EA_{14}$ | 27 |
| $\overline{CE}$ | 20 |
| $\overline{OE}$ | 22 |
| $V_{CC}$ | 28 |
| $V_{SS}$ | 14 |

Note: Pins not indicated in the figure should be left open.

**Figure 6.2 Socket Adapter Pin Correspondence (with HN27C256)**

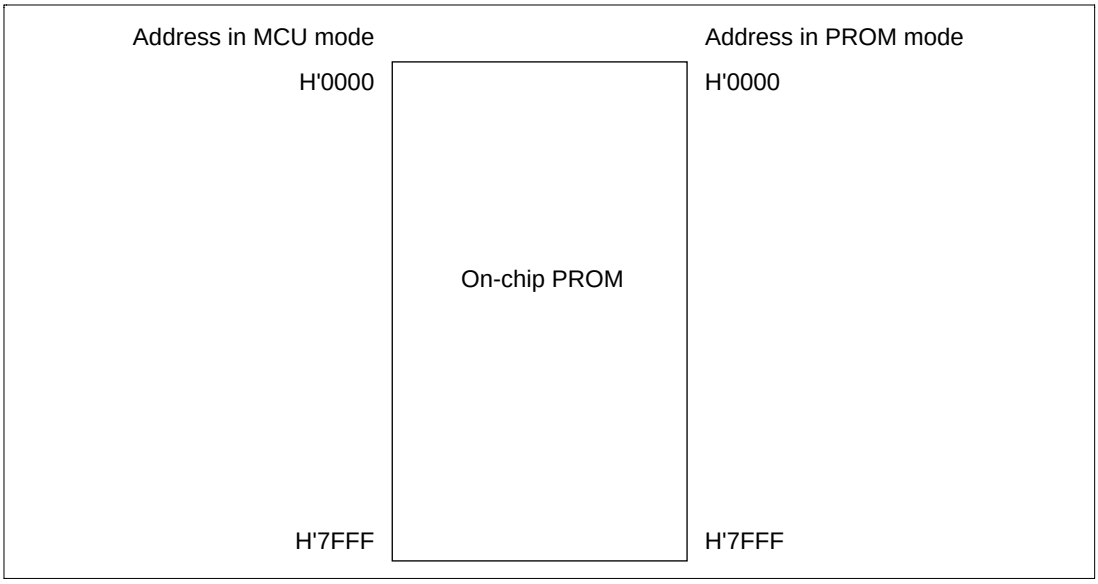



**Figure 6.3 H8/3834 Memory Map in PROM Mode**

Note: When programming with a PROM programmer, be sure to specify addresses from H'0000 to H'7FFF.

## 6.3 H8/3834 Programming

The write, verify, and other modes are selected as shown in table 6.3 in H8/3834 PROM mode.

**Table 6.3 Mode Selection in H8/3834 PROM Mode**

| | Pin | | | | | |
|---|---|---|---|---|---|---|
| **Mode** | $\overline{CE}$ | $\overline{OE}$ | $V_{PP}$ | $V_{CC}$ | $EO_7$ to $EO_0$ | $EA_{14}$ to $EA_0$ |
| Write | L | H | $V_{PP}$ | $V_{CC}$ | Data input | Address input |
| Verify | H | L | $V_{PP}$ | $V_{CC}$ | Data output | Address input |
| Programming disabled | H | H | $V_{PP}$ | $V_{CC}$ | High impedance | Address input |

Notation:
L: Low level
H: High level
$V_{PP}$: $V_{PP}$ level
$V_{CC}$: $V_{CC}$ level

The specifications for writing and reading the on-chip PROM are identical to those for the standard HN27C256 EPROM.

### 6.3.1 Writing and Verifying

An efficient, high-performance programming method is available for writing and verifying the PROM data. This method achieves high speed without voltage stress on the device and without lowering the reliability of written data. H'FF data is written in unused address areas.

The basic flow of this high-performance programming method is shown in figure 6.4. Table 6.4 and table 6.5 give the electrical characteristics in programming mode. Figure 6.5 shows a write/verify timing diagram.

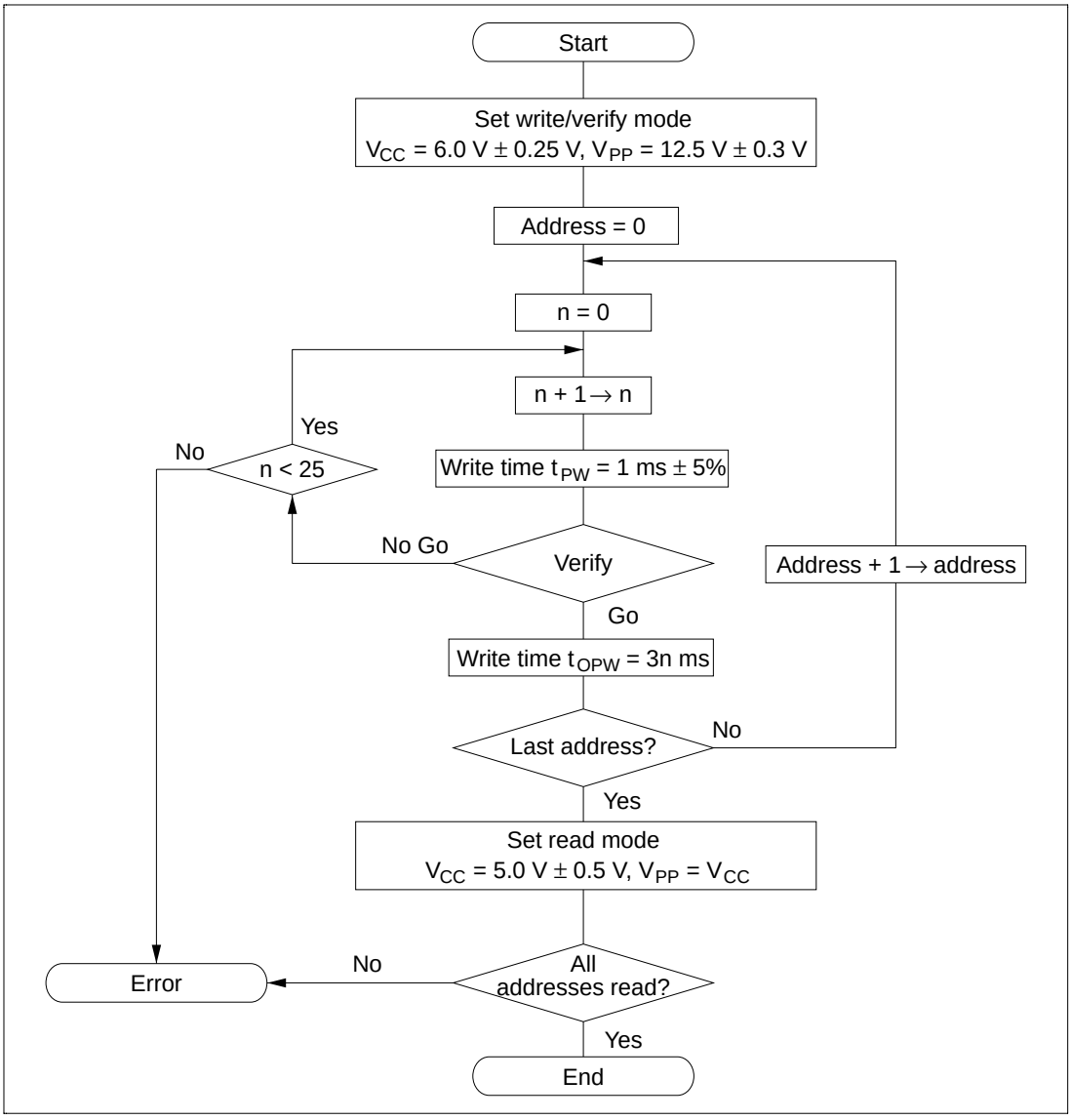



**Figure 6.4 High-Performance Programming Flowchart**

**Table 6.4 DC Characteristics**

(Conditions: $V_{CC}$ = 6.0 V ±0.25 V, $V_{PP}$ = 12.5 V ±0.3 V, $V_{SS}$ = 0 V, $T_a$ = 25°C ±5°C)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|---|
| Input high-level voltage | $EO_7$ to $EO_0$, $EA_{14}$ to $EA_0$, $\overline{OE}$, $\overline{CE}$ | $V_{IH}$ | 2.4 | — | $V_{CC}$ + 0.3 | V | |
| Input low-level voltage | $EO_7$ to $EO_0$, $EA_{14}$ to $EA_0$, $\overline{OE}$, $\overline{CE}$ | $V_{IL}$ | –0.3 | — | 0.8 | V | |
| Output high-level voltage | $EO_7$ to $EO_0$ | $V_{OH}$ | 2.4 | — | — | V | $I_{OH}$ = –200 µA |
| Output low-level voltage | $EO_7$ to $EO_0$ | $V_{OL}$ | — | — | 0.45 | V | $I_{OL}$ = 0.8 mA |
| Input leakage current | $EO_7$ to $EO_0$, $EA_{14}$ to $EA_0$, $\overline{OE}$, $\overline{CE}$ | $\|I_{LI}\|$ | — | — | 2 | µA | $V_{IN}$ = 5.25 V/ 0.5 V |
| $V_{CC}$ current | | $I_{CC}$ | — | — | 40 | mA | |
| $V_{PP}$ current | | $I_{PP}$ | — | — | 40 | mA | |

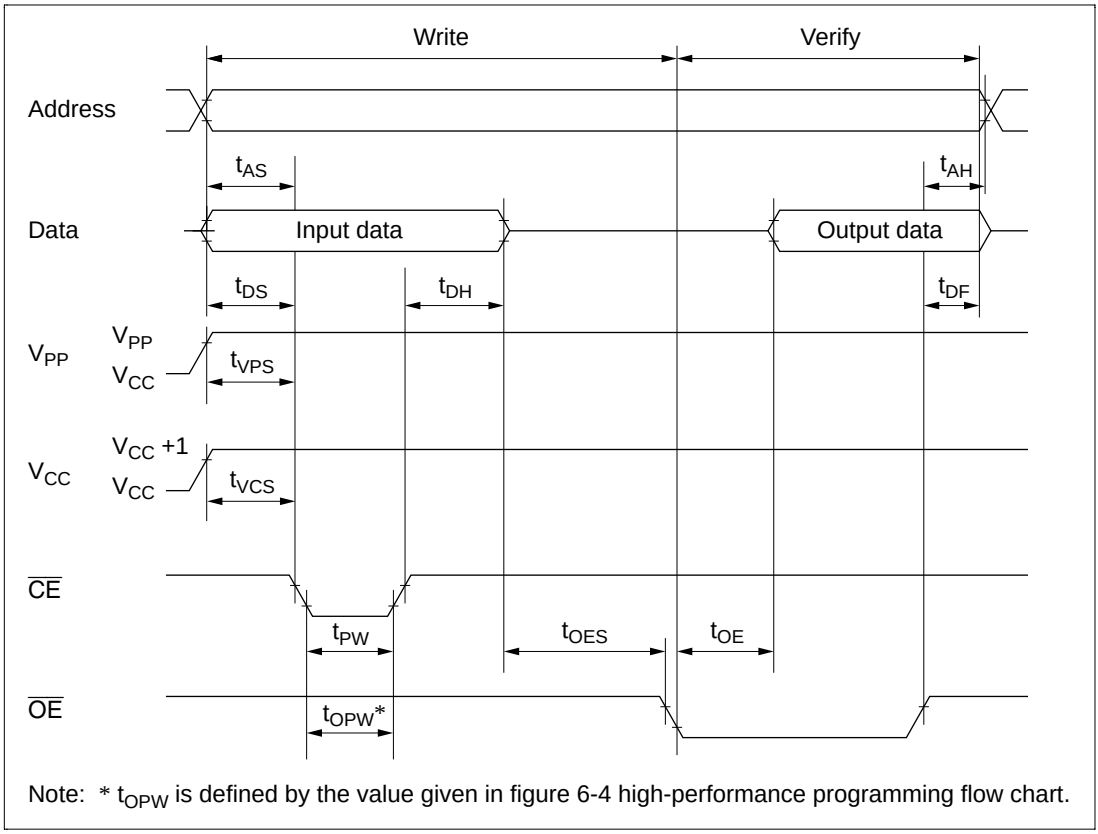
**Table 6.5 AC Characteristics**

(Conditions: $V_{CC}$ = 6.0 V ±0.25 V, $V_{PP}$ = 12.5 V ±0.3 V, $T_a$ = 25°C ±5°C)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| Address setup time | $t_{AS}$ | 2 | — | — | µs | Figure 6.5*[1] |
| $\overline{OE}$ setup time | $t_{OES}$ | 2 | — | — | µs | |
| Data setup time | $t_{DS}$ | 2 | — | — | µs | |
| Address hold time | $t_{AH}$ | 0 | — | — | µs | |
| Data hold time | $t_{DH}$ | 2 | — | — | µs | |
| Data output disable time | $t_{DF}$*[2] | 0 | — | 130 | ns | |
| $V_{PP}$ setup time | $t_{VPS}$ | 2 | — | — | µs | |
| Programming pulse width | $t_{PW}$ | 0.95 | 1.0 | 1.05 | ms | |
| $\overline{CE}$ pulse width for overwrite programming | $t_{OPW}$*[3] | 2.85 | — | 78.7 | ms | |
| $V_{CC}$ setup time | $t_{VCS}$ | 2 | — | — | µs | |
| Data output delay time | $t_{OE}$ | 0 | — | 500 | ns | |

Notes: 1. Input pulse level: 0.8 V to 2.2 V
Input rise time/fall time ≤ 20 ns
Timing reference levels: Input: 1.0 V, 2.0 V
Output: 0.8 V, 2.0 V

2. $t_{DF}$ is defined at the point at which the output is floating and the output level cannot be read.
3. $t_{OPW}$ is defined by the value given in figure 6.4 high-performance programming flow chart.




Note: * $t_{OPW}$ is defined by the value given in figure 6-4 high-performance programming flow chart.

**Figure 6.5 PROM Write/Verify Timing**

### 6.3.2 Programming Precautions

- Use the specified programming voltage and timing.

  The programming voltage in PROM mode ($V_{pp}$) is 12.5 V. Use of a higher voltage can permanently damage the chip. Be especially careful with respect to PROM programmer overshoot.

  Setting the PROM programmer to Hitachi specifications for the HN27C256 will result in a correct $V_{pp}$ of 12.5 V.

- Make sure the index marks on the PROM programmer socket, socket adapter, and chip are properly aligned. If they are not, the chip may be destroyed by excessive current flow. Before programming, be sure that the chip is properly mounted in the PROM programmer.

- Avoid touching the socket adapter or chip during programming, since this may cause contact faults and write errors.

## 6.4 H8/3837 PROM Mode

### 6.4.1 Setting to PROM Mode

If the on-chip ROM is PROM, setting the chip to PROM mode stops operation as a microcontroller and allows the PROM to be programmed in the same way as the standard HN27C101 EPROM. However, page programming is not supported. Table 6.6 shows how to set the chip to PROM mode.

**Table 6.6 Setting to PROM Mode**

| Pin Name | Setting |
|---|---|
| TEST | High level |
| $PB_4/AN_4$ | Low level |
| $PB_5/AN_5$ | |
| $PB_6/AN_6$ | High level |

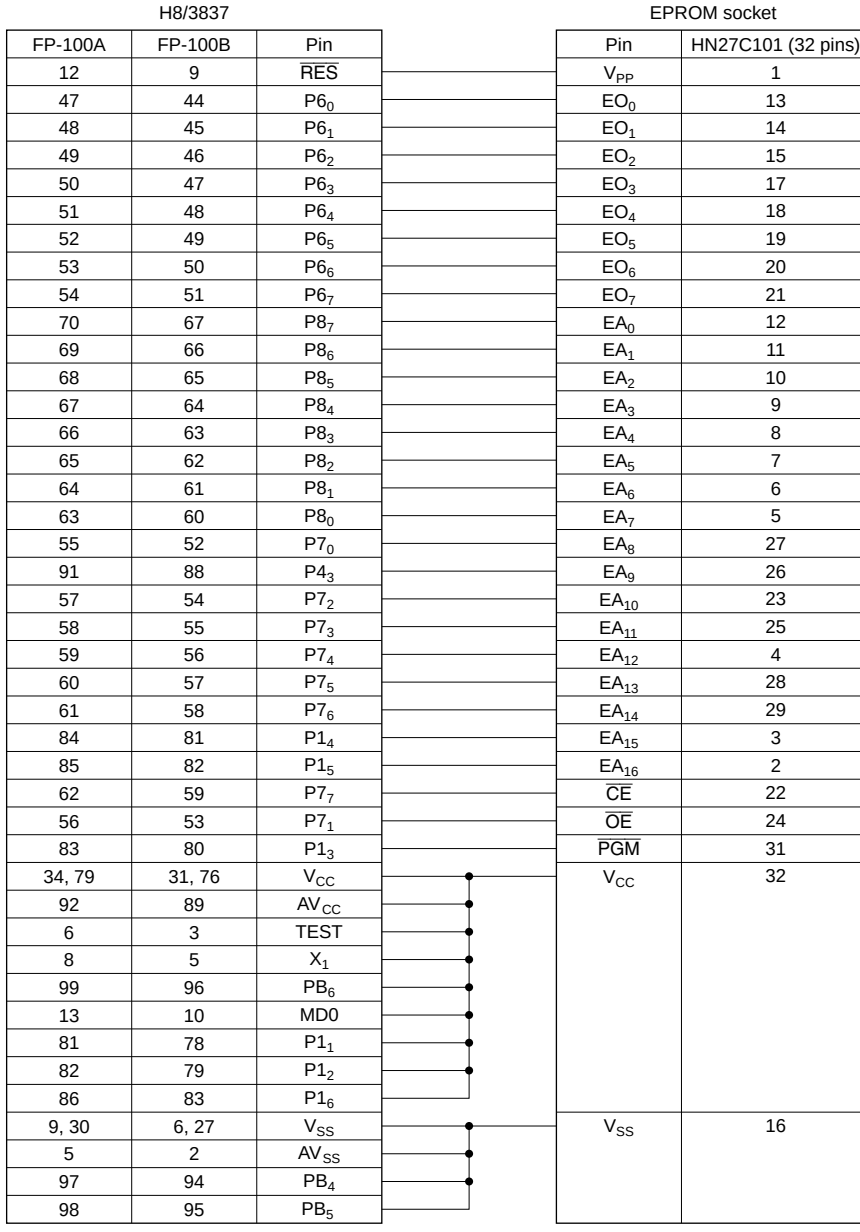
### 6.4.2 Socket Adapter Pin Arrangement and Memory Map

A standard PROM programmer can be used to program the PROM. A socket adapter is required for conversion to 32 pins, as listed in table 6.7.

Figure 6.6 shows the pin-to-pin wiring of the socket adapter. Figure 6.7 shows a memory map.

**Table 6.7 Socket Adapter**

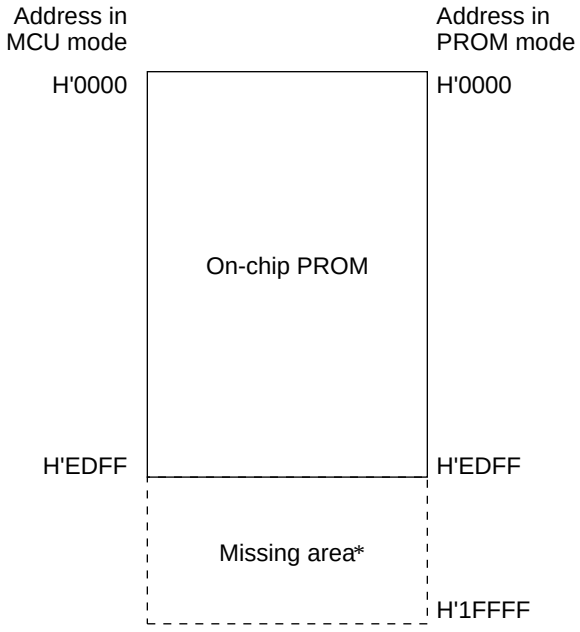
| Package | Socket Adapter |
|---|---|
| 100-pin (FP-100B) | HS3836ESH01H |
| 100-pin (FP-100A) | HS3836ESF01H |
| 100-pin (TFP-100B) | HS3836ESN01H |




| FP-100A | FP-100B | Pin | Pin | HN27C101 (32 pins) |
|---|---|---|---|---|
| 12 | 9 | $\overline{RES}$ | $V_{PP}$ | 1 |
| 47 | 44 | $P6_0$ | $EO_0$ | 13 |
| 48 | 45 | $P6_1$ | $EO_1$ | 14 |
| 49 | 46 | $P6_2$ | $EO_2$ | 15 |
| 50 | 47 | $P6_3$ | $EO_3$ | 17 |
| 51 | 48 | $P6_4$ | $EO_4$ | 18 |
| 52 | 49 | $P6_5$ | $EO_5$ | 19 |
| 53 | 50 | $P6_6$ | $EO_6$ | 20 |
| 54 | 51 | $P6_7$ | $EO_7$ | 21 |
| 70 | 67 | $P8_7$ | $EA_0$ | 12 |
| 69 | 66 | $P8_6$ | $EA_1$ | 11 |
| 68 | 65 | $P8_5$ | $EA_2$ | 10 |
| 67 | 64 | $P8_4$ | $EA_3$ | 9 |
| 66 | 63 | $P8_3$ | $EA_4$ | 8 |
| 65 | 62 | $P8_2$ | $EA_5$ | 7 |
| 64 | 61 | $P8_1$ | $EA_6$ | 6 |
| 63 | 60 | $P8_0$ | $EA_7$ | 5 |
| 55 | 52 | $P7_0$ | $EA_8$ | 27 |
| 91 | 88 | $P4_3$ | $EA_9$ | 26 |
| 57 | 54 | $P7_2$ | $EA_{10}$ | 23 |
| 58 | 55 | $P7_3$ | $EA_{11}$ | 25 |
| 59 | 56 | $P7_4$ | $EA_{12}$ | 4 |
| 60 | 57 | $P7_5$ | $EA_{13}$ | 28 |
| 61 | 58 | $P7_6$ | $EA_{14}$ | 29 |
| 84 | 81 | $P1_4$ | $EA_{15}$ | 3 |
| 85 | 82 | $P1_5$ | $EA_{16}$ | 2 |
| 62 | 59 | $P7_7$ | $\overline{CE}$ | 22 |
| 56 | 53 | $P7_1$ | $\overline{OE}$ | 24 |
| 83 | 80 | $P1_3$ | $\overline{PGM}$ | 31 |
| 34, 79 | 31, 76 | $V_{CC}$ | $V_{CC}$ | 32 |
| 92 | 89 | $AV_{CC}$ | | |
| 6 | 3 | TEST | | |
| 8 | 5 | $X_1$ | | |
| 99 | 96 | $PB_6$ | | |
| 13 | 10 | MD0 | | |
| 81 | 78 | $P1_1$ | | |
| 82 | 79 | $P1_2$ | | |
| 86 | 83 | $P1_6$ | | |
| 9, 30 | 6, 27 | $V_{SS}$ | $V_{SS}$ | 16 |
| 5 | 2 | $AV_{SS}$ | | |
| 97 | 94 | $PB_4$ | | |
| 98 | 95 | $PB_5$ | | |

Note: Pins not indicated in the figure should be left open.

**Figure 6.6 Socket Adapter Pin Correspondence (with HN27C101)**




Note: * If read in PROM mode, this address area returns unpredictable output data. When programming with a PROM programmer, be sure to specify addresses from H'0000 to H'EDFF.
If address H'EE00 and higher addresses are programmed by mistake, it may become impossible to program the PROM or verify the programmed data. When programming, assign H'FF data to this address area (H'EE00 to H'1FFFF).

**Figure 6.7 H8/3837 Memory Map in PROM Mode**

## 6.5 H8/3837 Programming

The write, verify, and other modes are selected as shown in table 6.8 in H8/3837 PROM mode.

**Table 6.8 Mode Selection in H8/3837 PROM Mode**

| | Pin | | | | | | |
|---|---|---|---|---|---|---|---|
| **Mode** | **$\overline{CE}$** | **$\overline{OE}$** | **PGM** | **$V_{PP}$** | **$V_{CC}$** | **$EO_7$ to $EO_0$** | **$EA_{16}$ to $EA_0$** |
| Write | L | H | L | $V_{PP}$ | $V_{CC}$ | Data input | Address input |
| Verify | L | L | H | $V_{PP}$ | $V_{CC}$ | Data output | Address input |
| Programming disabled | L | L | L | $V_{PP}$ | $V_{CC}$ | High impedance | Address input |
| | L | H | H | | | | |
| | H | L | L | | | | |
| | H | H | H | | | | |

Notation:
L: Low level
H: High level
$V_{PP}$: $V_{PP}$ level
$V_{CC}$: $V_{CC}$ level

The specifications for writing and reading the on-chip PROM are identical to those for the standard HN27C101 EPROM. Page programming is not supported, however. The PROM writer must not be set to page mode. A PROM programmer that provides only page programming mode cannot be used. When selecting a PROM programer, check that it supports a byte-by-byte high-speed, high-reliability programming method. Be sure to set the address range to H'0000 to H'EDFF.

### 6.5.1 Writing and Verifying

An efficient, high-speed, high-reliability method is available for writing and verifying the PROM data. This method achieves high speed without voltage stress on the device and without lowering the reliability of written data. The basic flow of this high-speed, high-reliability programming method is shown in figure 6.8.

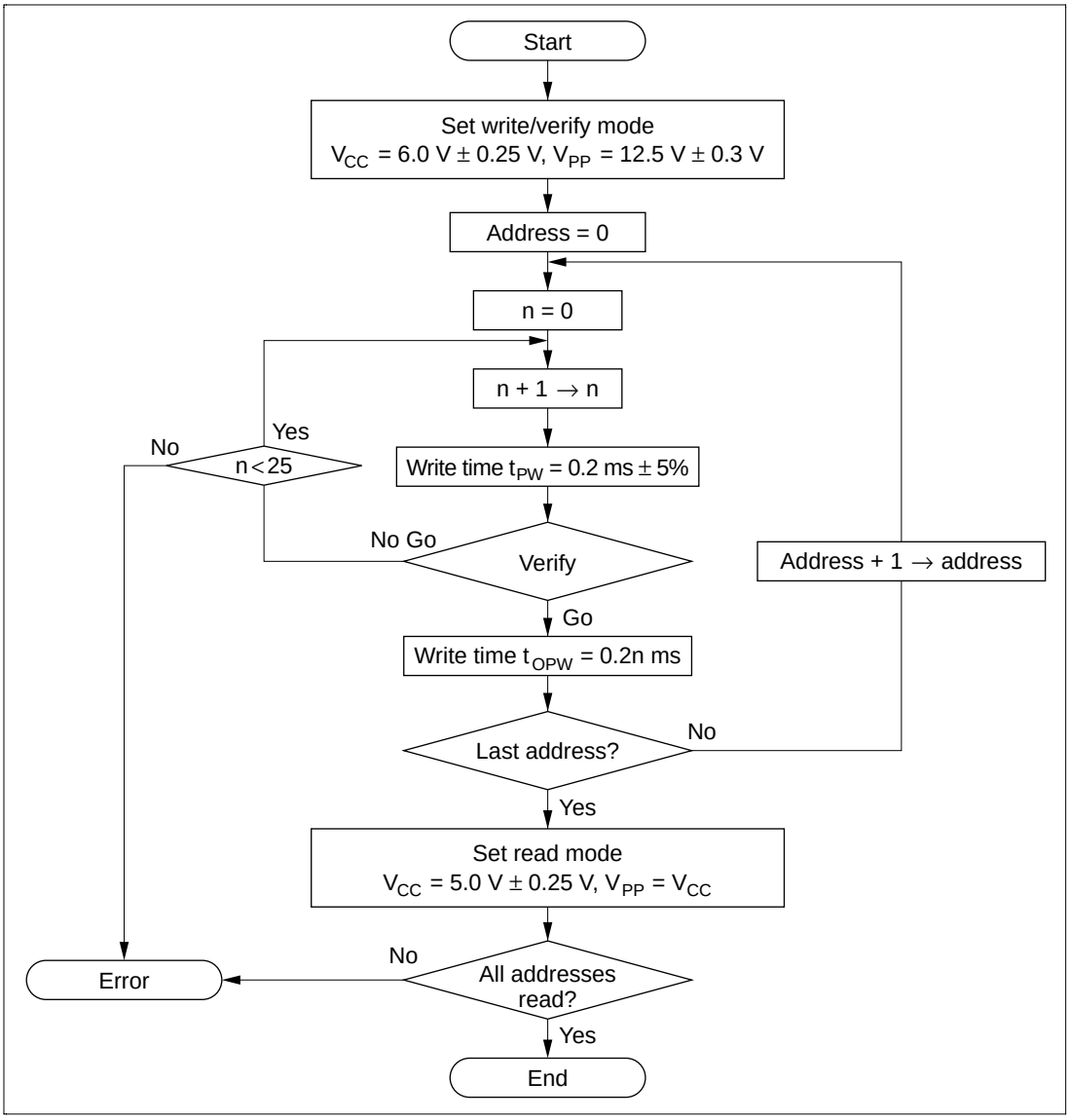



**Figure 6.8 High-Speed, High-Reliability Programming Flow Chart**

Table 6.9 and table 6.10 give the electrical characteristics in programming mode.

**Table 6.9 DC Characteristics (preliminary)**

(Conditions: $V_{CC}$ = 6.0 V ±0.25 V, $V_{PP}$ = 12.5 V ±0.3 V, $V_{SS}$ = 0 V, $T_a$ = 25°C ±5°C)

| Item | | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|---|
| Input high-level voltage | $EO_7$ to $EO_0$, $EA_{16}$ to $EA_0$, $\overline{OE}$, $\overline{CE}$, $\overline{PGM}$ | $V_{IH}$ | 2.4 | — | $V_{CC}$ + 0.3 | V | |
| Input low-level voltage | $EO_7$ to $EO_0$, $EA_{16}$ to $EA_0$, $\overline{OE}$, $\overline{CE}$, $\overline{PGM}$ | $V_{IL}$ | –0.3 | — | 0.8 | V | |
| Output high-level voltage | $EO_7$ to $EO_0$ | $V_{OH}$ | 2.4 | — | — | V | $I_{OH}$ = –200 µA |
| Output low-level voltage | $EO_7$ to $EO_0$ | $V_{OL}$ | — | — | 0.45 | V | $I_{OL}$ = 0.8 mA |
| Input leakage current | $EO_7$ to $EO_0$, $EA_{16}$ to $EA_0$, $\overline{OE}$, $\overline{CE}$, $\overline{PGM}$ | $\|I_{LI}\|$ | — | — | 2 | µA | $V_{in}$ = 5.25 V/ 0.5 V |
| $V_{CC}$ current | | $I_{CC}$ | — | — | 40 | mA | |
| $V_{PP}$ current | | $I_{PP}$ | — | — | 40 | mA | |

**Table 6.10 AC Characteristics**

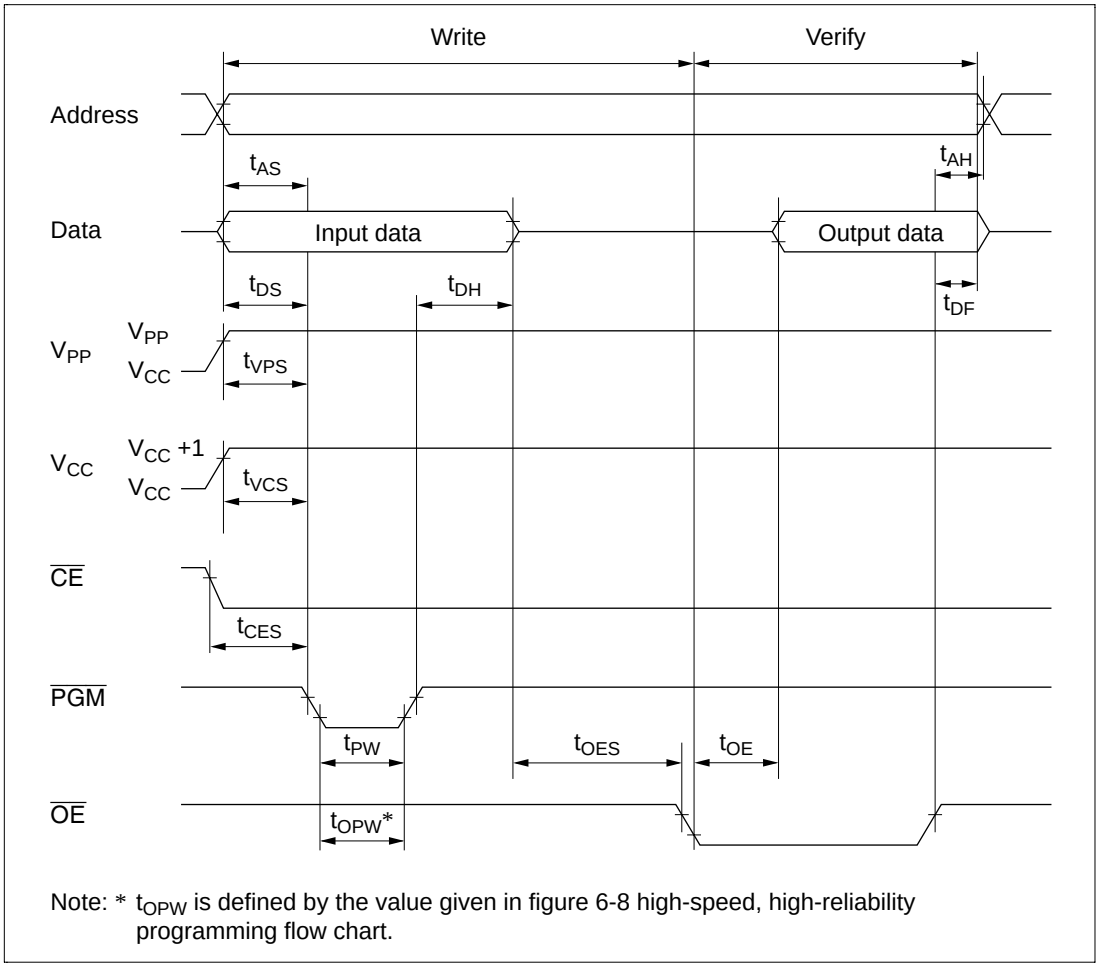
(Conditions: $V_{CC}$ = 6.0 V ±0.25 V, $V_{PP}$ = 12.5 V ±0.3 V, $T_a$ = 25°C ±5°C)

| Item | Symbol | Min | Typ | Max | Unit | Test Conditions |
|---|---|---|---|---|---|---|
| Address setup time | $t_{AS}$ | 2 | — | — | µs | Figure 6.9*[1] |
| $\overline{OE}$ setup time | $t_{OES}$ | 2 | — | — | µs | |
| Data setup time | $t_{DS}$ | 2 | — | — | µs | |
| Address hold time | $t_{AH}$ | 0 | — | — | µs | |
| Data hold time | $t_{DH}$ | 2 | — | — | µs | |
| Data output disable time | $t_{DF}$*[2] | — | — | 130 | ns | |
| $V_{PP}$ setup time | $t_{VPS}$ | 2 | — | — | µs | |
| Programming pulse width | $t_{PW}$ | 0.19 | 0.20 | 0.21 | ms | |
| $\overline{PGM}$ pulse width for overwrite programming | $t_{OPW}$*[3] | 0.19 | — | 5.25 | ms | |
| $V_{CC}$ setup time | $t_{VCS}$ | 2 | — | — | µs | |
| $\overline{CE}$ setup time | $t_{CES}$ | 2 | — | — | µs | |
| Data output delay time | $t_{OE}$ | 0 | — | 200 | ns | |

Notes: 1. Input pulse level: 0.45 V to 2.4 V
Input rise time/fall time ≤ 20 ns
Timing reference levels: Input: 0.8 V, 2.0 V
Output: 0.8 V, 2.0 V

2. $t_{DF}$ is defined at the point at which the output is floating and the output level cannot be read.
3. $t_{OPW}$ is defined by the value given in figure 6.8 high-speed, high-reliability programming flow chart.

Figure 6.9 shows a write/verify timing diagram.




Note: * $t_{OPW}$ is defined by the value given in figure 6-8 high-speed, high-reliability programming flow chart.

**Figure 6.9 PROM Write/Verify Timing**

### 6.5.2 Programming Precautions

- Use the specified programming voltage and timing.

  The programming voltage in PROM mode ($V_{PP}$) is 12.5 V. Use of a higher voltage can permanently damage the chip. Be especially careful with respect to PROM programmer overshoot.

  Setting the PROM programmer to Hitachi specifications for the HN27C101 will result in correct $V_{PP}$ of 12.5 V.

- Make sure the index marks on the PROM programmer socket, socket adapter, and chip are properly aligned. If they are not, the chip may be destroyed by excessive current flow. Before programming, be sure that the chip is properly mounted in the PROM programmer.

- Avoid touching the socket adapter or chip while programming, since this may cause contact faults and write errors.

- Select the programming mode carefully. The chip cannot be programmed in page programming mode.

- When programming with a PROM programmer, be sure to specify addresses from H'0000 to H'EDFF. If address H'EE00 and higher addresses are programmed by mistake, it may become impossible to program the PROM or verify the programmed data. When programming, assign H'FF data to the address area from H'EE00 to H'1FFFF.

## 6.6 Reliability of Programmed Data

A highly effective way of assuring data retention characteristics after programming is to screen the chips by baking them at a temperature of 150°C. High-temperature baking is a screening method that quickly eliminates PROM memory cells prone to initial data retention failure.

Figure 6.10 shows a flowchart of this screening procedure.

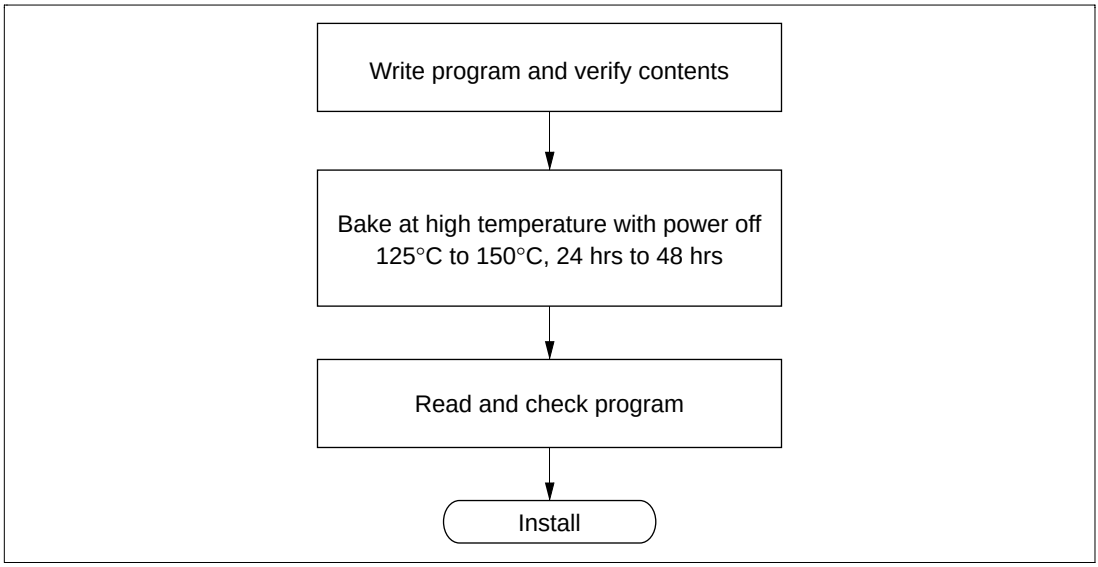



**Figure 6.10 Recommended Screening Procedure**

If write errors occur repeatedly while the same PROM programmer is being used, stop programming and check for problems in the PROM programmer and socket adapter, etc.

Please notify your Hitachi representative of any problems occurring during programming or in screening after high-temperature baking.

# Section 7 RAM

## 7.1 Overview

The H8/3832, H8/3833 and H8/3834 have 1 kbyte of high-speed static RAM on-chip, while the H8/3835, H8/3836, and H8/3837 each have 2 kbytes. The RAM is connected to the CPU by a 16-bit data bus, allowing high-speed 2-state access for both byte data and word data.

### 7.1.1 Block Diagram

Figure 7.1 shows a block diagram of the on-chip RAM.

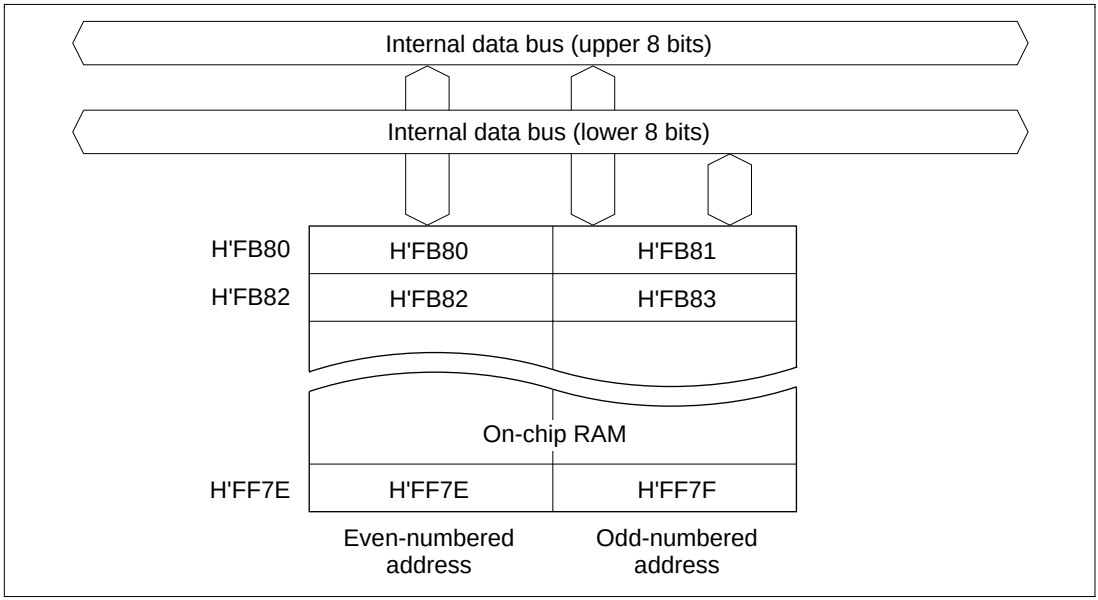



**Figure 7.1 RAM Block Diagram (H8/3834)**

# Section 8 I/O Ports

## 8.1 Overview

The H8/3834 Series is provided with eight 8-bit I/O ports, one 4-bit I/O port, one 3-bit I/O port, one 8-bit input-only port, one 4-bit input-only port, and one 1-bit input-only port. Table 8.1 indicates the functions of each port.

Each port has of a port control register (PCR) that controls input and output, and a port data register (PDR) for storing output data. Input or output can be assigned to individual bits. See 2.9.2, Notes on Bit Manipulation, for information on executing bit-manipulation instructions to write data in PCR or PDR.

Ports 5, 6, 7, 8, 9, and A double as liquid crystal display segment pins and common pins. The choice of pin functions can be made in 4-bit groupings.

Block diagrams of each port are given in Appendix C, I/O Port Block Diagrams.

**Table 8.1 Port Functions**

| Port | Description | Pins | Other Functions | Function Switching Register |
|---|---|---|---|---|
| Port 1 | • 8-bit I/O port<br>• Input pull-up MOS option | $P1_7$ to $P1_5$/ $\overline{IRQ_3}$ to $\overline{IRQ_1}$/ TMIF, TMIC, TMIB | External interrupts 3 to 1<br>Timer event input TMIF, TMIC, TMIB | PMR1<br>TCRF, TMC, TMB |
| | | $P1_4$/PWM | 14-bit PWM output | PMR1 |
| | | $P1_3$/TMIG | Timer G input capture | PMR1 |
| | | $P1_2$, $P1_1$/ TMOFH, TMOFL | Timer F output compare | PMR1 |
| | | $P1_0$/TMOW | Timer A clock output | PMR1 |
| Port 2 | • 8-bit I/O port | $P2_7$ to $P2_2$ | None | |
| | • Open drain output option | $P2_1$/UD | Timer C count-up/down selection | PMR2 |
| | • High-current port | $P2_0$/$\overline{IRQ_4}$/ $\overline{ADTRG}$ | External interrupt 4 and A/D converter external trigger | PMR2<br>AMR |

**Table 8.1 Port Functions (cont)**

| Port | Description | Pins | Other Functions | Function Switching Register |
|---|---|---|---|---|
| Port 3 | • 8-bit I/O port<br>• Input pull-up MOS option<br>• High-current port | $P3_7/\overline{CS}$<br>$P3_6$/STRB<br>$P3_5/SO_2$<br>$P3_4/SI_2$<br>$P3_3/SCK_2$ | SCI2 chip select input ($\overline{CS}$), strobe output (STRB), data output ($SO_2$), data input ($SI_2$), clock input/output ($SCK_2$) | PMR3 |
| | | $P3_2/SO_1$<br>$P3_1/SI_1$<br>$P3_0/SCK_1$ | SCI1 data output ($SO_1$), data input ($SI_1$), clock input/output ($SCK_1$) | PMR3 |
| Port 4 | • 1-bit input-only port | $P4_3/\overline{IRQ}_0$ | External interrupt 0 | PMR2 |
| | • 3-bit I/O port | $P4_2$/TXD<br>$P4_1$/RXD<br>$P4_0/SCK_3$ | SCI3 data output (TXD), data input (RXD), clock input/output ($SCK_3$) | SCR3<br>SMR3 |
| Port 5 | • 8-bit I/O port<br>• Input pull-up MOS option | $P5_7$ to $P5_0$/<br>$\overline{WKP}_7$ to $\overline{WKP}_0$/<br>$SEG_8$ to $SEG_1$ | • Wakeup input ($\overline{WKP}_7$ to $\overline{WKP}_0$)<br>• Segment output ($SEG_8$ to $SEG_1$) | PMR5<br>LPCR |
| Port 6 | • 8-bit I/O port<br>• Input pull-up MOS option | $P6_7$ to $P6_0$/<br>$SEG_{16}$ to $SEG_9$ | Segment output ($SEG_{16}$ to $SEG_9$) | LPCR |
| Port 7 | • 8-bit I/O port | $P7_7$ to $P7_0$/<br>$SEG_{24}$ to $SEG_{17}$ | Segment output ($SEG_{24}$ to $SEG_{17}$) | LPCR |
| Port 8 | • 8-bit I/O port | $P8_7$ to $P8_0$/<br>$SEG_{32}$ to $SEG_{25}$ | Segment output ($SEG_{32}$ to $SEG_{25}$) | LPCR |
| Port 9 | • 8-bit I/O port | $P9_7/SEG_{40}/CL_1$<br>$P9_6/SEG_{39}/CL_2$<br>$P9_5/SEG_{38}$/DO<br>$P9_4/SEG_{37}$/M<br>$P9_3$ to $P9_0$/<br>$SEG_{36}$ to $SEG_{33}$ | • Segment output ($SEG_{40}$ to $SEG_{37}$)<br>• Latch clock ($CL_1$), for external segment expansion, shift clock ($CL_2$), display data port (DO), and alternating signal (M)<br>• Segment output ($SEG_{36}$ to $SEG_{33}$) | LPCR |
| Port A | • 4-bit I/O port | $PA_3$ to $PA_0$/<br>$COM_4$ to $COM_1$ | Common output ($COM_4$ to $COM_1$) | LPCR |
| Port B | • 8-bit input port | $PB_7$ to $PB_0$/<br>$AN_7$ to $AN_0$ | A/D converter analog input | AMR |
| Port C | • 4-bit input port | $PC_3$ to $PC_0$/<br>$AN_{11}$ to $AN_8$ | A/D converter analog input | AMR |

## 8.2 Port 1

### 8.2.1 Overview

Port 1 is an 8-bit I/O port. Figure 8.1 shows its pin configuration.

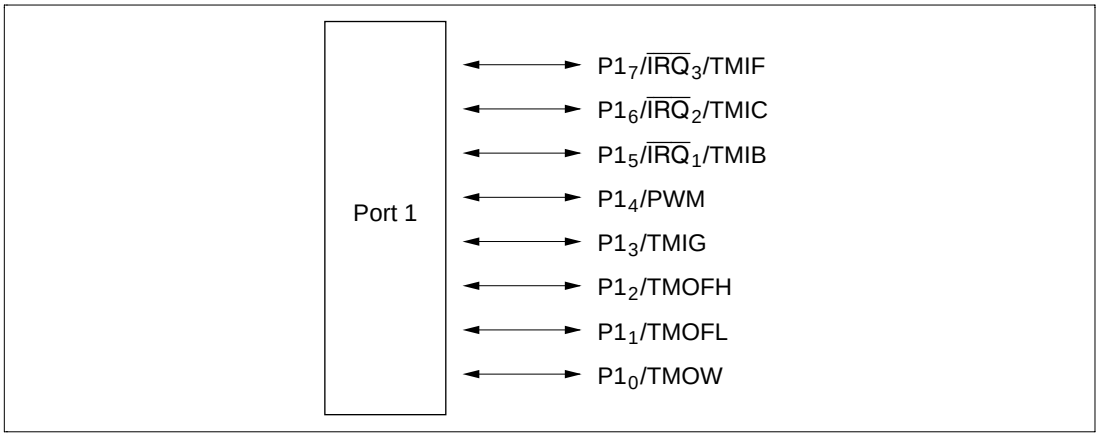



**Figure 8.1 Port 1 Pin Configuration**

### 8.2.2 Register Configuration and Description

Table 8.2 shows the port 1 register configuration.

**Table 8.2 Port 1 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 1 | PDR1 | R/W | H'00 | H'FFD4 |
| Port control register 1 | PCR1 | W | H'00 | H'FFE4 |
| Port pull-up control register 1 | PUCR1 | R/W | H'00 | H'FFE0 |
| Port mode register 1 | PMR1 | R/W | H'00 | H'FFC8 |

**Port Data Register 1 (PDR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P1_7$ | $P1_6$ | $P1_5$ | $P1_4$ | $P1_3$ | $P1_2$ | $P1_1$ | $P1_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR1 is an 8-bit register that stores data for pins $P1_7$ through $P1_0$. If port 1 is read while PCR1 bits are set to 1, the values stored in PDR1 are read, regardless of the actual pin states. If port 1 is read while PCR1 bits are cleared to 0, the pin states are read.

Upon reset, PDR1 is initialized to H'00.

**Port Control Register 1 (PCR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR1_7$ | $PCR1_6$ | $PCR1_5$ | $PCR1_4$ | $PCR1_3$ | $PCR1_2$ | $PCR1_1$ | $PCR1_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR1 is an 8-bit register for controlling whether each of the port 1 pins $P1_7$ to $P1_0$ functions as an input pin or output pin. Setting a PCR1 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR1 and in PDR1 are valid only when the corresponding pin is designated in PMR1 as a general I/O pin.

Upon reset, PCR1 is initialized to H'00.

PCR1 is a write-only register. All bits are read as 1.

**Port Pull-Up Control Register 1 (PUCR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR1_7$ | $PUCR1_6$ | $PUCR1_5$ | $PUCR1_4$ | $PUCR1_3$ | $PUCR1_2$ | $PUCR1_1$ | $PUCR1_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PUCR1 controls whether the MOS pull-up of each port 1 pin is on or off. When a PCR1 bit is cleared to 0, setting the corresponding PUCR1 bit to 1 turns on the MOS pull-up for the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR1 is initialized to H'00.

**Port Mode Register 1 (PMR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IRQ3 | IRQ2 | IRQ1 | PWM | TMIG | TMOFH | TMOFL | TMOW |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PMR1 is an 8-bit read/write register, controlling the selection of pin functions for port 1 pins.

Upon reset, PMR1 is initialized to H'00.

**Bit 7—$P1_7/\overline{IRQ_3}$/TMIF Pin Function Switch (IRQ3):** This bit selects whether pin $P1_7/\overline{IRQ_3}$/TMIF is used as $P1_7$ or as $\overline{IRQ_3}$/TMIF.

| Bit 7: IRQ3 | Description | |
|---|---|---|
| 0 | Functions as $P1_7$ I/O pin | (initial value) |
| 1 | Functions as $\overline{IRQ_3}$/TMIF input pin | |

Note: Rising or falling edge sensing can be designated for $\overline{IRQ_3}$/TMIF.
For details on TMIF pin settings, see 9.5.2 (3), timer control register F (TCRF).

**Bit 6—$P1_6/\overline{IRQ_2}$/TMIC Pin Function Switch (IRQ2):** This bit selects whether pin $P1_6/\overline{IRQ_2}$/TMIC is used as $P1_6$ or as $\overline{IRQ_2}$/TMIC.

| Bit 6: IRQ2 | Description | |
|---|---|---|
| 0 | Functions as $P1_6$ I/O pin | (initial value) |
| 1 | Functions as $\overline{IRQ_2}$/TMIC input pin | |

Note: Rising or falling edge sensing can be designated for $\overline{IRQ_2}$/TMIC.
For details on TMIC pin settings, see 9.4.2 (1), timer mode register C (TMC).

**Bit 5—$P1_5/\overline{IRQ_1}$/TMIB Pin Function Switch (IRQ1):** This bit selects whether pin $P1_5/\overline{IRQ_1}$/TMIB is used as $P1_5$ or as $\overline{IRQ_1}$/TMIB.

| Bit 5: IRQ1 | Description | |
|---|---|---|
| 0 | Functions as $P1_5$ I/O pin | (initial value) |
| 1 | Functions as $\overline{IRQ_1}$/TMIB input pin | |

Note: Rising or falling edge sensing can be designated for $\overline{IRQ_1}$/TMIB.
For details on TMIB pin settings, see 9.3.2 (1), timer mode register B (TMB).

**Bit 4—$P1_4$/PWM Pin Function Switch (PWM):** This bit selects whether pin $P1_4$/PWM is used as $P1_4$ or as PWM.

| Bit 4: PWM | Description | |
|---|---|---|
| 0 | Functions as $P1_4$ I/O pin | (initial value) |
| 1 | Functions as PWM output pin | |

**Bit 3—$P1_3$/TMIG Pin Function Switch (TMIG):** This bit selects whether pin $P1_3$/TMIG is used as $P1_3$ or as TMIG.

| Bit 3: TMIG | Description | |
|---|---|---|
| 0 | Functions as $P1_3$ I/O pin | (initial value) |
| 1 | Functions as TMIG input pin | |

**Bit 2—$P1_2$/TMOFH Pin Function Switch (TMOFH):** This bit selects whether pin $P1_2$/TMOFH is used as $P1_2$ or as TMOFH.

| Bit 2: TMOFH | Description | |
|---|---|---|
| 0 | Functions as $P1_2$ I/O pin | (initial value) |
| 1 | Functions as TMOFH output pin | |

**Bit 1:** $P1_1$/TMOFL Pin Function Switch (TMOFL)

This bit selects whether pin $P1_1$/TMOFL is used as $P1_1$ or as TMOFL.

| Bit 1: TMOFL | Description | |
|---|---|---|
| 0 | Functions as $P1_1$ I/O pin | (initial value) |
| 1 | Functions as TMOFL output pin | |

**Bit 0—$P1_0$/TMOW Pin Function Switch (TMOW):** This bit selects whether pin $P1_0$/TMOW is used as $P1_0$ or as TMOW.

| Bit 0: TMOW | Description | |
|---|---|---|
| 0 | Functions as $P1_0$ I/O pin | (initial value) |
| 1 | Functions as TMOW output pin | |

### 8.2.3 Pin Functions

Table 8.3 shows the port 1 pin functions.

**Table 8.3 Port 1 Pin Functions**

| Pin | Pin Functions and Selection Method |
|---|---|
| $P1_7/\overline{IRQ}_3$/TMIF | The pin function depends on bit IRQ3 in PMR1, bits CKSL2 to CKSL0 in TCRF, and bit $PCR1_7$ in PCR1. |

| IRQ3 | 0 | | 1 | |
|---|---|---|---|---|
| $PCR1_7$ | 0 | 1 | * | |
| CKSL2 to CKSL0 | * | | Not 0** | 0** |
| Pin function | $P1_7$ input pin | $P1_7$ output pin | $\overline{IRQ}_3$ input pin | $\overline{IRQ}_3$/TMIF input pin |

Note: When using as TMIF input pin, clear bit IEN3 in IENR1 to 0, disabling $IRQ_3$ interrupts.

$P1_6/\overline{IRQ}_2$/TMIC — The pin function depends on bit IRQ2 in PMR1, bits TMC2 to TMC0 in TMC, and bit $PCR1_6$ in PCR1.

| IRQ2 | 0 | | 1 | |
|---|---|---|---|---|
| $PCR1_6$ | 0 | 1 | * | |
| TMC2 to TMC0 | * | | Not 111 | 111 |
| Pin function | $P1_6$ input pin | $P1_6$ output pin | $\overline{IRQ}_2$ input pin | $\overline{IRQ}_2$/TMIC input pin |

Note: When using as TMIC input pin, clear bit IEN2 in IENR1 to 0, disabling $IRQ_2$ interrupts.

$P1_5/\overline{IRQ}_1$/TMIB — The pin function depends on bit IRQ1 in PMR1, bits TMB2 to TMB0 in TMB, and bit $PCR1_5$ in PCR1.

| IRQ1 | 0 | | 1 | |
|---|---|---|---|---|
| $PCR1_5$ | 0 | 1 | * | |
| TMB2 to TMB0 | * | | Not 111 | 111 |
| Pin function | $P1_5$ input pin | $P1_5$ output pin | $\overline{IRQ}_1$ input pin | $\overline{IRQ}_1$/TMIB input pin |

Note: When using as TMIB input pin, clear bit IEN1 in IENR1 to 0, disabling IRQ1 interrupts.

Note: * Don't care

**Table 8.3 Port 1 Pin Functions (cont)**

| Pin | Pin Functions and Selection Method | | |
|---|---|---|---|
| $P1_4$/PWM | The pin function depends on bit PWM in PMR1 and bit $PCR1_4$ in PCR1. | | |
| | PWM | 0 | 1 |
| | $PCR1_4$ | 0 | 1 | * |
| | Pin function | $P1_4$ input pin | $P1_4$ output pin | PWM output pin |
| $P1_3$/TMIG | The pin function depends on bit TMIG in PMR1 and bit $PCR1_3$ in PCR1. | | |
| | TMIG | 0 | 1 |
| | $PCR1_3$ | 0 | 1 | * |
| | Pin function | $P1_3$ input pin | $P1_3$ output pin | TMIG input pin |
| $P1_2$/TMOFH | The pin function depends on bit TMOFH in PMR1 and bit $PCR1_2$ in PCR1. | | |
| | TMOFH | 0 | 1 |
| | $PCR1_2$ | 0 | 1 | * |
| | Pin function | $P1_2$ input pin | $P1_2$ output pin | TMOFH output pin |
| $P1_1$/TMOFL | The pin function depends on bit TMOFL in PMR1 and bit $PCR1_1$ in PCR1. | | |
| | TMOFL | 0 | 1 |
| | $PCR1_1$ | 0 | 1 | * |
| | Pin function | $P1_1$ input pin | $P1_1$ output pin | TMOFL output pin |
| $P1_0$/TMOW | The pin function depends on bit TMOW in PMR1 and bit $PCR1_0$ in PCR1. | | |
| | TMOW | 0 | 1 |
| | $PCR1_0$ | 0 | 1 | * |
| | Pin function | $P1_0$ input pin | $P1_0$ output pin | TMOW output pin |

Note: * Don't care

### 8.2.4 Pin States

Table 8.4 shows the port 1 pin states in each operating mode.

**Table 8.4 Port 1 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P1_7/\overline{IRQ_3}$/TMIF<br>$P1_6/\overline{IRQ_2}$/TMIC<br>$P1_5/\overline{IRQ_1}$/TMIB<br>$P1_4$/PWM<br>$P1_3$/TMIG<br>$P1_2$/TMOFH<br>$P1_1$/TMOFL<br>$P1_0$/TMOW | High-impedance | Retains previous state | Retains previous state | High-impedance* | Retains previous state | Functional | Functional |

Note: * A high-level signal is output when the MOS pull-up is in the on state.

### 8.2.5 MOS Input Pull-Up

Port 1 has a built-in MOS input pull-up function that can be controlled by software. When a PCR1 bit is cleared to 0, setting the corresponding PUCR1 bit to 1 turns on the MOS input pull-up for that pin. The MOS input pull-up function is in the off state after a reset.

| $PCR1_n$ | 0 | | 1 |
|---|---|---|---|
| $PUCR1_n$ | 0 | 1 | * |
| MOS input pull-up | Off | On | Off |

Note: * Don't care
n = 7 to 4

## 8.3 Port 2

### 8.3.1 Overview

Port 2 is an 8-bit I/O port. Figure 8.2 shows its pin configuration.

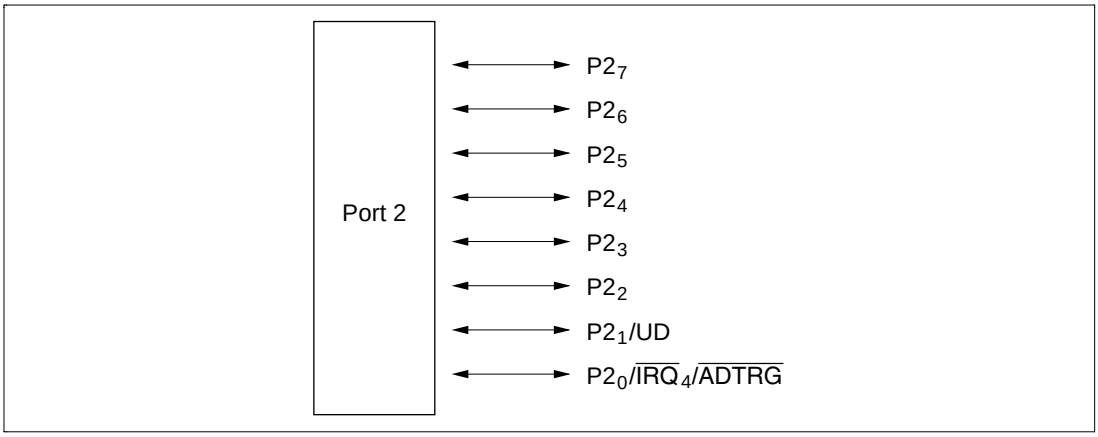



**Figure 8.2 Port 2 Pin Configuration**

### 8.3.2 Register Configuration and Description

Table 8.5 shows the port 2 register configuration.

**Table 8.5 Port 2 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 2 | PDR2 | R/W | H'00 | H'FFD5 |
| Port control register 2 | PCR2 | W | H'00 | H'FFE5 |
| Port mode register 2 | PMR2 | R/W | H'C0 | H'FFC9 |
| Port mode register 4 | PMR4 | R/W | H'00 | H'FFCB |

**Port Data Register 2 (PDR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P2_7$ | $P2_6$ | $P2_5$ | $P2_4$ | $P2_3$ | $P2_2$ | $P2_1$ | $P2_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR2 is an 8-bit register that stores data for pins $P2_7$ through $P2_0$. If port 2 is read while PCR2 bits are set to 1, the values stored in PDR2 are read, regardless of the actual pin states. If port 2 is read while PCR2 bits are cleared to 0, the pin states are read.

Upon reset, PDR2 is initialized to H'00.

**Port Control Register 2 (PCR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR2_7$ | $PCR2_6$ | $PCR2_5$ | $PCR2_4$ | $PCR2_3$ | $PCR2_2$ | $PCR2_1$ | $PCR2_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR2 is an 8-bit register for controlling whether each of the port 2 pins $P2_7$ to $P2_0$ functions as an input pin or output pin. Setting a PCR2 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR2 and in PDR2 are valid only when the corresponding pin is designated in PMR2 as a general I/O pin.

Upon reset, PCR2 is initialized to H'00.

PCR2 is a write-only register. All bits are read as 1.

**Port Mode Register 2 (PMR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | POF2 | NCS | IRQ0 | POF1 | UD | IRQ4 |
| Initial value | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | R/W | R/W | R/W | R/W | R/W | R/W |

PMR2 is an 8-bit read/write register, controlling the selection of pin functions for pins $P2_0$, $P2_1$,and $P4_3$, controlling the PMOS on/off option for pins $P3_5/SO_2$ and $P3_2/SO_1$, and controlling the TMIG input noise canceller.

Upon reset, PMR2 is initialized to H'C0.

**Bits 7 to 6—Reserved Bits:** Bits 7 to 6 are reserved; they are always read as 1, and cannot be modified.

**Bit 5—$P3_5/SO_2$ Pin PMOS Control (POF2):** This bit controls the on/off state of the PMOS transistor in the $P3_5/SO_2$ pin output buffer.

| Bit 5: POF2 | Description | |
|---|---|---|
| 0 | CMOS output | (initial value) |
| 1 | NMOS open-drain output | |

**Bit 4—TMIG Noise Canceller Select (NCS):** This bit controls the noise canceller circuit for input capture at pin TMIG.

| Bit 4: NCS | Description | |
|---|---|---|
| 0 | Noise canceller function not selected | (initial value) |
| 1 | Noise canceller function selected | |

**Bit 3—$P4_3/\overline{IRQ_0}$ Pin Function Switch (IRQ0):** This bit selects whether pin $P4_3/\overline{IRQ_0}$ is used as $P4_3$ or as $\overline{IRQ_0}$.

| Bit 3: IRQ0 | Description | |
|---|---|---|
| 0 | Functions as $P4_3$ input pin | (initial value) |
| 1 | Functions as $\overline{IRQ_0}$ input pin | |

**Bit 2—$P3_2/SO_1$ Pin PMOS Control (POF1):** This bit controls the on/off state of the PMOS transistor in the $P3_2/SO_1$ pin output buffer.

| Bit 2: POF1 | Description | |
|---|---|---|
| 0 | CMOS output | (initial value) |
| 1 | NMOS open-drain output | |

**Bit 1—$P2_1$/UD Pin Function Switch (UD):** This bit selects whether pin $P2_1$/UD is used as $P2_1$ or as UD.

| Bit 1: UD | Description | |
|---|---|---|
| 0 | Functions as $P2_1$ I/O pin | (initial value) |
| 1 | Functions as UD input pin | |

**Bit 0: $P2_0/\overline{IRQ_4}/\overline{ADTRG}$ Pin Function Switch (IRQ4):** This bit selects whether pin $P2_0/\overline{IRQ_4}/\overline{ADTRG}$ is used as $P2_0$ or as $\overline{IRQ_4}/\overline{ADTRG}$.

| Bit 0: IRQ4 | Description | |
|---|---|---|
| 0 | Functions as $P2_0$ I/O pin | (initial value) |
| 1 | Functions as $\overline{IRQ_4}/\overline{ADTRG}$ input pin | |

Note: See 12.3.2, Start of A/D Conversion by External Trigger Input, for the $\overline{ADTRG}$ pin setting.

**Port Mode Register 4 (PMR4)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $NMOD_7$ | $NMOD_6$ | $NMOD_5$ | $NMOD_4$ | $NMOD_3$ | $NMOD_2$ | $NMOD_1$ | $NMOD_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PMR4 is an 8-bit read/write register, used to select CMOS output or NMOS open drain output for each port 2 pin.

Upon reset, PMR4 is initialized to H'00.

**Bit n—NMOS Open-Drain Output Select (NMODn):** This bit selects NMOS open-drain output when pin $P2_n$ is used as an output pin.

| Bit n: NMODn | Description |
|---|---|
| 0 | CMOS output |
| 1 | NMOS open-drain outputNMOS open-drain output |

(n = 7 to 0)

### 8.3.3 Pin Functions

Table 8.6 shows the port 2 pin functions.

**Table 8.6 Port 2 Pin Functions**

| Pin | Pin Functions and Selection Method |
|---|---|
| $P2_7$ to $P2_2$ | Input or output is selected as follows by the bit settings in PCR2. (n = 2 to 7) |

| PCR2n | 0 | 1 |
|---|---|---|
| Pin function | $P2_n$ input pin | $P2_n$ output pin |

$P2_1$/UD — The pin function depends on bit UD in PMR2 and bit $PCR2_1$ in PCR2.

| UD | 0 | | 1 |
|---|---|---|---|
| $PCR2_1$ | 0 | 1 | * |
| Pin function | $P2_1$ input pin | $P2_1$ output pin | UD input pin |

$P2_0$/$\overline{IRQ_4}$/$\overline{ADTRG}$ — The pin function depends on bit IRQ4 in PMR2, bit TRGE in AMR, and bit $PCR2_0$ in PCR2.

| IRQ4 | 0 | | 1 | |
|---|---|---|---|---|
| $PCR2_0$ | 0 | 1 | * | |
| TRGE | * | | 0 | 1 |
| Pin function | $P2_0$ input pin | $P2_0$ output pin | $\overline{IRQ_4}$ input pin | $\overline{IRQ_4}$/$\overline{ADTRG}$ input pin |

Note: When using as $\overline{ADTRG}$ input pin, clear bit IEN4 in IENR1 to 0, disabling $\overline{IRQ_4}$ interrupts.

Note: * Don't care

### 8.3.4 Pin States

Table 8.7 shows the port 2 pin states in each operating mode.

**Table 8.7 Port 2 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P2_7$ to $P2_2$<br>$P2_1$/UD<br>$P2_0$/$\overline{IRQ_4}$/<br>$\overline{ADTRG}$ | High-impedance | Retains previous state | Retains previous state | High-impedance | Retains previous state | Functional | Functional |

## 8.4 Port 3

### 8.4.1 Overview

Port 3 is an 8-bit I/O port, configured as shown in figure 8.3.

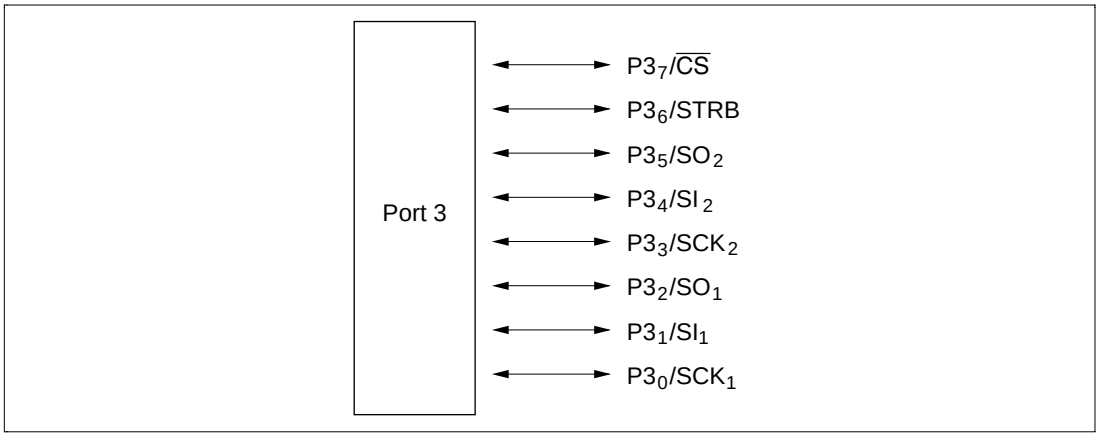



**Figure 8.3 Port 3 Pin Configuration**

### 8.4.2 Register Configuration and Description

Table 8.8 shows the port 3 register configuration.

**Table 8.8 Port 3 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 3 | PDR3 | R/W | H'00 | H'FFD6 |
| Port control register 3 | PCR3 | W | H'00 | H'FFE6 |
| Port pull-up control register 3 | PUCR3 | R/W | H'00 | H'FFE1 |
| Port mode register 3 | PMR3 | R/W | H'00 | H'FFCA |

**Port Data Register 3 (PDR3)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P3_7$ | $P3_6$ | $P3_5$ | $P3_4$ | $P3_3$ | $P3_2$ | $P3_1$ | $P3_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR3 is an 8-bit register that stores data for port 3 pins $P3_7$ to $P3_0$. If port 3 is read while PCR3 bits are set to 1, the values stored in PDR3 are read, regardless of the actual pin states. If port 3 is read while PCR3 bits are cleared to 0, the pin states are read.

Upon reset, PDR3 is initialized to H'00.

**Port Control Register 3 (PCR3)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR3_7$ | $PCR3_6$ | $PCR3_5$ | $PCR3_4$ | $PCR3_3$ | $PCR3_2$ | $PCR3_1$ | $PCR3_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR3 is an 8-bit register for controlling whether each of the port 3 pins $P3_7$ to $P3_0$ functions as an input pin or output pin. Setting a PCR3 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR3 and in PDR3 are valid only when the corresponding pin is designated in PMR3 as a general I/O pin.

Upon reset, PCR3 is initialized to H'00.

PCR3 is a write-only register. All bits are read as 1.

**Port Pull-Up Control Register 3 (PUCR3)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR3_7$ | $PUCR3_6$ | $PUCR3_5$ | $PUCR3_4$ | $PUCR3_3$ | $PUCR3_2$ | $PUCR3_1$ | $PUCR3_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PUCR3 bits control the on/off state of pin $P3_7$–$P3_0$ MOS pull-ups. When a PCR3 bit is cleared to 0, setting the corresponding PUCR3 bit to 1 turns on the MOS pull-up for the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR3 is initialized to H'00.

**Port Mode Register 3 (PMR3)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CS | STRB | SO2 | SI2 | SCK2 | SO1 | SI1 | SCK1 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PMR3 is an 8-bit read/write register, controlling the selection of pin functions for port 3 pins.

Upon reset, PMR3 is initialized to H'00.

**Bit 7—$P3_7/\overline{CS}$ Pin Function Switch (CS):** This bit selects whether pin $P3_7/\overline{CS}$ is used as $P3_7$ or as $\overline{CS}$.

| Bit 7: CS | Description | |
|---|---|---|
| 0 | Functions as $P3_7$ I/O pin | (initial value) |
| 1 | Functions as $\overline{CS}$ input pin | |

**Bit 6—$P3_6$/STRB Pin Function Switch (STRB):** This bit selects whether pin $P3_6$/STRB is used as $P3_6$ or as STRB.

| Bit 6: STRB | Description | |
|---|---|---|
| 0 | Functions as $P3_6$ I/O pin | (initial value) |
| 1 | Functions as STRB output pin | |

**Bit 5—$P3_5/SO_2$ Pin Function Switch (SO2):** This bit selects whether pin $P3_5/SO_2$ is used as $P3_5$ or as $SO_2$.

| Bit 5: SO2 | Description | |
|---|---|---|
| 0 | Functions as $P3_5$ I/O pin | (initial value) |
| 1 | Functions as $SO_2$ output pin | |

**Bit 4—$P3_4/SI_2$ Pin Function Switch (SI2):** This bit selects whether pin $P3_4/SI_2$ is used as $P3_4$ or as $SI_2$.

| Bit 4: SI2 | Description | |
|---|---|---|
| 0 | Functions as $P3_4$ I/O pin | (initial value) |
| 1 | Functions as $SI_2$ input pin | |

**Bit 3—$P3_3/SCK_2$ Pin Function Switch (SCK2):** This bit selects whether pin $P3_3/SCK_2$ is used as $P3_3$ or as $SCK_2$.

| Bit 3: SCK2 | Description | |
|---|---|---|
| 0 | Functions as $P3_3$ I/O pin | (initial value) |
| 1 | Functions as $SCK_2$ I/O pin | |

**Bit 2—$P3_2/SO_1$ Pin Function Switch (SO1):** This bit selects whether pin $P3_2/SO_1$ is used as $P3_2$ or as $SO_1$.

| Bit 2: SO1 | Description | |
|---|---|---|
| 0 | Functions as $P3_2$ I/O pin | (initial value) |
| 1 | Functions as $SO_1$ output pin | |

**Bit 1—$P3_1/SI_1$ Pin Function Switch (SI1):** This bit selects whether pin $P3_1/SI_1$ is used as $P3_1$ or as $SI_1$.

| Bit 1: SI1 | Description | |
|---|---|---|
| 0 | Functions as $P3_1$ I/O pin | (initial value) |
| 1 | Functions as $SI_1$ input pin | |

**Bit 0—$P3_0/SCK_1$ Pin Function Switch (SCK1):** This bit selects whether pin $P3_0/SCK_1$ is used as $P3_0$ or as $SCK_1$.

| Bit 0: SCK1 | Description | |
|---|---|---|
| 0 | Functions as $P3_0$ I/O pin | (initial value) |
| 1 | Functions as $SCK_1$ I/O pin | |

### 8.4.3 Pin Functions

Table 8.9 shows the port 3 pin functions.

**Table 8.9 Port 3 Pin Functions**

| Pin | Pin Functions and Selection Method | | | |
|---|---|---|---|---|
| $P3_7/\overline{CS}$ | The pin function depends on bit CS in PMR3 and bit $PCR3_7$ in PCR3. | | | |
| | CS | 0 | | 1 |
| | $PCR3_7$ | 0 | 1 | * |
| | Pin function | $P3_7$ input pin | $P3_7$ output pin | $\overline{CS}$ input pin |
| $P3_6$/STRB | The pin function depends on bit STRB in PMR3 and bit $PCR3_6$ in PCR3. | | | |
| | STRB | 0 | | 1 |
| | $PCR3_6$ | 0 | 1 | * |
| | Pin function | $P3_6$ input pin | $P3_6$ output pin | STRB output pin |
| $P3_5/SO_2$ | The pin function depends on bit SO2 in PMR3 and bit $PCR3_5$ in PCR3. | | | |
| | SO2 | 0 | | 1 |
| | $PCR3_5$ | 0 | 1 | * |
| | Pin function | $P3_5$ input pin | $P3_5$ output pin | $SO_2$ output pin |
| $P3_4/SI_2$ | The pin function depends on bit SI2 in PMR3 and bit $PCR3_4$ in PCR3. | | | |
| | SI2 | 0 | | 1 |
| | $PCR3_4$ | 0 | 1 | * |
| | Pin function | $P3_4$ input pin | $P3_4$ output pin | $SI_2$ input pin |
| $P3_3/SCK_2$ | The pin function depends on bit SCK2 in PMR3, bits CKS2 to 0 in SCR2, and bit $PCR3_3$ in PCR3. | | | |
| | SCK2 | 0 | | 1 | |
| | CKS2 to CKS0 | * | | Not 111 | 111 |
| | $PCR3_3$ | 0 | 1 | * | * |
| | Pin function | $P3_3$ input pin | $P3_3$ output pin | $SCK_2$ output pin | $SCK_2$ input pin |

Note: * Don't care

**Table 8.9 Port 3 Pin Functions (cont)**

| Pin | Pin Functions and Selection Method |
|---|---|
| $P3_2/SO_1$ | The pin function depends on bit SO1 in PMR3 and bit $PCR3_2$ in PCR3. |

| SO1 | 0 | | 1 |
|---|---|---|---|
| $PCR3_2$ | 0 | 1 | * |
| Pin function | $P3_2$ input pin | $P3_2$ output pin | $SO_1$ output pin |

| $P3_1/SI_1$ | The pin function depends on bit SI1 in PMR3 and bit $PCR3_1$ in PCR3. |
|---|---|

| SI1 | 0 | | 1 |
|---|---|---|---|
| $PCR3_1$ | 0 | 1 | * |
| Pin function | $P3_1$ input pin | $P3_1$ output pin | $SI_1$ input pin |

| $P3_0/SCK_1$ | The pin function depends on bit SCK1 in PMR3, bit CKS3 in SCR1, and bit $PCR3_0$ in PCR3. |
|---|---|

| SCK1 | 0 | | 1 | |
|---|---|---|---|---|
| CKS3 | * | | 0 | 1 |
| $PCR3_0$ | 0 | 1 | * | * |
| Pin function | $P3_0$ input pin | $P3_0$ output pin | $SCK_1$ output pin | $SCK_1$ input pin |

Note: * Don't care

### 8.4.4 Pin States

Table 8.10 shows the port 3 pin states in each operating mode.

**Table 8.10 Port 3 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P3_7/\overline{CS}$<br>$P3_6$/STRB<br>$P3_5/SO_2$<br>$P3_4/SI_2$<br>$P3_3/SCK_2$<br>$P3_2/SO_1$<br>$P3_1/SI_1$<br>$P3_0/SCK_1$ | High-impedance | Retains previous state | Retains previous state | High-impedance* | Retains previous state | Functional | Functional |

Note: * A high-level signal is output when the MOS pull-up is in the on state.

### 8.4.5 MOS Input Pull-Up

Port 3 has a built-in MOS input pull-up function that can be controlled by software. When a PCR3 bit is cleared to 0, setting the corresponding PUCR3 bit to 1 turns on the MOS pull-up for that pin. The MOS pull-up function is in the off state after a reset.

| $PCR3_n$ | 0 | | 1 |
|---|---|---|---|
| $PUCR3_n$ | 0 | 1 | * |
| MOS input pull-up | Off | On | Off |

Note: * Don't care
n = 7 to 0

## 8.5 Port 4

### 8.5.1 Overview

Port 4 consists of a 3-bit I/O port and a 1-bit input port, and is configured as shown in figure 8.4.

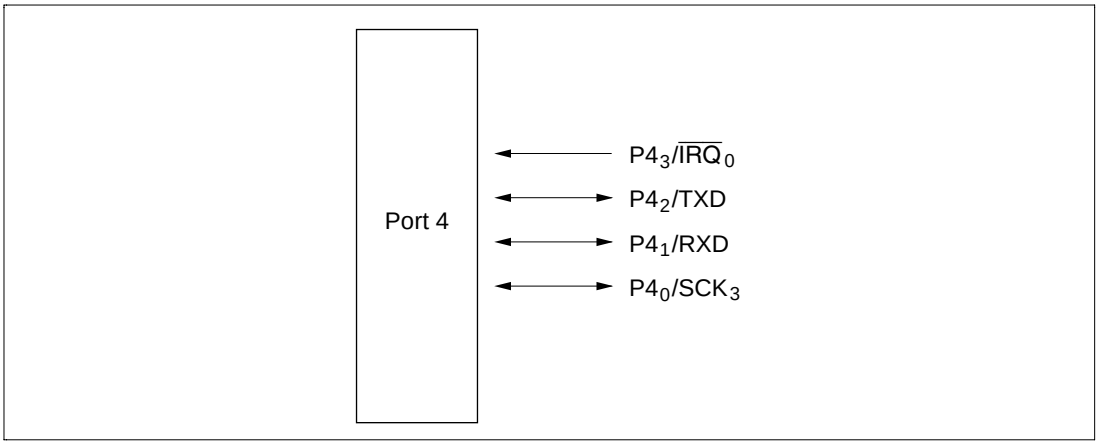



**Figure 8.4 Port 4 Pin Configuration**

### 8.5.2 Register Configuration and Description

Table 8.11 shows the port 4 register configuration.

**Table 8.11 Port 4 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 4 | PDR4 | R/W | H'F8 | H'FFD7 |
| Port control register 4 | PCR4 | W | H'F8 | H'FFE7 |

**Port Data Register 4 (PDR4)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $P4_3$ | $P4_2$ | $P4_1$ | $P4_0$ |
| Initial value | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | R | R/W | R/W | R/W |

PDR4 is an 8-bit register that stores data for port 4 pins $P4_2$ to $P4_0$. If port 4 is read while PCR4 bits are set to 1, the values stored in PDR4 are read, regardless of the actual pin states. If port 4 is read while PCR4 bits are cleared to 0, the pin states are read.

Upon reset, PDR4 is initialized to H'F8.

**Port Control Register 4 (PCR4)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | — | $PCR4_2$ | $PCR4_1$ | $PCR4_0$ |
| Initial value | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | — | W | W | W |

PCR4 controls whether each of the port 4 pins $P4_2$ to $P4_0$ functions as an input pin or output pin. Setting a PCR4 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR4 and in PDR4 are valid only when the corresponding pin is designated in SCR3 as a general I/O pin.

Upon reset, PCR4 is initialized to H'F8.

PCR4 is a write-only register. All bits are read as 1.

### 8.5.3 Pin Functions

Table 8.12 shows the port 4 pin functions.

**Table 8.12 Port 4 Pin Functions**

| Pin | Pin Functions and Selection Method | | | |
|---|---|---|---|---|
| $P4_3/\overline{IRQ}_0$ | The pin function depends on the IRQ0 bit setting in PMR2. | | | |
| | IRQ0 | 0 | | 1 |
| | Pin function | $P4_3$ input pin | | $\overline{IRQ}_0$ input pin |
| $P4_2$/TXD | The pin function depends on bit TE in SCR3 and bit $PCR4_2$ in PCR4. | | | |
| | TE | 0 | | 1 |
| | $PCR4_2$ | 0 | 1 | * |
| | Pin function | $P4_2$ input pin | $P4_2$ output pin | TXD output pin |
| $P4_1$/RXD | The pin function depends on bit RE in SCR3 and bit $PCR4_1$ in PCR4. | | | |
| | RE | 0 | | 1 |
| | $PCR4_1$ | 0 | 1 | * |
| | Pin function | $P4_1$ input pin | $P4_1$ output pin | RXD input pin |
| $P4_0/SCK_3$ | The pin function depends on bits CKE1 and CKE0 in SCR3, bit COM in SMR, and bit $PCR4_0$ in PCR4. | | | |
| | CKE1 | 0 | | 1 |
| | CKE0 | 0 | 1 | * |
| | COM | 0 | 1 | * |
| | $PCR4_0$ | 0 | 1 | * |
| | Pin function | $P4_0$ input pin | $P4_0$ output pin | $SCK_3$ output pin / $SCK_3$ input pin |

Note: * Don't care

### 8.5.4 Pin States

Table 8.13 shows the port 4 pin states in each operating mode.

**Table 8.13 Port 4 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P4_3/\overline{IRQ_0}$<br>$P4_2$/TXD<br>$P4_1$/RXD<br>$P4_0/SCK_3$ | High-impedance | Retains previous state | Retains previous state | High-impedance | Retains previous state | Functional | Functional |

## 8.6 Port 5

### 8.6.1 Overview

Port 5 is an 8-bit I/O port, configured as shown in figure 8.5.

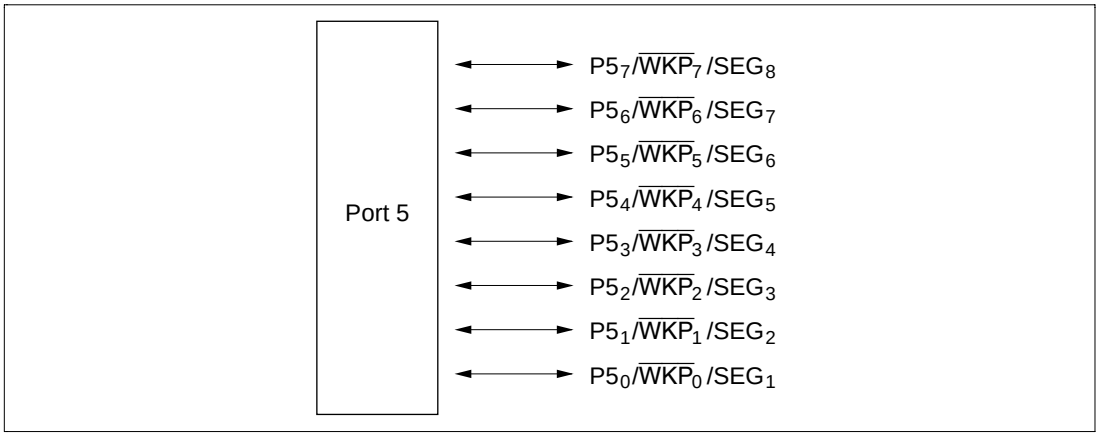



**Figure 8.5 Port 5 Pin Configuration**

### 8.6.2 Register Configuration and Description

Table 8.14 shows the port 5 register configuration.

**Table 8.14 Port 5 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 5 | PDR5 | R/W | H'00 | H'FFD8 |
| Port control register 5 | PCR5 | W | H'00 | H'FFE8 |
| Port pull-up control register 5 | PUCR5 | R/W | H'00 | H'FFE2 |
| Port mode register 5 | PMR5 | R/W | H'00 | H'FFCC |

**Port Data Register 5 (PDR5)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P5_7$ | $P5_6$ | $P5_5$ | $P5_4$ | $P5_3$ | $P5_2$ | $P5_1$ | $P5_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR5 is an 8-bit register that stores data for port 5 pins $P5_7$ to $P5_0$. If port 5 is read while PCR5 bits are set to 1, the values stored in PDR5 are read, regardless of the actual pin states. If port 5 is read while PCR5 bits are cleared to 0, the pin states are read.

Upon reset, PDR5 is initialized to H'00.

**Port Control Register 5 (PCR5)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR5_7$ | $PCR5_6$ | $PCR5_5$ | $PCR5_4$ | $PCR5_3$ | $PCR5_2$ | $PCR5_1$ | $PCR5_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR5 is an 8-bit register for controlling whether each of the port 5 pins $P5_7$ to $P5_0$ functions as an input pin or output pin. Setting a PCR5 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR5 and in PDR5 are valid only when the corresponding pin is designated as a general I/O pin in PMR5 and in bits SGS3 to SGS0 of LPCR.

Upon reset, PCR5 is initialized to H'00.

PCR5 is a write-only register. All bits are read as 1.

**Port Pull-up Control Register 5 (PUCR5)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR5_7$ | $PUCR5_6$ | $PUCR5_5$ | $PUCR5_4$ | $PUCR5_3$ | $PUCR5_2$ | $PUCR5_1$ | $PUCR5_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PUCR5 bits control the on/off state of pin $P5_7$–$P5_0$ MOS pull-ups. When a PCR5 bit is cleared to 0, setting the corresponding PUCR5 bit to 1 turns on the MOS pull-up for the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR5 is initialized to H'00.

**Port Mode Register 5 (PMR5)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WKP7 | WKP6 | WKP5 | WKP4 | WKP3 | WKP2 | WKP1 | WKP0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PMR5 is an 8-bit read/write register, controlling the selection of pin functions for port 5 pins.

Upon reset, PMR5 is initialized to H'00.

**Bit n—$P5_n/\overline{WKP}_n/SEG_{n+1}$ Pin Function Switch (WKPn):** When pin $P5_n/\overline{WKP}_n/SEG_{n+1}$ is not used as a $SEG_{n+1}$ pin, this bit selects whether it is used as $P5_n$ or as $\overline{WKP}_n$.

| Bit n: WKPn | Description | |
|---|---|---|
| 0 | Functions as $P5_n$ I/O pin | (initial value) |
| 1 | Functions as $\overline{WKP}_n$ input pin | |

(n = 7 to 0)

Note: For information on use as a $SEG_{n+1}$ pin, see 13.2.1, LCD Port Control Register (LPCR).

### 8.6.3 Pin Functions

Table 8.15 shows the port 5 pin functions.

**Table 8.15 Port 5 Pin Functions**

| Pin | Pin Functions and Selection Method |
|---|---|
| $P5_7/\overline{WKP}_7/SEG_8$ to $P5_4/\overline{WKP}_4/SEG_5$ | The pin function depends on bit WKPn in PMR5, bit $PCR5_n$ in PCR5, and bits SGS3 to SGS0 in LPCR. (n = 7 to 4) |

| SGS3 to SGS0 | 0*** | | | 1*** |
|---|---|---|---|---|
| WKPn | 0 | | 1 | * |
| $PCR5_n$ | 0 | 1 | * | * |
| Pin function | $P5_n$ input pin | $P5_n$ output pin | $\overline{WKP}_n$ input pin | $SEG_{n+1}$ output pin |

| $P5_3/\overline{WKP}_3/SEG_4$ to $P5_0/\overline{WKP}_0/SEG_1$ | The pin function depends on bit $WKP_n$ in PMR5, bit PCR5n in PCR5, and bits SGS3 to SGS0 in LPCR. (n = 3 to 0) |
|---|---|

| SGS3 to SGS0 | 0*** or 1**0 | | | 1**1 |
|---|---|---|---|---|
| WKPn | 0 | | 1 | * |
| PCR5n | 0 | 1 | * | * |
| Pin function | $P5_n$ input pin | $P5_n$ output pin | $\overline{WKP}_n$ input pin | $SEG_{n+1}$ output pin |

Note: * Don't care

### 8.6.4 Pin States

Table 8.16 shows the port 5 pin states in each operating mode.

**Table 8.16 Port 5 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P5_7/\overline{WKP}_7/SEG_8$ to $P5_0/\overline{WKP}_0/SEG_1$ | High-impedance | Retains previous state | Retains previous state | High-impedance* | Retains previous state | Functional | Functional |

Note: * A high-level signal is output when the MOS pull-up is in the on state.

### 8.6.5 MOS Input Pull-Up

Port 5 has a built-in MOS input pull-up function that can be controlled by software. When a PCR5 bit is cleared to 0, setting the corresponding PUCR5 bit to 1 turns on the MOS pull-up for that pin. The MOS pull-up function is in the off state after a reset.

| $PCR5_n$ | 0 | | 1 |
|---|---|---|---|
| $PUCR5_n$ | 0 | 1 | * |
| MOS input pull-up | Off | On | Off |

Note: * Don't care
n = 7 to 0

## 8.7 Port 6

### 8.7.1 Overview

Port 6 is an 8-bit I/O port, configured as shown in figure 8.6.

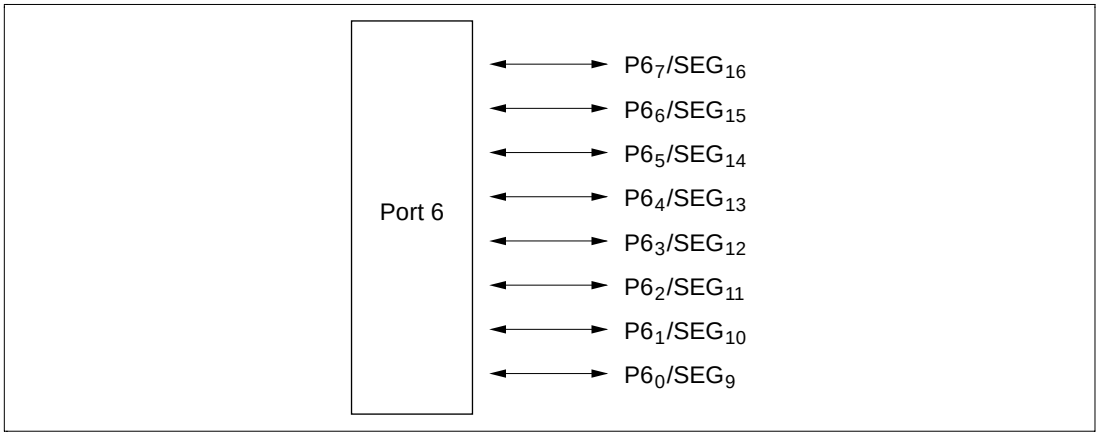



**Figure 8.6 Port 6 Pin Configuration**

### 8.7.2 Register Configuration and Description

Table 8.17 shows the port 6 register configuration.

**Table 8.17 Port 6 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 6 | PDR6 | R/W | H'00 | H'FFD9 |
| Port control register 6 | PCR6 | W | H'00 | H'FFE9 |
| Port pull-up control register 6 | PUCR6 | R/W | H'00 | H'FFE3 |

**Port Data Register 6 (PDR6)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P6_7$ | $P6_6$ | $P6_5$ | $P6_4$ | $P6_3$ | $P6_2$ | $P6_1$ | $P6_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR6 is an 8-bit register that stores data for port 6 pins $P6_7$ to $P6_0$. If port 6 is read while PCR6 bits are set to 1, the values stored in PDR6 are read, regardless of the actual pin states. If port 6 is read while PCR6 bits are cleared to 0, the pin states are read.

Upon reset, PDR6 is initialized to H'00.

**Port Control Register 6 (PCR6)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR6_7$ | $PCR6_6$ | $PCR6_5$ | $PCR6_4$ | $PCR6_3$ | $PCR6_2$ | $PCR6_1$ | $PCR6_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR6 is an 8-bit register for controlling whether each of the port 6 pins $P6_7$ to $P6_0$ functions as an input pin or output pin. Setting a PCR6 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR6 and in PDR6 are valid only when the corresponding pin is designated in bits SGS3 to SGS0 in LPCR as a general I/O pin.

Upon reset, PCR6 is initialized to H'00.

PCR6 is a write-only register. All bits are read as 1.

**Port Pull-Up Control Register 6 (PUCR6)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR6_7$ | $PUCR6_6$ | $PUCR6_5$ | $PUCR6_4$ | $PUCR6_3$ | $PUCR6_2$ | $PUCR6_1$ | $PUCR6_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PUCR6 controls whether the MOS pull-up of each port 6 pin $P6_7$–$P6_0$ is on or off. When a PCR6 bit is cleared to 0, setting the corresponding PUCR6 bit to 1 turns on the MOS pull-up for the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR6 is initialized to H'00.

### 8.7.3 Pin Functions

Table 8.18 shows the port 6 pin functions.

**Table 8.18 Port 6 Pin Functions**

<table>
<tr><th>Pin</th><th>Pin Functions and Selection Method</th></tr>
<tr><td>$P6_7$/$SEG_{16}$ to $P6_4$/$SEG_{13}$</td><td>The pin function depends on bit $PCR6_n$ in PCR6 and bits SGS3 to SGS0 in LPCR.<br>(n = 7 to 4)
<table>
<tr><td>SGS3 to SGS0</td><td colspan="2">00** or 010*</td><td>011* or 1***</td></tr>
<tr><td>PCR6n</td><td>0</td><td>1</td><td>*</td></tr>
<tr><td>Pin function</td><td>P6n input pin</td><td>P6n output pin</td><td>SEGn+9 output pin</td></tr>
</table></td></tr>
<tr><td>$P6_3$/$SEG_{12}$ to $P6_0$/$SEG_9$</td><td>The pin function depends on bit $PCR6_n$ in PCR6 and bits SGS3 to SGS0 in LPCR.<br>(n = 3 to 0)
<table>
<tr><td>SGS3 to SGS0</td><td colspan="2">00**, 010* or 0110</td><td>0111 or 1***</td></tr>
<tr><td>PCR6n</td><td>0</td><td>1</td><td>*</td></tr>
<tr><td>Pin function</td><td>P6n input pin</td><td>P6n output pin</td><td>SEGn+9 output pin</td></tr>
</table></td></tr>
</table>

Note: * Don’t care

### 8.7.4 Pin States

Table 8.19 shows the port 6 pin states in each operating mode.

**Table 8.19 Port 6 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P6_7/SEG_{16}$ to $P6_0/SEG_9$ | High-impedance | Retains previous state | Retains previous state | High-impedance* | Retains previous state | Functional | Functional |

Note: * A high-level signal is output when the MOS pull-up is in the on state.

### 8.7.5 MOS Input Pull-Up

Port 6 has a built-in MOS input pull-up function that can be controlled by software. When a PCR6 bit is cleared to 0, setting the corresponding PUCR6 bit to 1 turns on the MOS pull-up for that pin. The MOS pull-up function is in the off state after a reset.

| $PCR6_n$ | 0 | | 1 |
|---|---|---|---|
| $PUC6_n$ | 0 | 1 | * |
| MOS input pull-up | Off | On | Off |

Note: * Don't care
n = 7 to 0

## 8.8 Port 7

### 8.8.1 Overview

Port 7 is an 8-bit I/O port, configured as shown in figure 8.7.

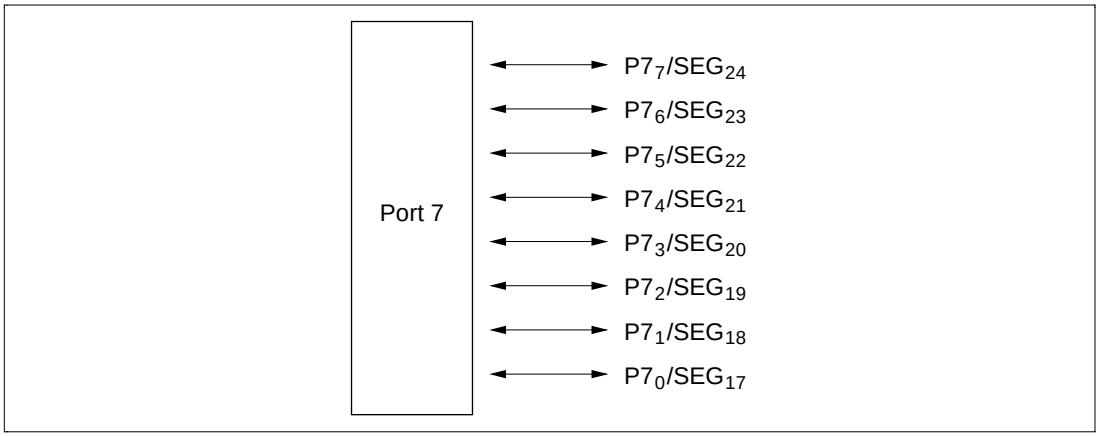



**Figure 8.7 Port 7 Pin Configuration**

### 8.8.2 Register Configuration and Description

Table 8.20 shows the port 7 register configuration.

**Table 8.20 Port 7 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 7 | PDR7 | R/W | H'00 | H'FFDA |
| Port control register 7 | PCR7 | W | H'00 | H'FFEA |

**Port Data Register 7 (PDR7)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | P7$_7$ | P7$_6$ | P7$_5$ | P7$_4$ | P7$_3$ | P7$_2$ | P7$_1$ | P7$_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR7 is an 8-bit register that stores data for port 7 pins P7$_7$ to P7$_0$. If port 7 is read while PCR7 bits are set to 1, the values stored in PDR7 are read, regardless of the actual pin states. If port 7 is read while PCR7 bits are cleared to 0, the pin states are read.

Upon reset, PDR7 is initialized to H'00.

**Port Control Register 7 (PCR7)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR7_7$ | $PCR7_6$ | $PCR7_5$ | $PCR7_4$ | $PCR7_3$ | $PCR7_2$ | $PCR7_1$ | $PCR7_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR7 is an 8-bit register for controlling whether each of the port 7 pins $P7_7$ to $P7_0$ functions as an input pin or output pin. Setting a PCR7 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR7 and in PDR7 are valid only when the corresponding pin is designated in bits SGS3 to SGS0 in LPCR as a general I/O pin.

Upon reset, PCR7 is initialized to H'00.

PCR7 is a write-only register. All bits are read as 1.

### 8.8.3 Pin Functions

Table 8.21 shows the port 7 pin functions.

**Table 8.21 Port 7 Pin Functions**

**Pin**: $P7_7/SEG_{24}$ to $P7_4/SEG_{21}$

**Pin Functions and Selection Method**: The pin function depends on bit $PCR7_n$ in PCR7 and bits SGS3 to SGS0 in LPCR.

(n = 7 to 4)

| SGS3 to SGS0 | 00** | | 01** or 1*** |
|---|---|---|---|
| PCR7n | 0 | 1 | * |
| Pin function | P7n input pin | P7n output pin | SEGn+17 output pin |

**Pin**: $P7_3/SEG_{20}$ to $P7_0/SEG_{17}$

**Pin Functions and Selection Method**: The pin function depends on bit $PCR7_n$ in PCR7 and bits SGS3 to SGS0 in LPCR.

(n = 3 to 0)

| SGS3 to SGS0 | 00** or 0100 | | 0101, 011* or 1**** |
|---|---|---|---|
| PCR7n | 0 | 1 | * |
| Pin function | P7n input pin | P7n output pin | SEGn+17 output pin |

Note: * Don't care

### 8.8.4 Pin States

Table 8.22 shows the port 7 pin states in each operating mode.

**Table 8.22 Port 7 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P7_7/SEG_{24}$ to $P7_0/SEG_{17}$ | High-impedance | Retains previous state | Retains previous state | High-impedance | Retains previous state | Functional | Functional |

## 8.9 Port 8

### 8.9.1 Overview

Port 8 is an 8-bit I/O port configured as shown in figure 8.9.

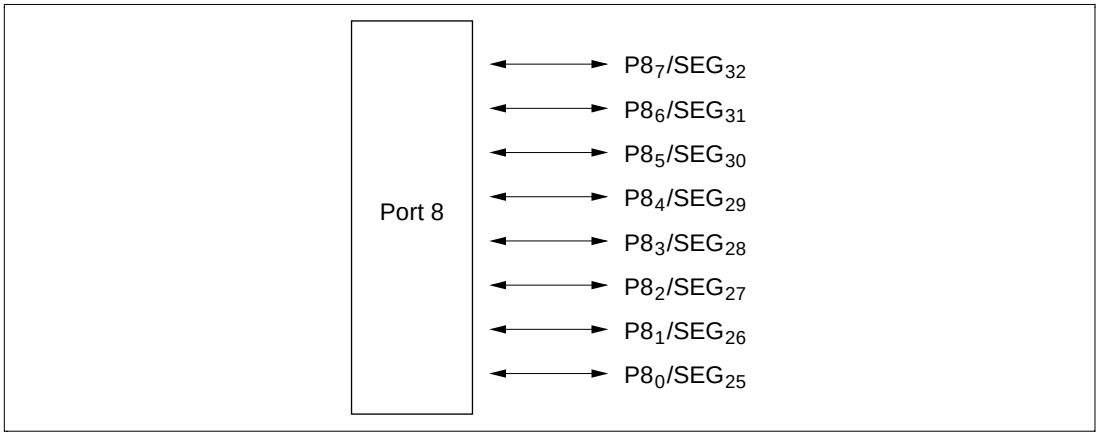



**Figure 8.8 Port 8 Pin Configuration**

### 8.9.2 Register Configuration and Description

Table 8.23 shows the port 8 register configuration.

**Table 8.23 Port 8 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 8 | PDR8 | R/W | H'00 | H'FFDB |
| Port control register 8 | PCR8 | W | H'00 | H'FFEB |

**Port Data Register 8 (PDR8)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P8_7$ | $P8_6$ | $P8_5$ | $P8_4$ | $P8_3$ | $P8_2$ | $P8_1$ | $P8_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR8 is an 8-bit register that stores data for port 8 pins $P8_7$ to $P8_0$. If port 8 is read while PCR8 bits are set to 1, the values stored in PDR8 are read, regardless of the actual pin states. If port 8 is read while PCR8 bits are cleared to 0, the pin states are read.

Upon reset, PDR8 is initialized to H'00.

**Port Control Register 8 (PCR8)**

| | $PCR8_7$ | $PCR8_6$ | $PCR8_5$ | $PCR8_4$ | $PCR8_3$ | $PCR8_2$ | $PCR8_1$ | $PCR8_0$ |
|---|---|---|---|---|---|---|---|---|
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR8 is an 8-bit register for controlling whether each of the port 8 pins $P8_7$ to $P8_0$ functions as an input or output pin. Setting a PCR8 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR8 and in PDR8 are valid only when the corresponding pin is designated in bits SGS3 to SGS0 in LPCR as a general I/O pin.

Upon reset, PCR8 is initialized to H'00.

PCR8 is a write-only register. All bits are read as 1.

### 8.9.3 Pin Functions

Table 8.24 shows the port 8 pin functions.

**Table 8.24 Port 8 Pin Functions**

| Pin | Pin Functions and Selection Method |
|---|---|
| $P8_7/SEG_{32}$ to $P8_4/SEG_{29}$ | The pin function depends on bit $PCR8_n$ in PCR8 and bits SGS3 to SGS0 in LPCR.<br>(n = 7 to 4) |

| SGS3 to SGS0 | 000* | | 001*, 01** or 1*** |
|---|---|---|---|
| PCR8n | 0 | 1 | * |
| Pin function | P8n input pin | P8n output pin | SEGn+25 output pin |

| $P8_3/SEG_{28}$ to $P8_0/SEG_{25}$ | The pin function depends on bit $PCR8_n$ in PCR8 and bits SGS3 to SGS0 in LPCR.<br>(n = 3 to 0) |
|---|---|

| SGS3 to SGS0 | 000* or 0010 | | 0011, 01** or 1*** |
|---|---|---|---|
| PCR8n | 0 | 1 | * |
| Pin function | P8n input pin | P8n output pin | SEGn+25 output pin |

Note: * Don't care

### 8.9.4 Pin States

Table 8.25 shows the port 8 pin states in each operating mode.

**Table 8.25 Port 8 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P8_7/SEG_{32}$ to $P8_0/SEG_{25}$ | High-impedance | Retains previous state | Retains previous state | High-impedance | Retains previous state | Functional | Functional |

## 8.10 Port 9

### 8.10.1 Overview

Port 9 is an 8-bit I/O port configured as shown in figure 8.9.

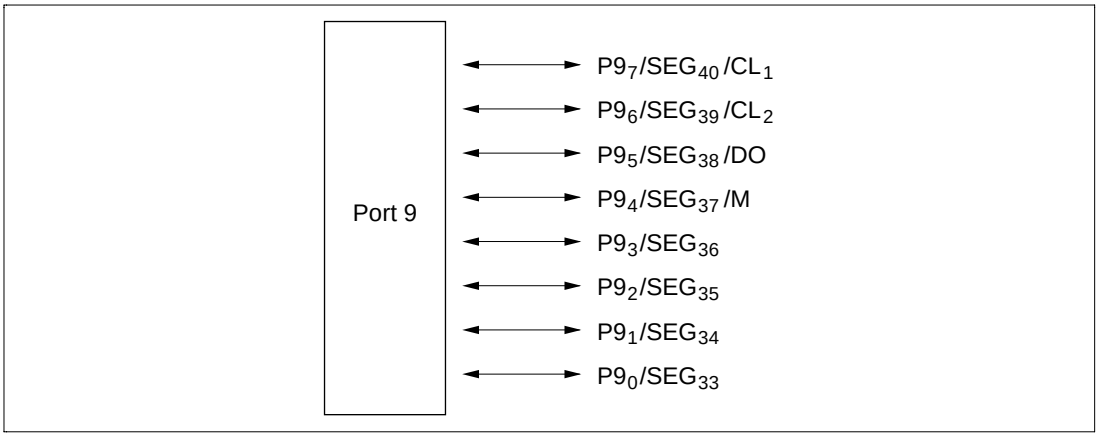



**Figure 8.9 Port 9 Pin Configuration**

### 8.10.2 Register Configuration and Description

Table 8.26 shows the port 9 register configuration.

**Table 8.26 Port 9 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register 9 | PDR9 | R/W | H'00 | H'FFDC |
| Port control register 9 | PCR9 | W | H'00 | H'FFEC |

**Port Data Register 9 (PDR9)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | P9$_7$ | P9$_6$ | P9$_5$ | P9$_4$ | P9$_3$ | P9$_2$ | P9$_1$ | P9$_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

PDR9 is an 8-bit register that stores data for port 9 pins P9$_7$ to P9$_0$. If port 9 is read while PCR9 bits are set to 1, the values stored in PDR9 are read, regardless of the actual pin states. If port 9 is read while PCR9 bits are cleared to 0, the pin states are read.

Upon reset, PDR9 is initialized to H'00.

**Port Control Register 9 (PCR9)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR9_7$ | $PCR9_6$ | $PCR9_5$ | $PCR9_4$ | $PCR9_3$ | $PCR9_2$ | $PCR9_1$ | $PCR9_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

PCR9 is an 8-bit register for controlling whether each of the port 9 pins $P9_7$ to $P9_0$ functions as an input or output pin. Setting a PCR9 bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCR9 and in PDR9 are valid only when the corresponding pin is designated in bits SGS3 to SGS0 in LPCR as a general I/O pin.

Upon reset, PCR9 is initialized to H'00.

PCR9 is a write-only register. All bits are read as 1.

### 8.10.3 Pin Functions

Table 8.27 shows the port 9 pin functions.

**Table 8.27 Port 9 Pin Functions**

| Pin | Pin Functions and Selection Method | | | |
|---|---|---|---|---|
| $P9_7/SEG_{40}/CL_1$ | The pin function depends on bit $PCR9_7$ in PCR9, and bits SGX and SGS3 to SGS0 in LPCR. | | | |
| | SGS3 to SGS0 | 0000 | Not 0000 | * |
| | SGX | 0 | 0 | 1 |
| | $PCR9_7$ | 0 / 1 | * | * |
| | Pin function | $P9_7$ input pin / $P9_7$ output pin | $SEG_{40}$ output pin | $CL_1$ output pin |
| $P9_6/SEG_{39}/CL_2$ | The pin function depends on bit $PCR9_6$ in PCR9, and bits SGX and SGS3 to SGS0 in LPCR. | | | |
| | SGS3 to SGS0 | 0000 | Not 0000 | * |
| | SGX | 0 | 0 | 1 |
| | $PCR9_6$ | 0 / 1 | * | * |
| | Pin function | $P9_6$ input pin / $P9_6$ output pin | $SEG_{39}$ output pin | $CL_2$ output pin |

Note: * Don’t care

**Table 8.27 Port 9 Pin Functions (cont)**

| Pin | Pin Functions and Selection Method |
|---|---|
| $P9_5/SEG_{38}/DO$ | The pin function depends on bit $PCR9_5$ in PCR9, and bits SGX and SGS3 to SGS0 in LPCR. |

| SGS3 to SGS0 | 0000 | | Not 0000 | * |
|---|---|---|---|---|
| SGX | 0 | | 0 | 1 |
| $PCR9_5$ | 0 | 1 | * | * |
| Pin function | $P9_5$ input pin | $P9_5$ output pin | $SEG_{38}$ output pin | DO output pin |

| $P9_4/SEG_{37}/M$ | The pin function depends on bit $PCR9_4$ in PCR9, and bits SGX and SGS3 to SGS0 in LPCR. |
|---|---|

| SGS3 to SGS0 | 0000 | | Not 0000 | * |
|---|---|---|---|---|
| SGX | 0 | | 0 | 1 |
| $PCR9_4$ | 0 | 1 | * | * |
| Pin function | $P9_4$ input pin | $P9_4$ output pin | $SEG3_7$ output pin | M output pin |

| $9_3/SEG_{36}$ to $P9_0/SEG_{33}$ | The pin function depends on bit $PCR9_n$ in PCR9 and bits SGS3 to SGS0 in LPCR. |
|---|---|

(n = 3 to 0)

| SGS3 to SGS0 | 0000 | | Not 0000 |
|---|---|---|---|
| $PCR9_n$ | 0 | 1 | * |
| Pin function | $P9_n$ input pin | $P9_n$ output pin | $SEG_{n+33}$ output pin |

Note: * Don't care

### 8.10.4 Pin States

Table 8.28 shows the port 9 pin states in each operating mode.

**Table 8.28 Port 9 Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P9_7/SEG_{40}/CL_1$<br>$P9_6/SEG_{39}/CL_2$<br>$P9_5/SEG_{38}/DO$<br>$P9_4/SEG_{37}/M$<br>$P9_3/SEG_{36}$ to $P9_0/SEG_{33}$ | High-impedance | Retains previous state | Retains previous state | High-impedance | Retains previous state | Functional | Functional |

## 8.11 Port A

### 8.11.1 Overview

Port A is a 4-bit I/O port, configured as shown in figure 8.10.

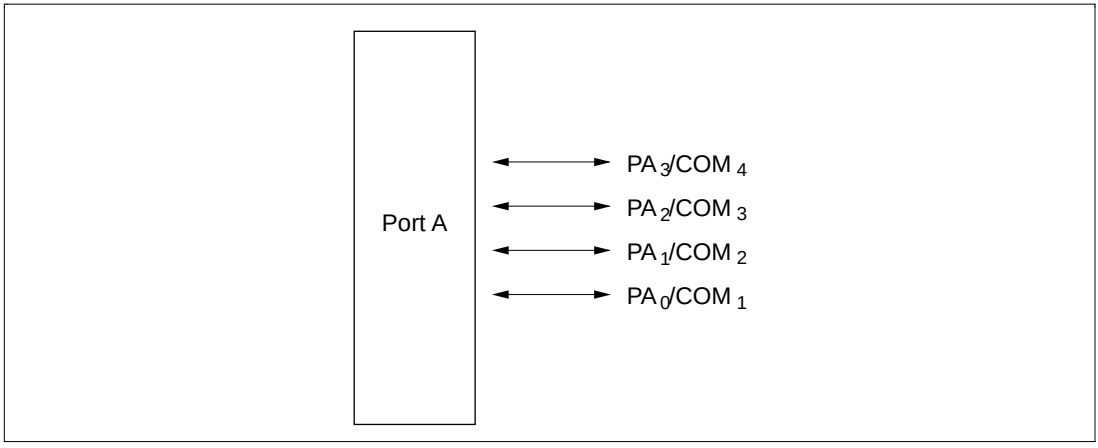



**Figure 8.10 Port A Pin Configuration**

### 8.11.2 Register Configuration and Description

Table 8.29 shows the port A register configuration.

**Table 8.29 Port A Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Port data register A | PDRA | R/W | H'F0 | H'FFDD |
| Port control register A | PCRA | W | H'F0 | H'FFED |

**Port Data Register A (PDRA)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $PA_3$ | $PA_2$ | $PA_1$ | $PA_0$ |
| Initial value | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | R/W | R/W | R/W | R/W |

PDRA is an 8-bit register that stores data for port A pins $PA_3$ to $PA_0$. If port A is read while PCRA bits are set to 1, the values stored in PDRA are read, regardless of the actual pin states. If port A is read while PCRA bits are cleared to 0, the pin states are read.

Upon reset, PDRA is initialized to H'F0.

**Port Control Register A (PCRA)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $PCRA_3$ | $PCRA_2$ | $PCRA_1$ | $PCRA_0$ |
| Initial value | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | W | W | W | W |

PCRA is an 8-bit register for controlling whether each of the port A pins $PA_3$ to $PA_0$ functions as an input or output pin. Setting a PCRA bit to 1 makes the corresponding pin an output pin, while clearing the bit to 0 makes the pin an input pin. The settings in PCRA and in PDRA are valid only when the corresponding pin is designated in LPCR as a general I/O pin.

Upon reset, PCRA is initialized to H'F0.

PCRA is a write-only register. All bits are read as 1.

### 8.11.3 Pin Functions

Table 8.30 gives the port A pin functions.

**Table 8.30 Port A Pin Functions**

| Pin | Pin Functions and Selection Method |
|---|---|
| $PA_3/COM_4$ | The pin function depends on bit $PCRA_3$ in PCRA and bits DTS1, DTS0, CMX, SGX, and SGS3 to SGS0 in LPCR. |

| CMX | * | 0 | | * | 0 | | 1 | | * | |
|---|---|---|---|---|---|---|---|---|---|---|
| DTS1, DTS0 | ** | Not 11 | | ** | Not 11 | | Not 11 | | 11 | |
| SGX | 0 | 1 | * | 0 | 1 | * | 1 | * | 1 | * |
| SGS3 to SGS0 | 0000 | | Not 0000 | 0000 | | Not 0000 | 0000 | Not 0000 | 0000 | Not 0000 |
| $PCRA_3$ | 0 | | | 1 | | | * | | | |
| Pin function | $PA_3$ input pin | | | $PA_3$ output pin | | | $COM_4$ output pin | | | |

| $PA_2/COM_3$ | The pin function depends on bit $PCRA_2$ in PCRA and bits DTS1, DTS0, CMX, SGX, and SGS3 to SGS0 in LPCR. |
|---|---|

| CMX | * | 0 | | * | 0 | | 1 | | * | |
|---|---|---|---|---|---|---|---|---|---|---|
| DTS1, DTS0 | ** | 00 or 01 | | ** | 00 or 01 | | 00 or 01 | | Not 00 or 01 | |
| SGX | 0 | 1 | * | 0 | 1 | * | 1 | * | 1 | * |
| SGS3 to SGS0 | 0000 | | Not 0000 | 0000 | | Not 0000 | 0000 | Not 0000 | 0000 | Not 0000 |
| $PCRA_2$ | 0 | | | 1 | | | * | | | |
| Pin function | $PA_2$ input pin | | | $PA_2$ output pin | | | $COM_3$ output pin | | | |

| $PA_1/COM_2$ | The pin function depends on bit $PCRA_1$ in PCRA and bits DTS1, DTS0, CMX, SGX, and SGS3 to SGS0 in LPCR. |
|---|---|

| CMX | * | 0 | | * | 0 | | 1 | | * | |
|---|---|---|---|---|---|---|---|---|---|---|
| DTS1, DTS0 | ** | 00 | | ** | 00 | | 00 | | Not 00 | |
| SGX | 0 | 1 | * | 0 | 1 | * | 1 | * | 1 | * |
| SGS3 to SGS0 | 0000 | | Not 0000 | 0000 | | Not 0000 | 0000 | Not 0000 | 0000 | Not 0000 |
| $PCRA_1$ | 0 | | | 1 | | | * | | | |
| Pin function | $PA_1$ input pin | | | $PA_1$ output pin | | | $COM_2$ output pin | | | |

Note: * Don't care

**Table 8.30 Port A Pin Functions (cont)**

| Pin | Pin Functions and Selection Method |
|---|---|
| $PA_0/COM_1$ | The pin function depends on bit PCRA0 in PCRA, and bits SGX and SGS3 to SGS0 in LPCR. |

| SGS3 to SGS0 | 0000 | | 0000 | Not 0000 |
|---|---|---|---|---|
| SGX | 0 | | 1 | * |
| $PCRA_0$ | 0 | 1 | * | |
| Pin function | $PA_0$ input pin | $PA_0$ output pin | $COM_1$ output pin | |

Note: * Don't care

### 8.11.4 Pin States

Table 8.31 shows the port A pin states in each operating mode.

**Table 8.31 Port A Pin States**

| Pins | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $PA_3/COM_4$<br>$PA_2/COM_3$<br>$PA_1/COM_2$<br>$PA_0/COM_1$ | High-impedance | Retains previous state | Retains previous state | High-impedance | Retains previous state | Functional | Functional |

## 8.12 Port B

### 8.12.1 Overview

Port B is an 8-bit input-only port, configured as shown in figure 8.11.

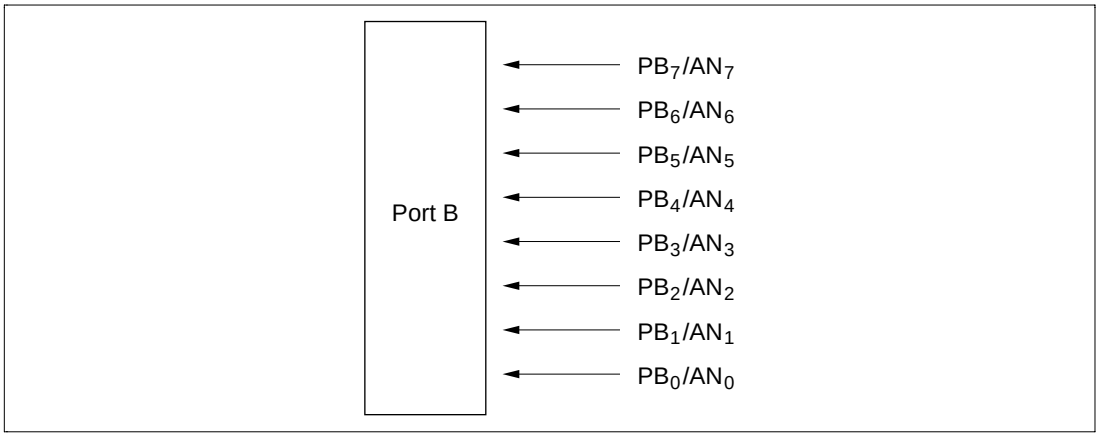



**Figure 8.11 Port B Pin Configuration**

### 8.12.2 Register Configuration and Description

Table 8.32 shows the port B register configuration.

**Table 8.32 Port B Register**

| Name | Abbrev. | R/W | Address |
|---|---|---|---|
| Port data register B | PDRB | R | H'FFDE |

**Port Data Register B (PDRB)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PB_7$ | $PB_6$ | $PB_5$ | $PB_4$ | $PB_3$ | $PB_2$ | $PB_1$ | $PB_0$ |
| Read/Write | R | R | R | R | R | R | R | R |

Reading PDRB always gives the pin states. However, if a port B pin is selected as an analog input channel for the A/D converter by AMR bits CH3 to CH0, that pin reads 0 regardless of the input voltage.

## 8.13 Port C

### 8.13.1 Overview

Port C is a 4-bit input-only port, configured as shown in figure 8.12.

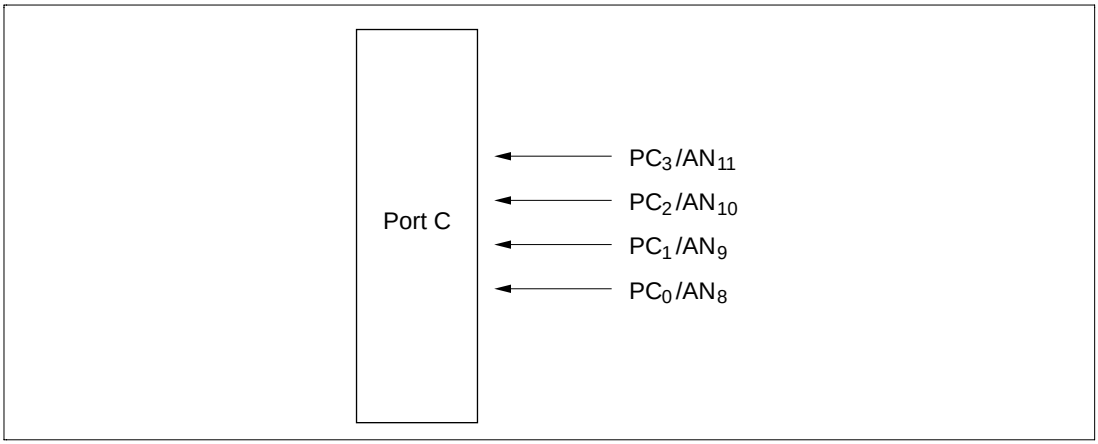



**Figure 8.12 Port C Pin Configuration**

### 8.13.2 Register Configuration and Description

Table 8.33 shows the port C register configuration.

**Table 8.33 Port C Register**

| Name | Abbrev. | R/W | Address |
|---|---|---|---|
| Port data register C | PDRC | R | H'FFDF |

**Port Data Register C (PDRC)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | PC3 | PC2 | PC1 | PC0 |
| Read/Write | — | — | — | — | R | R | R | R |

Reading PDRC always gives the pin states. However, if a port C pin is selected as an analog input channel for the A/D converter by AMR bits CH3 to CH0, that pin reads 0 regardless of the input voltage.

# Section 9 Timers

## 9.1 Overview

The H8/3834 Series provides five timers (timers A, B, C, F, and G) on-chip.

Table 9.1 outlines the functions of timers A, B, C, F, and G.

**Table 9.1 Timer Functions**

| Name | Functions | Internal Clock | Event Input Pin | Waveform Output Pin | Remarks |
|---|---|---|---|---|---|
| Timer A | • 8-bit timer<br>• Interval timer | $\phi$/8 to $\phi$/8192 (8 choices) | — | — | |
| | • 8-bit timer<br>• Time base | $\phi_w$/128 (choice of 4 overflow periods) | — | — | |
| | • 8-bit timer<br>• Clock output | $\phi$/4 to $\phi$/32, $\phi_w$/4 to $\phi_w$/32 (8 choices) | — | TMOW | |
| Timer B | • 8-bit timer<br>• Interval timer<br>• Event counter | $\phi$/4 to $\phi$/8192 (7 choices) | TMIB | — | |
| Timer C | • 8-bit timer<br>• Interval timer<br>• Event counter<br>• Choice of up- or down-counting | $\phi$/4 to $\phi$/8192, $\phi_w$/4 (7 choices) | TMIC | — | Counting direction can be controlled by software or hardware |
| Timer F | • 16-bit timer<br>• Event counter<br>• Can be used as two independent 8-bit timers<br>• Output compare | $\phi$/2 to $\phi$/32 (4 choices) | TMIF | TMOFL<br>TMOFH | |
| Timer G | • 8-bit timer<br>• Input capture<br>• Interval timer | $\phi$/2 to $\phi$/64, $\phi_w$/2 (4 choices) | TMIG | — | • Counter clear designation possible<br>• Built-in noise canceller circuit for input capture |

## 9.2 Timer A

### 9.2.1 Overview

Timer A is an 8-bit timer with interval timing and real-time clock time-base functions. The clock time-base function is available when a 32.768-kHz crystal oscillator is connected. A clock signal divided from 32.768 kHz or from the system clock can be output at the TMOW pin.

**Features**

Features of timer A are given below.

- Choice of eight internal clock sources ($\phi$/8192, $\phi$/4096, $\phi$/2048, $\phi$/512, $\phi$/256, $\phi$/128, $\phi$/32, $\phi$/8).
- Choice of four overflow periods (1 s, 0.5 s, 0.25 s, 31.25 ms) when timer A is used as a clock time base (using a 32.768 kHz crystal oscillator).
- An interrupt is requested when the counter overflows.
- Any of eight clock signals can be output from pin TMOW: 32.768 kHz divided by 32, 16, 8, or 4 (1 kHz, 2 kHz, 4 kHz, 8 kHz), or the system clock divided by 32, 16, 8, or 4.

**Block Diagram**

Figure 9.1 shows a block diagram of timer A.

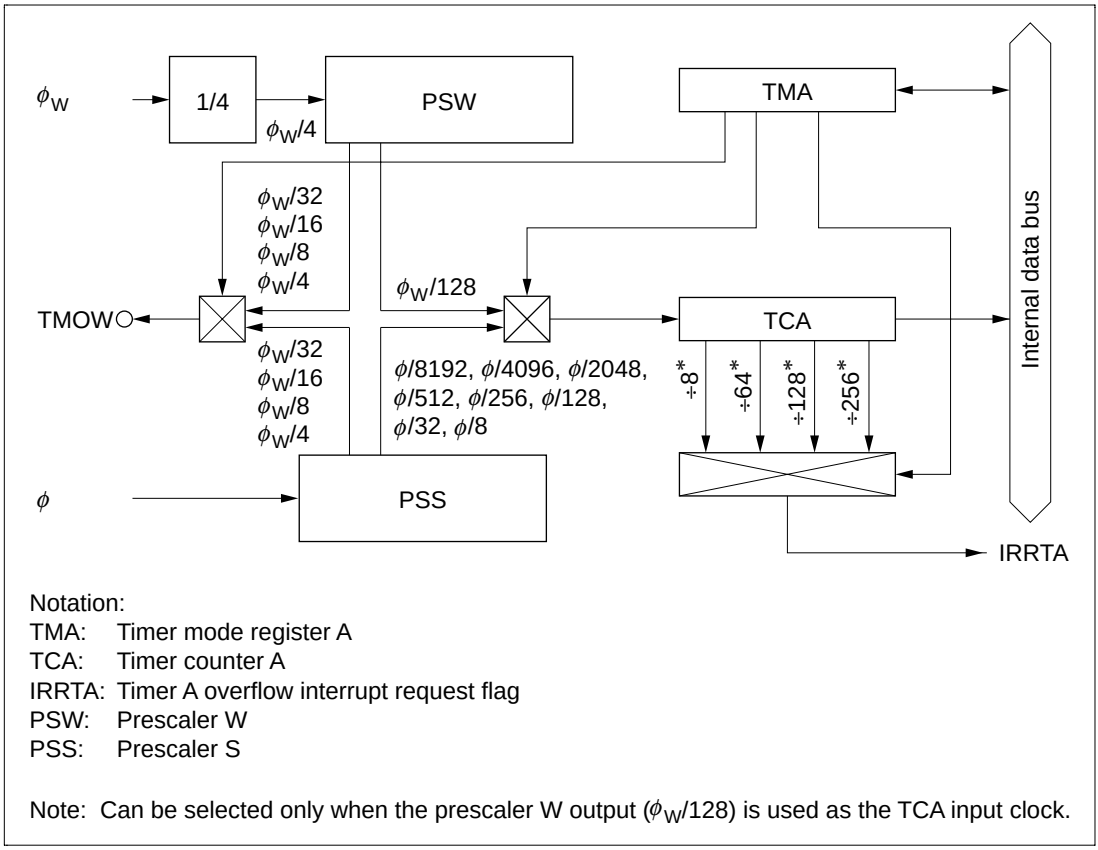



**Figure 9.1 Block Diagram of Timer A**

**Pin Configuration**

Table 9.2 shows the timer A pin configuration.

**Table 9.2 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| Clock output | TMOW | Output | Output of waveform generated by timer A output circuit |

**Register Configuration**

Table 9.3 shows the register configuration of timer A.

**Table 9.3 Timer A Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Timer mode register A | TMA | R/W | H'10 | H'FFB0 |
| Timer counter A | TCA | R | H'00 | H'FFB1 |

### 9.2.2 Register Descriptions

**Timer Mode Register A (TMA)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TMA7 | TMA6 | TMA5 | — | TMA3 | TMA2 | TMA1 | TMA0 |
| Initial value | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | — | R/W | R/W | R/W | R/W |

TMA is an 8-bit read/write register for selecting the prescaler, input clock, and output clock.

Upon reset, TMA is initialized to H'10.

**Bits 7 to 5—Clock Output Select (TMA7 to TMA5):** Bits 7 to 5 choose which of eight clock signals is output at the TMOW pin. The system clock divided by 32, 16, 8, or 4 can be output in active mode and sleep mode. A 32.768 kHz signal divided by 32, 16, 8, or 4 can be output in active mode, sleep mode, and subactive mode.

| Bit 7: TMA7 | Bit 6: TMA6 | Bit 5: TMA5 | Clock Output | |
|---|---|---|---|---|
| 0 | 0 | 0 | $\phi$/32 | (initial value) |
| | | 1 | $\phi$/16 | |
| | 1 | 0 | $\phi$/8 | |
| | | 1 | $\phi$/4 | |
| 1 | 0 | 0 | $\phi_w$/32 | |
| | | 1 | $\phi_w$/16 | |
| | 1 | 0 | $\phi_w$/8 | |
| | | 1 | $\phi_w$/4 | |

**Bit 4—Reserved Bit:** Bit 4 is reserved; it is always read as 1, and cannot be modified.

**Bits 3 to 0—Internal Clock Select (TMA3 to TMA0):** Bits 3 to 0 select the clock input to TCA.

| | | | | Description | | |
|---|---|---|---|---|---|---|
| **Bit 3: TMA3** | **Bit 2: TMA2** | **Bit 1: TMA1** | **Bit 0: TMA0** | **Prescaler and Divider Ratio or Overflow Period** | | **Function** |
| 0 | 0 | 0 | 0 | PSS, $\phi$/8192 | (initial value) | Interval timer |
| | | | 1 | PSS, $\phi$/4096 | | |
| | | 1 | 0 | PSS, $\phi$/2048 | | |
| | | | 1 | PSS, $\phi$/512 | | |
| | 1 | 0 | 0 | PSS, $\phi$/256 | | |
| | | | 1 | PSS, $\phi$/128 | | |
| | | 1 | 0 | PSS, $\phi$/32 | | |
| | | | 1 | PSS, $\phi$/8 | | |
| 1 | 0 | 0 | 0 | PSW, 1 s | | Clock time base |
| | | | 1 | PSW, 0.5 s | | |
| | | 1 | 0 | PSW, 0.25 s | | |
| | | | 1 | PSW, 0.03125 s | | |
| | 1 | 0 | 0 | PSW and TCA are reset | | |
| | | | 1 | | | |
| | | 1 | 0 | | | |
| | | | 1 | | | |

**Timer Counter A (TCA)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCA7 | TCA6 | TCA5 | TCA4 | TCA3 | TCA2 | TCA1 | TCA0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

TCA 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 TMA3 to TMA0 in timer mode register A (TMA). TCA values can be read by the CPU in active mode, but cannot be read in subactive mode. When TCA overflows, the IRRTA bit in interrupt request register 1 (IRR1) is set to 1.

TCA is cleared by setting bits TMA3 and TMA2 of TMA to 11.

Upon reset, TCA is initialized to H'00.

### 9.2.3 Timer Operation

**Interval Timer Operation:** When bit TMA3 in timer mode register A (TMA) is cleared to 0, timer A functions as an 8-bit interval timer.

Upon reset, TCA is cleared to H'00 and bit TMA3 is cleared to 0, so up-counting and interval timing resume immediately. The clock input to timer A is selected by bits TMA2 to TMA0 in TMA; any of eight internal clock signals output by prescaler S can be selected.

After the count value in TCA reaches H'FF, the next clock signal input causes timer A to overflow, setting bit IRRTA to 1 in interrupt request register 1 (IRR1). If IENTA = 1 in interrupt enable register 1 (IENR1), a CPU interrupt is requested.*

At overflow, TCA returns to H'00 and starts counting up again. In this mode timer A functions as an interval timer that generates an overflow output at intervals of 256 input clock pulses.

Note: * For details on interrupts, see 3.3, Interrupts.

**Real-Time Clock Time Base Operation:** When bit TMA3 in TMA is set to 1, timer A functions as a real-time clock time base by counting clock signals output by prescaler W.

The overflow period of timer A is set by bits TMA1 and TMA0 in TMA. A choice of four periods is available. In time base operation (TMA3 = 1), setting bit TMA2 to 1 clears both TCA and prescaler W to their initial values of H'00.

**Clock Output:** Setting bit TMOW in port mode register 1 (PMR1) to 1 causes a clock signal to be output at pin TMOW. Eight different clock output signals can be selected by means of bits TMA7 to TMA5 in TMA. The system clock divided by 32, 16, 8, or 4 can be output in active mode and sleep mode. A 32.768 kHz signal divided by 32, 16, 8, or 4 can be output in active mode, sleep mode, and subactive mode.

### 9.2.4 Timer A Operation States

Table 9.4 summarizes the timer A operation states.

**Table 9.4 Timer A Operation States**

| Operation Mode | | Reset | Active | Sleep | Watch | Sub-active | Sub-sleep | Standby |
|---|---|---|---|---|---|---|---|---|
| TCA | Interval | Reset | Functions | Functions | Halted | Halted | Halted | Halted |
| | Clock time base | Reset | Functions | Functions | Functions | Functions | Functions | Halted |
| TMA | | Reset | Functions | Retained | Retained | Functions | Retained | Retained |

Note: When real-time clock time base function is selected as the internal clock of TCA in active mode or sleep mode, the internal clock is not synchronous with the system clock, so it is synchronized by a synchronizing circuit. This may result in a maximum error of 1/$\phi$ (s) in the count cycle.

## 9.3 Timer B

### 9.3.1 Overview

Timer B is an 8-bit timer that increments each time a clock pulse is input. This timer has two operation modes, interval and auto reload.

**Features**

Features of timer B are given below.

- Choice of seven internal clock sources ($\phi$/8192, $\phi$/2048, $\phi$/512, $\phi$/256, $\phi$/64, $\phi$/16, $\phi$/4) or an external clock (can be used to count external events).
- An interrupt is requested when the counter overflows.

**Block Diagram**

Figure 9.2 shows a block diagram of timer B.

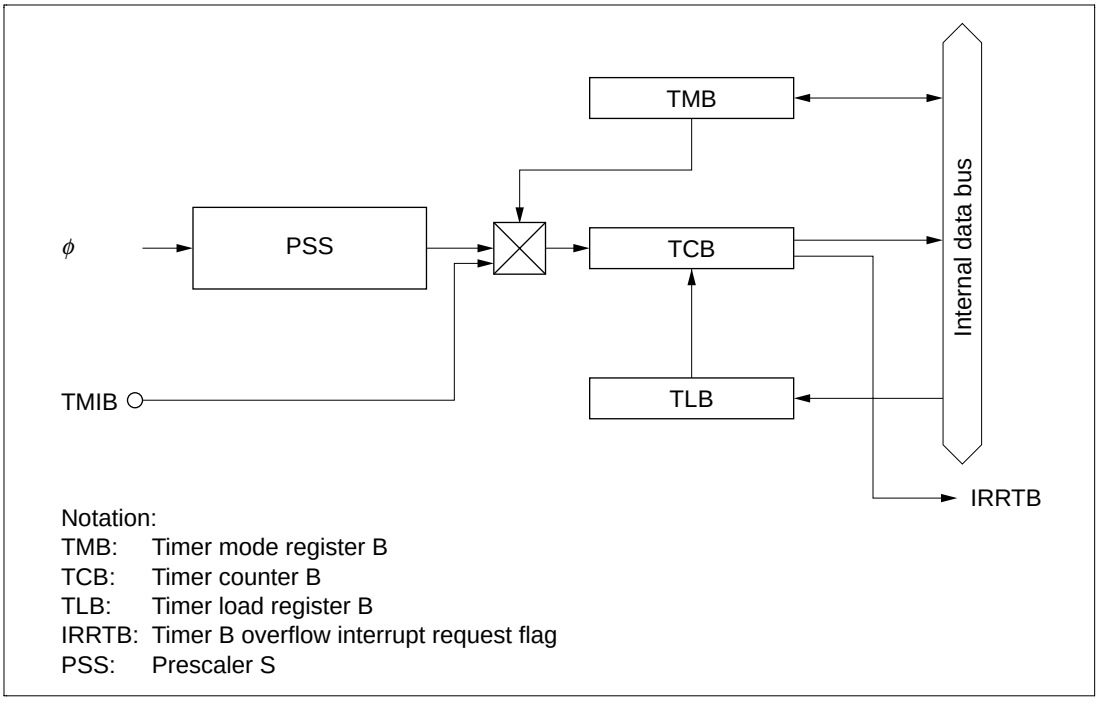



**Figure 9.2 Block Diagram of Timer B**

### Pin Configuration

Table 9.5 shows the timer B pin configuration.

**Table 9.5 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| Timer B event input | TMIB | Input | Event input to TCB |

### Register Configuration

Table 9.6 shows the register configuration of timer B.

**Table 9.6 Timer B Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Timer mode register B | TMB | R/W | H'78 | H'FFB2 |
| Timer counter B | TCB | R | H'00 | H'FFB3 |
| Timer load register B | TLB | W | H'00 | H'FFB3 |

### 9.3.2 Register Descriptions

**Timer Mode Register B (TMB)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TMB7 | — | — | — | — | TMB2 | TMB1 | TMB0 |
| Initial value | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | R/W | — | — | — | — | R/W | R/W | R/W |

TMB is an 8-bit read/write register for selecting the auto-reload function and input clock.

Upon reset, TMB is initialized to H'78.

**Bit 7—Auto-Reload Function Select (TMB7):** Bit 7 selects whether timer B is used as an interval timer or auto-reload timer.

| Bit 7: TMB7 | Description | |
|---|---|---|
| 0 | Interval timer function selected | (initial value) |
| 1 | Auto-reload function selected | |

**Bits 6 to 3—Reserved Bits:** Bits 6 to 3 are reserved; they always read 1, and cannot be modified.

**Bits 2 to 0—Clock Select (TMB2 to TMB0):** Bits 2 to 0 select the clock input to TCB. For external event counting, either the rising or falling edge can be selected.

| Bit 2: TMB2 | Bit 1: TMB1 | Bit 0: TMB0 | Description | |
|---|---|---|---|---|
| 0 | 0 | 0 | Internal clock: $\phi$/8192 | (initial value) |
| | | 1 | Internal clock: $\phi$/2048 | |
| | 1 | 0 | Internal clock: $\phi$/512 | |
| | | 1 | Internal clock: $\phi$/256 | |
| 1 | 0 | 0 | Internal clock: $\phi$/64 | |
| | | 1 | Internal clock: $\phi$/16 | |
| | 1 | 0 | Internal clock: $\phi$/4 | |
| | | 1 | External event (TMIB): rising or falling edge* | |

Note: * The edge of the external event signal is selected by bit IEG1 in the IRQ edge select register (IEGR). See 3.3.2, Interrupt Control Registers, for details on the IRQ edge select register. Be sure to set bit IRQ1 in port mode register 1 (PMR1) to 1 before setting bits TMB2 to TMB0 to 111.

**Timer Counter B (TCB)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCB7 | TCB6 | TCB5 | TCB4 | TCB3 | TCB2 | TCB1 | TCB0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

TCB is an 8-bit read-only up-counter, which is incremented by internal clock or external event input. The clock source for input to this counter is selected by bits TMB2 to TMB0 in timer mode register B (TMB). TCB values can be read by the CPU at any time.

When TCB overflows from H'FF to H'00 or to the value set in TLB, the IRRTB bit in interrupt request register 2 (IRR2) is set to 1.

TCB is allocated to the same address as timer load register B (TLB).

Upon reset, TCB is initialized to H'00.

**Timer Load Register B (TLB)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TLB7 | TLB6 | TLB5 | TLB4 | TLB3 | TLB2 | TLB1 | TLB0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

TLB is an 8-bit write-only register for setting the reload value of timer counter B.

When a reload value is set in TLB, the same value is loaded into timer counter B (TCB) as well, and TCB starts counting up from that value. When TCB overflows during operation in auto-reload mode, the TLB value is loaded into TCB. Accordingly, overflow periods can be set within the range of 1 to 256 input clocks.

The same address is allocated to TLB as to TCB.

Upon reset, TLB is initialized to H'00.

### 9.3.3 Timer Operation

**Interval timer Operation:** When bit TMB7 in timer mode register B (TMB) is cleared to 0, timer B functions as an 8-bit interval timer.

Upon reset, TCB is cleared to H'00 and bit TMB7 is cleared to 0, so up-counting and interval timing resume immediately. The clock input to timer B is selected from seven internal clock signals output by prescaler S, or an external clock input at pin TMIB. The selection is made by bits TMB2 to TMB0 of TMB.

After the count value in TCB reaches H'FF, the next clock signal input causes timer B to overflow, setting bit IRRTB to 1 in interrupt request register 2 (IRR2). If IENTB = 1 in interrupt enable register 2 (IENR2), a CPU interrupt is requested.*

At overflow, TCB returns to H'00 and starts counting up again.

During interval timer operation (TMB7 = 0), when a value is set in timer load register B (TLB), the same value is set in TCB.

Note: * For details on interrupts, see 3.3, Interrupts.

**Auto-Reload Timer Operation:** Setting bit TMB7 in TMB to 1 causes timer B to function as an 8-bit auto-reload timer. When a reload value is set in TLB, the same value is loaded into TCB, becoming the value from which TCB starts its count.

After the count value in TCB reaches H'FF, the next clock signal input causes timer B to overflow. The TLB value is then loaded into TCB, 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 TLB value.

The clock sources and interrupts in auto-reload mode are the same as in interval mode.

In auto-reload mode (TMB7 = 1), when a new value is set in TLB, the TLB value is also set in TCB.

**Event Counter Operation:** Timer B can operate as an event counter, counting rising or falling edges of an external event signal input at pin TMIB. External event counting is selected by setting bits TMB2 to TMB0 in timer mode register B to all 1s (111).

When timer B is used to count external event input, bit IRQ1 in port mode register 1 (PMR1) should be set to 1, and bit IEN1 in interrupt enable register 1 (IENR1) should be cleared to 0 to disable $IRQ_1$ interrupt requests.

### 9.3.4 Timer B Operation States

Table 9.7 summarizes the timer B operation states.

**Table 9.7 Timer B Operation States**

| Operation Mode | | Reset | Active | Sleep | Watch | Sub-active | Sub-sleep | Standby |
|---|---|---|---|---|---|---|---|---|
| TCB | Interval | Reset | Functions | Functions | Halted | Halted | Halted | Halted |
| | Auto reload | Reset | Functions | Functions | Halted | Halted | Halted | Halted |
| TMB | | Reset | Functions | Retained | Retained | Retained | Retained | Retained |

## 9.4 Timer C

### 9.4.1 Overview

Timer C is an 8-bit timer that increments or decrements each time a clock pulse is input. This timer has two operation modes, interval and auto reload.

**Features**

The main features of timer C are given below.

- Choice of seven internal clock sources ($\phi$/8192, $\phi$/2048, $\phi$/512, $\phi$/64, $\phi$/16, $\phi$/4, $\phi_W$/4) or an external clock (can be used to count external events).
- An interrupt is requested when the counter overflows.
- Can be switched between up- and down-counting by software or hardware.
- When $\phi_W$/4 is selected as the internal clock source, or when an external clock is selected, timer C can function in subactive mode and subsleep mode.

### Block Diagram

Figure 9.3 shows a block diagram of timer C.

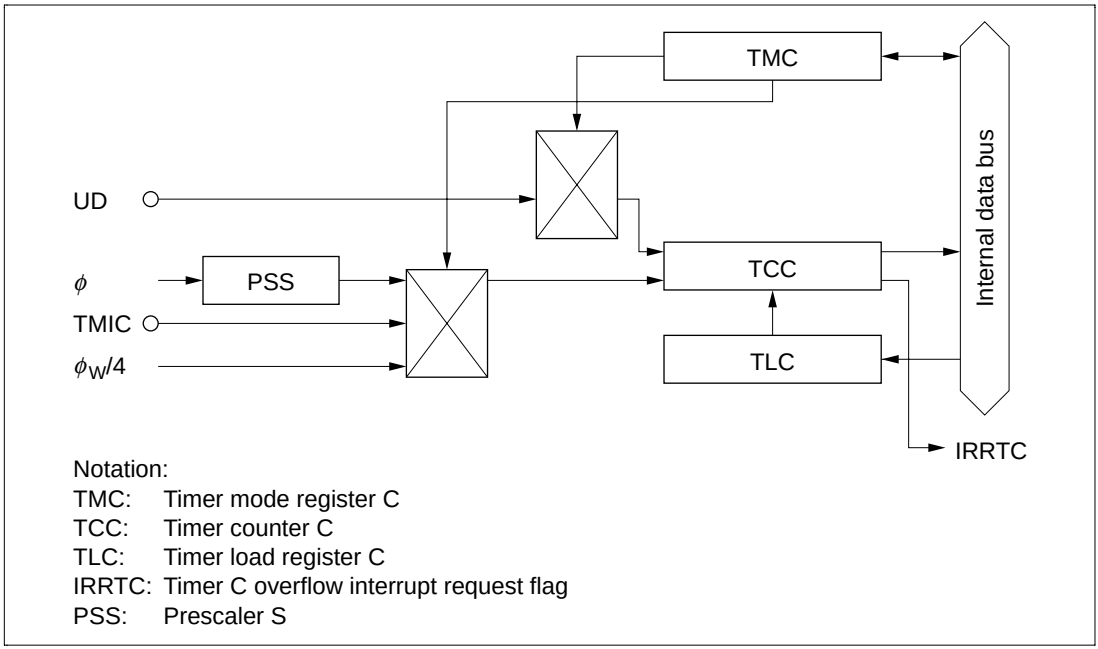



**Figure 9.3 Block Diagram of Timer C**

### Pin Configuration

Table 9.8 shows the timer C pin configuration.

**Table 9.8 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| Timer C event input | TMIC | Input | Event input to TCC |
| Timer C up/down control | UD | Input | Selection of counting direction |

### Register Configuration

Table 9.9 shows the register configuration of timer C.

**Table 9.9 Timer C Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Timer mode register C | TMC | R/W | H'18 | H'FFB4 |
| Timer counter C | TCC | R | H'00 | H'FFB5 |
| Timer load register C | TLC | W | H'00 | H'FFB5 |

### 9.4.2 Register Descriptions

**Timer Mode Register C (TMC)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TMC7 | TMC6 | TMC5 | — | — | TMC2 | TMC1 | TMC0 |
| Initial value | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | — | — | R/W | R/W | R/W |

TMC is an 8-bit read/write register for selecting the auto-reload function, counting direction, and input clock.

Upon reset, TMC is initialized to H'18.

**Bit 7—Auto-Reload Function Select (TMC7):** Bit 7 selects whether timer C is used as an interval timer or auto-reload timer.

| Bit 7: TMC7 | Description | |
|---|---|---|
| 0 | Interval timer function selected | (initial value) |
| 1 | Auto-reload function selected | |

**Bits 6 and 5—Counter Up/Down Control (TMC6 and TMC5):** These bits select the counting direction of timer counter C (TCC), or allow hardware to control the counting direction using pin UD.

| Bit 6: TMC6 | Bit 5: TMC5 | Description | |
|---|---|---|---|
| 0 | 0 | TCC is an up-counter | (initial value) |
| | 1 | TCC is a down-counter | |
| 1 | * | TCC up/down control is determined by input at pin UD. TCC is a down-counter if the UD input is high, and an up-counter if the UD input is low. | |

Note: * Don't care

**Bits 4 and 3—Reserved Bits:** Bits 4 and 3 are reserved; they are always read as 1, and cannot be modified.

**Bits 2 to 0—Clock Select (TMC2 to TMC0):** Bits 2 to 0 select the clock input to TCC. For external clock counting, either the rising or falling edge can be selected.

| Bit 2: TMC2 | Bit 1: TMC1 | Bit 0: TMC0 | Description | |
|---|---|---|---|---|
| 0 | 0 | 0 | Internal clock: $\phi$/8192 | (initial value) |
| | | 1 | Internal clock: $\phi$/2048 | |
| | 1 | 0 | Internal clock: $\phi$/512 | |
| | | 1 | Internal clock: $\phi$/64 | |
| 1 | 0 | 0 | Internal clock: $\phi$/16 | |
| | | 1 | Internal clock: $\phi$/4 | |
| | 1 | 0 | Internal clock: $\phi_W$/4 | |
| | | 1 | External event (TMIC): rising or falling edge* | |

Note: * The edge of the external event signal is selected by bit IEG2 in the IRQ edge select register (IEGR). See 3.3.2, for details on the IRQ edge select register. Be sure to set bit IRQ2 in port mode register 1 (PMR1) to 1 before setting bits TMC2 to TMC0 to 111.

**Timer Counter C (TCC)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCC7 | TCC6 | TCC5 | TCC4 | TCC3 | TCC2 | TCC1 | TCC0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

TCC is an 8-bit read-only up-/down-counter, which is incremented or decremented by internal or external clock input. The clock source for input to this counter is selected by bits TMC2 to TMC0 in timer mode register C (TMC). TCC values can be read by the CPU at any time.

When TCC overflows (from H'FF to H'00 or to the value set in TLC) or underflows (from H'00 to H'FF or to the value set in TLC), the IRRTC bit in interrupt request register 2 (IRR2) is set to 1.

TCC is allocated to the same address as timer load register C (TLC).

Upon reset, TCC is initialized to H'00.

**Timer Load Register C (TLC)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TLC7 | TLC6 | TLC5 | TLC4 | TLC3 | TLC2 | TLC1 | TLC0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

TLC is an 8-bit write-only register for setting the reload value of TCC.

When a reload value is set in TLC, the same value is loaded into timer counter C (TCC) as well, and TCC starts counting up or down from that value. When TCC overflows or underflows during operation in auto-reload mode, the TLC value is loaded into TCC. Accordingly, overflow and underflow periods can be set within the range of 1 to 256 input clocks.

The same address is allocated to TLC as to TCC.

Upon reset, TLC is initialized to H'00.

### 9.4.3 Timer Operation

**Interval Timer Operation:** When bit TMC7 in timer mode register C (TMC) is cleared to 0, timer C functions as an 8-bit interval timer.

Upon reset, timer counter C (TCC) is initialized to H'00 and TMC to H'18. After a reset, the counter continues uninterrupted incrementing as an interval up-counter. The clock input to timer C is selected from seven internal clock signals output by prescalers S and W, or an external clock input at pin TMIC. The selection is made by bits TMC2 to TMC0 in TMC.

Either software or hardware can control whether TCC counts up or down. The selection is made by TMC bits TMC6 and TMC5.

After the count value in TCC reaches H'FF (H'00), the next clock signal input causes timer C to overflow (underflow), setting bit IRRTC to 1 in interrupt request register 2 (IRR2). If IENTC = 1 in interrupt enable register 2 (IENR2), a CPU interrupt is requested.

At overflow or underflow, TCC returns to H'00 or H'FF and starts counting up or down again.

During interval timer operation (TMC7 = 0), when a value is set in timer load register C (TLC), the same value is set in TCC.

Note: * For details on interrupts, see 3.3, Interrupts.

**Auto-Reload Timer Operation:** Setting bit TMC7 in TMC to 1 causes timer C to function as an 8-bit auto-reload timer. When a reload value is set in TLC, the same value is loaded into TCC, becoming the value from which TCC starts its count.

After the count value in TCC reaches H'FF (H'00), the next clock signal input causes timer C to overflow (underflow). The TLC value is then loaded TCC, and the count continues from that value. The overflow (underflow) period can be set within a range from 1 to 256 input clocks, depending on the TLC value.

The clock sources, up/down control, and interrupts in auto-reload mode are the same as in interval mode.

In auto-reload mode (TMC7 = 1), when a new value is set in TLC, the TLC value is also set in TCC.

**Event Counter Operation:** Timer C can operate as an event counter, counting an event signal input at pin TMIC. External event counting is selected by setting TMC bits TMC2 to TMC0 to all 1s (111). TCC counts up or down at the rising or falling edge of the input at pin TMIC.

When timer C is used to count external event inputs, bit IRQ2 in port mode register 1 (PMR1) should be set to 1, and bit IEN2 in interrupt enable register 1 (IENR1) should be cleared to 0 to disable IRQ2 interrupt requests.

**TCC Up/Down Control by Hardware:** The counting direction of timer C can be controlled by input at pin UD. When bit TMC6 in TMC is set to 1, high-level input at the UD pin selects down-counting, while low-level input selects up-counting.

When using input at pin UD for this control function, set the UD bit in port mode register 2 (PMR2) to 1.

### 9.4.4 Timer C Operation States

Table 9.10 summarizes the timer C operation states.

**Table 9.10 Timer C Operation States**

| Operation Mode | | Reset | Active | Sleep | Watch | Sub-active | Sub-sleep | Standby |
|---|---|---|---|---|---|---|---|---|
| TCC | Interval | Reset | Functions | Functions | Halted | Functions/ Halted* | Functions/ Halted* | Halted |
| TCC | Auto reload | Reset | Functions | Functions | Halted | Functions/ Halted* | Functions/ Halted* | Halted |
| TMC | | Reset | Functions | Retained | Retained | Functions | Retained | Retained |

Note: When $\phi_W/4$ is selected as the internal clock of TCC in active mode or sleep mode, the internal clock is not synchronous with the system clock, so it is synchronized by a synchronizing circuit. This may result in a maximum error of $1/\phi$ (s) in the count cycle.

* When timer C is operated in subactive mode or subsleep mode, either an external clock or the $\phi_W/4$ internal clock must be selected. The counter will not operate in these modes if another clock is selected. If the internal $\phi_W/4$ clock is selected when $\phi_W/8$ is being used as the subclock $\phi_{SUB}$, the lower 2 bits of the counter will operate on the same cycle, with the least significant bit not being counted.

## 9.5 Timer F

### 9.5.1 Overview

Timer F is a 16-bit timer with an output compare function. Compare match signals can be used to reset the counter, request an interrupt, or toggle the output. Timer F can also be used for external event counting, and can operate as two independent 8-bit timers, timer FH and timer FL.

**Features**

Features of timer F are given below.

- Choice of four internal clock sources ($\phi/32$, $\phi/16$, $\phi/4$, $\phi/2$) or an external clock (can be used as an external event counter).
- Output from pin TMOFH is toggled by one compare match signal (the initial value of the toggle output can be set).
- Counter can be reset by the compare match signal.
- Two interrupt sources: counter overflow and compare match.
- Can operate as two independent 8-bit timers (timer FH and timer FL) in 8-bit mode.

Timer FH

- 8-bit timer (clocked by timer FL overflow signals when timer F operates as a 16-bit timer).
- Choice of four internal clocks ($\phi$/32, $\phi$/16, $\phi$/4, $\phi$/2).
- Output from pin TMOFH is toggled by one compare match signal (the initial value of the toggle output can be set).
- Counter can be reset by the compare match signal.
- Two interrupt sources: counter overflow and compare match.

Timer FL

- 8-bit timer/event counter
- Choice of four internal clocks ($\phi$/32, $\phi$/16, $\phi$/4, $\phi$/2) or event input at pin TMIF.
- Output from pin TMOFL is toggled by one compare match signal (the initial value of the toggle output can be set).
- Counter can be reset by the compare match signal.
- Two interrupt sources: counter overflow and compare match.

**Block Diagram**

Figure 9.4 shows a block diagram of timer F.

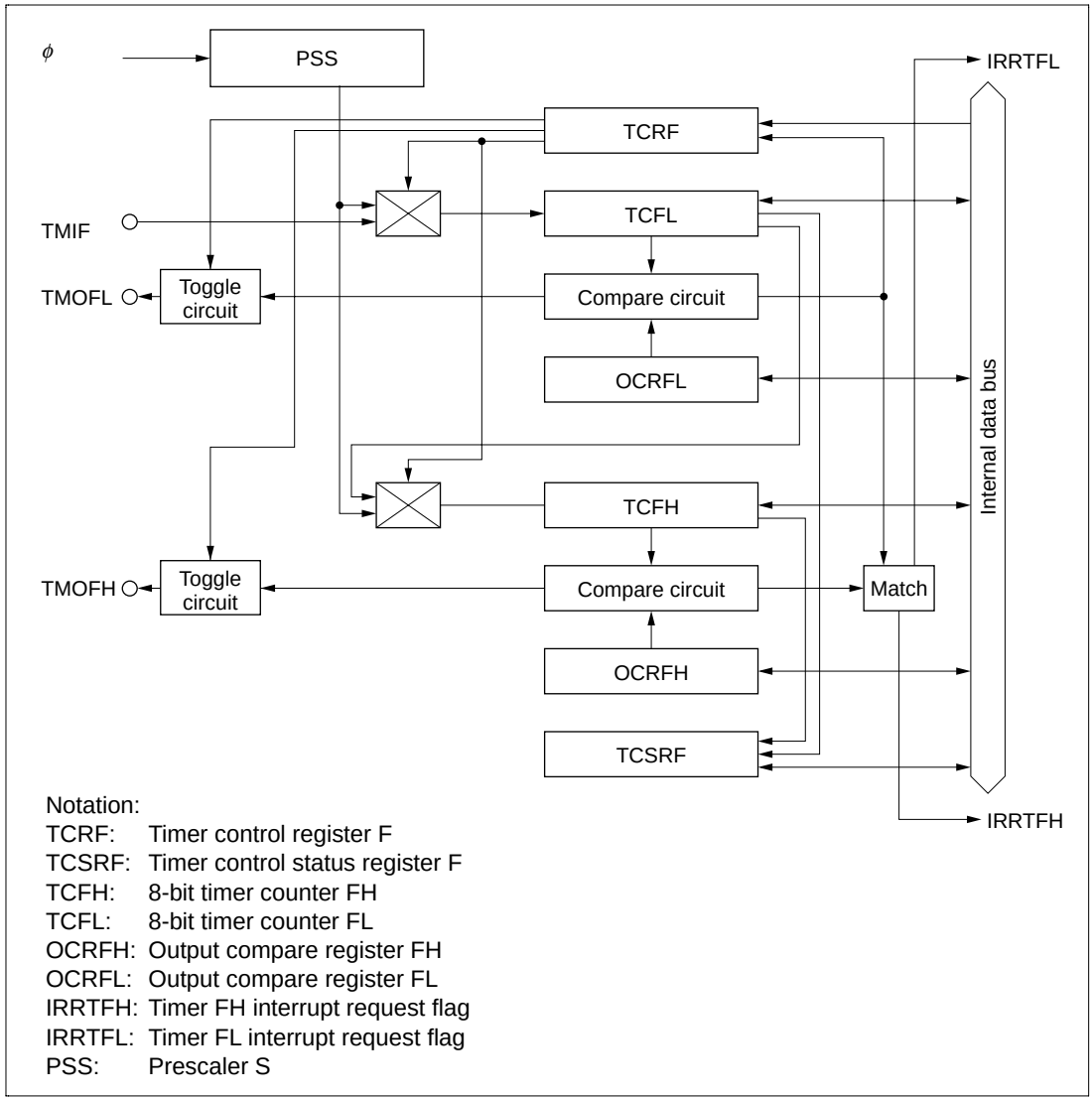



**Figure 9.4 Block Diagram of Timer F**

**Pin Configuration**

Table 9.11 shows the timer F pin configuration.

**Table 9.11 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| Timer F event input | TMIF | Input | Event input to TCFL |
| Timer FH output | TMOFH | Output | Timer FH toggle output |
| Timer FL output | TMOFL | Output | Timer FL toggle output |

**Register Configuration:**

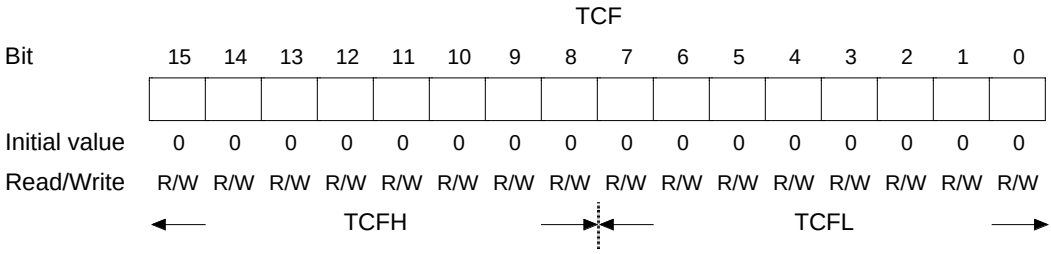
Table 9.12 shows the register configuration of timer F.

**Table 9.12 Timer F Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Timer control register F | TCRF | W | H'00 | H'FFB6 |
| Timer control/status register F | TCSRF | R/W | H'00 | H'FFB7 |
| 8-bit timer counter FH | TCFH | R/W | H'00 | H'FFB8 |
| 8-bit timer counter FL | TCFL | R/W | H'00 | H'FFB9 |
| Output compare register FH | OCRFH | R/W | H'FF | H'FFBA |
| Output compare register FL | OCRFL | R/W | H'FF | H'FFBB |

### 9.5.2 Register Descriptions

**16-Bit Timer Counter (TCF)**
**8-Bit Timer Counter (TCFH)**
**8-Bit Timer Counter (TCFL)**



TCF is a 16-bit read/write up-counter consisting of two cascaded 8-bit timer counters, TCFH and TCFL. TCF can be used as a 16-bit counter, with TCFH as the upper 8 bits and TCFL as the lower 8 bits of the counter, or TCFH and TCFL can be used as independent 8-bit counters.

TCFH and TCFL can be read and written by the CPU, but in 16-bit mode, data transfer with the CPU takes place via a temporary register (TEMP). For details see 9.5.3, Interface with the CPU.

Upon reset, TCFH and TCFL are each initialized to H'00.

- 16-bit mode (TCF)

  16-bit mode is selected by clearing bit CKSH2 to 0 in timer control register F (TCRF). The TCF input clock is selected by TCRF bits CKSL2 to CKSL0.

  Timer control status register F (TCSRF) can be set so that counter TCF will be cleared by compare match.

  When TCF overflows from H'FFFF to H'0000, the overflow flag (OVFH) in TCSRF is set to 1. If bit OVIEH in TCSRF is set to 1 when an overflow occurs, bit IRRTFH in interrupt request register 2 (IRR2) will be set to 1; and if bit IENTFH in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt will be requested.

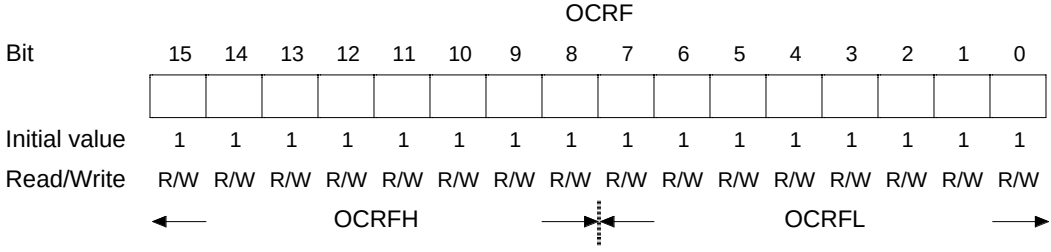
- 8-bit mode (TCFH, TCFL)

  When bit CKSH2 in timer control register F (TCRF) is set to 1, timer F functions as two separate 8-bit counters, TCFH and TCFL. The TCFH (TCFL) input clock is selected by TCRF bits CKSH2 to CKSH0 (CKSL2 to CKSL0).

  TCFH (TCFL) can be cleared by a compare match signal. This designation is made in bit CCLRH (CCLRL) in TCSRF.

  When TCFH (TCFL) overflows from H'FF to H'00, the overflow flag OVFH (OVFL) in TCSRF is set to 1. If bit OVIEH (OVIEL) in TCSRF is set to 1 when an overflow occurs, bit IRRTFH (IRRTHL) in interrupt request register 2 (IRR2) will be set to 1; and if bit IENTFH (IENTFL) in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt will be requested.

**16-Bit Output Compare Register (OCRF)**
**8-Bit Output Compare Register (OCRFH)**
**8-Bit Output Compare Register (OCRFL)**



OCRF is a 16-bit read/write output compare register consisting of two 8-bit read/write registers OCRFH and OCRFL. It can be used as a 16-bit output compare register, with OCRFH as the upper 8 bits and OCRFL as the lower 8 bits of the register, or OCRFH and OCRFL can be used as independent 8-bit registers.

OCRFH and OCRFL can be read and written by the CPU, but in 16-bit mode, data transfer with the CPU takes place via a temporary register (TEMP). For details see 9.5.3, Interface with the CPU.

Upon reset, OCRFH and OCRFL are each initialized to H'FF.

- 16-bit mode (OCRF)

  16-bit mode is selected by clearing bit CKSH2 to 0 in timer control register F (TCRF). The OCRF contents are always compared with the 16-bit timer counter (TCF). When the contents match, the compare match flag (CMFH) in TCSRF is set to 1. Also, IRRTFH in interrupt request register 2 (IRR2) is set to 1. If bit IENTFH in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt is requested.

  Output for pin TMOFH can be toggled by compare match. The output level can also be set to high or low by bit TOLH of timer control register F (TCRF).

- 8-bit mode (OCRFH, OCRFL)

  Setting bit CKSH2 in TCRF to 1 results in two 8-bit registers, OCRFH and OCRFL.

  The OCRFH contents are always compared with TCFH, and the OCRFL contents are always compared with TCFL. When the contents match, the compare match flag (CMFH or CMFL) in TCSRF is set to 1. Also, bit IRRTFH (IRRTFL) in interrupt request register 2 (IRR2) set to 1. If bit IENTFH (IENTFL) in interrupt enable register 2 (IENR2) is set to 1 at this time, a CPU interrupt is requested.

  The output at pin TMOFH (TMOFL) can be toggled by compare match. The output level can also be set to high or low by bit TOLH (TOLL) of the timer control register (TCRF).

**Timer Control Register F (TCRF)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TOLH | CKSH2 | CKSH1 | CKSH0 | TOLL | CKSL2 | CKSL1 | CKSL0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

TCRF is an 8-bit write-only register. It is used to switch between 16-bit mode and 8-bit mode, to select among four internal clocks and an external clock, and to select the output level at pins TMOFH and TMOFL.

Upon reset, TCRF is initialized to H'00.

**Bit 7—Toggle Output Level H (TOLH):** Bit 7 sets the output level at pin TMOFH. The setting goes into effect immediately after this bit is written.

| Bit 7: TOLH | Description | |
|---|---|---|
| 0 | Low level | (initial value) |
| 1 | High level | |

**Bits 6 to 4—Clock Select H (CKSH2 to CKSH0):** Bits 6 to 4 select the input to TCFH from four internal clock signals or the overflow of TCFL.

| Bit 6: CKSH2 | Bit 5: CKSH1 | Bit 4: CKSH0 | Description |
|---|---|---|---|
| 0 | * | * | 16-bit mode selected. TCFL overflow signals are counted (initial value) |
| 1 | 0 | 0 | Internal clock: $\phi$/32 |
| | | 1 | Internal clock: $\phi$/16 |
| | 1 | 0 | Internal clock: $\phi$/4 |
| | | 1 | Internal clock: $\phi$/2 |

Note: * Don't care

**Bit 3—Toggle Output Level L (TOLL):** Bit 3 sets the output level at pin TMOFL. The setting goes into effect immediately after this bit is written.

| Bit 3: TOLL | Description | |
|---|---|---|
| 0 | Low level | (initial value) |
| 1 | High level | |

**Bits 2 to 0—Clock Select L (CKSL2 to CKSL0):** Bits 2 to 0 select the input to TCFL from four internal clock signals or external event input.

| Bit 2: CKSL2 | Bit 1: CKSL1 | Bit 0: CKSL0 | Description |
|---|---|---|---|
| 0 | * | * | External event (TMIF). Rising or falling edge is counted*[1] (initial value) |
| 1 | 0 | 0 | Internal clock: $\phi$/32 |
| | | 1 | Internal clock: $\phi$/16 |
| | 1 | 0 | Internal clock: $\phi$/4 |
| | | 1 | Internal clock: $\phi$/2 |

Note: 1. The edge of the external event signal is selected by bit IEG3 in the IRQ edge select register (IEGR). See 3.3.2, for details on the IRQ edge select register. Note that switching the TMIF pin function by changing bit IRQ3 in port mode register 1 (PMR1) from 0 to 1 or from 1 to 0 while the TMIF pin is at the low level may cause the timer F counter to be incremented.

* Don't care

**Timer Control/Status Register F (TCSRF)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVFH | CMFH | OVIEH | CCLRH | OFL | CMFL | OVIEL | CCLRL |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | R/W | R/W | R/W* | R/W* | R/W | R/W |

Note: * Only 0 can be written, to clear flag.

TCSRF is an 8-bit read/write register. It is used for counter clear selection, overflow and compare match indication, and enabling of interrupts caused by timer overflow.

Upon reset, TCSRF is initialized to H'00.

**Bit 7—Timer overflow flag H (OVFH):** Bit 7 is a status flag indicating TCFH overflow (H'FF to H'00). This flag is set by hardware and cleared by software. It cannot be set by software.

| Bit 7: OVFH | Description |
|---|---|
| 0 | Clearing conditions:<br>After reading OVFH = 1, cleared by writing 0 to OVFH (initial value) |
| 1 | Setting conditions:<br>Set when the value of TCFH goes from H'FF to H'00 |

**Bit 6—Compare Match Flag H (CMFH):** Bit 6 is a status flag indicating a compare match between TCFH and OCRFH. This flag is set by hardware and cleared by software. It cannot be set by software.

| Bit 6: CMFH | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading CMFH = 1, cleared by writing 0 to CMFH | (initial value) |
| 1 | Setting conditions:<br>Set when the TCFH value matches OCRFH value | |

**Bit 5—Timer Overflow Interrupt Enable H (OVIEH):** Bit 5 enables or disables TCFH overflow interrupts.

| Bit 5: OVIEH | Description | |
|---|---|---|
| 0 | TCFH overflow interrupt disabled | (initial value) |
| 1 | TCFH overflow interrupt enabled | |

**Bit 4—Counter Clear H (CCLRH):** In 16-bit mode, bit 4 selects whether or not TCF is cleared when a compare match occurs between TCF and OCRF.

In 8-bit mode, bit 4 selects whether or not TCFH is cleared when a compare match occurs between TCFH and OCRFH.

| Bit 4: CCLRH | Description | |
|---|---|---|
| 0 | 16-bit mode: TCF clearing by compare match disabled<br>8-bit mode: TCFH clearing by compare match disabled | (initial value) |
| 1 | 16-bit mode: TCF clearing by compare match enabled<br>8-bit mode: TCFH clearing by compare match enabled | |

**Bit 3—Timer Overflow Flag L (OVFL):** Bit 3 is a status flag indicating TCFL overflow (H'FF to H'00). This flag is set by hardware and cleared by software. It cannot be set by software.

| Bit 3: OVFL | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading OVFL = 1, cleared by writing 0 to OVFL | (initial value) |
| 1 | Setting conditions:<br>Set when the value of TCFL goes from H'FF to H'00 | |

**Bit 2—Compare Match Flag L (CMFL):** Bit 2 is a status flag indicating a compare match between TCFL and OCRFL. This flag is set by hardware and cleared by software. It cannot be set by software.

| Bit 2: CMFL | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading CMFL = 1, cleared by writing 0 to CMFL | (initial value) |
| 1 | Setting conditions:<br>Set when the TCFL value matches the OCRFL value | |

**Bit 1—Timer Overflow Interrupt Enable L (OVIEL):** Bit 1 enables or disables TCFL overflow interrupts.

| Bit 1: OVIEL | Description | |
|---|---|---|
| 0 | TCFL overflow interrupt disabled | (initial value) |
| 1 | TCFL overflow interrupt enabled | |

**Bit 0—Counter Clear L (CCLRL):** Bit 0 selects whether or not TCFL is cleared when a compare match occurs between TCFL and OCRFL.

| Bit 0: CCLRL | Description | |
|---|---|---|
| 0 | TCFL clearing by compare match disabled | (initial value) |
| 1 | TCFL clearing by compare match enabled | |

### 9.5.3 Interface with the CPU

TCF and OCRF are 16-bit read/write registers, whereas the data bus between the CPU and on-chip peripheral modules has an 8-bit width. For this reason, when the CPU accesses TCF or OCRF, it makes use of an 8-bit temporary register (TEMP).

In 16-bit mode, when reading or writing TCF or writing OCRF, always use two consecutive byte size MOV instructions, and always access the upper byte first. Data will not be transferred properly if only the upper byte or only the lower byte is accessed. In 8-bit mode there is no such restriction on the order of access.

**Write Access:** When the upper byte is written, the upper-byte data is loaded into the TEMP register. Next when the lower byte is written, the data in TEMP goes to the upper byte of the register, and the lower-byte data goes directly to the lower byte of the register. Figure 9.5 shows a TCF write operation when H'AA55 is written to TCF.

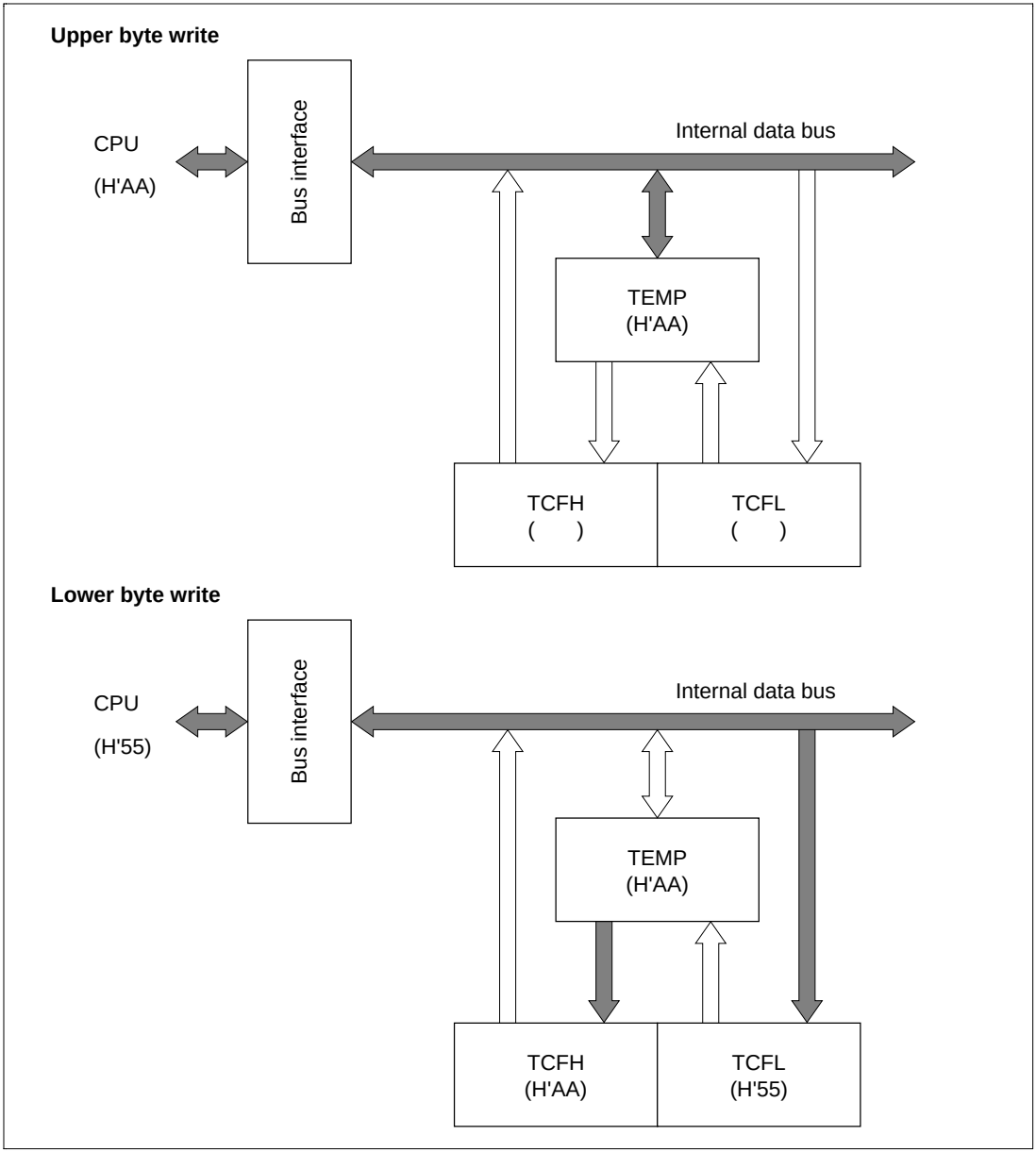
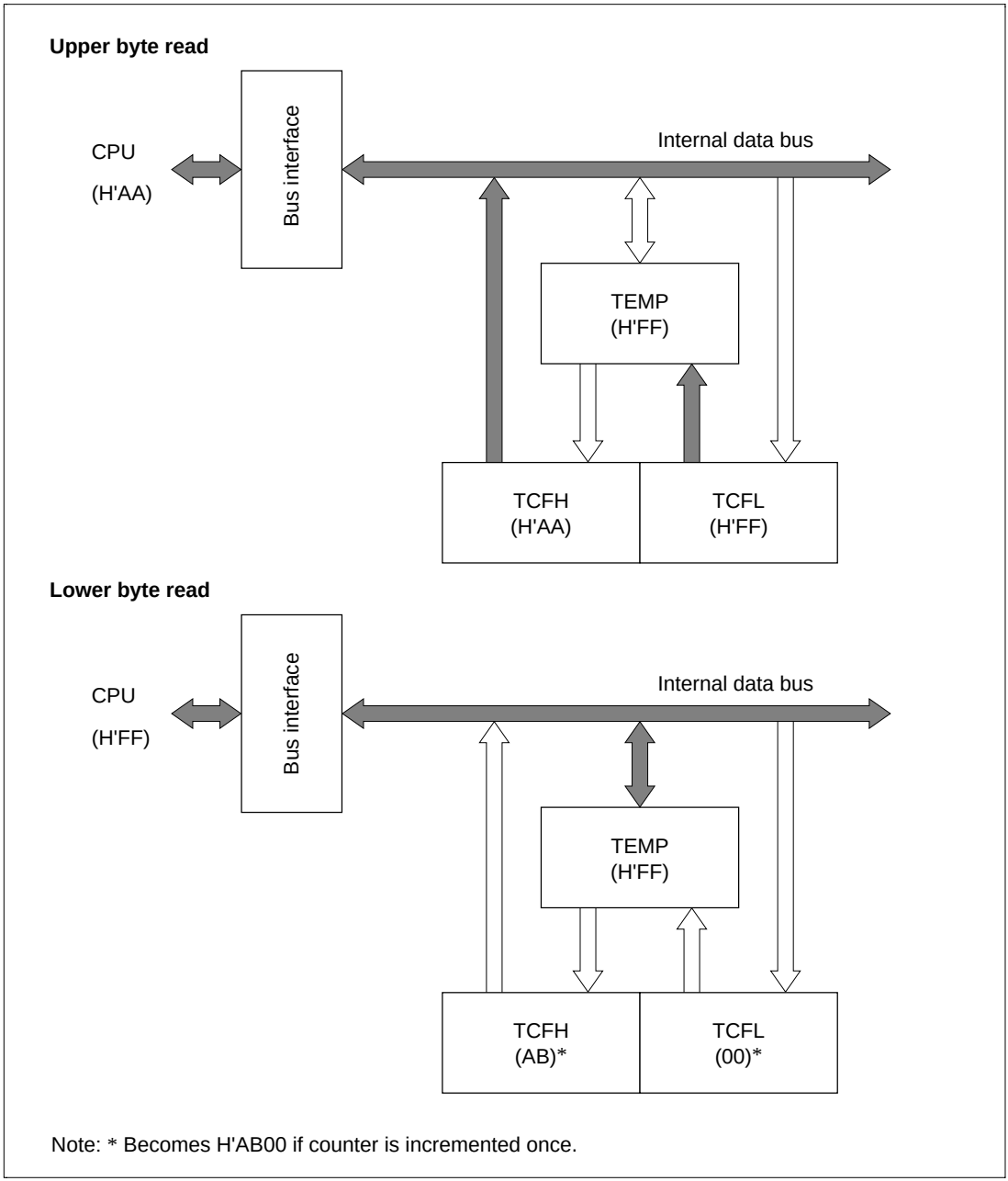



**Figure 9.5 TCF Write Operation (CPU → TCF)**

**Read Access:** When the upper byte of TCF is read, the upper-byte data is sent directly to the CPU, and the lower byte is loaded into TEMP. Next when the lower byte is read, the lower byte in TEMP is sent to the CPU.

When the upper byte of OCRF is read, the upper-byte data is sent directly to the CPU. Next when the lower byte is read, the lower-byte data is sent directly to the CPU.

Figure 9.6 shows a TCF read operation when H'AAFF is read from TCF.




Note: * Becomes H'AB00 if counter is incremented once.

**Figure 9.6 TCF Read Operation (TCF → CPU)**

### 9.5.4 Timer Operation

Timer F is a 16-bit timer/counter that increments with each input clock. The value set in output compare register F is constantly compared with the value of timer counter F, and when they match the counter can be cleared, an interrupt can be requested, and the port output can be toggled. Timer F can also be used as two independent 8-bit timers.

**Timer F Operation:** Timer F can operate in either 16-bit timer mode or 8-bit timer mode. These modes are described below.

- 16-bit timer mode

  Timer F operates in 16-bit timer mode when the CKSH2 bit in timer control register F (TCRF) is cleared to 0.

  A reset initializes timer counter F (TCF) to H'0000, output compare register F (OCRF) to H'FFFF, and timer control register F (TCRF) and timer control status register F (TCSRF) to H'00. Timer F begins counting external event input signals (TMIF). The edge of the external event signal is selected by the IEG3 bit in the IRQ edge select register (IEGR).

  Any of four internal clocks output by prescaler S, or an external clock, can be selected as the timer F operating clock by bits CKSL2 to CKSL0 in TCRF.

  TCF is continuously compared with the contents of OCRF. When these two values match, the CMFH bit in TCSRF is set to 1. At this time if IENTFH of IENR2 is 1, a CPU interrupt is requested and the output at pin TMOFH is toggled. If the CCLRH bit in TCSRF is 1, timer F is cleared. The output at pin TMOFH can also be set by the TOLH bit in TCRF.

  If timer F overflows (from H'FFFF to H'0000), the OVFH bit in TCSRF is set. At this time, if the OVIEH bit in TCSRF and the IENTFH bit in IENR2 are both 1, a CPU interrupt is requested.

- 8-bit timer mode

  When the CKSH2 bit in TCRF is set to 1, timer F operates as two independent 8-bit timers, TCFH and TCFL. The input clock of TCFH/TCFL is selected by bits CKSH2 to CKSH0/CKSL2 to CKSL0 in TCRF.

  When TCFH/TCFL and the contents of OCRFH/OCRFL match, the CMFH/CMFL bit in TCSRF is set to 1. If the IENTFH/IENTFL bit in IENR2 is 1, a CPU interrupt is requested and the output at pin TMOFH/TMOFL is toggled. If the CCLRH/CCLRL bit in TCRF is 1, TCFH/TCFL is cleared. The output at pin TMOFH/TMOFL can also be set by the TOLH/TOLL bit in TCRF.

  When TCFH/TCFL overflows from H'FF to H'00, the OVFH/OVFL bit in TCSRF is set to 1. At this time, if the OVIEH/OVIEL bit in TCSRF and the IENTFH/IENTFL bit in IENR2 are both 1, a CPU interrupt is requested.

**TCF Count Timing:** TCF is incremented by each pulse of the input clock (internal or external clock).

- Internal clock

  The settings of bits CKSH2 to CKSH0 or bits CKSL2 to CKSL0 in TCRF select one of four internal clock signals divided from the system clock ($\phi$), namely, $\phi/32$, $\phi/16$, $\phi/4$, or $\phi/2$.
- External clock

  External clock input is selected by clearing bit CKSL2 to 0 in TCRF. Either rising or falling edges of the clock input can be counted. The edge of an external event is selected by bit IEG3 in the interrupt controller's IEGR register. An external event pulse width of at least two system clock ($\phi$) cycles is necessary for correct operation of the counter.

**TMOFH and TMOFL Output Timing:** The outputs at pins TMOFH and TMOFL are the values set in bits TOLH and TOLL in TCRF. When a compare match occurs, the output value is inverted. Figure 9.7 shows the output timing.

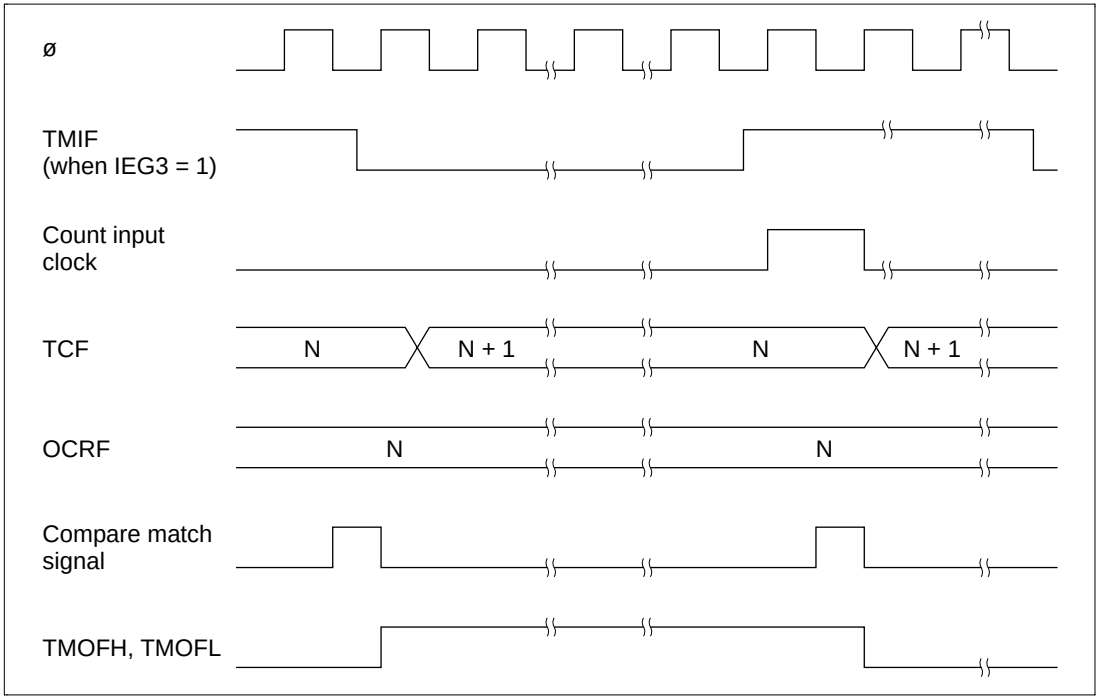



**Figure 9.7 TMOFH, TMOFL Output Timing**

**TCF Clear Timing:** TCF can be cleared at compare match with OCRF.

**Timer Overflow Flag (OVF) Set Timing:** OVF is set to 1 when TCF overflows (goes from H'FFFF to H'0000).

**Compare Match Flag Set Timing:** The compare match flags (CMFH or CMFL) are set to 1 when a compare match occurs between TCF and OCRF. A compare match signal is generated in the final state in which the values match (when TCF changes from the matching count value to the next value). When TCF and OCRF match, a compare match signal is not generated until the next counter clock pulse.

**Timer F Operation States:** Table 9.13 summarizes the timer F operation states.

**Table 9.13 Timer F Operation States**

| Operation Mode | Reset | Active | Sleep | Watch | Sub-active | Sub-sleep | Standby |
|---|---|---|---|---|---|---|---|
| TCF | Reset | Functions | Functions | Halted | Halted | Halted | Halted |
| OCRF | Reset | Functions | Retained | Retained | Retained | Retained | Retained |
| TCRF | Reset | Functions | Retained | Retained | Retained | Retained | Retained |
| TCSRF | Reset | Functions | Retained | Retained | Retained | Retained | Retained |

### 9.5.5 Application Notes

The following conflicts can arise in timer F operation.

- 16-bit timer mode

  The output at pin TMOFH toggles when all 16 bits match and a compare match signal is generated. If the compare match signal occurs at the same time as new data is written in TCRF by a MOV instruction, however, the new value written in bit TOLH will be output at pin TMOFH. The TMOFL output in 16-bit mode is indeterminate, so this output should not be used. Use the pin as a general input or output port.

  If an OCRFL write occurs at the same time as a compare match signal, the compare match signal is inhibited. If a compare match occurs between the written data and the counter value, however, a compare match signal will be generated at that point. The compare match signal is output in synchronization with the TCFL clock, so if this clock is stopped no compare match signal will be generated, even if a compare match occurs.

  Compare match flag CMFH is set when all 16 bits match and a compare match signal is generated; bit CMFL is set when the setting conditions are met for the lower 8 bits.

  The overflow flag (OVFH) is set when TCF overflows; bit OVFL is set if the setting conditions are met when the lower 8 bits overflow. If a write to TCFL occurs at the same time as an overflow signal, the overflow signal is not output.

- 8-bit timer mode
  - TCFH and OCRFH

    The output at pin TMOFH toggles when there is a compare match. If the compare match signal occurs at the same time as new data is written in TCRF by a MOV instruction, however, the new value written in bit TOLH will be output at pin TMOFH.

    If an OCRFH write occurs at the same time as a compare match signal, the compare match signal is inhibited. If a compare match occurs between the written data and the counter value, however, a compare match signal will be generated at that point. The compare match signal is output in synchronization with the TCFH clock.

    If a TCFH write occurs at the same time as an overflow signal, the overflow signal is not output.

  - TCFL and OCRFL

    The output at pin TMOFL toggles when there is a compare match. If the compare match signal occurs at the same time as new data is written in TCRF by a MOV instruction, however, the new value written in bit TOLL will be output at pin TMOFL.

    If an OCRFL write occurs at the same time as a compare match signal, the compare match signal is inhibited. If a compare match occurs between the written data and the counter value, however, a compare match signal will be generated at that point. The compare match signal is output in synchronization with the TCFL clock, so if this clock is stopped no compare match signal will be generated, even if a compare match occurs.

    If a TCFL write occurs at the same time as an overflow signal, the overflow signal is not output.

## 9.6 Timer G

### 9.6.1 Overview

Timer G is an 8-bit timer, with input capture functions for separately capturing the rising edge and falling edge of pulses input at the input capture pin (input capture input signal). Timer G has a built-in noise canceller circuit that can eliminate high-frequency noise from the input capture signal, enabling accurate measurement of its duty cycle. When timer G is not used for input capture, it functions as an 8-bit interval timer.

**Features**

Features of timer G are given below.

- Choice of four internal clock sources ($\phi$/64, $\phi$/32, $\phi$/2, $\phi_W$/2)
- Input capture function
  Separate input capture registers are provided for the rising and falling edges.
- Counter overflow detection
  Can detect whether overflow occurred when the input capture signal was high or low.
- Choice of counter clear triggers
  The counter can be cleared at the rising edge, falling edge, or both edges of the input capture signal.
- Two interrupt sources
  Interrupts can be requested by input capture and by overflow. For input capture, the rising or falling edge can be selected.
- Built-in noise-canceller circuit
  The noise canceller circuit can eliminate high-frequency noise in the input capture signal.
- Operates in subactive and subsleep modes
  When $\phi_W$/2 is selected as the internal clock source, timer G can operate in the subactive and subsleep modes.

### Block Diagram

Figure 9.8 shows a block diagram of timer G.

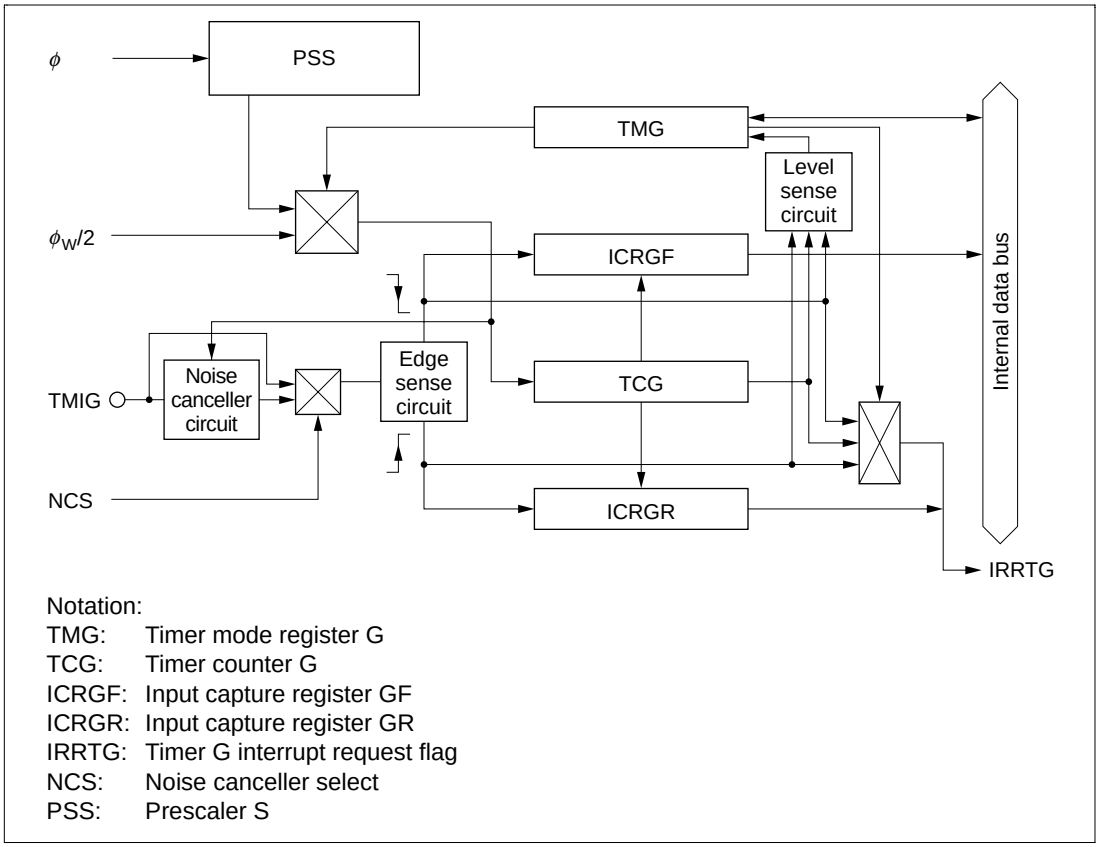



**Figure 9.8 Block Diagram of Timer G**

### Pin Configuration

Table 9.14 shows the timer G pin configuration.

**Table 9.14 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| Timer G capture input | TMIG | Input | Timer G capture input |

**Register Configuration**

Table 9.15 shows the register configuration of timer G.

**Table 9.15 Timer G Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Timer mode register G | TMG | R/W | H'00 | H'FFBC |
| Timer counter G | TCG | — | H'00 | — |
| Input capture register GF | ICRGF | R | H'00 | H'FFBD |
| Input capture register GR | ICRGR | R | H'00 | H'FFBE |

### 9.6.2 Register Descriptions

**Timer Counter G (TCG)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCG7 | TCG6 | TCG5 | TCG4 | TCG3 | TCG2 | TCG1 | TCG0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | — | — | — | — |

Timer counter G (TCG) is an 8-bit up-counter which is incremented by an input clock. The input clock signal is selected by bits CKS1 and CKS0 in timer mode register G (TMG).

To use TCG as an input capture timer, set bit TMIG to 1 in PMR1; to use TCG as an interval timer, clear bit TMIG to 0.* When TCG is used as an input capture timer, the TCG value can be cleared at the rising edge, falling edge, or both edges of the input capture signal, depending on settings in TMG.

When TCG overflows (goes from H'FF to H'00), if the timer overflow interrupt enable bit (OVIE) is set to 1 in TMG, bit IRRTG in interrupt request register 2 (IRR2) is set to 1. If in addition bit IENTG in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt is requested. Details on interrupts are given in 3.3, Interrupts.

TCG cannot be read or written by the CPU.

Upon reset, TCG is initialized to H'00.

Note: * An input capture signal may be generated when TMIG is rewritten.

**Input capture register GF (ICRGF)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ICRGF7 | ICRGF6 | ICRGF5 | ICRGF4 | ICRGF3 | ICRGF2 | ICRGF1 | ICRGF0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

ICRGF is an 8-bit read-only register. When the falling edge of the input capture signal is detected, the TCG value at that time is transferred to ICRGF. If the input capture interrupt select bit (IIEGS) is set to 1 in TMG, bit IRRTG in interrupt request register 2 (IRR2) is set to 1. If in addition bit IENTG in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt is requested. Details on interrupts are given in 3.3, Interrupts.

To ensure proper input capture when the noise canceller is not used, the pulse width of the input capture signal should be at least $2\phi$ or $2\phi_{SUB}$.

Upon reset, ICRGF is initialized to H'00.

**Input Capture Register GR (ICRGR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ICRGR7 | ICRGR6 | ICRGR5 | ICRGR4 | ICRGR3 | ICRGR2 | ICRGR1 | ICRGR0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

ICRGR is an 8-bit read-only register. When the rising edge of the input capture signal is detected, the TCG value at that time is sent to ICRGR. If the IIEGS bit is cleared to 0 in TMG, bit IRRTG in interrupt request register 2 (IRR2) is set to 1. If in addition bit IENTG in interrupt enable register 2 (IENR2) is set to 1, a CPU interrupt is requested. Details on interrupts are given in 3.3, Interrupts.

To ensure proper input capture when the noise canceller is not used, the pulse width of the input capture signal should be at least $2\phi$ or $2\phi_{SUB}$.

Upon reset, ICRGR is initialized to H'00.

**Timer Bode Register G (TMG)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVFH | OVFL | OVIE | IIEGS | CCLR1 | CCLR0 | CKS1 | CKS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | R/W | R/W | R/W | R/W | R/W | R/W |

Note: * Only 0 can be written, to clear flag.

TMG is an 8-bit read/write register. It controls the choice of four input clocks, counter clear selection, and edge selection for input capture interrupt requests. It also indicates overflow status and enables or disables overflow interrupt requests.

Upon reset, TMG is initialized to H'00.

**Bit 7—Timer Overflow Flag H (OVFH):** Bit 7 is a status flag indicating that TCG overflowed (from H'FF to H'00) when the input capture signal was high. This flag is set by hardware and cleared by software. It cannot be set by software.

| Bit 7: OVFH | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading OVFH = 1, cleared by writing 0 to OVFH | (initial value) |
| 1 | Setting conditions:<br>Set when the value of TCG overflows from H'FF to H'00 | |

**Bit 6—Timer Overflow Flag L (OVFL):** Bit 6 is a status flag indicating that TCG overflowed (from H'FF to H'00) when the input capture signal was low, or in interval timer operation. This flag is set by hardware and cleared by software. It cannot be set by software.

| Bit 6: OVFL | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading OVFL = 1, cleared by writing 0 to OVFL | (initial value) |
| 1 | Setting conditions:<br>Set when the value of TCG overflows from H'FF to H'00 | |

**Bit 5—Timer Overflow Interrupt Enable (OVIE):** Bit 5 enables or disables TCG overflow interrupts.

| Bit 5: OVIE | Description | |
|---|---|---|
| 0 | TCG overflow interrupt disabled | (initial value) |
| 1 | TCG overflow interrupt enabled | |

**Bit 4—Input Capture Interrupt Edge Select (IIEGS):** Bit 4 selects the input signal edge at which input capture interrupts are requested.

| Bit 4: IIEGS | Description |
|---|---|
| 0 | Interrupts are requested at the rising edge of the input capture signal (initial value) |
| 1 | Interrupts are requested at the falling edge of the input capture signal |

**Bits 3, 2—Counter Clear 1, 0 (CCLR1, CCLR0):** Bits 3 and 2 designate whether TCG is cleared at the rising, falling, or both edges of the input capture signal, or is not cleared.

| Bit 3: CCLR1 | Bit 2: CCLR0 | Description |
|---|---|---|
| 0 | 0 | TCG is not cleared (initial value) |
| | 1 | TCG is cleared at the falling edge of the input capture signal |
| 1 | 0 | TCG is cleared at the rising edge of the input capture signal |
| | 1 | TCG is cleared at both edges of the input capture signal |

**Bits 1, 0—Clock Select (CKS1, CKS0):** Bits 1 and 0 select the clock input to TCG from four internal clock signals.

| Bit 1: CKS1 | Bit 0: CKS0 | Description |
|---|---|---|
| 0 | 0 | Internal clock: $\phi$/64 (initial value) |
| | 1 | Internal clock: $\phi$/32 |
| 1 | 0 | Internal clock: $\phi$/2 |
| | 1 | Internal clock: $\phi_W$/2 |

### 9.6.3 Noise Canceller Circuit

The noise canceller circuit built into the H8/3834 Series is a digital low-pass filter that rejects high-frequency pulse noise in the input at the input capture pin. The noise canceller circuit is enabled by the noise canceller select (NCS)* bit in port mode register 2 (PMR2).

Figure 9.9 shows a block diagram of the noise canceller circuit.

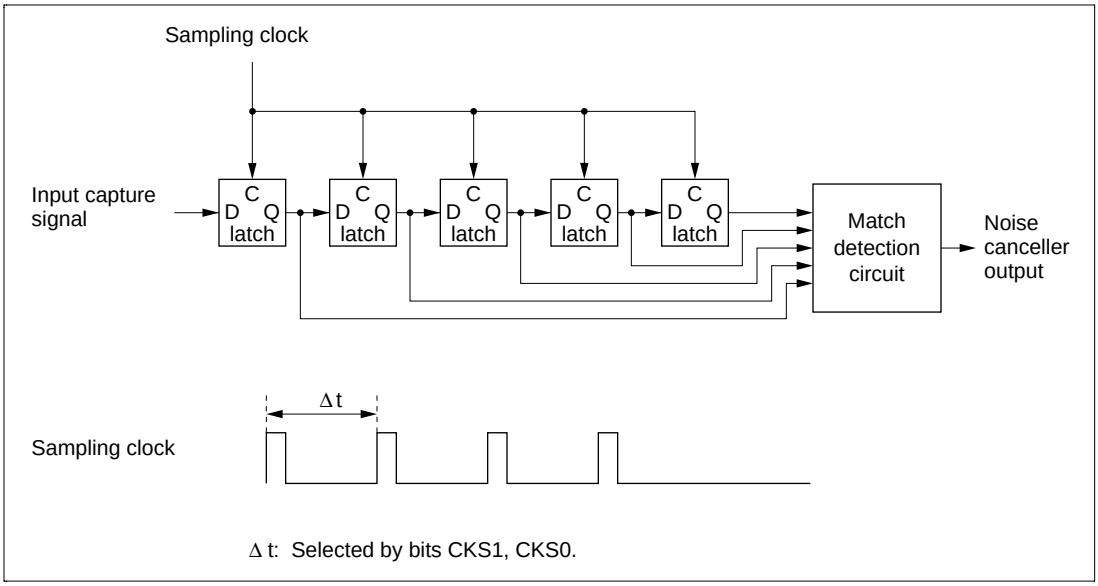



**Figure 9.9 Block Diagram of Noise Canceller Circuit**

The noise canceller consists of five latch circuits connected in series, and a match detection circuit. When the noise canceller function is disabled (NCS = 0), the system clock is selected as the sampling clock. When the noise canceller is enabled (NCS = 1), the internal clock selected by bits CKS1 and CKS0 in TMG becomes the sampling clock. The input signal is sampled at the rising edge of this clock pulse. Data is considered correct when the outputs of all five latch circuits match. If they do not match, the previous value is retained. Upon reset, the noise canceller output is initialized after the falling edge of the input capture signal has been sampled five times. Accordingly, after the noise canceller function is enabled, pulses that have a pulse width five times greater than the sampling clock will be recognized as input capture signals.

If the noise canceller circuit is not used, the input capture signal pulse width must be at least $2\phi$ or $2\phi_{SUB}$ in order to ensure proper input capture operation.

Note: * Rewriting the NCS bit may cause an internal input capture signal to be generated.

Figure 9.10 shows a typical timing diagram for the noise canceller circuit. In this example, a high-level input at the input capture pin is rejected as noise because its pulse width is less than five sampling clock $\phi$ cycles.

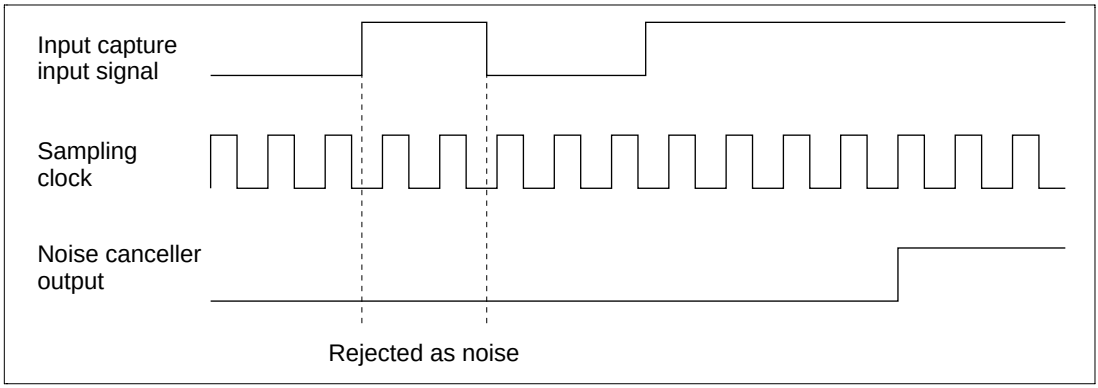



**Figure 9.10 Noise Canceller Circuit Timing (Example)**

### 9.6.4 Timer Operation

Timer G is an 8-bit timer with input capture and interval timer functions.

**Timer G Functions:** Timer G is an 8-bit timer/counter that functions as an input capture timer or an interval timer. These two functions are described below.

- Input capture timer operation

  Timer G functions as an input capture timer when bit TMIG of port mode register 1 (PMR1) is set to 1.*

  At reset, timer mode register G (TMG), timer counter G (TCG), input capture register GF (ICRGF), and input capture register GR (ICRGR) are all initialized to H'00.

  Immediately after reset, TCG begins counting an internal clock with a frequency of $\phi$ divided by 64 ($\phi$/64). Four other internal clocks can be selected using bits CKS1 and CKS0 of TMG.

  At the rising edge/falling edge of the input capture signal input to pin TMIG, the value of TCG is copied into ICRGR/ICRGF. If the input edge is the same as the edge selected by the IIEGS bit of TMG, then bit IRRTG is set to 1 in IRR2. If bit IENTG is also set to 1 in IENR2, a CPU interrupt is requested. For details on interrupts, see section 3.3, Interrupts.

  TCG can be cleared to 0 at the rising edge, falling edge, or both edges of the input capture signal as determined with bits CCLR1 and CCLR0 of TMG. If TCG overflows while the input capture signal is high, bit OVFH of TMG is set. If TCG overflows while the input capture signal is low, bit OVFL of TMG is set. When either of these bits is set, if bit OVIE of TMG is currently set to 1, then bit IRRTG is set to 1 in IRR2. If bit IENTG is also set to 1 in IENR2, then timer G requests a CPU interrupt. For further details see 3.3, Interrupts.

  Timer G has a noise canceller circuit that rejects high-frequency pulse noise in the input to pin TMIG. See 9.6.3, Noise Canceller Circuit, for details.

Note: * Rewriting the TMIG bit may cause an internal input capture signal to be generated.

- Interval timer operation

  Timer G functions as an interval timer when bit TMIG is cleared to 0 in PMR1. Following a reset, TCG starts counting cycles of the $\phi$/64 internal clock. This is one of four internal clock sources that can be selected by bits CKS1 and CKS0 of TMG. TCG counts up according to the selected clock source. When it overflows from H'FF to H'00, bit OVFL of TMG is set to 1. If bit OVIE of TMG is currently set to 1, then bit IRRTG is set to 1 in IRR2. If bit IENTG is also set to 1 in IENR2, then timer G requests a CPU interrupt. For further details see 3.3, Interrupts.

**Count Timing:** TCG is incremented by input pulses from an internal clock. TMG bits CKS1 and CKS0 select one of four internal clocks ($\phi$/64, $\phi$/32, $\phi$/2, $\phi_W$/2) derived by dividing the system clock ($\phi$) or the watch clock ($\phi_W$).

**Timing of Internal Input Capture Signals:**

- Timing with noise canceller function disabled

  Separate internal input capture signals are generated from the rising and falling edges of the external input signal.

  Figure 9.11 shows the timing of these signals.

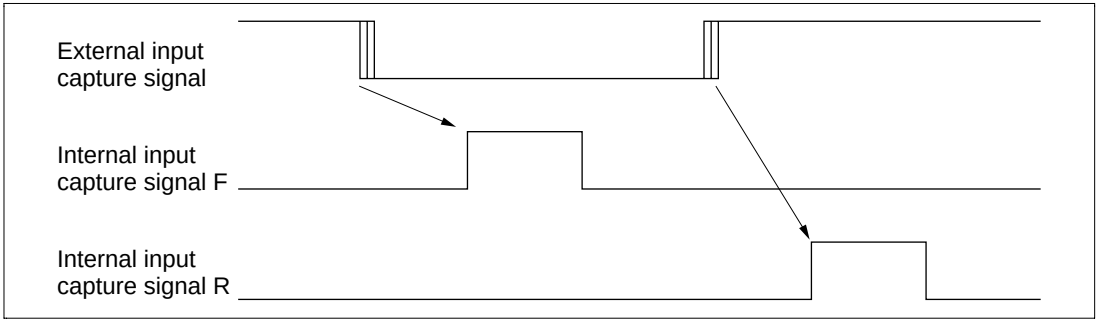



**Figure 9.11 Input Capture Signal Timing (Noise Canceller Function Disabled)**

- Timing with noise canceller function enabled

  When input capture noise cancelling is enabled, the external input capture signal is routed via the noise canceller circuit, so the internal signals are delayed from the input edge by five sampling clock cycles. Figure 9.12 shows the timing.

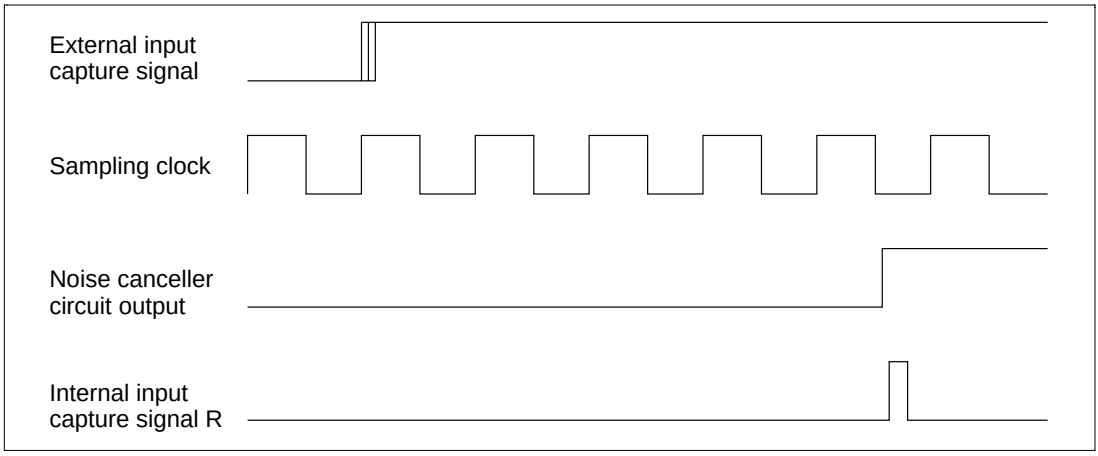



**Figure 9.12 Input Capture Signal Timing (Noise Canceller Function Enabled)**

**Timing of Input Capture:** Figure 9.13 shows the input capture timing in relation to the internal input capture signal.

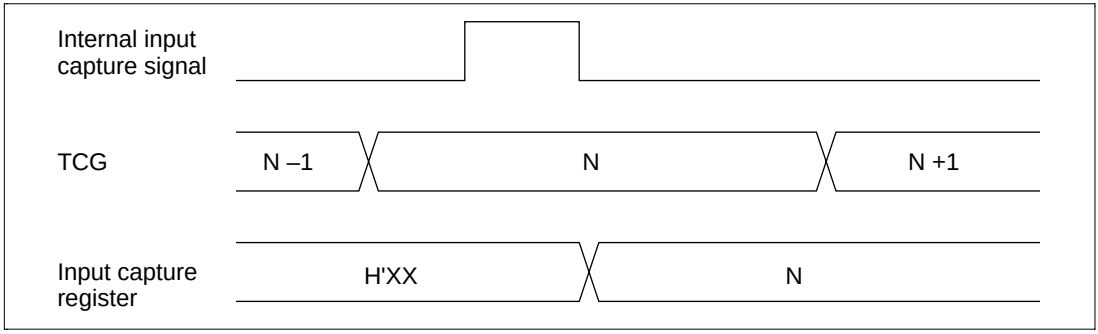



**Figure 9.13 Input Capture Timing**

**TCG Clear Timing:** TCG can be cleared at the rising edge, falling edge, or both edges of the external input capture signal. Figure 9.14 shows the timing for clearing at both edges.

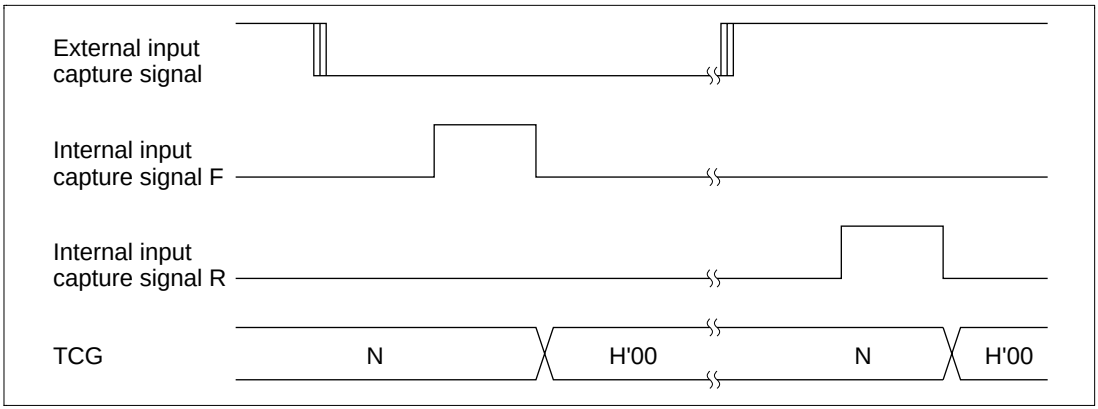



**Figure 9.14 TCG Clear Timing**

**Timer G Operation States:** Table 9.16 summarizes the timer G operation states.

**Table 9.16 Timer G Operation States**

| Operation Mode | | Reset | Active | Sleep | Watch | Sub-active | Sub-sleep | Standby |
|---|---|---|---|---|---|---|---|---|
| TCG | Input capture | Reset | Functions* | Functions* | Halted | Functions/ Halted* | Functions/ Halted* | Halted |
| | Interval | Reset | Functions* | Functions* | Retained | Functions/ Halted* | Functions/ Halted* | Halted |
| ICRGF | | Reset | Functions* | Functions* | Retained | Functions/ Halted* | Functions/ Halted* | Retained |
| ICRGR | | Reset | Functions* | Functions* | Retained | Functions/ Halted* | Functions/ Halted* | Retained |
| TMG | | Reset | Functions | Retained | Retained | Functions | Retained | Retained |

Note: * In active mode and sleep mode, if $\phi_W/2$ is selected as the TCG internal clock, since the system clock and internal clock are not synchronized with each other, a synchronization circuit is used. This may result in a count cycle error of up to $1/\phi$ (s). In subactive mode and subsleep mode, if $\phi_W/2$ is selected as the TCG internal clock, regardless of the subclock $\phi/_{SUB}$ ($\phi_W/2$, $\phi_W/4$, $\phi_W/8$) TCG and the noise canceller circuit run on an internal clock of $\phi_W/2$. If any other internal clock is chosen, TCG and the noise canceller circuit will not run, and the input capture function will not operate.

### 9.6.5 Application Notes

**Input Clock Switching and TCG Operation:** Depending on when the input clock is switched, there will be cases in which TCG is incremented in the process. Table 9.17 shows the relation between internal clock switchover timing (selected in bits CKS1 and CKS0) and TCG operation. If an internal clock (derived from the system clock $\phi$ or subclock $\phi_{SUB}$) is used, an increment pulse is generated when a falling edge of the internal clock is detected. For this reason, in a case like No. 3 in table 9.17, where the clock is switched at a time such that the clock signal goes from high level before switching to low level after switching, the switchover is seen as a falling edge, a count clock pulse is generated, and TCG is incremented.

**Table 9.17 Internal Clock Switching and TCG Operation**

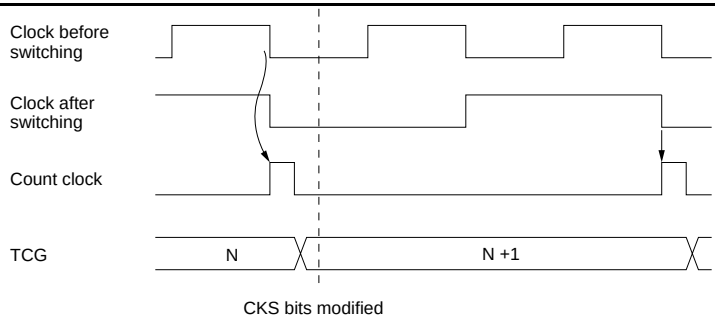
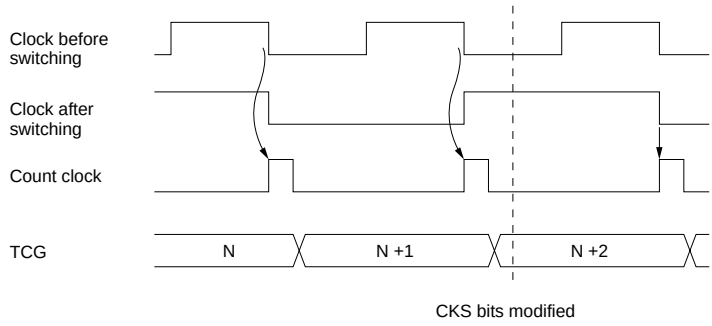
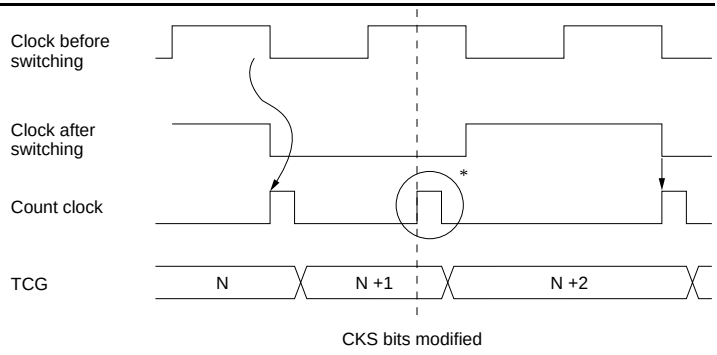
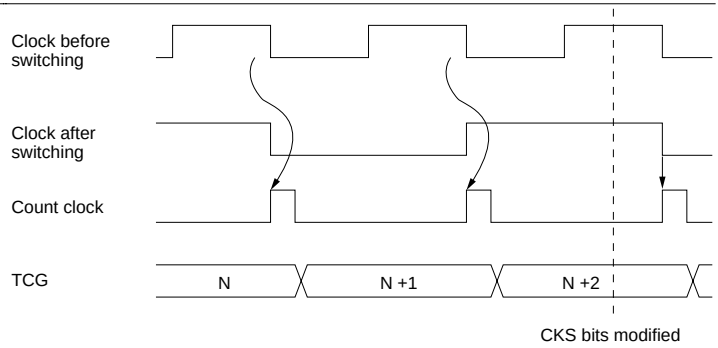
| No. | Clock Level Before and After Modifying Bits CKS1 and CKS0 | TCG Operation |
|---|---|---|
| 1 | Goes from low level to low level |  |
| 2 | Goes from low level to high level |  |

**Table 9.17 Internal Clock Switching and TCG Operation**

| No. | Clock Level Before and After Modifying Bits CKS1 and CKS0 | TCG Operation |
|---|---|---|
| 3 | Goes from high level to low level |  |
| 4 | Goes from high level to high level |  |

Note: * The switchover is seen as a falling edge of the clock pulse, and TCG is incremented.

**Note on Rewriting Port Mode Registers:** When a port mode register setting is modified to enable or disable the input capture function or input capture noise canceling function, note the following points.

- Switching the function of the input capture pin

  When the function of the input capture pin is switched by modifying the TMIG bit in port mode register 1 (PMR1) an input capture edge may be recognized even though no valid signal edge has been input. This occurs under the conditions listed in table 9.18.

**Table 9.18 Input Capture Input Signal Input Edges and Conditions by Switching of Input Capture Pin Function**

| Input Capture Edge | Conditions |
|---|---|
| Rising edge recognized | TMIG pin level is high, and TMIG bit is changed from 0 to 1 |
| | TMIG pin level is high and NCS bit is changed from 0 to 1, then TMIG bit is changed from 0 to 1 before noise canceller circuit completes five samples |
| Falling edge recognized | TMIG pin level is high, and TMIG bit is changed from 1 to 0 |
| | TMIG pin level is low and NCS bit is changed from 0 to 1, then TMIG bit is changed from 0 to 1 before noise canceller circuit completes five samples |
| | TMIG pin level is high and NCS bit is changed from 0 to 1, then TMIG bit is changed from 1 to 0 before noise canceller circuit completes five samples |

Note: When pin $P1_3$ is not used for input capture, the input capture signal input to timer G is low.

- Switching the input capture noise canceling function

  When modifying the NCS bit in port mode register 2 (PMR2) to enable or disable the input capture noise canceling function, first clear the TMIG bit to 0. Otherwise an input capture edge may be recognized even though no valid signal edge has been input. This occurs under the conditions listed in table 9.19.

**Table 9.19 Input Capture Input Signal Input Edges and Conditions by Switching of Noise Canceling Function**

| Input Capture Edge | Conditions |
|---|---|
| Rising edge recognized | TMIG bit is set to 1 and TMIG pin level changes from low to high, then NCS bit is changed from 1 to 0 before noise canceller circuit completes five samples |
| Falling edge recognized | TMIG bit is set to 1 and TMIG pin level changes from high to low, then NCS bit is changed from 1 to 0 before noise canceller circuit completes five samples |

If switching of the pin function generates a false input capture edge matching the edge selected by the input capture interrupt edge select bit (IIEGS), the interrupt request flag will be set to 1, making it necessary to clear this flag to 0 before using the interrupt function. Figure 9.15 shows the procedure for modifying port mode register settings and clearing the interrupt request flag. The first step is to mask interrupts before modifying the port mode register. After modifying the port mode register setting, wait long enough for an input capture edge to be recognized (at least two system clocks when noise canceling is disabled; at least five sampling clocks when noise canceling is enabled), then clear the interrupt request flag to 0 (assuming it has been set to 1). An alternative procedure is to avoid having the interrupt request flag set

when the pin function is switched, either by controlling the level of the input capture pin so that it does not satisfy the conditions in tables 9.18 and 9.19, or by setting the IIEGS bit of TMG to select the edge opposite to the falsely generated edge.

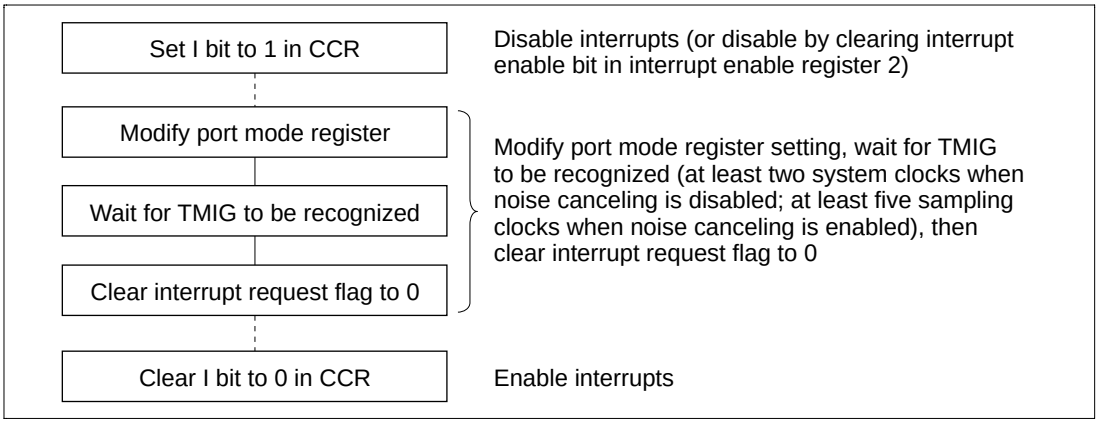



**Figure 9.15 Procedure for Modifying Port Mode Register and Clearing Interrupt Request Flag**

### 9.6.6 Sample Timer G Application

The absolute values of the high and low widths of the input capture signal can be measured by using timer G. The CCLR1 and CCLR0 bits of TMG should be set to 1. Figure 9.16 shows an example of this operation.

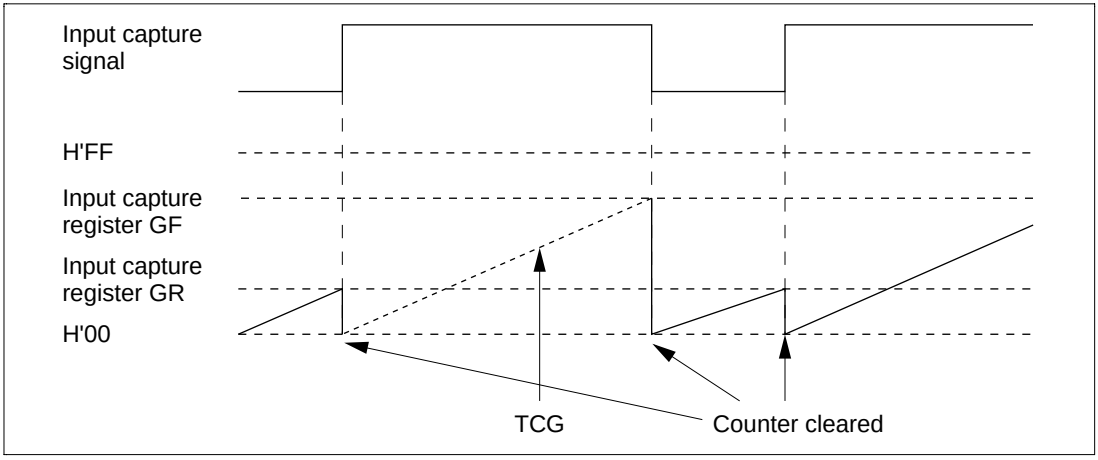



**Figure 9.16 Sample Timer G Application**

# Section 10 Serial Communication Interface

## 10.1 Overview

The H8/3834 Series is provided with a three-channel serial communication interface (SCI). Table 10.1 summarizes the functions and features of the three SCI channels.

**Table 10.1 Serial Communication Interface Functions**

| Channel | Functions | Features |
|---|---|---|
| SCI1 | Synchronous serial transfer<br>• Choice of 8-bit or 16-bit data length<br>• Continuous clock output | • Choice of 8 internal clocks ($\phi$/1024 to $\phi$/2) or external clock<br>• Open drain output possible<br>• Interrupt requested at completion of transfer |
| SCI2 | Synchronous serial transfer<br>• Automatic transfer of up to 32 bytes of data (send, receive, or simultaneous send/receive)<br>• Chip select input<br>• Strobe pulse output | • Choice of 7 internal clocks ($\phi$/256 to $\phi$/2) or external clock<br>• Open drain output possible<br>• Interrupt requested at completion of transfer or error |
| SCI3 | Synchronous serial transfer<br>• 8-bit data transfer<br>• Send, receive, or simultaneous send/receive<br>Asynchronous serial transfer<br>• Multiprocessor communication function<br>• Choice of 7-bit or 8-bit data length<br>• Choice of 1-bit or 2-bit stop bit length<br>• Parity addition | • Built-in baud rate generator<br>• Receive error detection<br>• Break detection<br>• Interrupt requested at completion of transfer or error |

## 10.2 SCI1

### 10.2.1 Overview

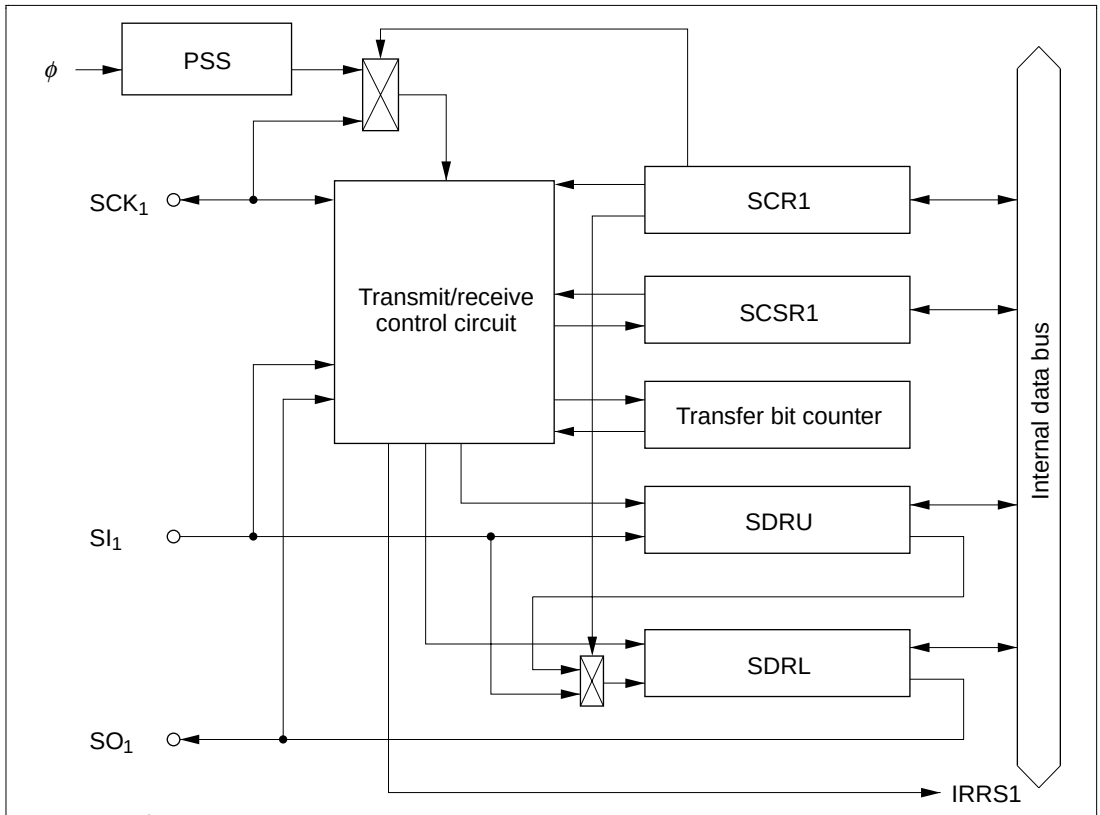
Serial communication interface 1 (SCI1) performs synchronous serial transfer of 8-bit or 16-bit data.

**Features**

- Choice of 8-bit or 16-bit data length
- Choice of eight internal clock sources ($\phi$/1024, $\phi$/256, $\phi$/64, $\phi$/32, $\phi$/16, $\phi$/8, $\phi$/4, $\phi$/2) or an external clock
- Interrupt requested at completion of transfer

**Block Diagram**

Figure 10.1 shows a block diagram of SCI1.




Notation:
SCR1: Serial control register 1
SCSR1: Serial control/status register 1
SDRU: Serial data register U
SDRL: Serial data register L
IRRS1: SCI1 interrupt request flag
PSS: Prescaler S

**Figure 10.1 SCI1 Block Diagram**

### Pin Configuration

Table 10.2 shows the SCI1 pin configuration.

**Table 10.2 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| SCI1 clock pin | $SCK_1$ | I/O | SCI1 clock input or output |
| SCI1 data input pin | $SI_1$ | Input | SCI1 receive data input |
| SCI1 data output pin | $SO_1$ | Output | SCI1 transmit data output |

### Register Configuration

Table 10.3 shows the SCI1 register configuration.

**Table 10.3 SCI1 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Serial control register 1 | SCR1 | R/W | H'00 | H'FFA0 |
| Serial control status register 1 | SCSR1 | R/W | H'80 | H'FFA1 |
| Serial data register U | SDRU | R/W | Not fixed | H'FFA2 |
| Serial data register L | SDRL | R/W | Not fixed | H'FFA3 |

## 10.2.2 Register Descriptions

### Serial Control Register 1 (SCR1)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SNC1 | SNC0 | — | — | CKS3 | CKS2 | CKS1 | CKS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

SCR1 is an 8-bit read/write register for selecting the operation mode, the transfer clock source, and the prescaler division ratio.

Upon reset, SCR1 is initialized to H'00. Writing to this register during a transfer stops the transfer.

**Bits 7 and 6—Operation Mode Select 1, 0 (SNC1, SNC0):** Bits 7 and 6 select the operation mode.

| Bit 7: SNC1 | Bit 6: SNC0 | Description | |
|---|---|---|---|
| 0 | 0 | 8-bit synchronous transfer mode | (initial value) |
| | 1 | 16-bit synchronous transfer mode | |
| 1 | 0 | Continuous clock output mode*[1] | |
| | 1 | Reserved*[2] | |

Notes: 1. Pins $SI_1$ and $SO_1$ should be used as general input or output ports.
2. Don't set bits SNC1 and SNC0 to 11.

**Bits 5 and 4—Reserved Bits:** Bits 5 and 4 are reserved, but they can be written and read.

**Bit 3—Clock Source Select 3 (CKS3):** Bit 3 selects the clock source and sets pin $SCK_1$ as an input or output pin.

| Bit 3: CKS3 | Description | |
|---|---|---|
| 0 | Clock source is prescaler S, and pin $SCK_1$ is output pin | (initial value) |
| 1 | Clock source is external clock, and pin $SCK_1$ is input pin | |

**Bits 2 to 0—Clock Select 2 to 0 (CKS2 to CKS 0):** When CKS3 = 0, bits 2 to 0 select the prescaler division ratio and the serial clock cycle.

| | | | | Serial Clock Cycle | |
|---|---|---|---|---|---|
| **Bit 2: CKS2** | **Bit 1: CKS1** | **Bit 0: CKS0** | **Prescaler Division** | **$\phi$ = 5 MHz** | **$\phi$ = 2.5 MHz** |
| 0 | 0 | 0 | $\phi$/1024 (initial value) | 204.8 µs | 409.6 µs |
| | | 1 | $\phi$/256 | 51.2 µs | 102.4 µs |
| | 1 | 0 | $\phi$/64 | 12.8 µs | 25.6 µs |
| | | 1 | $\phi$/32 | 6.4 µs | 12.8 µs |
| 1 | 0 | 0 | $\phi$/16 | 3.2 µs | 6.4 µs |
| | | 1 | $\phi$/8 | 1.6 µs | 3.2 µs |
| | 1 | 0 | $\phi$/4 | 0.8 µs | 1.6 µs |
| | | 1 | $\phi$/2 | — | 0.8 µs |

**Serial Control/Status Register 1 (SCSR1)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | SOL | ORER | — | — | — | — | STF |
| Initial value | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | R/W | R/(W)* | — | — | — | R/W | R/W |

Note: * Only a write of 0 for flag clearing is possible.

SCSR1 is an 8-bit read/write register indicating operation status and error status.

Upon reset, SCSR1 is initialized to H'80.

**Bit 7—Reserved Bit:** Bit 7 is reserved; it is always read as 1, and cannot be modified.

**Bit 6—Extended Data Bit (SOL):** Bit 6 sets the $SO_1$ output level. When read, SOL returns the output level at the $SO_1$ pin. After completion of a transmission, $SO_1$ continues to output the value of the last bit of transmitted data. The $SO_1$ output can be changed by writing to SOL before or after a transmission. The SOL bit setting remains valid only until the start of the next transmission. To control the level of the $SO_1$ pin after transmission ends, it is necessary to write to the SOL bit at the end of each transmission. Do not write to this register while transmission is in progress, because that may cause a malfunction.

| Bit 6: SOL | Description | |
|---|---|---|
| 0 | Read: $SO_1$ pin output level is low | (initial value) |
| | Write: $SO_1$ pin output level changes to low | |
| 1 | Read: $SO_1$ pin output level is high | |
| | Write: $SO_1$ pin output level changes to high | |

**Bit 5—Overrun Error Flag (ORER):** When an external clock is used, bit 5 indicates the occurrence of an overrun error. If a clock pulse is input after transfer completion, this bit is set to 1 indicating an overrun. If noise occurs during a transfer, causing an extraneous pulse to be superimposed on the normal serial clock, incorrect data may be transferred.

| Bit 5: ORER | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading ORER = 1, cleared by writing 0 to ORER | (initial value) |
| 1 | Setting conditions:<br>Set if a clock pulse is input after transfer is complete, when an external clock is used | |

**Bits 4 to 2—Reserved Bits:** Bits 4 to 2 are reserved; they are always read as 0, and cannot be modified.

**Bit 1—Reserved Bit:** Bit 1 is reserved; it should always be cleared to 0.

**Bit 0—Start Flag (STF):** Bit 0 controls the start of a transfer. Setting this bit to 1 causes SCI1 to start transferring data.

During the transfer or while waiting for the first clock pulse, this bit remains set to 1. It is cleared to 0 upon completion of the transfer. It can therefore be used as a busy flag.

| Bit 0: STF | Description | |
|---|---|---|
| 0 | Read: Indicates that transfer is stopped | (initial value) |
| | Write: Invalid | |
| 1 | Read: Indicates transfer in progress | |
| | Write: Starts a transfer operation | |

**Serial Data Register U (SDRU)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SDRU7 | SDRU6 | SDRU5 | SDRU4 | SDRU3 | SDRU2 | SDRU1 | SDRU0 |
| Initial value | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

SDRU is an 8-bit read/write register. It is used as the data register for the upper 8 bits in 16-bit transfer (SDRL is used for the lower 8 bits).

Data written to SDRU is output to SDRL starting from the least significant bit (LSB). This data is then replaced by LSB-first data input at pin $SI_1$, which is shifted in the direction from the most significant bit (MSB) toward the LSB.

SDRU must be written or read only after data transmission or reception is complete. If this register is written or read while a data transfer is in progress, the data contents are not guaranteed.

The SDRU value upon reset is not fixed.

**Serial Data Register L (SDRL)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SDRL7 | SDRL6 | SDRL5 | SDRL4 | SDRL3 | SDRL2 | SDRL1 | SDRL0 |
| Initial value | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

SDRL is an 8-bit read/write register. It is used as the data register in 8-bit transfer, and as the data register for the lower 8 bits in 16-bit transfer (SDRU is used for the upper 8 bits).

In 8-bit transfer, data written to SDRL is output from pin $SO_1$ starting from the least significant bit (LSB). This data is than replaced by LSB-first data input at pin $SI_1$, which is shifted in the direction from the most significant bit (MSB) toward the LSB.

In 16-bit transfer, operation is the same as for 8-bit transfer, except that input data is fed in via SDRU.

SDRL must be written or read only after data transmission or reception is complete. If this register is read or written while a data transfer is in progress, the data contents are not guaranteed.

The SDRL value upon reset is not fixed.

### 10.2.3 Operation

Data can be sent and received in an 8-bit or 16-bit format, synchronized to an internal or external serial clock. Overrun errors can be detected when an external clock is used.

**Clock**

The serial clock can be selected from a choice of eight internal clocks and an external clock. When an internal clock source is selected, pin $SCK_1$ becomes the clock output pin. When continuous clock output mode is selected (SCR1 bits SNC1 and SNC0 are set to 10), the clock signal ($\phi$/1024 to $\phi$/2) selected in bits CKS2 to CKS0 is output continuously from pin $SCK_1$. When an external clock is used, pin $SCK_1$ is the clock input pin.

**Data Transfer Format**

Figure 10.2 shows the data transfer format. Data is sent and received starting from the least significant bit, in LSB-first format. Transmit data is output from one falling edge of the serial clock until the next falling edge. Receive data is latched at the rising edge of the serial clock.

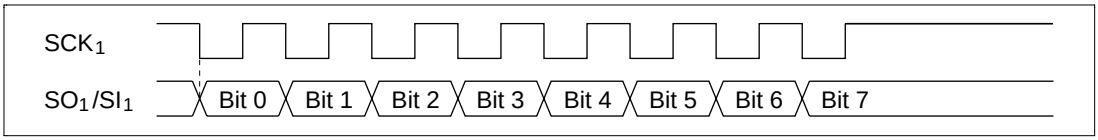



**Figure 10.2 Transfer Format**

**Data Transfer Operations**

**Transmitting:** A transmit operation is carried out as follows.

- Set bits SO1 and SCK1 in PMR3 TO 1 so that the respective pins function as $SO_1$ and $SCK_1$. If necessary, set bit POF1 in port mode register 2 (PMR2) for NMOS open drain output at pin $SO_1$.
- Clear bit SNC1 in SCR1 to 0, and set bit SNC0 to 1 or 0, designating 8- or 16-bit synchronous transfer mode. Select the serial clock in bits CKS3 to CKS0. Writing data to SCR1 initializes the internal state of SCI1.
- Write transmit data in SDRL and SDRU, as follows.
  - 8-bit transfer mode: SDRL
  - 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL
- Set the SCSR1 start flag (STF) to 1. SCI1 starts operating and outputs transmit data at pin $SO_1$.
- After data transmission is complete, bit IRRS1 in interrupt request register 1 (IRR1) is set to 1.

When an internal clock is used, a serial clock is output from pin $SCK_1$ in synchronization with the transmit data. After data transmission is complete, the serial clock is not output until the next time the start flag is set to 1. During this time, pin $SO_1$ continues to output the value of the last bit transmitted.

When an external clock is used, data is transmitted in synchronization with the serial clock input at pin $SCK_1$. After data transmission is complete, an overrun occurs if the serial clock continues to be input; no data is transmitted and the SCSR1 overrun error flag (bit ORER) is set to 1.

While transmission is stopped, the output value of pin $SO_1$ can be changed by rewriting bit SOL in SCSR1.

**Receiving:** A receive operation is carried out as follows.

- Set bits SI1 and SCK1 in PMR3 TO 1 so that the respective pins function as $SI_1$ and $SCK_1$.
- Clear bit SNC1 in SCR1 to 0, and set bit SNC0 to 1 or 0, designating 8- or 16-bit synchronous transfer mode. Select the serial clock in bits CKS3 to CKS0. Writing data to SCR1 initializes the internal state of SCI1.
- Set the SCSR1 start flag (STF) to 1. SCI1 starts operating and receives data at pin $SI_1$.
- After data reception is complete, bit IRRS1 in interrupt request register 1 (IRR1) is set to 1.

- Read the received data from SDRL and SDRU, as follows.
  — 8-bit transfer mode: SDRL
  — 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL
- After data reception is complete, an overrun occurs if the serial clock continues to be input; no data is received and the SCSR1 overrun error flag (bit ORER) is set to 1.

**Simultaneous transmit/receive:** A simultaneous transmit/receive operation is carried out as follows.

- Set bits SO1, SI1, and SCK1 in PMR3 to 1 so that the respective pins function as $SO_1$, $SI_1$, and $SCK_1$. If necessary, set bit POF1 in port mode register 2 (PMR2) for NMOS open drain output at pin $SO_1$.
- Clear bit SNC1 in SCR1 to 0, and set bit SNC0 to 1 or 0, designating 8- or 16-bit synchronous transfer mode. Select the serial clock in bits CKS3 to CKS0. Writing data to SCR1 initializes the internal state of SCI1.
- Write transmit data in SDRL and SDRU, as follows.
  — 8-bit transfer mode: SDRL
  — 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL
- Set the SCSR1 start flag (STF) to 1. SCI1 starts operating. Transmit data is output at pin $SO_1$. Receive data is input at pin $SI_1$.
- After data transmission and reception are complete, bit IRRS1 in IRR1 is set to 1.
- Read the received data from SDRL and SDRU, as follows.
  — 8-bit transfer mode: SDRL
  — 16-bit transfer mode: Upper byte in SDRU, lower byte in SDRL

When an internal clock is used, a serial clock is output from pin $SCK_1$ in synchronization with the transmit data. After data transmission is complete, the serial clock is not output until the next time the start flag is set to 1. During this time, pin $SO_1$ continues to output the value of the last bit transmitted.

When an external clock is used, data is transmitted and received in synchronization with the serial clock input at pin $SCK_1$. After data transmission and reception are complete, an overrun occurs if the serial clock continues to be input; no data is transmitted or received and the SCSR1 overrun error flag (bit ORER) is set to 1.

While transmission is stopped, the output value of pin $SO_1$ can be changed by rewriting bit SOL in SCSR1.

### 10.2.4 Interrupts

SCI1 can generate an interrupt at the end of a data transfer.

When an SCI1 transfer is complete, bit IRRS1 in interrupt request register 1 (IRR1) is set to 1. SCI1 interrupt requests can be enabled or disabled by bit IENS1 of interrupt enable register 1 (IENR1).

For further details, see 3.3, Interrupts.

### 10.2.5 Application Notes

When an external clock is input at pin $SCK_1$, and an external clock is selected for use as the clock source bit STF in SCSR1 must first be set to 1 to start data transfer before inputting the external clock.

## 10.3 SCI2

### 10.3.1 Overview

Serial communication interface 2 (SCI2) has a 32-bit data buffer for synchronous serial transfer of up to 32 bytes of data in one operation.

**Features**

Features of SCI are listed below.

- Automatic transfer of up to 32 bytes of data
- Choice of seven internal clock sources ($\phi$/256, $\phi$/64, $\phi$/32, $\phi$/16, $\phi$/8, $\phi$/4, $\phi$/2) or an external clock
- Interrupts requested at completion of transfer or when an error occurs
- Gaps of 56, 24, or 8 internal clock cycles can be inserted between successive bytes of transferred data.
- Transfer can be started by chip select input.
- A strobe pulse can be output for each byte transferred.

**Block Diagram**

Figure 10.3 shows a block diagram of SCI2.

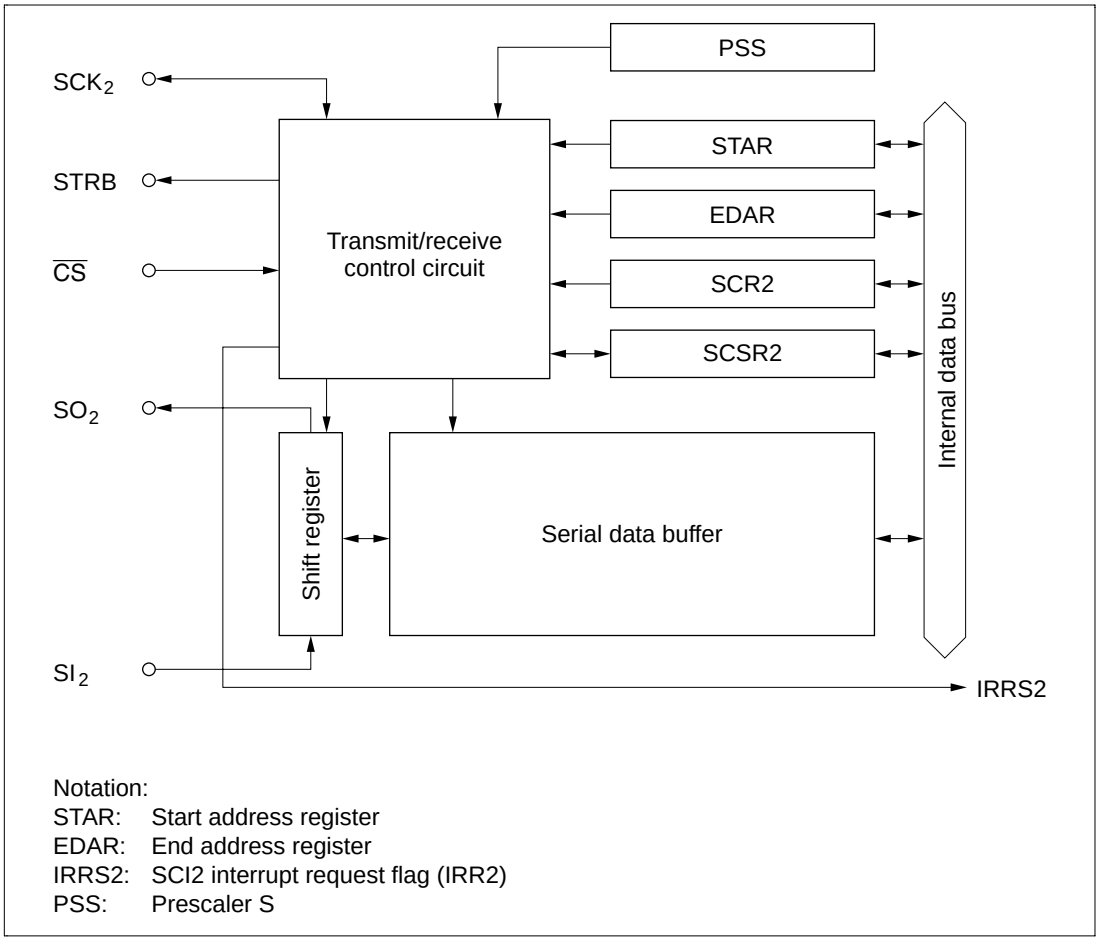



**Figure 10.3 SCI2 Block Diagram**

**Pin Configuration**

Table 10.4 shows the SCI2 pin configuration.

**Table 10.4 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| SCI2 clock pin | $SCK_2$ | I/O | SCI2 clock input/output |
| SCI2 data input pin | $SI_2$ | Input | SCI2 receive data input |
| SCI2 data output pin | $SO_2$ | Output | SCI2 transmit data output |
| SCI2 strobe pin | STRB | Output | SCI2 strobe signal output |
| SCI2 chip select pin | $\overline{CS}$ | Input | SCI2 chip select signal input |

### Register Configuration

Table 10.5 shows the SCI2 register configuration.

**Table 10.5 SCI2 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Start address register | STAR | R/W | H'E0 | H'FFA4 |
| End address register | EDAR | R/W | H'E0 | H'FFA5 |
| Serial control register 2 | SCR2 | R/W | H'E0 | H'FFA6 |
| Serial control/status register 2 | SCSR2 | R/W | H'E0 | H'FFA7 |
| Serial data buffer (32 bytes) | — | R/W | Not fixed | H'FF80 to H'FF9F |

## 10.3.2 Register Descriptions

### Start Address Register (STAR)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | STA4 | STA3 | STA2 | STA1 | STA0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

STAR is an 8-bit read/write register, for designating a transfer start address in the address space (H'FF80 to H'FF9F) allocated to the 32-byte data buffer. The lower 5 bits of STAR correspond to the lower 5 bits of the address. The extent of continuous data transfer is defined in STAR and in the end address register (EDAR). If the same value is designated by STAR and EDAR, only 1 byte of data is transferred.

Bits 7 to 5 are reserved; they are always read as 1, and cannot be modified.

Upon reset, STAR is initialized to H'E0.

**End Address Register (EDAR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | EDA4 | EDA3 | EDA2 | EDA1 | EDA0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

EDAR is an 8-bit read/write register, for designating a transfer end address in the address space (H'FF80 to H'FF9F) allocated to the 32-byte data buffer. The lower 5 bits of EDAR correspond to the lower 5 bits of the address. The extent of continuous data transfer is defined in STAR and in EDAR. If the same value is designated by STAR and EDAR, only 1 byte of data is transferred.

Bits 7 to 5 are reserved; they are always read as 1, and cannot be modified.

Upon reset, EDAR is initialized to H'E0.

**Serial Control Register 2 (SCR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | GAP1 | GAP0 | CKS2 | CKS1 | CKS0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

SCR2 is an 8-bit read/write register for selecting the serial clock, and for setting the gap inserted between data during continuous transfer when SCI2 uses an internal clock.

Upon reset, SCR2 is initialized to H'E0.

**Bits 7 to 5—Reserved Bits:** Bits 7 to 5 are reserved; they are always read as 1, and cannot be modified.

**Bits 4 and 3—Gap Select 1, 0 (GAP1 to GAP0):** When SCI2 uses an internal clock, gaps can be inserted between successive data bytes. Bits 4 and 3 designate the length of these gaps. During a gap, pin $SCK_2$ remains at the high level. When no gap is inserted, the STRB signal stays at the low level.

| Bit 4: GAP1 | Bit 3: GAP0 | Description |
|---|---|---|
| 0 | 0 | No gaps between bytes (initial value) |
| | 1 | A gap of 8 clock cycles is inserted between bytes |
| 1 | 0 | A gap of 24 clock cycles is inserted between bytes |
| | 1 | A gap of 56 clock cycles is inserted between bytes |

**Bits 2 to 0—Clock Select 2 to 0 (CKS2 to CKS0):** Bits 2 to 0 select one of seven internal clock sources or an external clock as the transfer clock.

| Bit 2: CKS2 | Bit 1: CKS1 | Bit 0: CKS0 | Pin $SCK_2$ | Clock Source | Prescaler Division | Serial Clock Cycle | |
|---|---|---|---|---|---|---|---|
| | | | | | | $\phi$ = 5 MHz | $\phi$ = 2.5 MHz |
| 0 | 0 | 0 | $SCK_2$ output | Prescaler S | $\phi$/256 (initial value) | 51.2 µs | 102.4 µs |
| | | 1 | | | $\phi$/64 | 12.8 µs | 25.6 µs |
| | 1 | 0 | | | $\phi$/32 | 6.4 µs | 12.8 µs |
| | | 1 | | | $\phi$/16 | 3.2 µs | 6.4 µs |
| 1 | 0 | 0 | | | $\phi$/8 | 1.6 µs | 3.2 µs |
| | | 1 | | | $\phi$/4 | 0.8 µs | 1.6 µs |
| | 1 | 0 | | | $\phi$/2 | — | 0.8 µs |
| | | 1 | $SCK_2$ input | External clock | — | — | — |

**Serial Control/Status Register 2 (SCSR2)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | SOL | ORER | WT | ABT | STF |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/(W)* | R/(W)* | R/(W)* | R/W |

Note: * Only a write of 0 for flag clearing is possible.

SCSR2 is an 8-bit register indicating SCI2 operation status and error status.

Upon reset, SCSR2 is initialized to H'E0.

**Bits 7 to 5—Reserved Bits:** Bits 7 to 5 are reserved; they are always read as 1, and cannot be modified.

**Bit 4—Extended Data Bit (SOL):** Bit 4 sets the $SO_2$ output level. When read, SOL returns the transmitted data output at the $SO_2$ pin. After completion of a transmission, $SO_2$ continues to output the value of the last bit of transmitted data. The $SO_2$ output can be changed by writing to SOL before or after a transmission. The SOL bit setting remains valid only until the start of the next transmission. To control the level of the $SO_2$ pin after transmission ends, it is necessary to write to the SOL bit at the end of each transmission. Note that if the STF bit is cleared to 0 to terminate a transmission in progress, the transmitted data will be modified when the bit is cleared.

| Bit 4: SOL | Description | |
|---|---|---|
| 0 | Read: $SO_2$ pin output level is low | (initial value) |
| | Write: $SO_2$ pin output level changes to low | |
| 1 | Read: $SO_2$ pin output level is high | |
| | Write: $SO_2$ pin output level changes to high | |

**Bit 3—Overrun Error Flag (ORER):** When an external clock is used, bit 3 indicates the occurrence of an overrun error. If a clock pulse is input after transfer completion, this bit is set to 1 indicating an overrun. If noise occurs during a transfer, causing an extraneous pulse to be superimposed on the normal serial clock, incorrect data may be transferred. Overrun errors are not detected while pin $\overline{CS}$ is at the high level.

| Bit 3: ORER | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading ORER = 1, cleared by writing 0 to ORER | (initial value) |
| 1 | Setting conditions:<br>Set if a clock pulse is input after transfer is complete, when an external clock is used | |

**Bit 2—Wait Flag (WT):** Bit 2 indicates that an attempt was made to read or write the 32-byte serial data buffer while a transfer was in progress, or while waiting for $\overline{CS}$ input. The read or write access is not carried out, and this bit is set to 1.

| Bit 2: WT | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading WT = 1, cleared by writing 0 to WT | (initial value) |
| 1 | Setting conditions:<br>An attempt was made to read or write the (32-byte) serial data buffer during a transfer operation or while waiting for $\overline{CS}$ input | |

**Bit 1—Abort Flag (ABT):** Bit 1 indicates that $\overline{CS}$ went to high during data transfer. When the $\overline{CS}$ input function is selected, if a high-level signal is detected at pin $\overline{CS}$ during a transfer, the transfer is immediately aborted and this bit is set to 1. At the same time bit IRRS2 in interrupt request register 2 (IRR2) is set
to 1, and pins $SCK_2$ and $SO_2$ go to the high-impedance state. Data in the (32-byte) serial data buffer and values in the internal registers other than SCSR2 remain unchanged.

Transfer cannot take place while this bit is set to 1. It must be cleared to 0 before resuming the transfer.

| Bit 1: ABT | Description | |
|---|---|---|
| 0 | Clearing conditions:<br>After reading ABT = 1, cleared by writing 0 to ABT | (initial value) |
| 1 | Setting conditions:<br>When pin $\overline{CS}$ goes high during a transfer | |

**Bit 0—Start Flag (STF):** Bit 0 controls the start of a transfer. If bit CS = 0 in PMR2, setting bit STF to 1 causes SCI2 to start transferring data. If bit CS = 1 in PMR2, SCI2 starts transferring data when $\overline{CS}$ goes low. This bit stays at 1 during the transfer or while waiting for $\overline{CS}$ input; it is cleared to 0 after the transfer is completed or when the transfer is aborted by $\overline{CS}$. It can therefore be used as a busy flag.

Clearing this bit to 0 during a transfer aborts the transfer. The contents of the (32-byte) serial data buffer and of internal registers other than SCSR2 remain unchanged.

| Bit 0: STF | Description | |
|---|---|---|
| 0 | Read: Indicates that transfer is stopped | (initial value) |
| | Write: Stops a transfer operation | |
| 1 | Read: Indicates transfer in progress or waiting for $\overline{CS}$ input | |
| | Write: Starts a transfer operation | |

### 10.3.3 Operation

SCI2 has a 32-byte serial data buffer, making possible continuous transfer of up to 32 bytes of data with one operation. SCI2 transmits and receives data in synchronization with clock pulses. Depending on register settings, it can transmit, receive, or transmit and receive simultaneously. When transmission is set, the serial data buffer values are retained after the transmission is completed.

Either an internal clock or external clock may be selected as the serial clock. When an internal clock is selected, gaps may be inserted between the data bytes. It is also possible to output a strobe signal at pin STRB. When an external clock is selected, the overrun flag allows detection of erroneous operation due to unwanted clock input.

Transfers can be started or aborted by input at pin $\overline{CS}$. Abort is indicated by means of an abort flag.

**Clock**

The serial clock can be selected from a choice of six internal clock sources or an external clock. When an internal clock source is selected, pin $SCK_2$ becomes the clock output pin.

**Data Transfer Format**

Figure 10.4 and figure 10.5 show the SCI2 data transfer format. Data is sent and received starting from the least significant bit, in LSB-first format. Transmit data is output from one falling edge of the serial clock until the next falling edge. Receive data is latched at the rising edge of the serial clock.

When SCI2 operates on an internal clock, a gap can be inserted between each byte of transferred data and the next, as shown in figure 10.5. During this gap, pin $SCK_2$ output remains high. Also, a strobe pulse can be output at pin STRB.

The length of the gap is designated in bits GAP1 and GAP0 in serial control register 2 (SCR2).

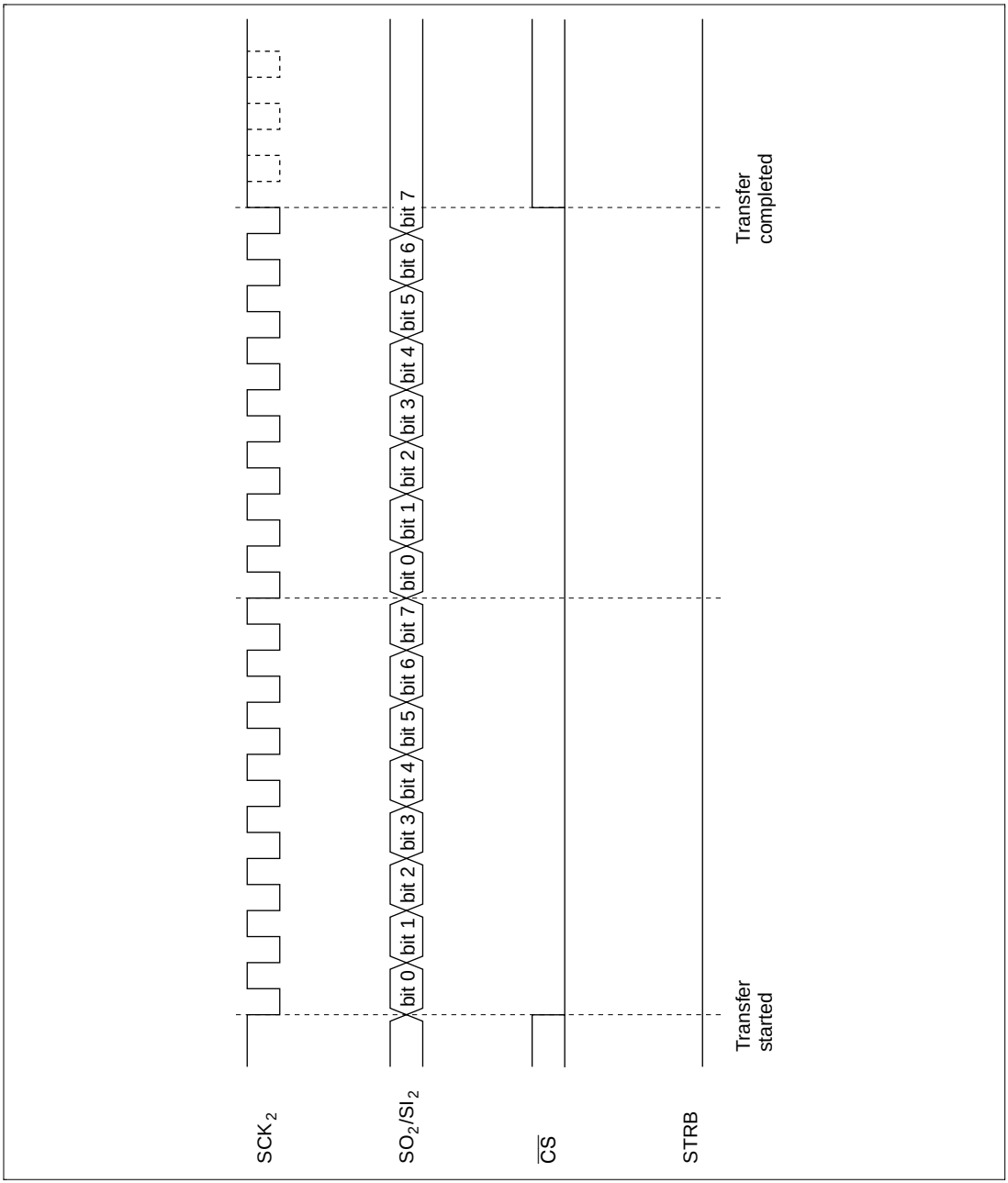



**Figure 10.4 Data Transfer Format (No Gaps between Data)**

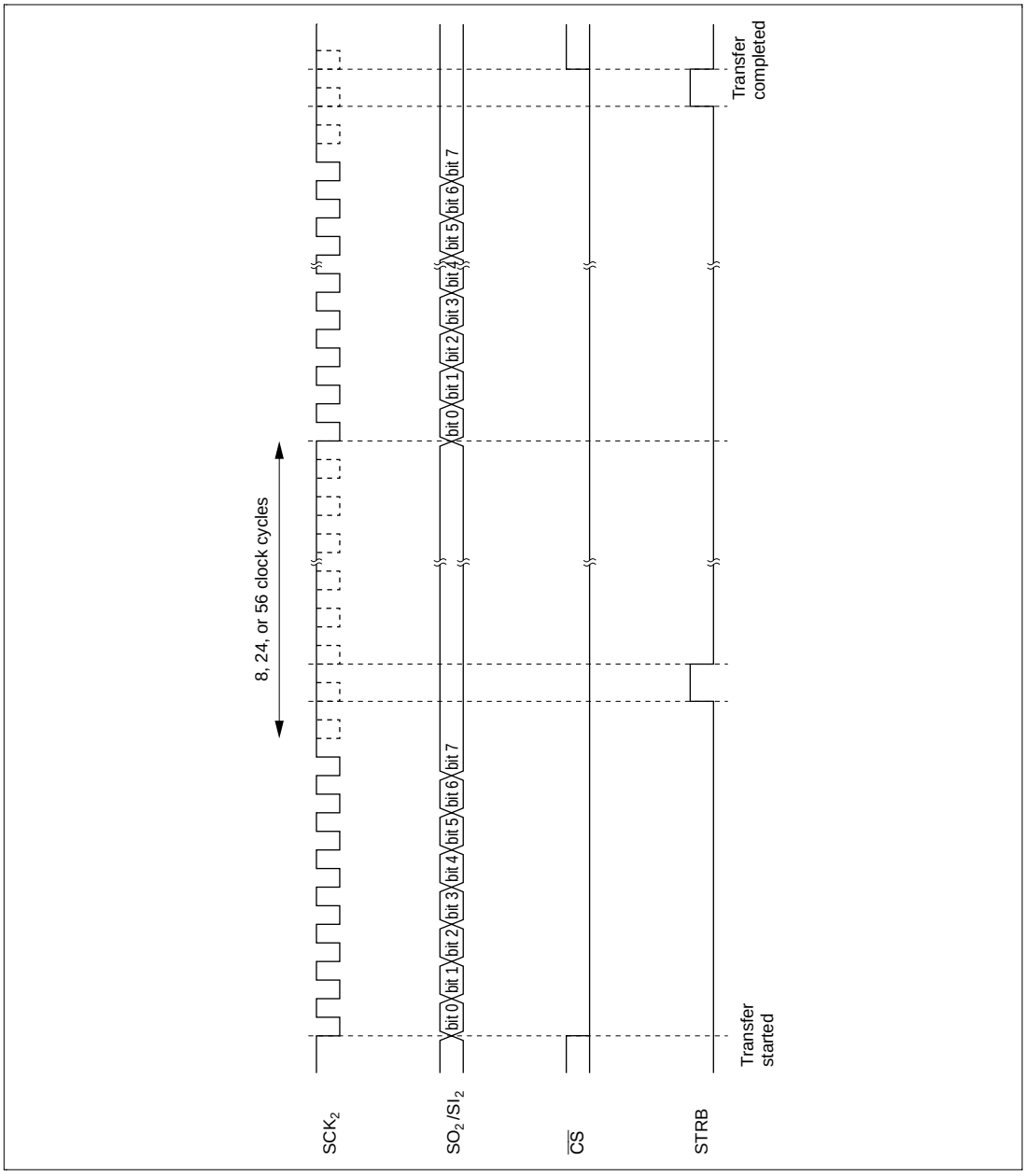



**Figure 10.5 Data Transfer Format (Gap Inserted between Data)**

**Data Transfer Operations**

**SCI2 Initialization:** Data transfer on SCI2 first of all requires that SCI2 be initialized by software as follows.

- With bit STF cleared to 0 in SCSR2, select pin functions and the transfer mode in registers PMR2, PMR3, STAR, EDAR, and SCR2.
- The SCI2 pins double as general input/output ports. Switching between port and SCI2 functions is controlled in PMR3. CMOS output or NMOS open drain output can be selected in PMR2. The serial clock and gaps between transferred bytes are set in SCR2.
- The start and end addresses of the transfer data area are set in STAR and EDAR. If the end address is set smaller than the start address, as shown in figure 10.6, the transfer wraps around from H'FF9F to H'FF80 and continues to the end address. If the start address and end address are the same, only one byte of data will be transferred.

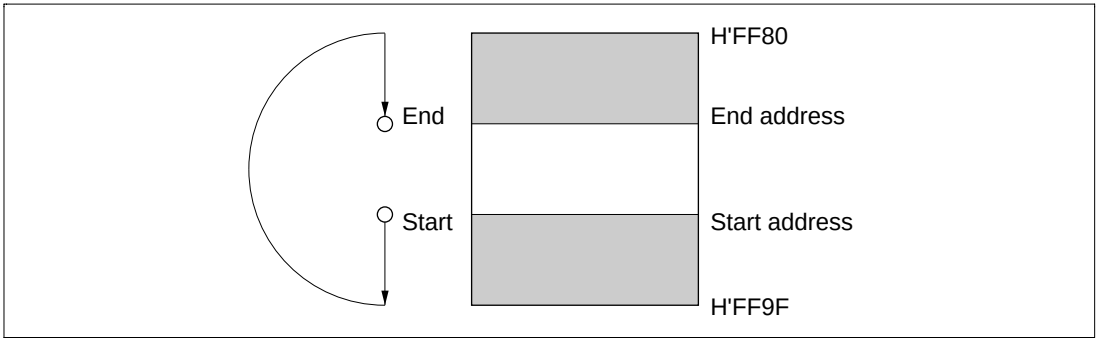



**Figure 10.6 Operation When End Address is Smaller than Start Address**

**Transmitting:** A transmit operation is carried out as follows.

- Set bits SO2 and SCK2 in PMR3 TO 1 so that the respective pins function as $SO_2$ and $SCK_2$. If necessary, set bit POF2 in port mode register 2 (PMR2) for NMOS open-drain output at pin $SO_2$, and set bits CS and STRB in PMR3 to designate use of the $\overline{CS}$ and STRB pin functions.
- Select the serial clock and, in the case of internal clock operation, the data gap in SCR2.
- Write transmit data in the serial data buffer. This data will remain in the data buffer after completion of the transfer. It is not necessary to rewrite the buffer when the same data is retransmitted.
- Set the start address in the lower 5 bits of STAR, and the end address in the lower 5 bits of EDAR.
- Set the start/busy flag (STF) to 1. If bit CS = 0 in PMR3, transmission starts as soon as STF is set to 1. If CS = 1 in PMR3, transmission starts when $\overline{CS}$ goes low.
- After data transmission is complete, bit IRRS2 in interrupt request register 2 (IRR2) is set to 1, and bit STF is cleared to 0.

When an internal clock is used, a serial clock is output from pin $SCK_2$ in synchronization with the transmit data. After data transmission is completed, the serial clock is not output until bit STF is set again. During this time, pin $SO_2$ continues to output the value of the last bit transmitted.

When an external clock is used, data is transmitted in synchronization with the serial clock input at pin $SCK_2$. After data transmission is completed, an overrun occurs if the serial clock continues to be input; no data is transmitted and the SCSR2 overrun error flag (bit ORER) is set to 1. Pin $SO_2$ continues to output the value of the last preceding bit. Overrun errors are not detected when both pin $\overline{CS}$ is at the high level and PMR3 bit CS = 1.

While transmission is stopped, the output value of pin $SO_2$ can be changed by rewriting bit SOL in SCSR2.

During a transmission or while waiting for $\overline{CS}$ input, the CPU cannot read or write the data buffer. If a read instruction is executed, H'FF will be read; if a write instruction is executed, the buffer contents will not change. In either case the wait flag (bit WT) in SCSR2 will be set to 1.

If bit CS = 1 in PMR3 and during transmission a high-level signal is detected at pin $\overline{CS}$, the transmit operation will immediately be aborted, setting the abort flag (bit ABT) to 1. At the same time bit IRRS2 in interrupt request register 2 (IRR2) will be set to 1, and bit STF will be cleared to 0. Pins $SCK_2$ and $SO_2$ will go to the high-impedance state. Data transfer is not possible while bit ABT is set to 1. It must be cleared before resuming the transfer.

**Receiving:** A receive operation is carried out as follows.

- Set bits SI2 and SCK2 in port mode register 3 (PMR3) to 1, designating use of the $SI_2$ and $SCK_2$ pin functions. If necessary, set bit CS in PMR3 to select the $\overline{CS}$ pin function.
- Select the serial clock and, in the case of internal clock operation, the data gap in SCR2.
- Allocate an area to hold the received data in the serial data buffer by designating the receive start address in the lower 5 bits of the start address register (STAR) and the receive end address in the lower 5 bits of the end address register (EDAR).
- Set the start/busy flag (bit STF) to 1. If bit CS = 0 in PMR3, receiving starts as soon as STF is set. If CS = 1 in PMR3, receiving starts when $\overline{CS}$ goes low.
- After receiving is completed, bit IRRS2 in interrupt request register 2 (IRR2) is set to 1, and bit STF is cleared to 0.
- Read the received data from the serial data buffer.

If an internal clock is used, a serial clock is output from pin $SCK_2$ when the receive operation starts. After receiving is completed, the serial clock is not output until bit STF is set again. When an external clock source is used, data is received in synchronization with the clock input at pin $SCK_2$. After receiving is completed, an overrun occurs if the serial clock continues to be input; no further data is received and the SCSR2 overrun error flag (bit ORER) is set to 1. Overrun errors are not detected when both pin $\overline{CS}$ is high and bit CS = 1 in PMR3.

While receiving or while waiting for $\overline{CS}$ input, the CPU cannot read or write the data buffer. If a read instruction is executed, H'FF will be read; if a write instruction is executed the buffer contents will not change. In either case the wait flag (bit WT) in SCSR2 will be set.

If bit CS = 1 in PMR3 and a high-level signal is detected at pin $\overline{CS}$ during receiving, the receive operation will immediately be aborted, setting the abort flag (bit ABT) to 1. At the same time bit IRRS2 in interrupt request register 2 (IRR2) will be set to 1, and bit STF will be cleared to 0. Pins $SCK_2$ and $SO_2$ will go to the high-impedance state. Data transfer is not possible while bit ABT is set to 1. It must be cleared before resuming the transfer.

**Simultaneous Transmit/Receive:** A simultaneous transmit/receive operation is carried out as follows.

- Set bits SO2, SI2, and SCK2 in PMR3 to 1, designating use of the $SO_2$, $SI_2$, and $SCK_2$ pin functions. If necessary, set bit POF2 in port mode register 2 (PMR2) for NMOS open-drain output at pin $SO_2$, and set bits CS and STRB to designate use of the $\overline{CS}$ and STRB pin functions.
- Select the transfer clock and, in the case of internal clock operation, the data gap in SCR2.
- Write transmit data in the serial data buffer. In simultaneous transmit/receive, received data replaces transmitted data at the same buffer addresses.
- Set the transfer start address in the lower 5 bits of STAR, and the transfer end address in the lower 5 bits of EDAR.
- Set the start/busy flag (bit STF) to 1. If bit CS = 0 in PMR3, the transfer starts as soon as STF is set. If CS = 1 in PMR3, transfer operations start when $\overline{CS}$ goes low.
- After data transfer is completed, bit IRRS2 in interrupt request register 2 (IRR2) is set to 1, and bit STF is cleared to 0.
- Read the received data from the serial data buffer.

If an internal clock is used, a serial clock is output from pin $SCK_2$ when the transfer begins. After the transfer is completed, the serial clock is not output until bit STF is set again. During this time, pin $SO_2$ continues to output the value of the last bit transmitted.

When an external clock is used, data is transferred in synchronization with the serial clock input at pin $SCK_2$. After the transfer is completed, an overrun occurs if the serial clock continues to be input; no transfer operation takes place and the SCSR2 overrun error flag (bit ORER) is set to 1. Pin $SO_2$ continues to output the value of the last transmitted bit. Overrun errors are not detected when both pin $\overline{CS}$ is high and bit CS = 1 in PMR3.

While data transfer is stopped, the output value of pin $SO_2$ can be changed by rewriting bit SOL in SCSR2.

During a transfer or while waiting for $\overline{CS}$ input, the CPU cannot read or write the data buffer. If a read instruction is executed, H'FF will be read; if a write instruction is executed the buffer contents will not change. In either case the wait flag (bit WT) in SCSR2 will be set.

If bit CS = 1 in PMR3 and during the transfer a high-level signal is detected at pin CS, the transfer will immediately be aborted, setting the abort flag (bit ABT) to 1. At the same time bit IRRS2 in interrupt request register 2 (IRR2) will be set to 1, and bit STF will be cleared to 0. Pins $SCK_2$ and $SO_2$ will go to the high-impedance state. Data transfer is not possible while bit ABT is set to 1. It must be cleared before resuming the transfer.

### 10.3.4 Interrupts

SCI2 can generate interrupts when a transfer is completed or when a transfer is aborted by CS. These interrupts have the same vector address.

When the above conditions occur, bit IRRS2 in interrupt request register 2 (IRR2) is set to 1. SCI2 interrupt requests can be enabled or disabled in bit IENS2 of interrupt enable register 2 (IENR2). For further details, see 3.3, Interrupts.

When a transfer is aborted by $\overline{CS}$, an overrun error occurs, or a read or write of the serial data buffer is attempted during a transfer or while waiting for $\overline{CS}$ input, the ABT, ORER, or WT bit in SCSR2 is set to 1. These bits can be used to determine the cause of the error.

### 10.3.5 Application Notes

When an external clock is input at pin $SCK_2$, and an external clock is selected for use as the clock source bit STF in SCSR2 must first be set to 1 to start data transfer before inputting the external clock.

## 10.4 SCI3

### 10.4.1 Overview

Serial communication interface 3 (SCI3) has both synchronous and asynchronous serial data communication capabilities. It also has a multiprocessor communication function for serial data communication among two or more processors.

**Features**

SCI3 features are listed below.

- Selection of asynchronous or synchronous mode
  - — Asynchronous mode

    Serial data communication is performed using an asynchronous method in which synchronization is established character by character.

    SCI3 can communicate with a UART (universal asynchronous receiver/transmitter), ACIA (asynchronous communication interface adapter), or other chip that employs standard asynchronous serial communication. It can also communicate with two or more other processors using the multiprocessor communication function. There are twelve selectable serial data communication formats.
    - Data length: seven or eight bits
    - Stop bit length: one or two bits
    - Parity: even, odd, or none
    - Multiprocessor bit: one or none
    - Receive error detection: parity, overrun, and framing errors
    - Break detection: by reading the RXD level directly when a framing error occurs
  - — Synchronous mode

    Serial data communication is synchronized with a clock signal. SCI3 can communicate with other chips having a clocked synchronous communication function.
    - Data length: eight bits
    - Receive error detection: overrun errors
- Full duplex communication

  The transmitting and receiving sections are independent, so SCI3 can transmit and receive simultaneously. Both sections use double buffering, so continuous data transfer is possible in both the transmit and receive directions.
- Built-in baud rate generator with selectable bit rates.
- Internal or external clock may be selected as the transfer clock source.
- There are six interrupt sources: transmit end, transmit data empty, receive data full, overrun error, framing error, and parity error.

**Block Diagram**

Figure 10.7 shows a block diagram of SCI3.

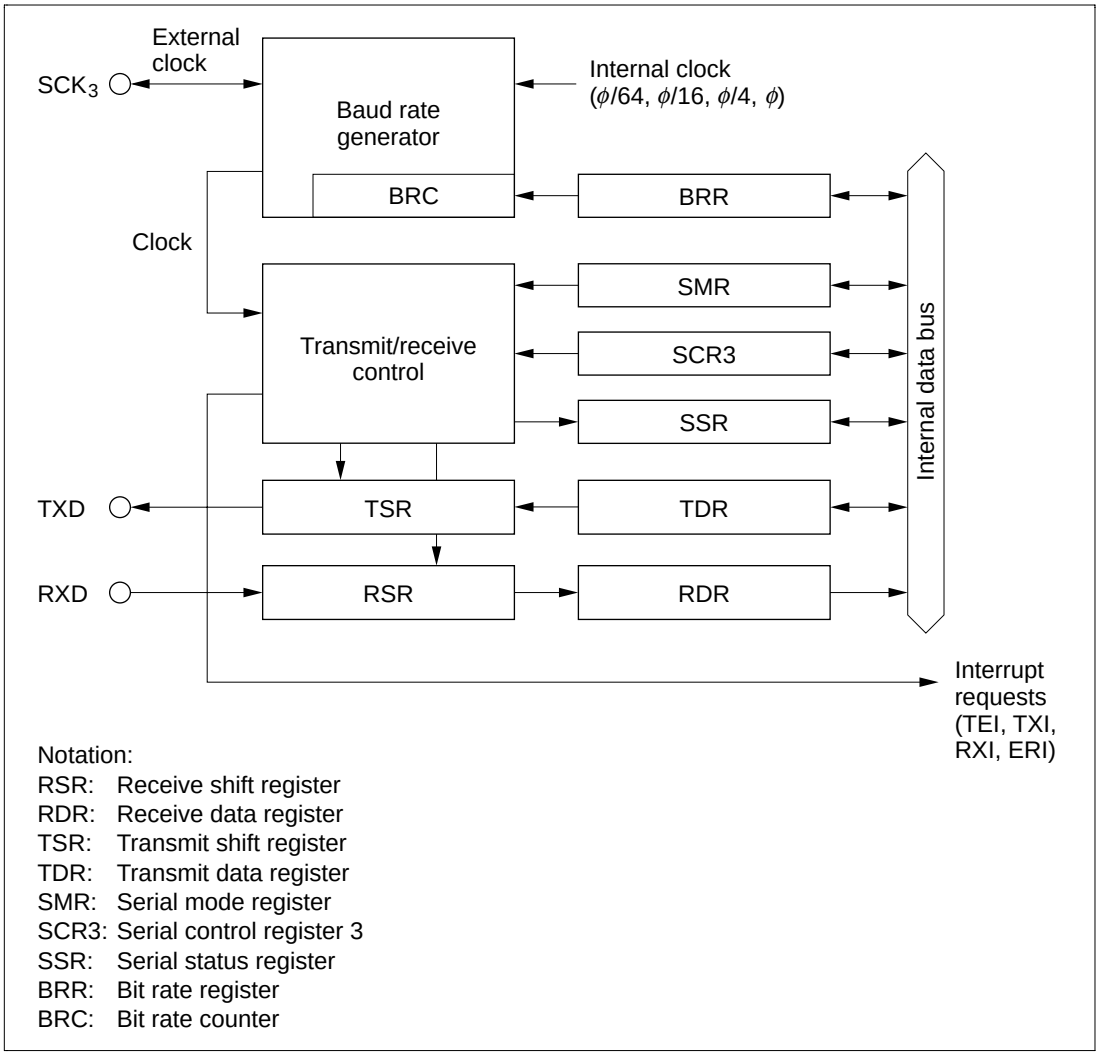



**Figure 10.7 SCI3 Block Diagram**

**Pin Configuration**

Table 10.6 shows the SCI3 pin configuration.

**Table 10.6 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| SCI3 clock | $SCK_3$ | 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 |

**Register Configuration**

Table 10.7 shows the SCI3 internal register configuration.

**Table 10.7 SCI3 Registers**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| Serial mode register | SMR | R/W | H'00 | H'FFA8 |
| Bit rate register | BRR | R/W | H'FF | H'FFA9 |
| Serial control register 3 | SCR3 | R/W | H'00 | H'FFAA |
| Transmit data register | TDR | R/W | H'FF | H'FFAB |
| Serial status register | SSR | R/W | H'84 | H'FFAC |
| Receive data register | RDR | R | H'00 | H'FFAD |
| Transmit shift register | TSR | * | — | — |
| Receive shift register | RSR | * | — | — |
| Bit rate counter | BRC | * | — | — |

Note: —: Cannot be read or written.

### 10.4.2 Register Descriptions

**Receive Shift Register (RSR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Read/Write | — | — | — | — | — | — | — | — |

The receive shift register (RSR) is for receiving serial data.

Serial data is input in LSB (bit 0) order into RSR from pin RXD, converting it to parallel data. After each byte of data has been received, the byte is automatically transferred to the receive data register (RDR).

RSR cannot be read or written directly by the CPU.

**Receive Data Register (RDR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RDR7 | RDR6 | RDR5 | RDR4 | RDR3 | RDR2 | RDR1 | RDR0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

The receive data register (RDR) is an 8-bit register for storing received serial data.

Each time a byte of data is received, the received data is transferred from the receive shift register (RSR) to RDR, completing a receive operation. Thereafter RSR again becomes ready to receive new data. RSR and RDR form a double buffer mechanism that allows data to be received continuously.

RDR is exclusively for receiving data and cannot be written by the CPU.

RDR is initialized to H'00 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

**Transmit Shift Register (TSR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | | |
| Read/Write | — | — | — | — | — | — | — | — |

The transmit shift register (TSR) is for transmitting serial data.

Transmit data is first transferred from the transmit data register (TDR) to TSR, then is transmitted from pin TXD, starting from the LSB (bit 0).

After one byte of data has been sent, the next byte is automatically transferred from TDR to TSR, and the next transmission begins. If no data has been written to TDR (1 is set in TDRE), there is no data transfer from TDR to TSR.

TSR cannot be read or written directly by the CPU.

**Transmit Data Register (TDR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TDR7 | TDR6 | TDR5 | TDR4 | TDR3 | TDR2 | TDR1 | TDR0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

The transmit data register (TDR) is an 8-bit register for holding transmit data.

When SCI3 detects that the transmit shift register (TSR) is empty, it shifts transmit data written in TDR to TSR and starts serial data transmission. While TSR is transmitting serial data, the next byte to be transmitted can be written to TDR, realizing continuous transmission.

TDR can be read or written by the CPU at all times.

TDR is initialized to H'FF upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

**Serial Mode Register (SMR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COM | CHR | PE | PM | STOP | MP | CKS1 | CKS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

The serial mode register (SMR) is an 8-bit register for setting the serial data communication format and for selecting the clock source of the baud rate generator. SMR can be read and written by the CPU at any time.

SMR is initialized to H'00 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

**Bit 7—Communication Mode (COM):** Bit 7 selects asynchronous mode or synchronous mode as the serial data communication mode.

| Bit 7: COM | Description | |
|---|---|---|
| 0 | Asynchronous mode | (initial value) |
| 1 | Synchronous mode | |

**Bit 6—Character Length (CHR):** Bit 6 selects either 7 bits or 8 bits as the data length in asynchronous mode. In synchronous mode the data length is always 8 bits regardless of the setting here.

| Bit 6: CHR | Description | |
|---|---|---|
| 0 | 8-bit data | (initial value) |
| 1 | 7-bit data* | |

Note: * When 7-bit data is selected as the character length in asynchronous mode, the MSB (bit 7) in the transmit data register is not transmitted.

**Bit 5—Parity Enable (PE):** In asynchronous mode, bit 5 selects whether or not a parity bit is to be added to transmitted data and checked in received data. In synchronous mode there is no adding or checking of parity regardless of the setting here.

| Bit 5: PE | Description | |
|---|---|---|
| 0 | Parity bit adding and checking disabled | (initial value) |
| 1 | Parity bit adding and checking enabled* | |

Note: * When PE is set to 1, then either odd or even parity is added to transmit data, depending on the setting of the parity mode bit (PM). When data is received, it is checked for odd or even parity as designated in bit PM.

**Bit 4—Parity Mode (PM):** In asynchronous mode, bit 4 selects whether odd or even parity is to be added to transmitted data and checked in received data. The PM setting is valid only if bit PE is set to 1, enabling parity adding/checking. In synchronous mode, or if parity adding/checking is disabled in asynchronous mode, the PM setting is invalid.

| Bit 4: PM | Description | |
|---|---|---|
| 0 | Even parity*[1] | (initial value) |
| 1 | Odd parity*[2] | |

Notes: 1. When even parity is designated, a parity bit is added to the transmitted data so that the sum of 1s in the resulting data is an even number. When data is received, the sum of 1s in the data plus parity bit is checked to see if the result is an even number.

2. When odd parity is designated, a parity bit is added to the transmitted data so that the sum of 1s in the resulting data is an odd number. When data is received, the sum of 1s in the data plus parity bit is checked to see if the result is an odd number.

**Bit 3—Stop Bit Length (STOP):** Bit 3 selects 1 bit or 2 bits as the stop bit length in asynchronous mode. This setting is valid only in asynchronous mode. In synchronous mode a stop bit is not added, so this bit is ignored.

| Bit 3: STOP | Description | |
|---|---|---|
| 0 | 1 stop bit*[1] | (initial value) |
| 1 | 2 stop bits*[2] | |

Notes: 1. When data is transmitted, one 1 bit is added at the end of each transmitted character as the stop bit.
2. When data is transmitted, two 1 bits are added at the end of each transmitted character as the stop bits.

When data is received, only the first stop bit is checked regardless of the stop bit length. If the second stop bit value is 1 it is treated as a stop bit; if it is 0, it is treated as the start bit of the next character.

**Bit 2—Multiprocessor Mode (MP):** Bit 2 enables or disables the multiprocessor communication function. When the multiprocessor communication function is enabled, the parity enable (PE) and parity mode (PM) settings are ignored. The MP bit is valid only in asynchronous mode; it should be cleared to 0 in synchronous mode.

See 10.4.6, for details on the multiprocessor communication function.

| Bit 2: MP | Description | |
|---|---|---|
| 0 | Multiprocessor communication function disabled | (initial value) |
| 1 | Multiprocessor communication function enabled | |

**Bits 1 and 0—Clock Select 1, 0 (CKS1, CKS0):** Bits 1 and 0 select the clock source for the built-in baud rate generator. A choice of $\phi/64$, $\phi/16$, $\phi/4$, or $\phi$ is made in these bits.

See 8, Bit rate register (BRR), below for information on the clock source and bit rate register settings, and their relation to the baud rate.

| Bit 1: CKS1 | Bit 0: CKS0 | Description | |
|---|---|---|---|
| 0 | 0 | $\phi$ clock | (initial value) |
| | 1 | $\phi/4$ clock | |
| 1 | 0 | $\phi/16$ clock | |
| | 1 | $\phi/64$ clock | |

**Serial Control Register 3 (SCR3)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Serial control register 3 (SCR3) is an 8-bit register that controls SCI3 transmit and receive operations, enables or disables serial clock output in asynchronous mode, enables or disables interrupts, and selects the serial clock source. SCR3 can be read and written by the CPU at any time.

SCR3 is initialized to H'00 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

**Bit 7—Transmit Interrupt Enable (TIE):** Bit 7 enables or disables the transmit data empty interrupt (TXI) request when data is transferred from TDR to TSR and the transmit data register empty bit (TDRE) in the serial status register (SSR) is set to 1. The TXI interrupt can be cleared by clearing bit TDRE to 0, or by clearing bit TIE to 0.

| Bit 7: TIE | Description | |
|---|---|---|
| 0 | Transmit data empty interrupt request (TXI) disabled | (initial value) |
| 1 | Transmit data empty interrupt request (TXI) enabled | |

**Bit 6—Receive Interrupt Enable (RIE):** Bit 6 enables or disables the receive error interrupt (ERI), and the receive data full interrupt (RXI) requested when data is transferred from RSR to RDR and the receive data register full bit (RDRF) in the serial status register (SSR) is set to 1. There are three kinds of receive error: overrun, framing, and parity. RXI and ERI interrupts can be cleared by clearing SSR flag RDRF, or flags FER, PER, and OER to 0, or by clearing bit RIE to 0.

| Bit 6: RIE | Description | |
|---|---|---|
| 0 | Receive data full interrupt request (RXI) and receive error interrupt request (ERI) disabled | (initial value) |
| 1 | Receive data full interrupt request (RXI) and receive error interrupt request (ERI) enabled | |

**Bit 5—Transmit Enable (TE):** Bit 5 enables or disables the start of a transmit operation.

| Bit 5: TE | Description | |
|---|---|---|
| 0 | Transmit operation disabled*[1] (TXD is a general I/O port) | (initial value) |
| 1 | Transmit operation enabled*[2] (TXD is the transmit data pin) | |

Notes: 1. The transmit data register empty bit (TDRE) in the serial status register (SSR) is fixed at 1.
2. In this state, writing transmit data in TDR clears bit TDRE in SSR to 0 and starts serial data transmission.
Before setting TE to 1 it is necessary to set the transmit format in SMR.

**Bit 4—Receive Enable (RE):** Bit 4 enables or disables the start of a receive operation.

| Bit 4: RE | Description | |
|---|---|---|
| 0 | Receive operation disabled*[1] (RXD is a general I/O port) | (initial value) |
| 1 | Receive operation enabled*[2] (RXD is the receive data pin) | |

Notes: 1. When RE is cleared to 0, this has no effect on the SSR flags RDRF, FER, PER, and OER, which retain their states.
2. Serial data receiving begins when, in this state, a start bit is detected in asynchronous mode, or serial clock input is detected in synchronous mode.
Before setting RE to 1 it is necessary to set the receive format in SMR.

**Bit 3—Multiprocessor Interrupt Enable (MPIE):** Bit 3 enables or disables multiprocessor interrupt requests. This setting is valid only in asynchronous mode, and only when the multiprocessor mode bit (MP) in the serial mode register (SMR) is set to 1. This bit is ignored when COM is set to 1 or when bit MP is cleared to 0.

| Bit 3: MPIE | Description | |
|---|---|---|
| 0 | Multiprocessor interrupt request disabled (ordinary receive operation) | (initial value) |
| | Clearing condition:<br>Multiprocessor bit receives a data value of 1 | |
| 1 | Multiprocessor interrupt request enabled* | |

Note: * SCI3 does not transfer receive data from RSR to RDR, does not detect receive errors, and does not set status flags RDRF, FER, and OER in SSR. Until a multiprocessor bit value of 1 is received, the receive data full interrupt (RXI) and receive error interrupt (ERI) are disabled and serial status register (SSR) flags RDRF, FER, and OER are not set. When the multiprocessor bit receives a 1, the MPBR bit of SSR is set to 1, MPIE is automatically cleared to 0, RXI and ERI interrupts are enabled (provided bits TIE and RIE in SCR3 are set to 1), and setting of the RDRF, FER, and OER flags is enabled.

**Bit 2—Transmit End Interrupt Enable (TEIE):** Bit 2 enables or disables the transmit end interrupt (TEI) requested if there is no valid transmit data in TDR when the MSB is transmitted.

| Bit 2: TEIE | Description | |
|---|---|---|
| 0 | Transmit end interrupt (TEI) disabled | (initial value) |
| 1 | Transmit end interrupt (TEI) enabled* | |

Note: * A TEI interrupt can be cleared by clearing the SSR bit TDRE to 0 and clearing the transmit end bit (TEND) to 0, or by clearing bit TEIE to 0.

**Bits 1 and 0—Clock Enable 1, 0 (CKE1, CKE0):** Bits 1 and 0 select the clock source and enable or disable clock output at pin $SCK_3$. The combination of bits CKE1 and CKE0 determines whether pin $SCK_3$ is a general I/O port, a clock output pin, or a clock input pin.

Note that the CKE0 setting is valid only when operation is in asynchronous mode using an internal clock (CKE1 = 0). This bit is invalid in synchronous mode or when using an external clock (CKE1 = 1). In synchronous mode and in external clock mode, clear CKE0 to 0. After setting bits CKE1 and CKE0, the operation mode must first be set in the serial mode register (SMR).

See table 10.9 in 10.4.3, Operation, for details on clock source selection.

| Bit 1: CKE1 | Bit 0: CKE0 | Communication Mode | Clock Source | $SCK_3$ Pin Function |
|---|---|---|---|---|
| 0 | 0 | Asynchronous | Internal clock | I/O port*[1] |
| | | Synchronous | Internal clock | Serial clock output*[1] |
| 0 | 1 | Asynchronous | Internal clock | Clock output*[2] |
| | | Synchronous | Reserved | Reserved |
| 1 | 0 | Asynchronous | External clock | Clock input*[3] |
| | | Synchronous | External clock | Serial clock input |
| 1 | 1 | Asynchronous | Reserved | Reserved |
| | | Synchronous | Reserved | Reserved |

Notes: 1. Initial value
2. A clock is output with the same frequency as the bit rate.
3. Input a clock with a frequency 16 times the bit rate.

**Serial Status Register (SSR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TDRE | RDRF | OER | FER | PER | TEND | MPBR | MPBT |
| Initial value | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Read/Write | R/(W)* | R/(W)* | R/(W)* | R/(W)* | R/(W)* | R | R | R/W |

Note: * Only 0 can be written for flag clearing.

The serial status register (SSR) is an 8-bit register containing status flags for indicating SCI3 states, and containing the multiprocessor bits.

SSR can be read and written by the CPU at any time, but the CPU cannot write a 1 to the status flags TDRE, RDRF, OER, PER, and FER. To clear these flags to 0 it is first necessary to read a 1. Bit 2 (TEND) and bit 1 (MPBR) are read-only bits and cannot be modified.

SSR is initialized to H'84 upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

**Bit 7—Transmit Data Register Empty (TDRE):** Bit 7 is a status flag indicating that data has been transferred from TDR to TSR.

| Bit 7: TDRE | Description |
|---|---|
| 0 | Indicates that transmit data written to TDR has not been transferred to TSR<br>Clearing conditions:<br>After reading TDRE = 1, cleared by writing 0 to TDRE.<br>When data is written to TDR by an instruction. |
| 1 | Indicates that no transmit data has been written to TDR, or the transmit data written to TDR has been transferred to TSR (initial value)<br>Setting conditions:<br>When bit TE in SCR3 is cleared to 0.<br>When data is transferred from TDR to TSR. |

**Bit 6—Receive Data Register Full (RDRF):** Bit 6 is a status flag indicating whether there is receive data in RDR.

| Bit 6: RDRF | Description |
|---|---|
| 0 | Indicates there is no receive data in RDR (initial value)<br>Clearing conditions:<br>After reading RDRF = 1, cleared by writing 0 to RDRF.<br>When data is read from RDR by an instruction. |
| 1 | Indicates that there is receive data in RDR<br>Setting condition:<br>When receiving ends normally, with receive data transferred from RSR to RDR |

Note: If a receive error is detected at the end of receiving, or if bit RE in serial control register 3 (SCR3) is cleared to 0, RDR and RDRF are unaffected and keep their previous states. An overrun error (OER) occurs if receiving of data is completed while bit RDRF remains set to 1. If this happens, receive data will be lost.

**Bit 5—Overrun Error (OER):** Bit 5 is a status flag indicating that an overrun error has occurred during data receiving.

| Bit 5: OER | Description |
|---|---|
| 0 | Indicates that data receiving is in progress or has been completed*1 (initial value)<br>Clearing condition:<br>After reading OER = 1, cleared by writing 0 to OER |
| 1 | Indicates that an overrun error occurred in data receiving*2<br>Setting condition:<br>When data receiving is completed while RDRF is set to 1 |

Notes: 1. When bit RE in serial control register 3 (SCR3) is cleared to 0, OER is unaffected and keeps its previous state.
2. RDR keeps the data received prior to the overrun; data received after that is lost. While OER is set to 1, data receiving cannot be continued. In synchronous mode, data transmitting cannot be continued either.

**Bit 4—Framing Error (FER):** Bit 4 is a status flag indicating that a framing error has occurred during asynchronous receiving.

| Bit 4: FER | Description |
|---|---|
| 0 | Indicates that data receiving is in progress or has been completed*1 (initial value)<br>Clearing condition:<br>After reading FER = 1, cleared by writing 0 to FER |
| 1 | Indicates that a framing error occurred in data receiving<br>Setting condition:<br>The stop bit at the end of receive data is checked for a value of 1 and found to be 0*2 |

Notes: 1. When bit RE in serial control register 3 (SCR3) is cleared to 0, FER is unaffected and keeps its previous state.
2. When two stop bits are used only the first stop bit is checked, not the second. When a framing error occurs, receive data is transferred to RDR but RDRF is not set. While FER is set to 1, data receiving cannot be continued. In synchronous mode, data transmitting cannot be continued either.

**Bit 3—Parity Error (PER):** Bit 3 is a status flag indicating that a parity error has occurred during asynchronous receiving.

| Bit 3: PER | Description |
|---|---|
| 0 | Indicates that data receiving is in progress or has been completed*[1] (initial value)<br>Clearing condition:<br>After reading PER = 1, cleared by writing 0 to PER |
| 1 | Indicates that a parity error occurred in data receiving*[2]<br>Setting condition:<br>When the sum of 1s in received data plus the parity bit does not match the parity mode bit (PM) setting in the serial mode register (SMR) |

Notes: 1. When bit RE in serial control register 3 (SCR3) is cleared to 0, PER is unaffected and keeps its previous state.
2. When a parity error occurs, receive data is transferred to RDR but RDRF is not set. While PER is set to 1, data receiving cannot be continued. In synchronous mode, data transmitting cannot be continued either.

**Bit 2—Transmit End (TEND):** Bit 2 is a status flag indicating that TDRE was set to 1 when the last bit of a transmitted character was sent. TEND is a read-only bit and cannot be modified directly.

| Bit 2: TEND | Description |
|---|---|
| 0 | Indicates that transmission is in progress<br>Clearing conditions:<br>After reading TDRE = 1, cleared by writing 0 to TDRE.<br>When data is written to TDR by an instruction. |
| 1 | Indicates that a transmission has ended (initial value)<br>Setting conditions:<br>When bit TE in SCR3 is cleared to 0.<br>If TDRE is set to 1 when the last bit of a transmitted character is sent. |

**Bit 1—Multiprocessor Bit Receive (MPBR):** Bit 1 holds the multiprocessor bit in data received in asynchronous mode using a multiprocessor format. MPBR is a read-only bit and cannot be modified.

| Bit 1: MPBR | Description |
|---|---|
| 0 | Indicates reception of data in which the multiprocessor bit is 0* (initial value) |
| 1 | Indicates reception of data in which the multiprocessor bit is 1 |

Note: * If bit RE is cleared to 0 while a multiprocessor format is in use, MPBR retains its previous state.

**Bit 0—Multiprocessor Bit Transmit (MPBT):** Bit 0 holds the multiprocessor bit to be added to transmitted data when a multiprocessor format is used in asynchronous mode. Bit MPBT is ignored when synchronous mode is chosen, when the multiprocessor communication function is disabled, or when data transmission is disabled.

| Bit 0: MPBT | Description | |
|---|---|---|
| 0 | The multiprocessor bit in transmit data is 0 | (initial value) |
| 1 | The multiprocessor bit in transmit data is 1 | |

**Bit Rate Register (BRR)**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BRR7 | BRR6 | BRR5 | BRR4 | BRR3 | BRR2 | BRR1 | BRR0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

The bit rate register (BRR) is an 8-bit register which, together with the baud rate generator clock selected by bits CKS1 and CKS0 in the serial mode register (SMR), sets the transmit/receive bit rate.

BRR can be read or written by the CPU at any time.

BRR is initialized to H'FF upon reset or in standby mode, watch mode, subactive mode, or subsleep mode.

Table 10.8 gives examples of how BRR is set in asynchronous mode. The values in table 10.8 are for active (high-speed) mode.

**Table 10.8 BRR Settings and Bit Rates in Asynchronous Mode**

| | OSC (MHz) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 2 | | | 2.4576 | | | 4 | | | 4.194304 | | |
| Bit Rate (bits/s) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 1 | 70 | +0.03 | 1 | 86 | +0.31 | 1 | 141 | +0.03 | 1 | 148 | –0.04 |
| 150 | 0 | 207 | +0.16 | 0 | 255 | 0 | 1 | 103 | +0.16 | 1 | 108 | +0.21 |
| 300 | 0 | 103 | +0.16 | 0 | 127 | 0 | 0 | 207 | +0.16 | 0 | 217 | +0.21 |
| 600 | 0 | 51 | +0.16 | 0 | 63 | 0 | 0 | 103 | +0.16 | 0 | 108 | +0.21 |
| 1200 | 0 | 25 | +0.16 | 0 | 31 | 0 | 0 | 51 | +0.16 | 0 | 54 | –0.70 |
| 2400 | 0 | 12 | +0.16 | 0 | 15 | 0 | 0 | 25 | +0.16 | 0 | 26 | +1.14 |
| 4800 | — | — | — | 0 | 7 | 0 | 0 | 12 | +0.16 | 0 | 13 | –2.48 |
| 9600 | — | — | — | 0 | 3 | 0 | — | — | — | 0 | 6 | –2.48 |
| 19200 | — | — | — | 0 | 1 | 0 | — | — | — | — | — | — |
| 31250 | 0 | 0 | 0 | — | — | — | 0 | 1 | 0 | — | — | — |
| 38400 | — | — | — | 0 | 0 | 0 | — | — | — | — | — | — |

**Table 10.8 BRR Settings and Bit Rates in Asynchronous Mode (cont)**

| | OSC (MHz) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 4.9152 | | | 6 | | | 7.3728 | | | 8 | | |
| Bit Rate (bits/s) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) | n | N | Error (%) |
| 110 | 1 | 174 | –0.26 | 1 | 212 | +0.03 | 2 | 64 | +0.70 | 2 | 70 | +0.03 |
| 150 | 1 | 127 | 0 | 1 | 155 | +0.16 | 1 | 191 | 0 | 1 | 207 | +0.16 |
| 300 | 0 | 255 | 0 | 1 | 77 | +0.16 | 1 | 95 | 0 | 1 | 103 | +0.16 |
| 600 | 0 | 127 | 0 | 0 | 155 | +0.16 | 0 | 191 | 0 | 0 | 207 | +0.16 |
| 1200 | 0 | 63 | 0 | 0 | 77 | +0.16 | 0 | 95 | 0 | 0 | 103 | +0.16 |
| 2400 | 0 | 31 | 0 | 0 | 38 | +0.16 | 0 | 47 | 0 | 0 | 51 | +0.16 |
| 4800 | 0 | 15 | 0 | 0 | 19 | –2.34 | 0 | 23 | 0 | 0 | 25 | +0.16 |
| 9600 | 0 | 7 | 0 | 0 | 9 | –2.34 | 0 | 11 | 0 | 0 | 12 | +0.16 |
| 19200 | 0 | 3 | 0 | 0 | 4 | –2.34 | 0 | 5 | 0 | — | — | — |
| 31250 | — | — | — | 0 | 2 | 0 | — | — | — | 0 | 3 | 0 |
| 38400 | 0 | 1 | 0 | — | — | — | 0 | 2 | 0 | — | — | — |

**Table 10.8 BRR Settings and Bit Rates in Asynchronous Mode (cont)**

| | OSC (MHz) | | | | | |
|---|---|---|---|---|---|---|
| | 9.8304 | | | 10 | | |
| Bit Rate (bits/s) | n | N | Error (%) | n | N | Error (%) |
| 110 | 2 | 86 | +0.31 | 2 | 88 | –0.25 |
| 150 | 1 | 255 | 0 | 2 | 64 | +0.16 |
| 300 | 1 | 127 | 0 | 1 | 129 | +0.16 |
| 600 | 0 | 255 | 0 | 1 | 64 | +0.16 |
| 1200 | 0 | 127 | 0 | 0 | 129 | +0.16 |
| 2400 | 0 | 63 | 0 | 0 | 64 | +0.16 |
| 4800 | 0 | 31 | 0 | 0 | 32 | –1.36 |
| 9600 | 0 | 15 | 0 | 0 | 15 | +1.73 |
| 19200 | 0 | 7 | 0 | 0 | 7 | +1.73 |
| 31250 | 0 | 4 | –1.70 | 0 | 4 | 0 |
| 38400 | 0 | 3 | 0 | 0 | 3 | +1.73 |

Notes: 1. Settings should be made so that error is within 1%.

2. BRR setting values are derived by the following equation.

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

B: Bit rate (bits/s)
N: BRR baud rate generator setting ($0 \leq N \leq 255$)
OSC: Value of $\phi_{OSC}$ (MHz)
n: Baud rate generator input clock number (n = 0, 1, 2, 3)

3. The error values in table 10.8 were derived by performing the following calculation and rounding off to two decimal places.

$$\text{Error (\%)} = \frac{B - R}{R} \times 100$$

B: Bit rate found from n, N, and OSC
R: Bit rate listed in left column of table 10.8

The meaning of n is shown in table 10.9.

**Table 10.9 Relation between n and Clock**

| | | SMR Setting | |
|---|---|---|---|
| **n** | **Clock** | **CKS1** | **CKS0** |
| 0 | $\phi$ | 0 | 0 |
| 1 | $\phi$/4 | 0 | 1 |
| 2 | $\phi$/16 | 1 | 0 |
| 3 | $\phi$/64 | 1 | 1 |

Table 10.10 shows the maximum bit rate for selected frequencies in asynchronous mode. Values in table 10.10 are for active (high-speed) mode.

**Table 10.10 Maximum Bit Rate at Selected Frequencies (Asynchronous Mode)**

| | | Setting | |
|---|---|---|---|
| **OSC (MHz)** | **Maximum Bit Rate (bits/s)** | **n** | **N** |
| 2 | 31250 | 0 | 0 |
| 2.4576 | 38400 | 0 | 0 |
| 4 | 62500 | 0 | 0 |
| 4.194304 | 65536 | 0 | 0 |
| 4.9152 | 76800 | 0 | 0 |
| 6 | 93750 | 0 | 0 |
| 7.3728 | 115200 | 0 | 0 |
| 8 | 125000 | 0 | 0 |
| 9.8304 | 153600 | 0 | 0 |
| 10 | 156250 | 0 | 0 |

Table 10.11 shows typical BRR settings in synchronous mode. Values in table 10.11 are for active (high-speed) mode.

**Table 10.11 Typical BRR Settings and Bit Rates (Synchronous Mode)**

| Bit Rate (bits/s) | OSC (MHz) 2 | | 4 | | 8 | | 10 | |
|---|---|---|---|---|---|---|---|---|
| | n | N | n | N | n | N | n | N |
| 110 | — | — | — | — | — | — | — | — |
| 250 | 1 | 249 | 2 | 124 | 2 | 249 | — | — |
| 500 | 1 | 124 | 1 | 249 | 2 | 124 | — | — |
| 1K | 0 | 249 | 1 | 124 | 1 | 249 | — | — |
| 2.5K | 0 | 99 | 0 | 199 | 1 | 99 | 1 | 124 |
| 5K | 0 | 49 | 0 | 99 | 0 | 199 | 0 | 249 |
| 10K | 0 | 24 | 0 | 49 | 0 | 99 | 0 | 124 |
| 25K | 0 | 9 | 0 | 19 | 0 | 39 | 0 | 49 |
| 50K | 0 | 4 | 0 | 9 | 0 | 19 | 0 | 24 |
| 100K | — | — | 0 | 4 | 0 | 9 | — | — |
| 250K | 0 | 0* | 0 | 1 | 0 | 3 | 0 | 4 |
| 500K | | | 0 | 0* | 0 | 1 | — | — |
| 1M | | | | | 0 | 0* | — | — |
| 2.5M | | | | | | | | |

Note: Blank: Cannot be set
—: Can be set, but error will result
*: Continuous transfer not possible at this setting

BRR setting values are derived by the following equation.

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

B: Bit rate (bits/s)
N: BRR baud rate generator setting ($0 \leq N \leq 255$)
OSC: Value of $\phi_{OSC}$ (MHz)
n: Baud rate generator input clock number (n = 0, 1, 2, 3)

The meaning of n is shown in table 10.12.

**Table 10.12 Relation between n and Clock**

| | | SMR Setting | |
|---|---|---|---|
| **n** | **Clock** | **CKS1** | **CKS0** |
| 0 | $\phi$ | 0 | 0 |
| 1 | $\phi$/4 | 0 | 1 |
| 2 | $\phi$/16 | 1 | 0 |
| 3 | $\phi$/64 | 1 | 1 |

### 10.4.3 Operation

SCI3 supports serial data communication in both asynchronous mode, where each character transferred is synchronized separately, and synchronous mode, where transfer is synchronized by clock pulses.

The choice of asynchronous mode or synchronous mode, and the communication format, is made in the serial mode register (SMR), as shown in table 10.13. The SCI3 clock source is determined by bit COM in SMR and bits CKE1 and CKE0 in serial control register 3 (SCR3), as shown in table 10.14.

**Asynchronous Mode:**

- Data length: choice of 7 bits or 8 bits
- Transmit/receive format options include addition of parity bit, multiprocessor bit, and one or two stop bits (character length depends on this combination of options).
- Framing error (FER), parity error (PER), overrun error (OER), and line breaks can be detected when data is received.
- Clock source: Choice of internal clocks or an external clock

  When an internal clock is selected: Operates on baud rate generator clock. A clock can be output with the same frequency as the bit rate.

  When an external clock is selected: A clock input with a frequency 16 times the bit rate is required (internal baud rate generator is not used).

**Synchronous Mode:**

- Transfer format: 8 bits
- Overrun error can be detected when data is received.
- Clock source: Choice of internal clocks or an external clock

  When an internal clock is selected: Operates on baud rate generator clock, and outputs a serial clock.

  When an external clock is selected: The internal baud rate generator is not used. Operation is synchronous with the input clock.

**Table 10.13 SMR Settings and SCI3 Communication Format**

| SMR Setting | | | | | | Communication Format | | | |
|---|---|---|---|---|---|---|---|---|---|
| **Bit 7: COM** | **Bit 6: CHR** | **Bit 2: MP** | **Bit 5: PE** | **Bit 3: STOP** | **Mode** | **Data Length** | **Multipro-cessor Bit** | **Parity Bit** | **Stop Bit Length** |
| 0 | 0 | 0 | 0 | 0 | Asynchronous mode | 8-bit data | No | No | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | | | 1 | 0 | | | | Yes | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | 1 | | 0 | 0 | | 7-bit data | | No | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | | | 1 | 0 | | | | Yes | 1 bit |
| | | | | 1 | | | | | 2 bits |
| | 0 | 1 | * | 0 | Asynchronous mode (multiprocessor format) | 8-bit data | Yes | No | 1 bit |
| | | | * | 1 | | | | | 2 bits |
| | 1 | | * | 0 | | 7-bit data | | | 1 bit |
| | | | * | 1 | | | | | 2 bits |
| 1 | * | 0 | * | * | Synchronous mode | 8-bit data | No | | None |

Note: * Don't care

**Table 10.14 SMR and SCR3 Settings and Clock Source Selection**

| SMR | SCR3 | | | Transmit/Receive Clock | |
|---|---|---|---|---|---|
| Bit 7: COM | Bit 1: CKE1 | Bit 0: CKE0 | Mode | Clock Source | Pin $SCK_3$ Function |
| 0 | 0 | 0 | Asynchronous mode | Internal | I/O port ($SCK_3$ function not used) |
| | | 1 | | | Outputs clock with same frequency as bit rate |
| | 1 | 0 | | External | Clock should be input with frequency 16 times the desired bit rate |
| 1 | 0 | 0 | Synchronous mode | Internal | Outputs a serial clock |
| | 1 | 0 | | External | Inputs a serial clock |
| 0 | 1 | 1 | Reserved | (illegal settings) | |
| 1 | 1 | 0 | | | |
| 1 | 1 | 1 | | | |

**Continuous Transmit/Receive Operation Using Interrupts:** Continuous transmit and receive operations are possible with SCI3, using the RXI or TXI interrupts. Table 10.15 explains this use of these interrupts.

**Table 10.15 Transmit/Receive Interrupts**

| Interrupt | Flag | Interrupt Conditions | Remarks |
|---|---|---|---|
| RXI | RDRF<br>RIE | When serial data is received normally and receive data is transferred from RSR to RDR, RDRF is set to 1. If RIE is 1 at this time, RXI is enabled and an interrupt occurs. (See figure 10.8 (a).) | The RXI interrupt handler routine should read the receive data from RDR and clear RDRF to 0. Continuous receiving is possible if these operations are completed before the next data has been completely received in RSR. |
| TXI | TDRE<br>TIE | When TSR empty (previous transmission complete) is detected and the transmit data set in TDR is transferred to TSR, TDRE is set to 1. If TIE is 1 at this time, TXI is enabled and an interrupt occurs. (See figure 10.8 (b).) | The TXI interrupt handler routine should write the next transmit data to TDR and clear TDRE to 0.Continuous transmission is possible if these operations are completed before the data transferred to TSR has been completely transmitted. |
| TEI | TEND<br>TEIE | When the last bit of the TSR transmit character has been sent, if TDRE is 1, then 1 is set in TEND. If TEIE is 1 at this time, TEI is enabled and an interrupt occurs. (See figure 10.8 (c).) | TEI indicates that, when the last bit of the TSR transmit character was sent, the next transmit data had not been written to TDR. |

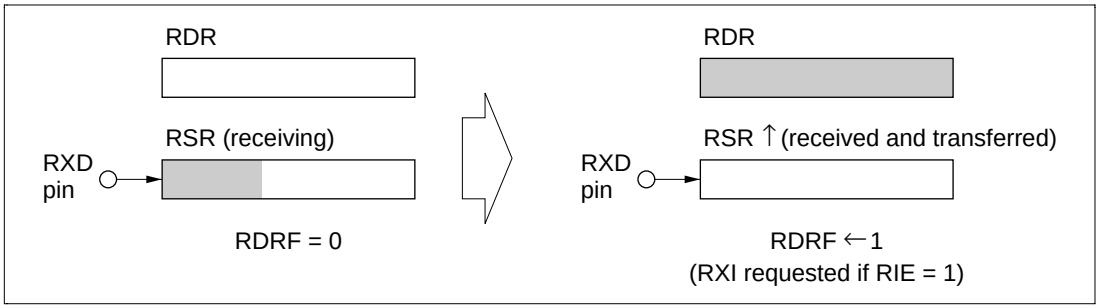



**Figure 10.8 (a) RDRF Setting and RXI Interrupt**

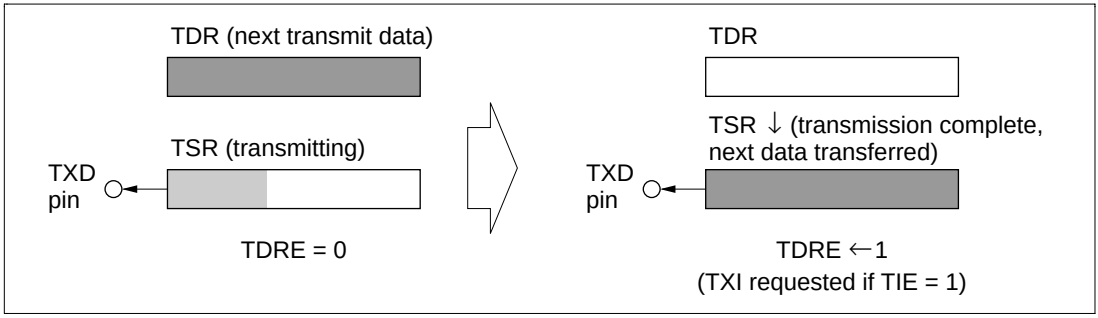



**Figure 10.8 (b) TDRE Setting and TXI Interrupt**

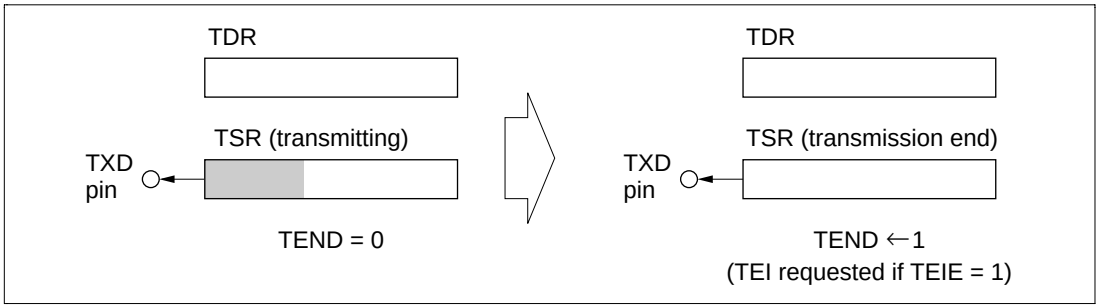



**Figure 10.8 (c) TEND Setting and TEI Interrupt**

### 10.4.4 Operation in Asynchronous Mode

In asynchronous communication mode, a start bit indicating the start of communication and a stop bit indicating the end of communication are added to each character that is sent. In this way synchronization is achieved for each character as a self-contained unit.

SCI3 consists of independent transmit and receive modules, giving it the capability of full duplex communication. Both the transmit and receive modules have a double-buffer configuration, allowing data to be read or written during communication operations so that data can be transmitted and received continuously.

**Transmit/Receive Formats**

Figure 10.9 shows the general format for asynchronous serial communication.

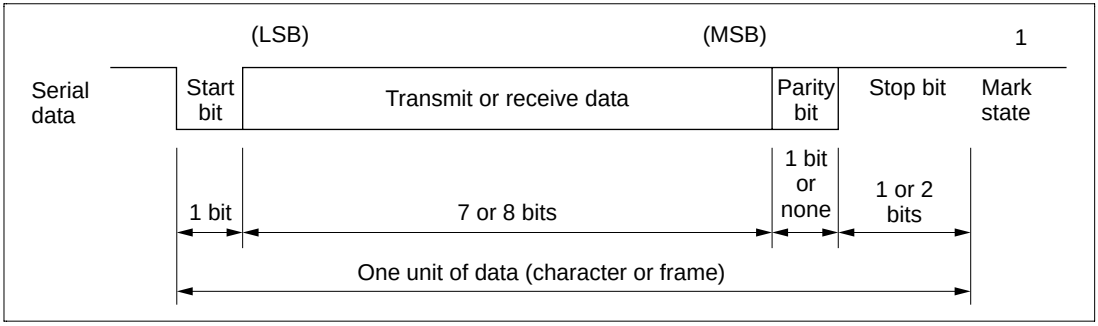



**Figure 10.9 Data Format in Asynchronous Serial Communication Mode**

The communication line in asynchronous communication mode normally stays at the high level, in the “mark” state. SCI3 monitors the communication line, and begins serial data communication when it detects a “space” (low-level signal), which is regarded as a start bit.

One character consists of a start bit (low level), transmit/receive data (in LSB-first order: starting with the least significant bit), a parity bit (high or low level), and finally a stop bit (high level), in this order.

In asynchronous data receiving, synchronization is with the falling edge of the start bit. SCI3 samples data on the 8th pulse of a clock that has 16 times the frequency of the bit rate, so each bit of data is latched at its center.

Table 10.16 shows the 12 transmit/receive formats formats that can be selected in asynchronous mode. The format is selected in the serial mode register (SMR).

**Table 10.16 Serial Communication Formats in Asynchronous Mode**

| SMR Settings | | | | Serial Transfer Format and Frame Length |
|---|---|---|---|---|
| CHR | PE | MP | STOP | 1 2 3 4 5 6 7 8 9 10 11 12 |
| 0 | 0 | 0 | 0 | S 8-bit data STOP |
| 0 | 0 | 0 | 1 | S 8-bit data STOP STOP |
| 0 | 1 | 0 | 0 | S 8-bit data P STOP |
| 0 | 1 | 0 | 1 | S 8-bit data P STOP STOP |
| 1 | 0 | 0 | 0 | S 7-bit data STOP |
| 1 | 0 | 0 | 1 | S 7-bit data STOP STOP |
| 1 | 1 | 0 | 0 | S 7-bit data P STOP |
| 1 | 1 | 0 | 1 | S 7-bit data P STOP STOP |
| 0 | * | 1 | 0 | S 8-bit data MPB STOP |
| 0 | * | 1 | 1 | S 8-bit data MPB STOP STOP |
| 1 | * | 1 | 0 | S 7-bit data MPB STOP |
| 1 | * | 1 | 1 | S 7-bit data MPB STOP STOP |

Notation:

S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multiprocessor bit

Note: * Don”t care

**Clock**

The clock source is determined by bit COM in SMR and bits CKE1 and CKE0 in serial control register 3 (SCR3). See table 10.14 for the settings. Either an internal clock source can be used to run the built-in baud rate generator, or an external clock source can be input at pin $SCK_3$.

When an external clock is input at pin $SCK_3$, it should have a frequency 16 times the desired bit rate.

When an internal clock source is used, $SCK_3$ is used as the clock output pin. The clock output has the same frequency as the serial bit rate, and is synchronized as in figure 10.10 so that the rising edge of the clock occurs in the center of each bit of transmit/receive data.

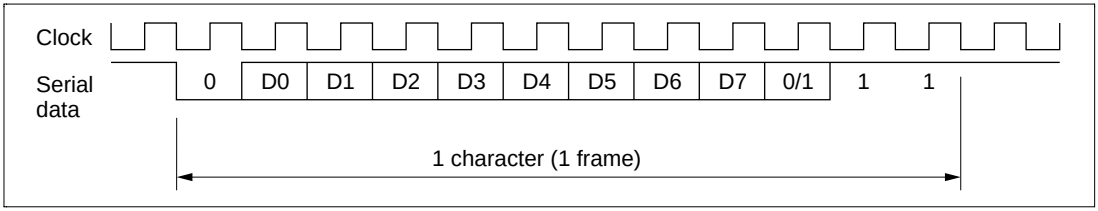



**Figure 10.10 Phase Relation of Output Clock and Communication Data in Asynchronous Mode (8-Bit Data, Parity Bit Added, and 2 Stop Bits)**

**Data Transmit/Receive Operations**

**SCI3 Initialization:** Before data is sent or received, bits TE and RE in serial control register 3 (SCR3) must be cleared to 0, after which initialization can be performed using the procedure shown in figure 10.11.

Note: When modifying the operation mode, transfer format or other settings, always be sure to clear bits TE and RE first. When TE is cleared to 0, bit TDRE will be set to 1. Clearing RE does not clear the status flags RDRF, PER, FER, or OER, or alter the contents of the receive data register (RDR).

When an external clock is used in asynchronous mode, do not stop the clock during operation, including during initialization. When an external clock is used in synchronous mode, do not supply the clock during initialization.

Figure 10.11 shows a typical flow chart for SCI3 initialization.

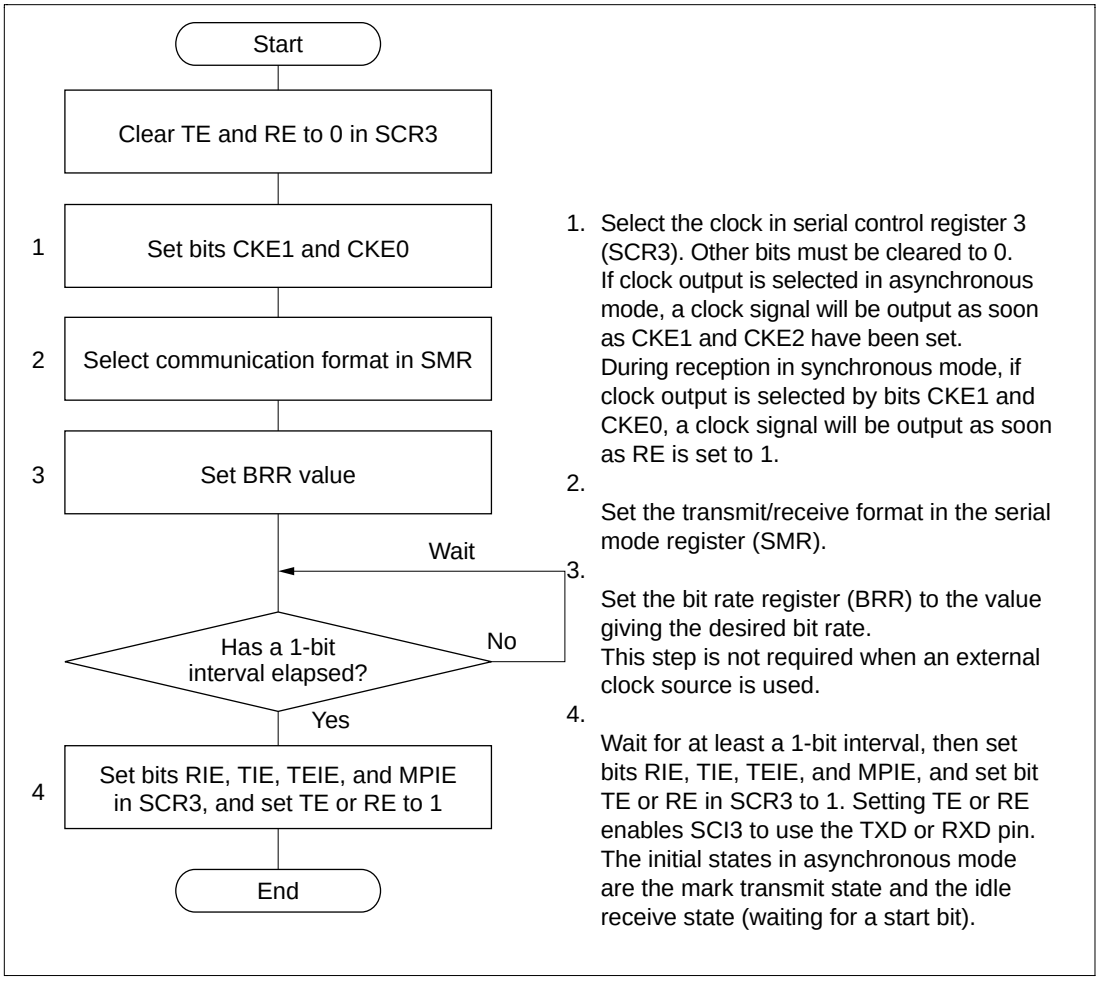



**Figure 10.11 Typical Flow Chart when SCI3 Is Initialized**

**Transmitting:** Figure 10.12 shows a typical flow chart for data transmission. After SCI3 initialization, follow the procedure below.

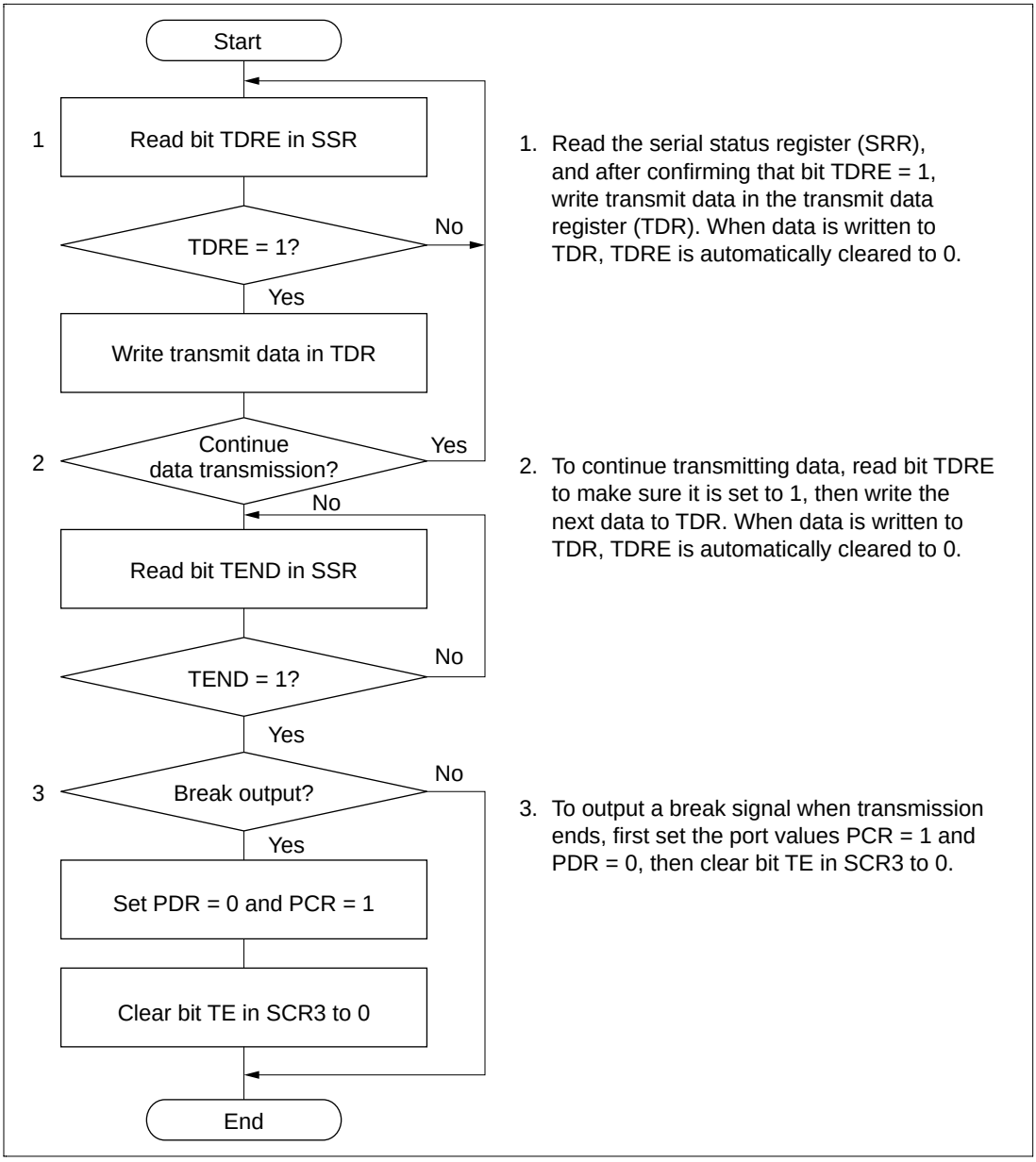



**Figure 10.12 Typical Data Transmission Flow Chart (Asynchronous Mode)**

SCI3 operates as follows during data transmission.

SCI3 monitors bit TDRE in SSR. When this bit is cleared to 0, SCI3 recognizes that there is data written in the transmit data register (TDR), which it transfers to the transmit shift register (TSR). Then TDRE is set to 1 and transmission starts. If bit TIE in SCR3 is set to 1, a TXI interrupt is requested.

Serial data is transmitted from pin TXD using the communication format outlined in table 10.16. Next, TDRE is checked as the stop bit is being transmitted.

If TDRE is 0, data is transferred from TDR to TSR, and after the stop bit is sent, transmission of the next frame starts. If TDRE is 1, the TEND bit in SSR is set to 1, and after the stop bit is sent the output remains at 1 (mark state). A TEI interrupt is requested in this state if bit TEIE in SCR3 is set to 1.

Figure 10.13 shows a typical operation in asynchronous transmission mode.

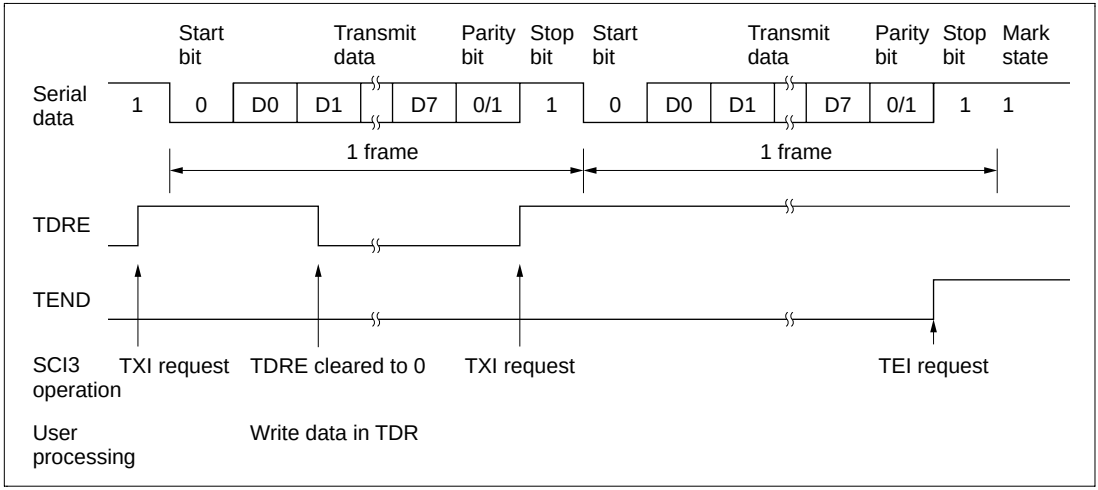
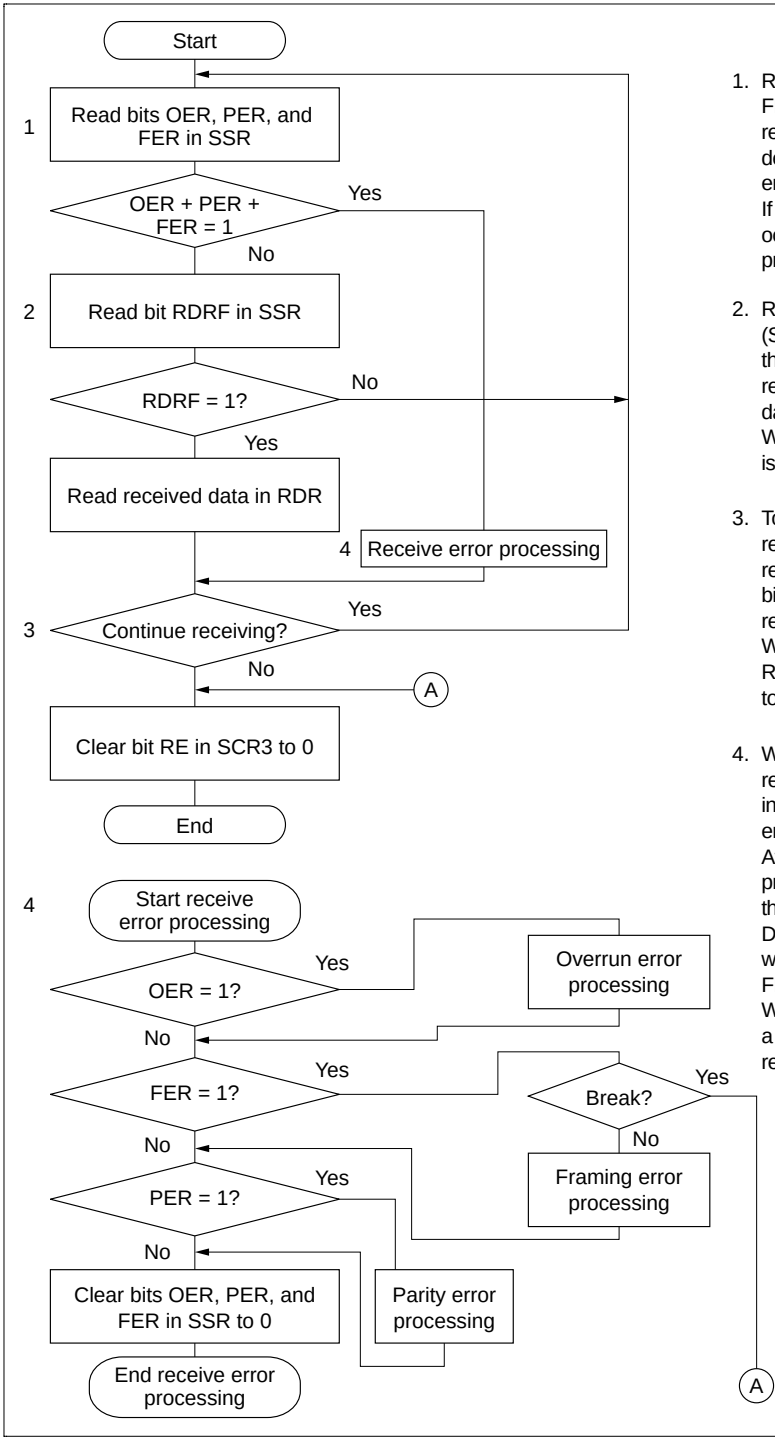



**Figure 10.13 Typical Transmit Operation in Asynchronous Mode (8-Bit Data, Parity Bit Added, and 1 Stop Bit)**

**Receiving:** Figure 10.14 shows a typical flow chart for receiving serial data. After SCI3 initialization, follow the procedure below.




1. Read bits OER, PER, and FER in the serial status register (SSR) to determine if a receive error has occurred. If a receive error has occurred, receive error processing is executed.

2. Read the serial status register (SSR), and after confirming that bit RDRF = 1, read received data from the receive data register (RDR). When RDR data is read, RDRF is automatically cleared to 0.

3. To continue receiving data, read bit RDRF and finish reading RDR before the stop bit of the present frame is received. When data is read from RDR, RDRF is automatically cleared to 0.

4. When a receive error occurs, read bits OER, PER, and FER in SSR to determine which error (s) occurred. After the necessary error processing, be sure to clear the above bits all to 0. Data receiving cannot be resumed while any of bits OER, PER, or FER is set to 1. When a framing error occurs, a break can be detected by reading the RXD pin value.

**Figure 10.14 Typical Serial Data Receiving Flow Chart in Asynchronous Mode**

SCI3 operates as follows when receiving serial data in asynchronous mode.

SCI3 monitors the communication line, and when a start bit (0) is detected it performs internal synchronization and starts receiving. The communication format for data receiving is as outlined in table 10.16. Received data is set in RSR from LSB to MSB, then the parity bit and stop bit(s) are received. After receiving the data, SCI3 performs the following checks:

- Parity check: The number of 1s received is checked to see if it matches the odd or even parity selected in bit PM of SMR.
- Stop bit check: The stop bit is checked for a value of 1. If there are two stop bits, only the first bit is checked.
- Status check: The RDRF bit is checked for a value of 0 to make sure received data can be transferred from RSR to RDR.

If no receive error is detected by the above checks, bit RDRF is set to 1 and the received data is stored in RDR. At that time, if bit RIE in SCR3 is set to 1, an RXI interrupt is requested. If the error check detects a receive error, the appropriate error flag (OER, PER, or FER) is set to 1. RDRF retains the same value as before the data was received. If at this time bit RIE in SCR3 is set to 1, an ERI interrupt is requested.

Table 10.17 gives the receive error detection conditions and the processing of received data in each case.

Note: Data receiving cannot be continued while a receive error flag is set. Before continuing the receive operation it is necessary to clear the OER, FER, PER, and RDRF flags to 0.

**Table 10.17 Receive Error Conditions and Received Data Processing**

| Receive Error | Abbrev. | Detection Conditions | Received Data Processing |
|---|---|---|---|
| Overrun error | OER | Receiving of the next data ends while bit RDRF in SSR is still set to 1 | Received data is not transferred from RSR to RDR |
| Framing error | FER | Stop bit is 0 | Received data is transferred from RSR to RDR |
| Parity error | PER | Received data does not match the parity (odd/even) set in SMR | Received data is not transferred from RSR to RDR |

Figure 10.15 shows a typical SCI3 data receive operation in asynchronous mode.

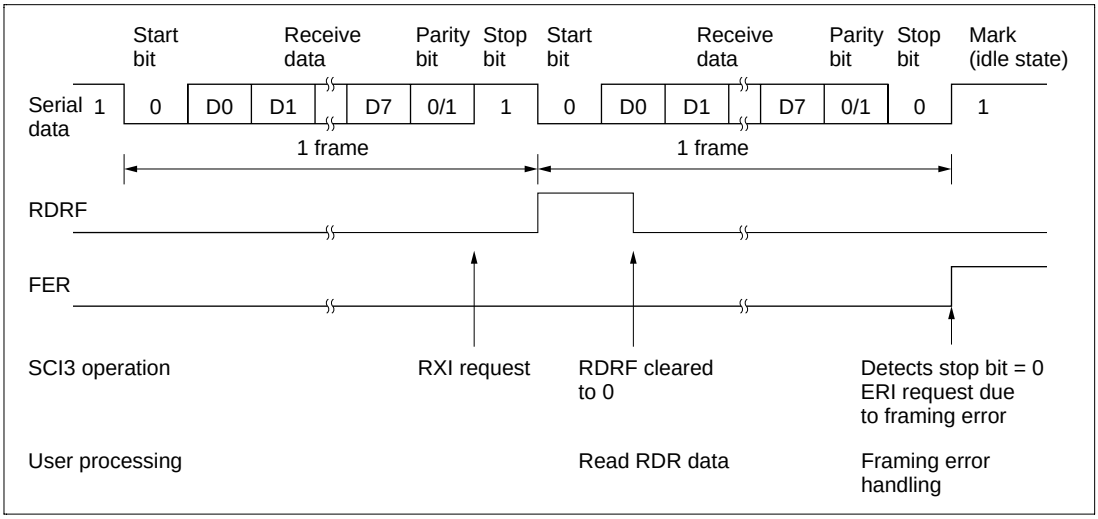



**Figure 10.15 Typical Receive Operation in Asynchronous Mode (8-Bit Data, Parity Bit Added, and 1 Stop Bit)**

### 10.4.5 Operation in Synchronous Mode

In synchronous mode, data is sent or received in synchronization with clock pulses. This mode is suited to high-speed serial communication.

SCI3 consists of independent transmit and receive modules, so full duplex communication is possible, sharing the same clock between both modules. Both the transmit and receive modules have a double-buffer configuration. This allows data to be written during a transmit operation so that data can be transmitted continuously, and enables data to be read during a receive operation so that data can be received continuously.

**Transmit/Receive Format**

Figure 10.16 shows the general communication data format for synchronous communication.

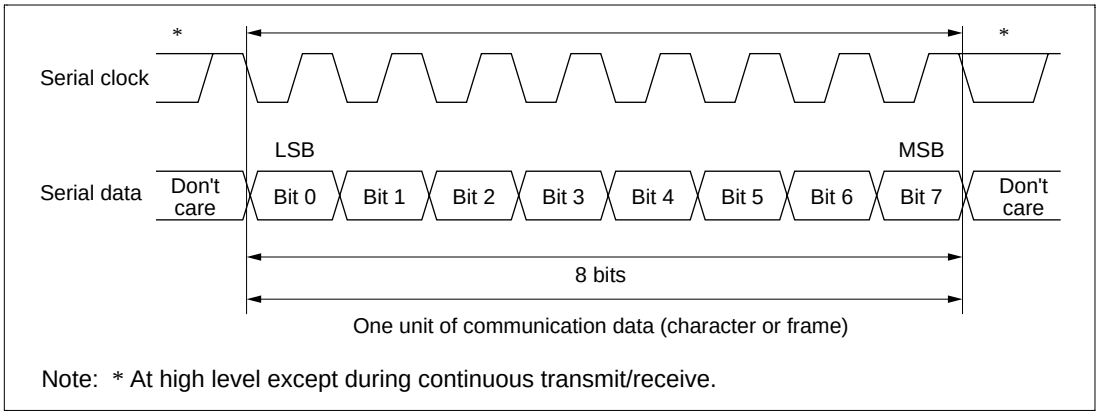



**Figure 10.16 Data Format in Synchronous Communication Mode**

In synchronous communication, data on the communication line is output from one falling edge of the serial clock until the next falling edge. Data is guaranteed valid at the rising edge of the serial clock.

One character of data starts from the LSB and ends with the MSB. The communication line retains the MSB state after the MSB is output.

In synchronous receive mode, SCI3 latches receive data in synchronization with the rising edge of the serial clock.

The transmit/receive format is fixed at 8-bit data. No parity bit or multiprocessor bit is added in this mode.

**Clock**

Either an internal clock from the built-in baud rate generator is used, or an external clock is input at pin $SCK_3$. The choice of clock sources is designated by bit COM in SMR and bits CKE1 and CKE0 in serial control register 3 (SCR3). See table 10.14 for details on selecting the clock source.

When operation is based on an internal clock, a serial clock is output at pin $SCK_3$. Eight clock pulses are output per character of transmit/receive data. When no transmit or receive operation is being performed, the pin is held at the high level.

**Data Transmit/Receive Operations**

**SCI3 Initialization:** Before transmitting or receiving data, follow the SCI3 initialization procedure explained under 10.4.4, SCI3 Initialization, and illustrated in figure 10.10.

**Transmitting:** Figure 10.17 shows a typical flow chart for data transmission. After SCI3 initialization, follow the procedure below.

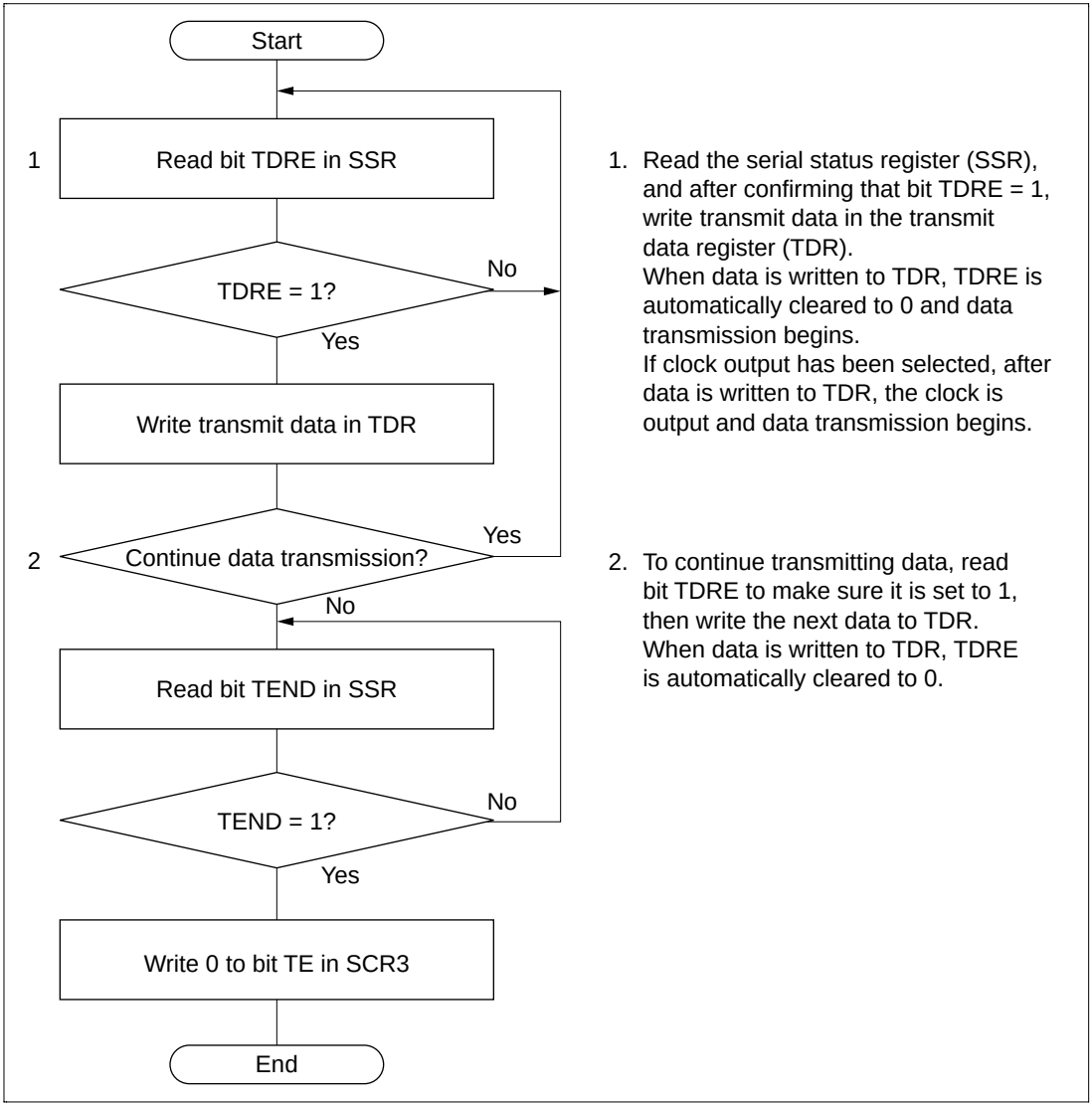



**Figure 10.17 Typical Data Transmission Flow Chart in Synchronous Mode**

SCI3 operates as follows during data transmission in synchronous mode.

SCI3 monitors bit TDRE in SSR. When this bit is cleared to 0, SCI3 recognizes that there is data written in the transmit data register (TDR), which it transfers to the transmit shift register (TSR). Then TDRE is set to 1 and transmission starts. If bit TIE in SCR3 is set to 1, a TXI interrupt is requested.

If clock output is selected, SCI3 outputs eight serial clock pulses. If an external clock is used, data is output in synchronization with the clock input.

Serial data is transmitted from pin TXD in order from LSB (bit 0) to MSB (bit 7).

Then TDRE is checked as the MSB (bit 7) is being transmitted. If TDRE is 0, data is transferred from TDR to TSR, and after the MSB (bit 7) is sent, transmission of the next frame starts. If TDRE is 1, the TEND bit in SSR is set to 1, and after the MSB (bit 7) has been sent, the MSB state is maintained. A TEI interrupt is requested in this state if bit TEIE in SCR3 is set to 1.

After data transmission ends, pin $SCK_3$ is held at the high level.

Note: Data transmission cannot take place while any of the receive error flags (OER, FER, PER) is set to 1. Be sure to confirm that these error flags are cleared to 0 before starting transmission.

Figure 10.18 shows a typical SCI3 transmit operation in synchronous mode.

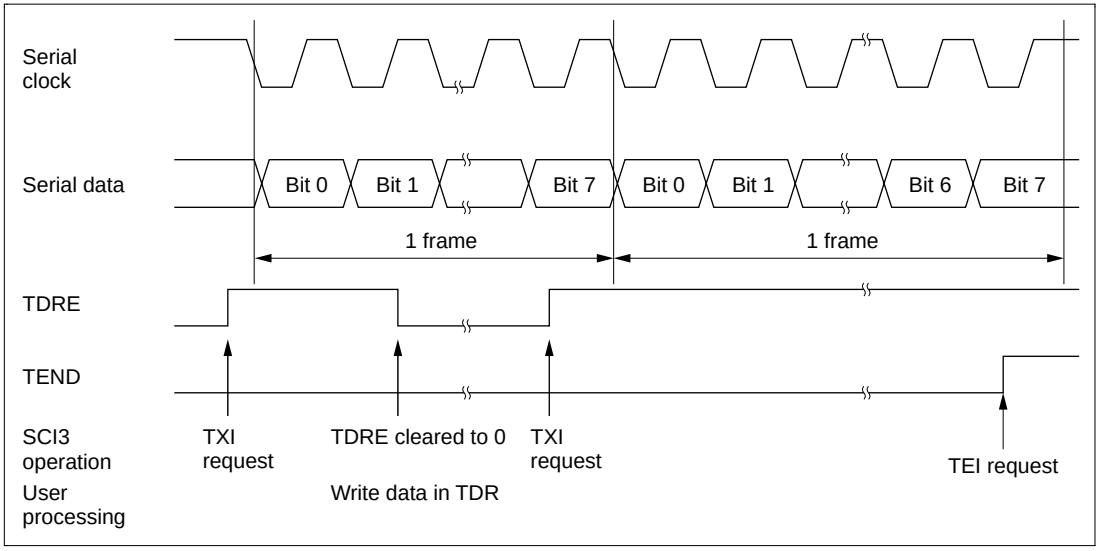



**Figure 10.18 Typical SCI3 Transmit Operation in Synchronous Mode**

**Receiving:** Figure 10.19 shows a typical flow chart for receiving data. After SCI3 initialization, follow the procedure below.

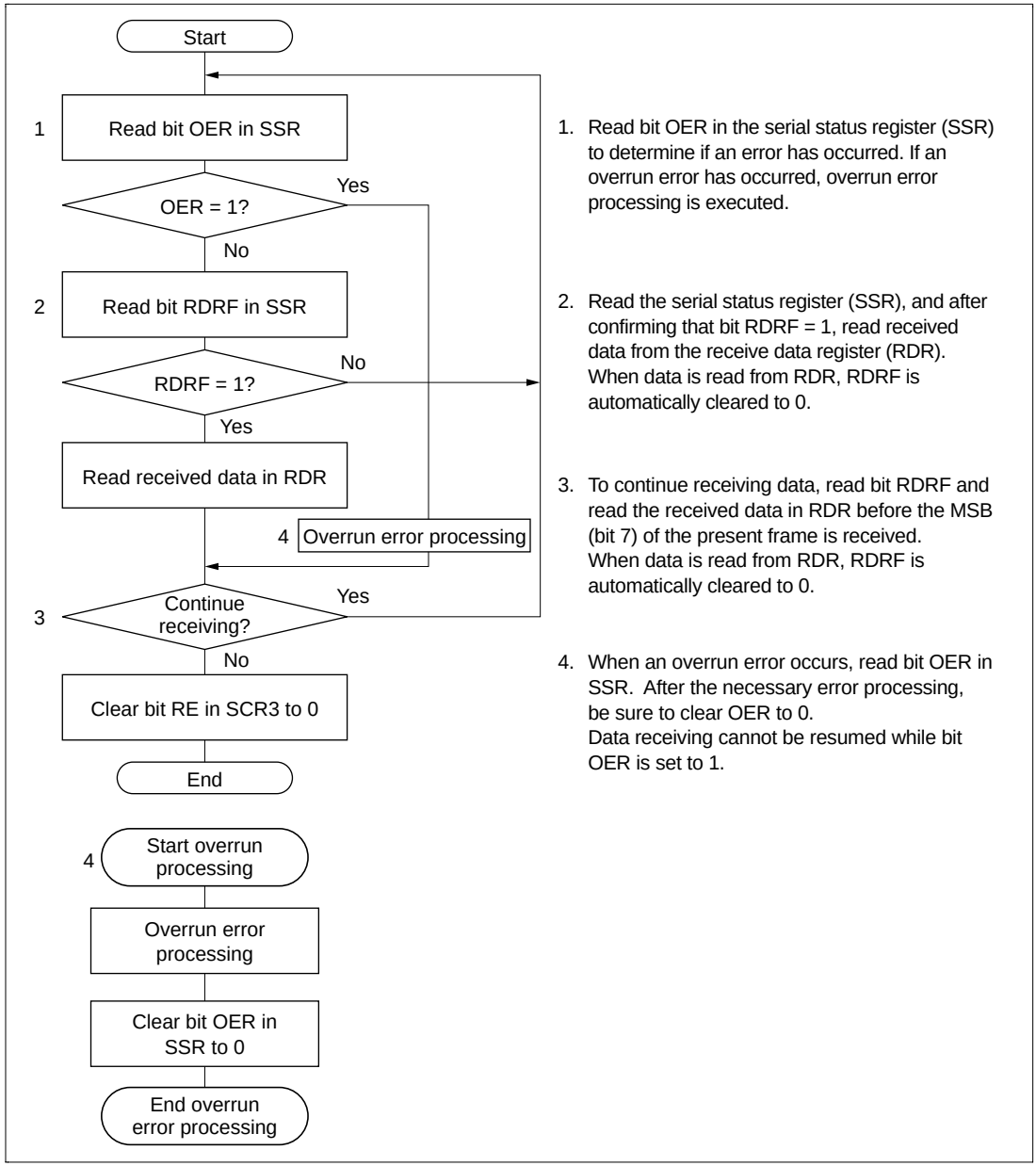



**Figure 10.19 Typical Data Receiving Flow Chart in Synchronous Mode**

SCI3 operates as follows when receiving serial data in synchronous mode.

SCI3 synchronizes internally with the input or output of the serial clock and starts receiving. Received data is set in RSR from LSB to MSB.

After data has been received, SCI3 checks to confirm that the value of bit RDRF is 0 indicating that received data can be transferred from RSR to RDR. If this check passes, RDRF is set to 1 and the received data is stored in RDR. At this time, if bit RIE in SCR3 is set to 1, an RXI interrupt is requested. If an overrun error is detected, OER is set to 1 and RDRF remains set to 1. Then if bit RIE in SCR3 is set to 1, an ERI interrupt is requested.

For the overrun error detection conditions and receive data processing, see table 10.17.

Note: Data receiving cannot be continued while a receive error flag is set. Before continuing the receive operation it is necessary to clear the OER, FER, PER, and RDRF flags to 0.

Figure 10.20 shows a typical receive operation in synchronous mode.

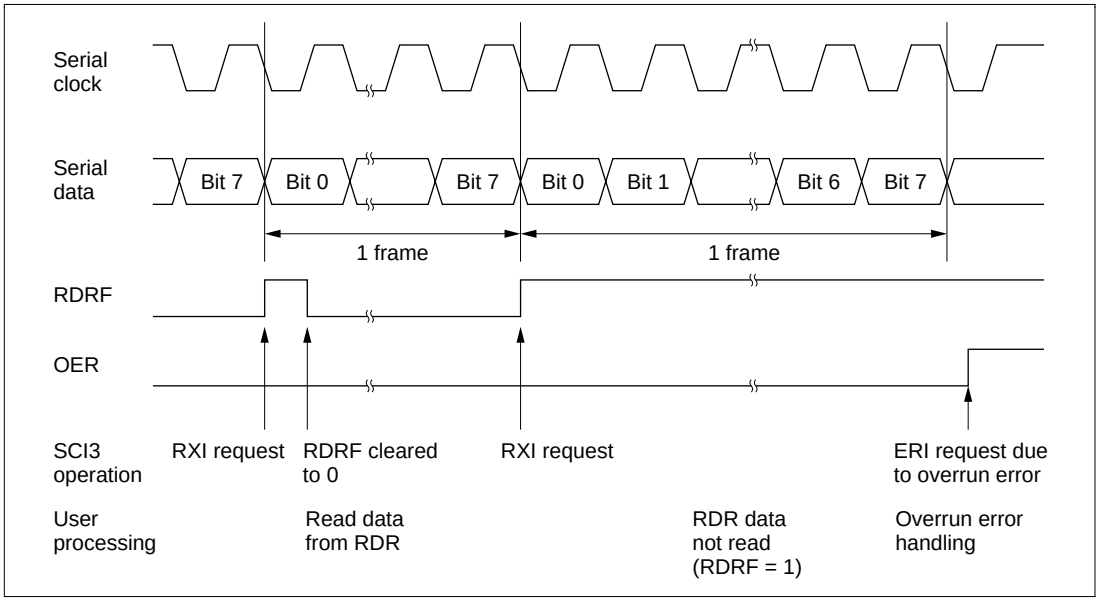
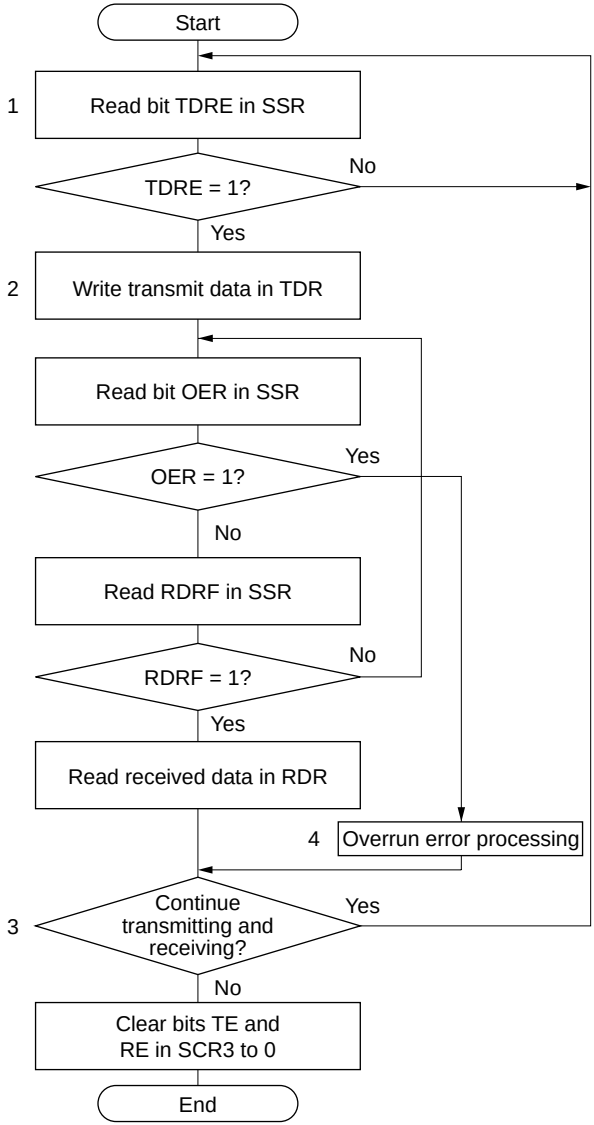



**Figure 10.20 Typical Receive Operation in Synchronous Mode**

**Simultaneous Transmit/Receive:** Figure 10.21 shows a typical flow chart for transmitting and receiving simultaneously. After SCI3 synchronization, follow the procedure below.




1. Read the serial status register (SSR), and after confirming that bit TDRE = 1, write transmit data in the transmit data register (TDR). When data is written to TDR, TDRE is automatically cleared to 0.
2. Read the serial status register (SSR), and after confirming that bit RDRF = 1, read the received data from the receive data register (RDR). When data is read from RDR, RDRF is automatically cleared to 0.
3. To continue transmitting and receiving serial data, read bit RDRF and finish reading RDR before the MSB (bit 7) of the present frame is received. Also read bit TDRE, check that it is set to 1, and write the next data in TDR before the MSB of the current frame has been transmitted. When data is written to TDR, TDRE is automatically cleared to 0; and when data is read from RDR, RDRF is automatically cleared to 0.
4. When an overrun error occurs, read bit OER in SSR. After the necessary error processing, be sure to clear OER to 0. Data transmission and reception cannot take place while bit OER is set to 1. See figure 10.19 for overrun error processing.

**Figure 10.21 Simultaneous Transmit/Receive Flow Chart in Synchronous Mode**

Notes: 1. To switch from transmitting to simultaneous transmitting and receiving, use the following procedure.
- First confirm that TDRE and TEND are both set to 1 and that SCI3 has finished transmitting. Next clear TE to 0. Then set both TE and RE to 1.

2. To switch from receiving to simultaneous transmitting and rceiving, use the following procedure.
- After confirming that SCI3 has finished receiving, clear RE to 0. Next, after confirming that RDRF and the error flags (OER FER, PER) are all 0, set both TE and RE to 1.

### 10.4.6 Multiprocessor Communication Function

The multiprocessor communication function enables several processors to share a single serial communication line. The processors communicate in asynchronous mode using a format with an additional multiprocessor bit (multiprocessor format).

In multiprocessor communication, each receiving processor is addressed by an ID code. A serial communication cycle consists of two cycles: an ID-sending cycle that identifies the receiving processor, and a data-sending cycle. The ID-sending cycle and data-sending cycle are differentiated by the multiprocessor bit. The multiprocessor bit is 1 in an ID-sending cycle, and 0 in a data-sending cycle.

The transmitting processor starts by sending the ID of the receiving processor with which it wants to communicate as data with the multiprocessor bit set to 1. Next the transmitting processor sends transmit data with the multiprocessor bit cleared to 0. When a receiving processor receives data with the multiprocessor bit set to 1, it compares the data with its own ID. If the data matches its ID, the receiving processor continues to receive incoming data. If the data does not match its ID, the receiving processor skips further incoming data until it again receives data with the multiprocessor bit set to 1. Multiple processors can send and receive data in this way.

Figure 10.22 shows an example of communication among different processors using a multiprocessor format.

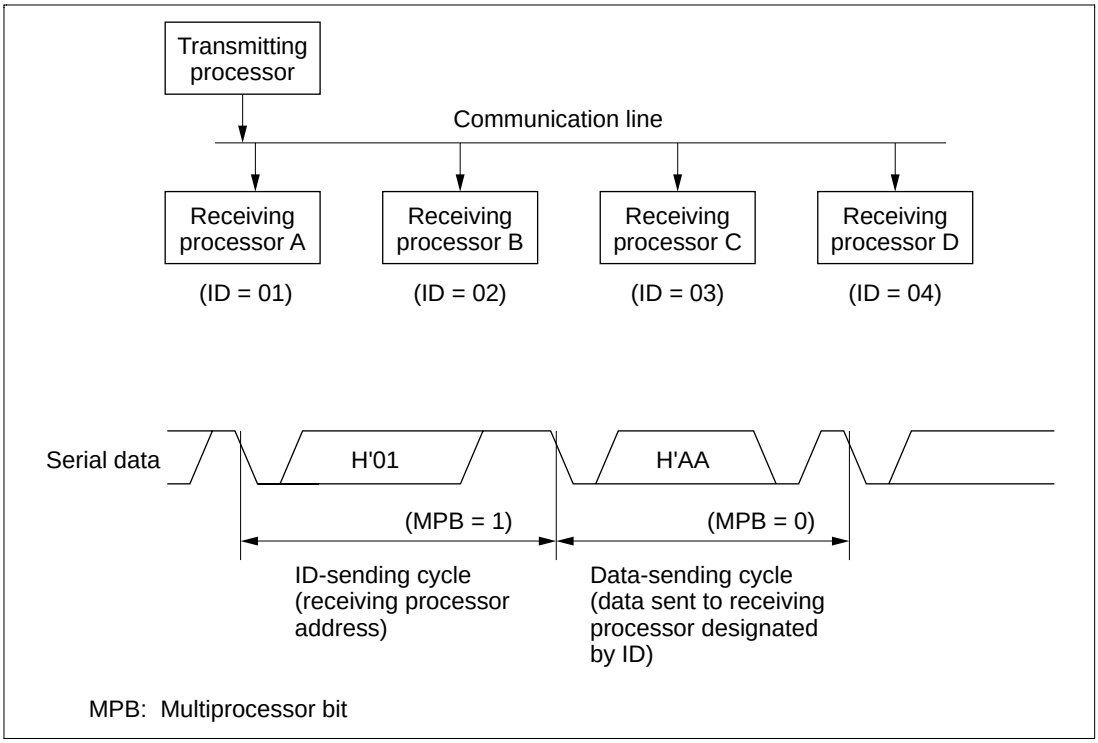



**Figure 10.22 Example of Interprocessor Communication Using Multiprocessor Format (Data H'AA Sent to Receiving Processor A)**

Four communication formats are available. Parity-bit settings are ignored when a multiprocessor format is selected. For details see table 10.16.

For a description of the clock used in multiprocessor communication, see 10.4.4, Operation in Asynchronous Mode.

**Transmitting Multiprocessor Data:** Figure 10.23 shows a typical flow chart for multiprocessor serial data transmission. After SCI3 initialization, follow the procedure below.

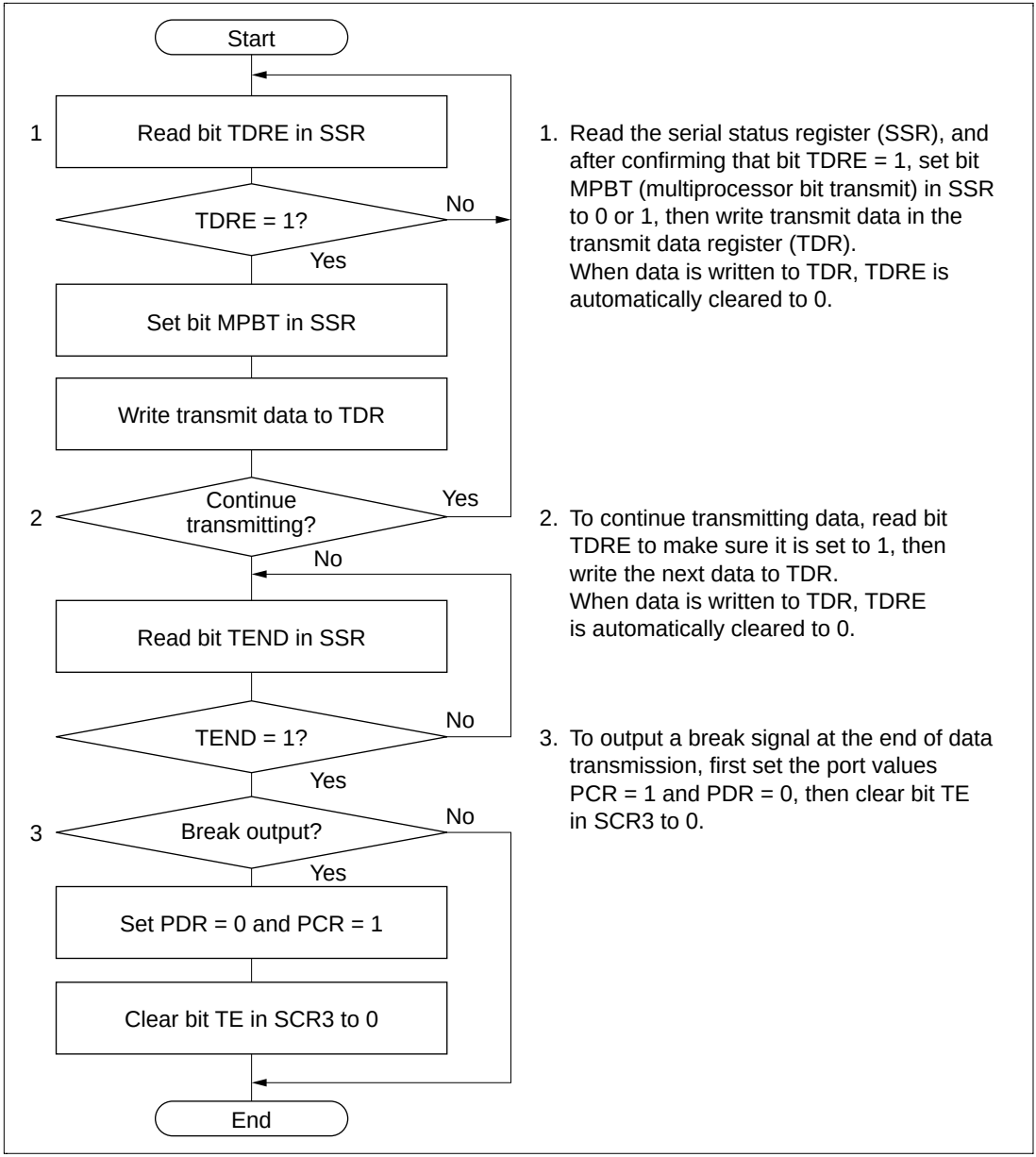



**Figure 10.23 Typical Multiprocessor Data Transmission Flow Chart**

SCI3 operates as follows during data transmission using a multiprocessor format.

SCI3 monitors bit TDRE in SSR. When this bit is cleared to 0, SCI3 recognizes that there is data written in the transmit data register (TDR), which it transfers to the transmit shift register (TSR). Then TDRE is set to 1 and transmission starts. If bit TIE in SCR3 is set to 1, a TXI interrupt is requested.

Serial data is transmitted from pin TXD using the communication format outlined in table 10.16.

Next, TDRE is checked as the stop bit is being transmitted. If TDRE is 0, data is transferred from TDR to TSR, and after the stop bit is sent, transmission of the next frame starts. If TDRE is 1, the TEND bit in SSR is set to 1, and after the stop bit is sent the output remains at 1 (mark state). A TEI interrupt is requested in this state if bit TEIE in SCR3 is set to 1.

Figure 10.24 shows a typical SCI3 operation in multiprocessor communication mode.

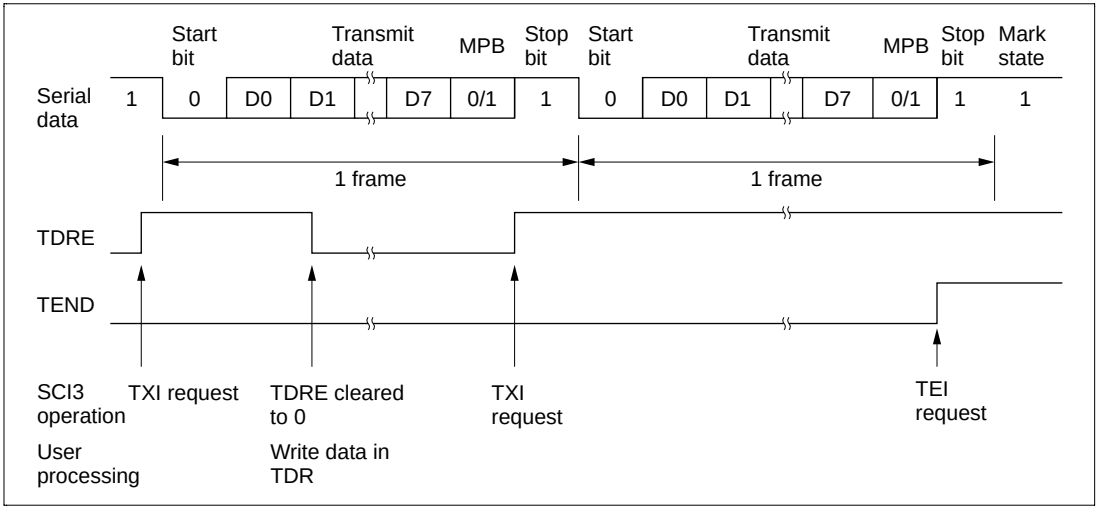



**Figure 10.24 Typical Multiprocessor Format Transmit Operation (8-Bit Data, Multiprocessor Bit Added, and 1 Stop Bit)**

**Receiving Multiprocessor Data:** Figure 10.25 shows a typical flow chart for receiving data using a multiprocessor format. After SCI3 initialization, follow the procedure below.

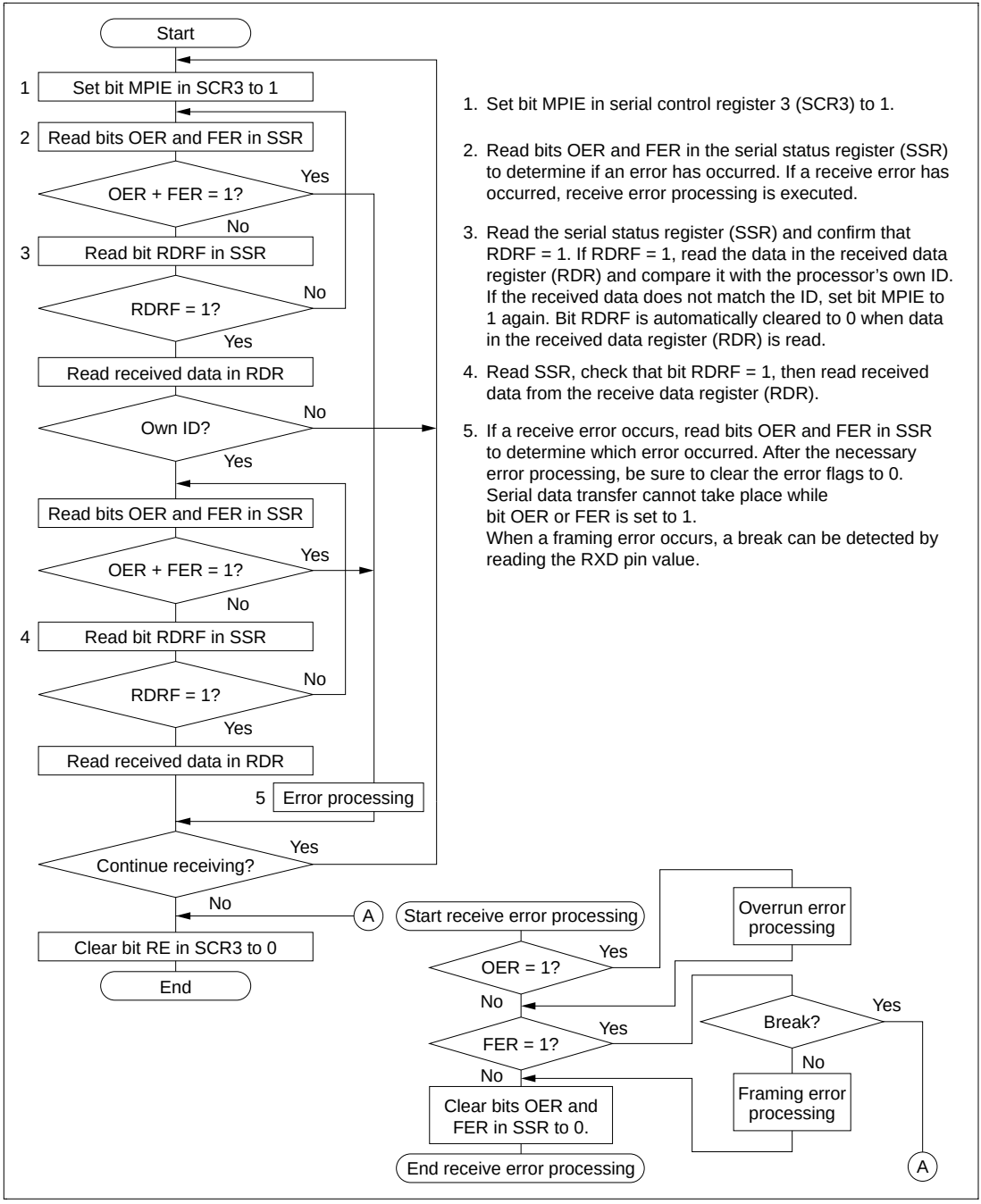



**Figure 10.25 Typical Flow Chart for Receiving Serial Data Using Multiprocessor Format**

Figure 10.26 gives an example of data reception using a multiprocessor format.

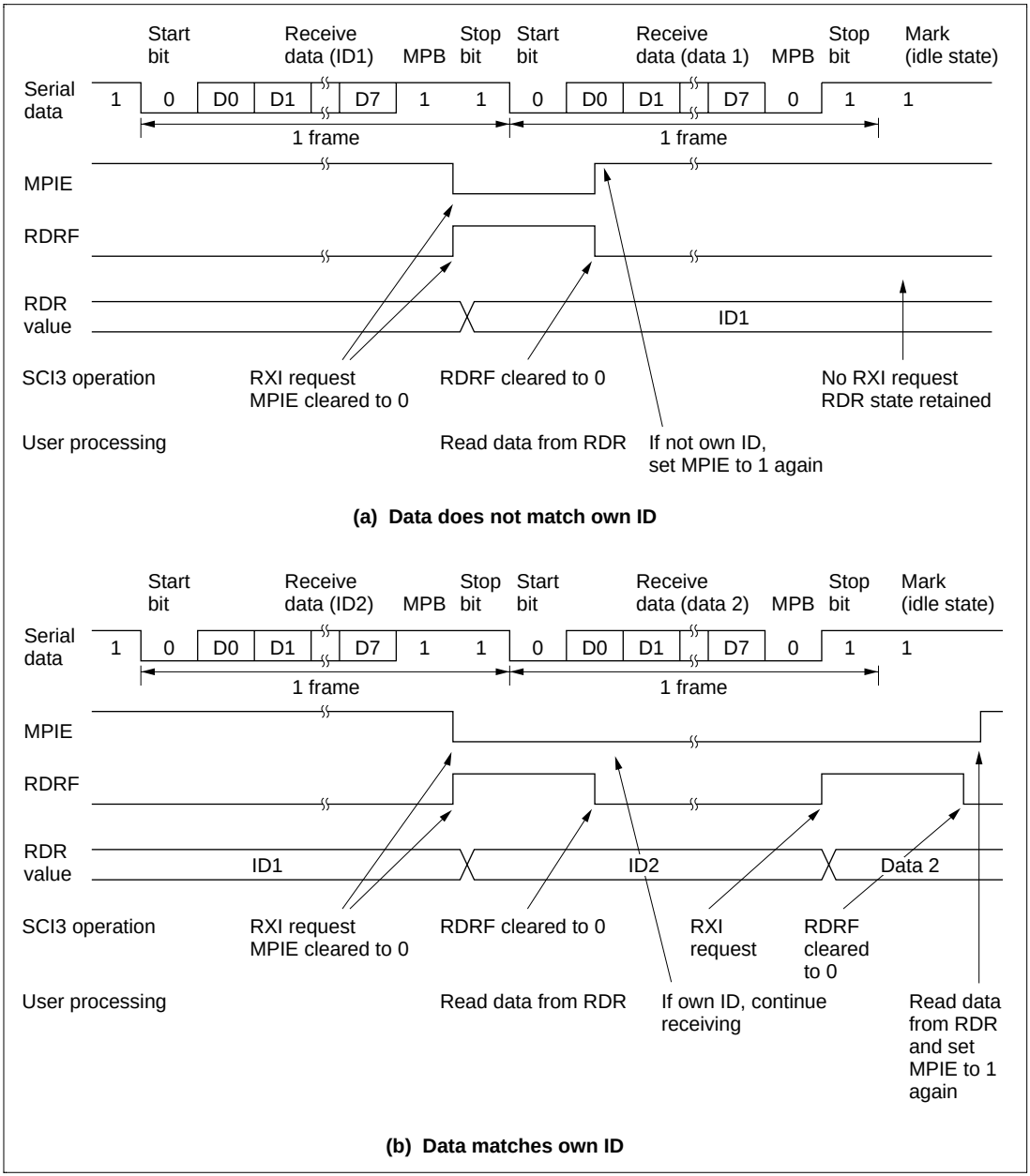



**Figure 10.26 Example of Multiprocessor Format Receive Operation (8-Bit Data, Multiprocessor Bit Added, and 1 Stop Bit)**

### 10.4.7 Interrupts

SCI3 has six interrupt sources: transmit end, transmit data empty, receive data full, and the three receive error interrupts (overrun error, framing error, and parity error). All share a common interrupt vector. Table 10.18 describes each interrupt.

**Table 10.18 SCI3 Interrupts**

| Interrupt | Description | Vector Address |
|---|---|---|
| RXI | Interrupt request due to receive data register full (RDRF) | H'0024 |
| TXI | Interrupt request due to transmit data register empty (TDRE) | |
| TEI | Interrupt request due to transmit end (TEND) | |
| ERI | Interrupt request due to receive error (OER, FER, or PER) | |

The interrupt requests are enabled and disabled by bits TIE and RIE of SCR3.

When bit TDRE in SSR is set to 1, TXI is requested. When bit TEND in SSR is set to 1, TEI is requested. These two interrupt requests occur during data transmission.

The initial value of bit TDRE is 1. Accordingly, if the transmit data empty interrupt request (TXI) is enabled by setting bit TIE to 1 in SCR3 before placing transmit data in TDR, TXI will be requested even though no transmit data has been readied.

Likewise, the initial value of bit TEND is 1. Accordingly, if the transmit end interrupt request (TEI) is enabled by setting bit TEIE to 1 in SCR3 before placing transmit data in TDR, TEI will be requested even though no data has been transmitted.

These interrupt features can be used to advantage by programming the interrupt handler to move the transmit data into TDR. When this technique is not used, the interrupt enable bits (TIE and TEIE) should not be set to 1 until after TDR has been loaded with transmit data, to avoid unwanted TXI and TEI interrupts.

When bit RDRF in SSR is set to 1, RXI is requested. When any of SSR bits OER, FER, or PER is set to 1, ERI is requested. These two interrupt requests occur during the receiving of data.

Details on interrupts are given in 3.3, Interrupts.

### 10.4.8 Application Notes

When using SCI3, attention should be paid to the following matters.

**Relation between Bit TDRE and Writing Data to TDR:** Bit TDRE in the serial status register (SSR) is a status flag indicating that TDR does not contain new transmit data. TDRE is automatically cleared to 0 when data is written to TDR. When SCI3 transfers data from TDR to TSR, bit TDRE is set to 1.

Data can be written to TDR regardless of the status of bit TDRE. However, if new data is written to TDR while TDRE is cleared to 0, assuming the data held in TDR has not yet been shifted to TSR, it will be lost. For this reason it is advisable to confirm that bit TDRE is set to 1 before each write to TDR and not write to TDR more than once without checking TDRE in between.

**Operation when Multiple Receive Errors Occur at the Same Time:** When two or more receive errors occur at the same time, the status flags in SSR are set as shown in table 10.19. If an overrun error occurs, data is not transferred from RSR to RDR, and receive data is lost.

**Table 10.19 SSR Status Flag States and Transfer of Receive Data**

| SSR Status Flags | | | | Receive Error Status | |
|---|---|---|---|---|---|
| RDRF* | OER | FER | PER | (RSR → RDR) | Receive Data Transfer |
| 1 | 1 | 0 | 0 | × | Overrun error |
| 0 | 0 | 1 | 0 | O | Framing error |
| 0 | 0 | 0 | 1 | O | Parity error |
| 1 | 1 | 1 | 0 | × | Overrun error + framing error |
| 1 | 1 | 0 | 1 | × | Overrun error + parity error |
| 0 | 0 | 1 | 1 | O | Framing error + parity error |
| 1 | 1 | 1 | 1 | × | Overrun error + framing error + parity error |

Notation:
O: Receive data transferred from RSR to RDR
×: Receive data not transferred from RSR to RDR

Note: * RDRF keeps the same state as before the data was received. However, if due to a late read of received data in one frame an overrun error occurs in the next frame, RDRF is cleared to 0 when RDR is read.

**Break Detection and Processing:** Break signals can be detected by reading the RXD pin directly when a framing error (FER) is detected. In the break state the input from the RXD pin consists of all 0s, so FER is set and the parity error flag (PER) may also be set. In the break state SCI3 continues to receive, so if the FER bit is cleared to 0 it will be set to 1 again.

**Sending a Mark or Break Signal:** When TE is cleared to 0 the TXD pin becomes an I/O port, the level and direction (input or output) of which are determined by the PDR and PCR bits. This feature can be used to place the TXD pin in the mark state or send a break signal.

To place the serial communication line in the mark (1) state before TE is set to 1, set the PDR and PCR bits both to 1. Since TE is cleared to 0, TXD becomes a general output port outputting the value 1.

To send a break signal during data transmission, set the PCR bit to 1 and clear the PDR bit to 0, then clear TE to 0. When TE is cleared to 0 the transmitter is initialized, regardless of its current state, so the TXD pin becomes an output port outputting the value 0.

**Receive Error Flags and Transmit Operation (Sysnchronous Mode Only):** When a receive error flag (ORER, PER, or FER) is set to 1, SCI3 will not start transmitting even if TDRE is cleared to 0. Be sure to clear the receive error flags to 0 when starting to transmit. Note that clearing RE to 0 does not clear the receive error flags.

**Receive Data Sampling Timing and Receive Margin in Asynchronous Mode:** In asynchronous mode SCI3 operates on a base clock with 16 times the bit rate frequency. In receiving, SCI3 synchronizes internally with the falling edge of the start bit, which it samples on the base clock. Receive data is latched at the rising edge of the eighth base clock pulse. See figure 10.27.

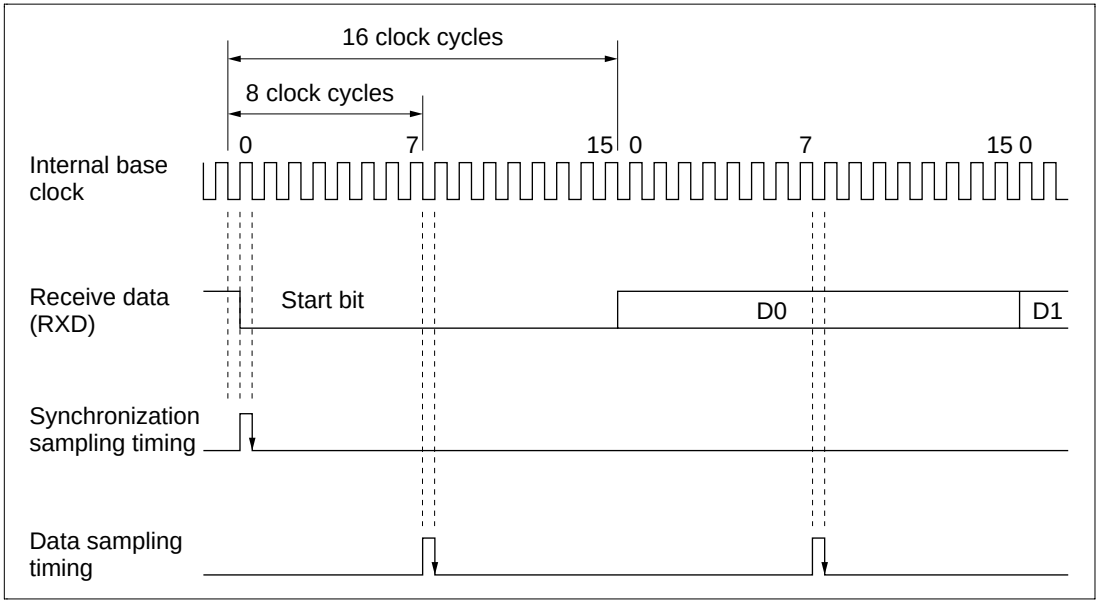



**Figure 10.27 Receive Data Sampling Timing in Asynchronous Mode**

The receive margin in asynchronous mode can therefore be derived from the following equation.

$$M = \{(0.5 - 1/2N) - (D - 0.5) / N - (L - 0.5) F\} \times 100\% \quad \text{Equation (1)}$$

M: Receive margin (%)
N: Ratio of clock frequency to bit rate (N = 16)
D: Clock duty cycle (D = 0.5 to 1)
L: Frame length (L = 9 to 12)
F: Absolute value of clock frequency error

In equation (1), if F (absolute value of clock frequency error) = 0 and D (clock duty cycle) = 0.5, the receive margin is 46.875% as given by equation (2) below.

When D = 0.5 and F = 0,

$$M = \{0.5 - 1/(2 \times 16)\} \times 100\% = 46.875\% \quad \text{Equation (2)}$$

This value is theoretical. In actual system designs a margin of from 20 to 30 percent should be allowed.

**Relationship between Bit RDRF and Reading RDR:** While SCI3 is receiving, it checks the RDRF flag. When a frame of data has been received, if the RDRF flag is cleared to 0, data receiving ends normally. If RDRF is set to 1, an overrun error occurs.

RDRF is automatically cleared to 0 when the contents of RDR are read. If RDR is read more than once, the second and later reads will be performed with RDRF cleared to 0. While RDRF is 0, if RDR is read when reception of the next frame is just ending, data from the next frame may be read. This is illustrated in figure 10.28.

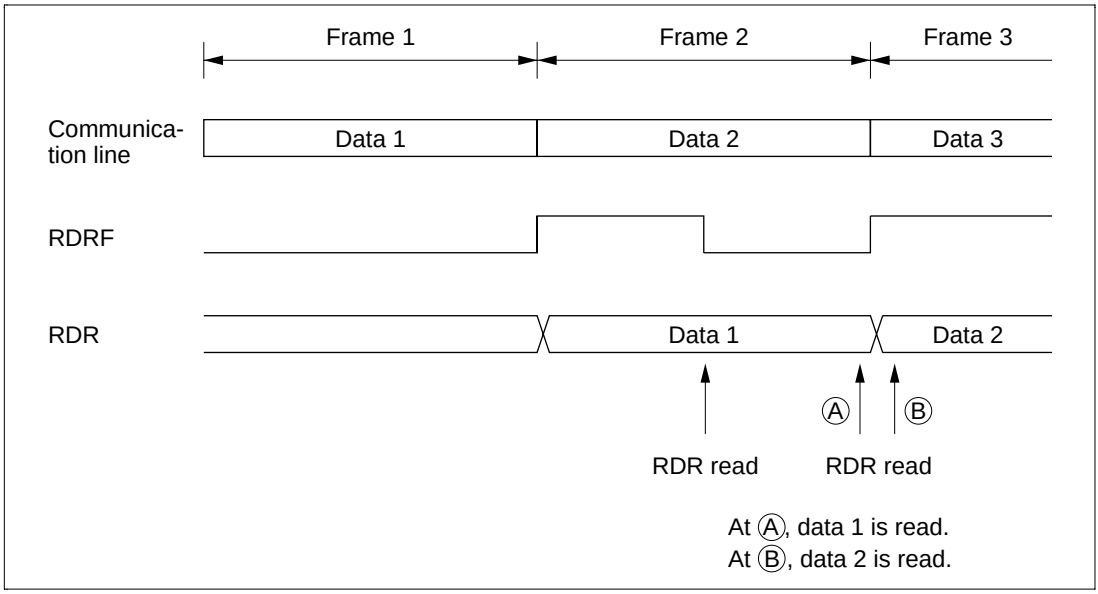



**Figure 10.28 Relationship between Data and RDR Read Timing**

To avoid the situation described above, after RDRF is confirmed to be 1, RDR should only be read once and should not be read twice or more.

When the same data must be read more than once, the data read the first time should be copied to RAM, for example, and the copied data should be used. An alternative is to read RDR but leave a safe margin of time before reception of the next frame is completed. In synchronous mode, all reads of RDR should be completed before bit 7 is received. In asynchronous mode, all reads of RDR should be completed before the stop bit is received.

**Caution on Switching of $SCK_3$ Function:** If pin $SCK_3$ is used as a clock output pin by SCI3 in synchronous mode and is then switched to a general input/output pin (a pin with a different function), the pin outputs a low level signal for half a system clock ($\phi$) cycle immediately after it is switched.

This can be prevented by either of the following methods according to the situation.

1. When an $SCK_3$ function is switched from clock output to non clock-output

   When stopping data transfer, issue one instruction to clear bits TE and RE to 0 and to set bits CKE1 and CKE0 in SCR3 to 1 and 0, respectively. In this case, bit COM in SMR should be left 1. The above prevents $SCK_3$ from being used as a general input/output pin. To avoid an intermediate level of voltage from being applied to $SCK_3$, the line connected to $SCK_3$ should be pulled up to the $V_{CC}$ level via a resistor, or supplied with output from an external device.

2. When an $SCK_3$ function is switched from clock output to general input/output

   When stopping data transfer,

   a. Issue one instruction to clear bits TE and RE to 0 and to set bits CKE1 and CKE0 in SCR3 to 1 and 0, respectively.
   b. Clear bit COM in SCR3 to 0
   c. Clear bits CKE1 and CKE0 in SCR3 to 0

Note that special care is also needed here to avoid an intermediate level of voltage from being applied to $SCK_3$.

**Caution on Switching TxD Function:** If pin TxD is used as a data output pin by SCI3 in synchronous mode and is then switched to a general input/output pin (a pin with a different function), the pin outputs a high level signal for one system clock ($\phi$) cycle immediately after it is switched.

## 11.1 Overview

The H8/3834 Series is provided with a 14-bit PWM (pulse width modulator) on-chip, which can be used as a D/A converter by connecting a low-pass filter.

### 11.1.1 Features

Features of the 14-bit PWM are as follows.

- Choice of two conversion periods
  A conversion period of 3,268/$\phi$, with a minimum modulation width of 2/$\phi$ (PWCR0 = 1), or a conversion period of 16,384/$\phi$, with a minimum modulation width of 1/$\phi$ (PWCR0 = 0), can be chosen.
- Pulse division method for less ripple

### 11.1.2 Block Diagram

Figure 11.1 shows a block diagram of the 14-bit PWM.

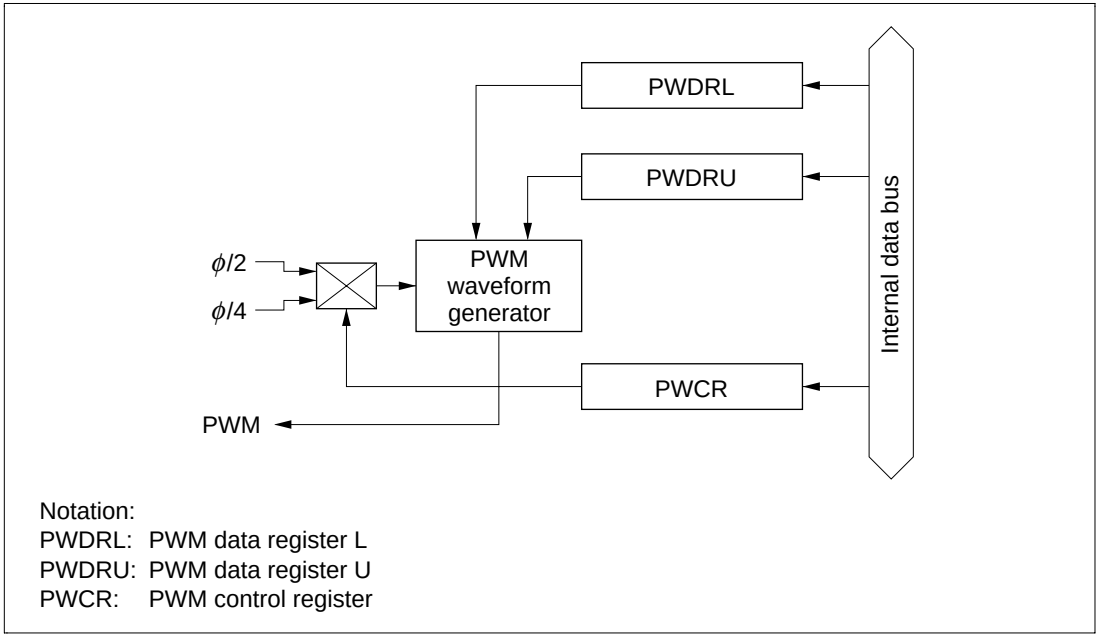



**Figure 11.1 Block Diagram of the 14 bit PWM**

### 11.1.3 Pin Configuration

Table 11.1 shows the output pin assigned to the 14-bit PWM.

**Table 11.1 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| PWM output pin | PWM | Output | Pulse-division PWM waveform output |

### 11.1.4 Register Configuration

Table 11.2 shows the register configuration of the 14-bit PWM.

**Table 11.2 Register Configuration**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| PWM control register | PWCR | W | H'FE | H'FFD0 |
| PWM data register U | PWDRU | W | H'C0 | H'FFD1 |
| PWM data register L | PWDRL | W | H'00 | H'FFD2 |

## 11.2 Register Descriptions

### 11.2.1 PWM Control Register (PWCR)

PWCR is an 8-bit write-only register for input clock selection.

Upon reset, PWCR is initialized to H'FE.

**Bits 7 to 1—Reserved Bits:** Bits 7 to 1 are reserved; they are always read as 1, and cannot be modified.

**Bit 0—Clock Select 0 (PWCR0):** Bit 0 selects the clock supplied to the 14-bit PWM. This bit is a write-only bit; it is always read as 1.

| Bit 0: PWCR0 | Description |
|---|---|
| 0 | The input clock is $\phi/2$ ($t_\phi^* = 2/\phi$). The conversion period is $16,384/\phi$, with a minimum modulation width of $1/\phi$. (initial value) |
| 1 | The input clock is $\phi/4$ ($t_\phi^* = 4/\phi$). The conversion period is $32,768/\phi$, with a minimum modulation width of $2/\phi$. |

Note: $t_\phi$: Period of PWM input clock

### 11.2.2 PWM Data Registers U and L (PWDRU, PWDRL)

PWDRU and PWDRL form a 14-bit write-only register, with the upper 6 bits assigned to PWDRU and the lower 8 bits to PWDRL. The value written to PWDRU and PWDRL gives the total high-level width of one PWM waveform cycle.

When 14-bit data is written to PWDRU and PWDRL, the register contents are latched in the PWM waveform generator, updating the PWM waveform generation data. The 14-bit data should always be written in the following sequence, first to PWDRL and then to PWDRU.

1. Write the lower 8 bits to PWDRL.
2. Write the upper 6 bits to PWDRU.

PWDRU and PWDRL are write-only registers. If they are read, all bits are read as 1.

Upon reset, PWDRU and PWDRL are initialized to H'C000.

## 11.3 Operation

When using the 14-bit PWM, set the registers in the following sequence.

1. Set bit PWM in port mode register 1 (PMR1) to 1 so that pin $P1_4$/PWM is designated for PWM output.

2. Set bit PWCR0 in the PWM control register (PWCR) to select a conversion period of either 32,768/$\phi$ (PWCR0 = 1) or 16,384/$\phi$ (PWCR0 = 0).

3. Set the output waveform data in PWM data registers U and L (PWDRU/L). Be sure to write in the correct sequence, first PWDRL then PWDRU. When data is written to PWDRU, the data in these registers will be latched in the PWM waveform generator, updating the PWM waveform generation in synchronization with internal signals.

   One conversion period consists of 64 pulses, as shown in figure 11.2. The total of the high-level pulse widths during this period ($T_H$) corresponds to the data in PWDRU and PWDRL. This relation can be represented as follows.

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

   where $t\phi$ is the PWM input clock period, either 2/$\phi$ (bit PWCR0 = 0) or 4/$\phi$ (bit PWCR0 = 1).

   Example: Settings in order to obtain a conversion period of 8,192 µs:

   When bit PWCR0 = 0, the conversion period is 16,384/$\phi$, so $\phi$ must be 2 MHz. In this case $t_{fn}$ = 128 µs, with 1/$\phi$ (resolution) = 0.5 µs.

   When bit PWCR0 = 1, the conversion period is 32,768/$\phi$, so $\phi$ must be 4 MHz. In this case $t_{fn}$ = 128 µs, with 2/$\phi$ (resolution) = 0.5 µs.

Accordingly, for a conversion period of 8,192 µs, the system clock frequency ($\phi$) must be 2 MHz or 4 MHz.

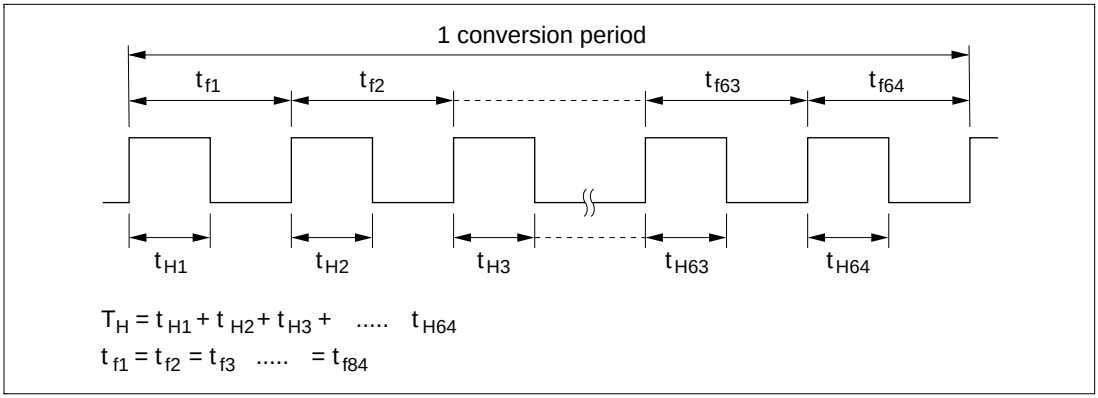



**Figure 11.2 PWM Output Waveform**

# Section 12 A/D Converter

## 12.1 Overview

The H8/3834 Series includes on-chip a resistance-ladder-based successive-approximation analog-to-digital converter, and can convert up to 12 channels of analog input.

### 12.1.1 Features

The A/D converter has the following features.

- 8-bit resolution
- 12 input channels
- Conversion time: approx. 12.4 µs per channel (at 5 MHz operation)
- Built-in sample-and-hold function
- Interrupt requested on completion of A/D conversion
- A/D conversion can be started by external trigger input

### 12.1.2 Block Diagram

Figure 12.1 shows a block diagram of the A/D converter.

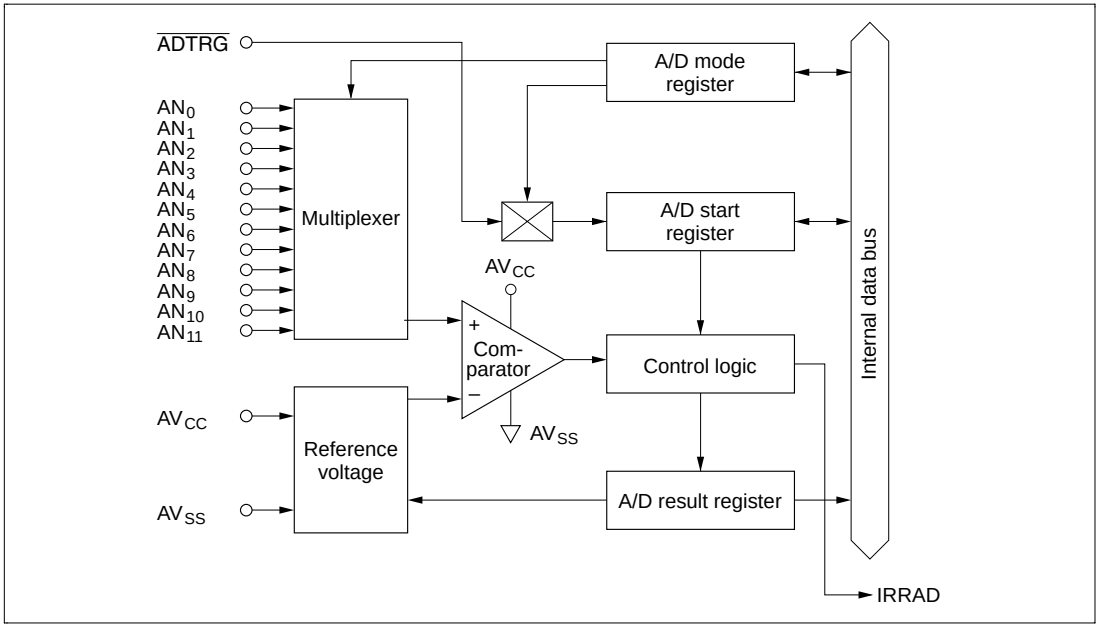



**Figure 12.1 Block Diagram of the A/D Converter**

### 12.1.3 Pin Configuration

Table 12.1 shows the A/D converter pin configuration.

**Table 12.1 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| Analog power supply pin | $AV_{CC}$ | Input | Power supply and reference voltage of analog part |
| Analog ground pin | $AV_{SS}$ | Input | Ground and reference voltage of analog part |
| Analog input pin 0 | $AN_0$ | Input | Analog input channel 0 |
| Analog input pin 1 | $AN_1$ | Input | Analog input channel 1 |
| Analog input pin 2 | $AN_2$ | Input | Analog input channel 2 |
| Analog input pin 3 | $AN_3$ | Input | Analog input channel 3 |
| Analog input pin 4 | $AN_4$ | Input | Analog input channel 4 |
| Analog input pin 5 | $AN_5$ | Input | Analog input channel 5 |
| Analog input pin 6 | $AN_6$ | Input | Analog input channel 6 |
| Analog input pin 7 | $AN_7$ | Input | Analog input channel 7 |
| Analog input pin 8 | $AN_8$ | Input | Analog input channel 8 |
| Analog input pin 9 | $AN_9$ | Input | Analog input channel 9 |
| Analog input pin 10 | $AN_{10}$ | Input | Analog input channel 10 |
| Analog input pin 11 | $AN_{11}$ | Input | Analog input channel 11 |
| External trigger input pin | $\overline{\text{ADTRG}}$ | Input | External trigger input for starting A/D conversion |

### 12.1.4 Register Configuration

Table 12.2 shows the A/D converter register configuration.

**Table 12.2 Register Configuration**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| A/D mode register | AMR | R/W | H'30 | H'FFC4 |
| A/D start register | ADSR | R/W | H'7F | H'FFC6 |
| A/D result register | ADRR | R | Not fixed | H'FFC5 |

## 12.2 Register Descriptions

### 12.2.1 A/D Result Register (ADRR)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADR7 | ADR6 | ADR5 | ADR4 | ADR3 | ADR2 | ADR1 | ADR0 |
| Initial value | — | — | — | — | — | — | — | — |
| Read/Write | R | R | R | R | R | R | R | R |

The A/D result register (ADRR) is an 8-bit read-only register for holding the results of analog-to-digital conversion.

ADRR can be read by the CPU at any time, but the ADRR values during A/D conversion are not fixed.

After A/D conversion is complete, the conversion result is stored in ADRR as 8-bit data; this data is held in ADRR until the next conversion operation starts.

ADRR is not cleared on reset.

### 12.2.2 A/D Mode Register (AMR)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CKS | TRGE | — | — | CH3 | CH2 | CH1 | CH0 |
| Initial value | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | — | — | R/W | R/W | R/W | R/W |

AMR is an 8-bit read/write register for specifying the A/D conversion speed, external trigger option, and the analog input pins.

Upon reset, AMR is initialized to H'30.

**Bit 7—Clock Select (CKS):** Bit 7 sets the A/D conversion speed.

| | | Conversion Time | |
|---|---|---|---|
| **Bit 7: CKS** | **Conversion Period** | **$\phi$ = 2 MHz** | **$\phi$ = 5 MHz** |
| 0 | 62/$\phi$ (initial value) | 31 µs | 12.4 µs |
| 1 | 31/$\phi$ | 15.5 µs | —* |

Note: * Operation is not guaranteed if the conversion time is less than 12.4 µs. Set bit 7 for a value of at least 12.4 µs.

**Bit 6—External Trigger Select (TRGE):** Bit 6 enables or disables the start of A/D conversion by external trigger input.

| **Bit 6: TRGE** | **Description** |
|---|---|
| 0 | Disables start of A/D conversion by external trigger (initial value) |
| 1 | Enables start of A/D conversion by rising or falling edge of external trigger at pin $\overline{\text{ADTRG}}$* |

Note: * The external trigger ($\overline{\text{ADTRG}}$) edge is selected by bit INTEG4 of the IRQ edge select register (IEGR). See 3.3.2, Interrupt Edge Select Register (IEGR), for details.

**Bits 5 and 4—Reserved Bits:** Bits 5 and 4 are reserved; they are always read as 1, and cannot be modified.

**Bits 3 to 0—Channel Select (CH3 to CH0):** Bits 3 to 0 select the analog input channel.

The channel selection should be made while bit ADSF is cleared to 0.

| Bit 3: CH3 | Bit 2: CH2 | Bit 1: CH1 | Bit 0: CH0 | Analog Input Channel | |
|---|---|---|---|---|---|
| 0 | 0 | * | * | No channel selected | (initial value) |
| | 1 | 0 | 0 | $AN_0$ | |
| | | | 1 | $AN_1$ | |
| | | 1 | 0 | $AN_2$ | |
| | | | 1 | $AN_3$ | |
| 1 | 0 | 0 | 0 | $AN_4$ | |
| | | | 1 | $AN_5$ | |
| | | 1 | 0 | $AN_6$ | |
| | | | 1 | $AN_7$ | |
| | 1 | 0 | 0 | $AN_8$ | |
| | | | 1 | $AN_9$ | |
| | | 1 | 0 | $AN_{10}$ | |
| | | | 1 | $AN_{11}$ | |

Note: * Don't care

### 12.2.3 A/D Start Register (ADSR)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADSF | — | — | — | — | — | — | — |
| Initial value | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | — | — | — | — | — | — | — |

The A/D start register (ADSR) is an 8-bit read/write register for starting and stopping A/D conversion.

A/D conversion is started by writing 1 to the A/D start flag (ADSF) or by input of the designated edge of the external trigger signal, which also sets ADSF to 1. When conversion is complete, the converted data is set in the A/D result register (ADRR), and at the same time ADSF is cleared to 0.

**Bit 7—A/D Start Flag (ADSF):** Bit 7 controls and indicates the start and end of A/D conversion.

| Bit 7: ADSF | Description | |
|---|---|---|
| 0 | Read: Indicates the completion of A/D conversion | (initial value) |
| | Write: Stops A/D conversion | |
| 1 | Read: Indicates A/D conversion in progress | |
| | Write: Starts A/D conversion | |

**Bits 6 to 0—Reserved Bits:** Bits 6 to 0 are reserved; they are always read as 1, and cannot be modified.

## 12.3 Operation

### 12.3.1 A/D Conversion Operation

The A/D converter operates by successive approximations, and yields its conversion result as 8-bit data.

A/D conversion begins when software sets the A/D start flag (bit ADSF) to 1. Bit ADSF keeps a value of 1 during A/D conversion, and is cleared to 0 automatically when conversion is complete.

The completion of conversion also sets bit IRRAD in interrupt request register 2 (IRR2) to 1. An A/D conversion end interrupt is requested if bit IENAD in interrupt enable register 2 (IENR2) is set to 1.

If the conversion time or input channel needs to be changed in the A/D mode register (AMR) during A/D conversion, bit ADSF should first be cleared to 0, stopping the conversion operation, in order to avoid malfunction.

### 12.3.2 Start of A/D Conversion by External Trigger Input

The A/D converter can be made to start A/D conversion by input of an external trigger signal. External trigger input is enabled at pin $\overline{\text{ADTRG}}$ when bit IRQ4 in port mode register 2 (PMR2) is set to 1, and bit TRGE in AMR is set to 1. Then when the input signal edge designated in bit IEG4 of the IRQ edge select register (IEGR) is detected at pin $\overline{\text{ADTRG}}$, bit ADSF in ADSR will be set to 1, starting A/D conversion.

Figure 12.2 shows the timing.

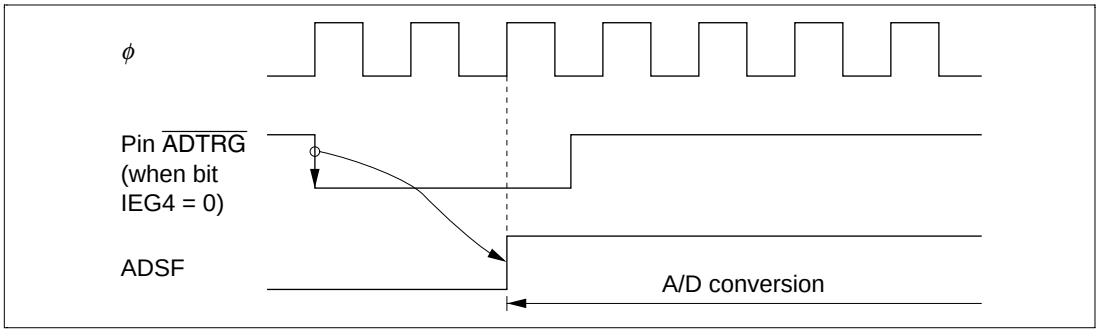



**Figure 12.2 External Trigger Input Timing**

## 12.4 Interrupts

When A/D conversion ends (ADSF changes from 1 to 0), bit IRRAD in interrupt request register 2 (IRR2) is set to 1.

A/D conversion end interrupts can be enabled or disabled by means of bit IENAD in interrupt enable register 2 (IENR2).

For further details see 3.3, Interrupts.

## 12.5 Typical Use

An example of how the A/D converter can be used is given below, using channel 1 (pin AN1) as the analog input channel. Figure 12.3 shows the operation timing.

- Bits CH3 to CH0 of the A/D mode register (AMR) are set to 0101, making pin $AN_1$ the analog input channel. A/D interrupts are enabled by setting bit IENAD to 1, and A/D conversion is started by setting bit ADSF to 1.
- When A/D conversion is complete, bit IRRAD is set to 1, and the A/D conversion result is stored in the A/D result register (ADRR). At the same time ADSF is cleared to 0, and the A/D converter goes to the idle state.
- Bit IENAD = 1, so an A/D conversion end interrupt is requested.
- The A/D interrupt handling routine starts.
- The A/D conversion result is read and processed.
- The A/D interrupt handling routine ends.

If ADSF is set to 1 again afterward, A/D conversion starts and steps 2 through 6 take place.

Figures 12.4 and 12.5 show flow charts of procedures for using the A/D converter.

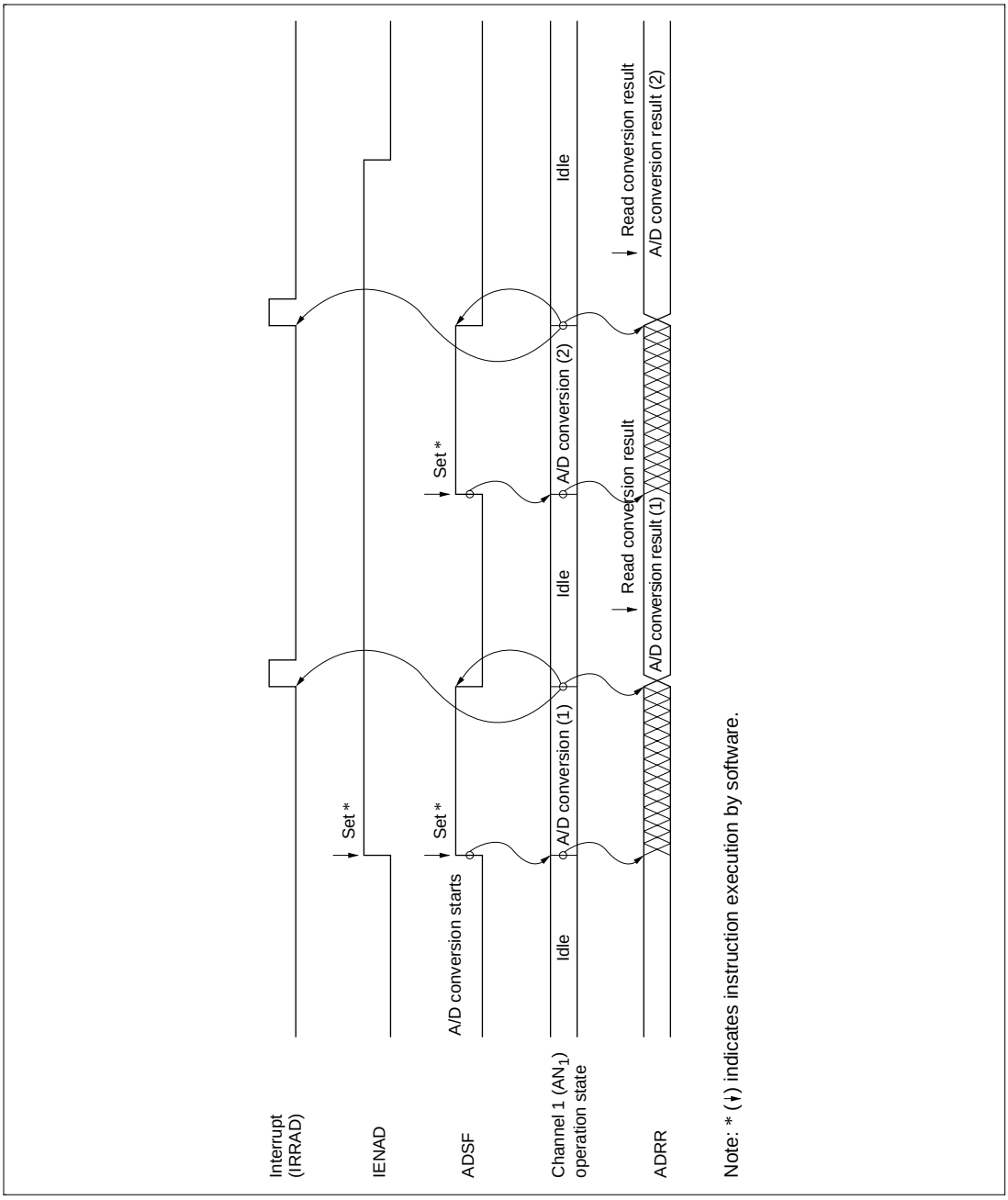



**Figure 12.3 Typical A/D Converter Operation Timing**

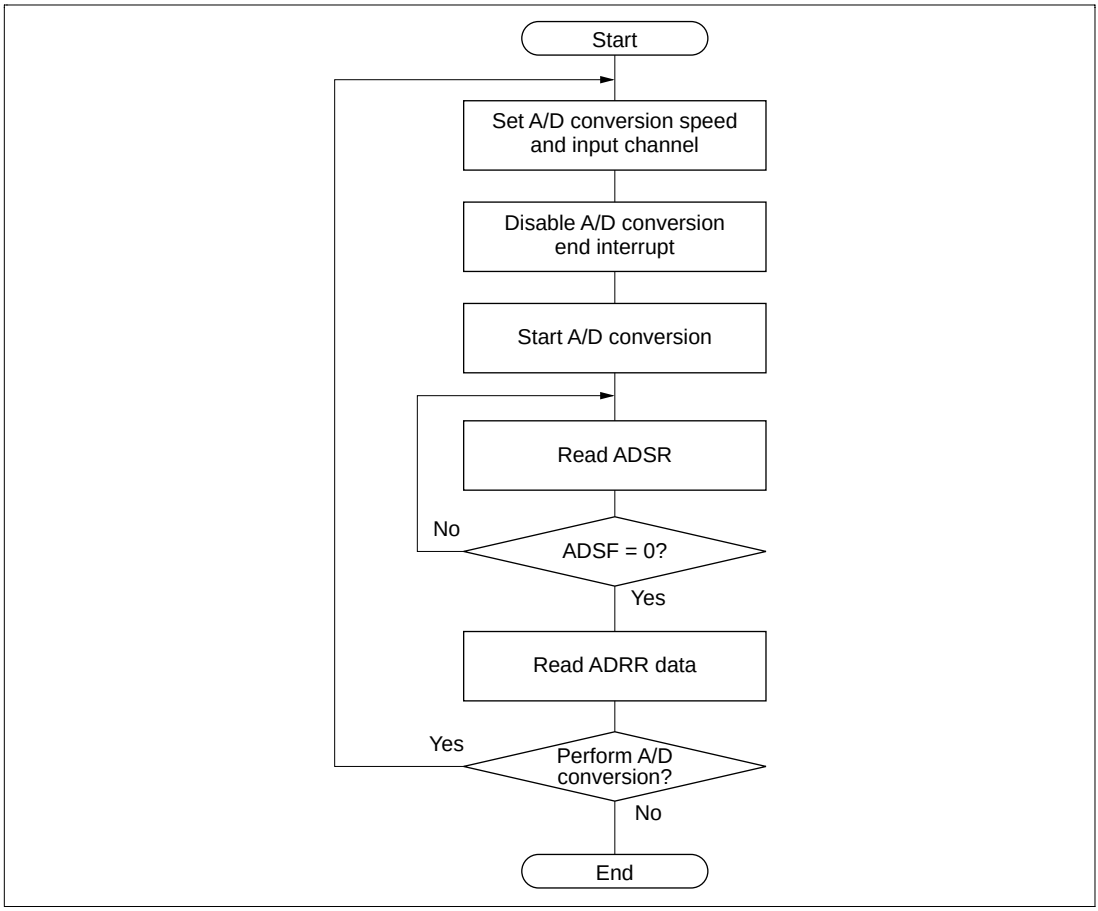



**Figure 12.4 Flow Chart of Procedure for Using A/D Converter (1) (Polling by Software)**

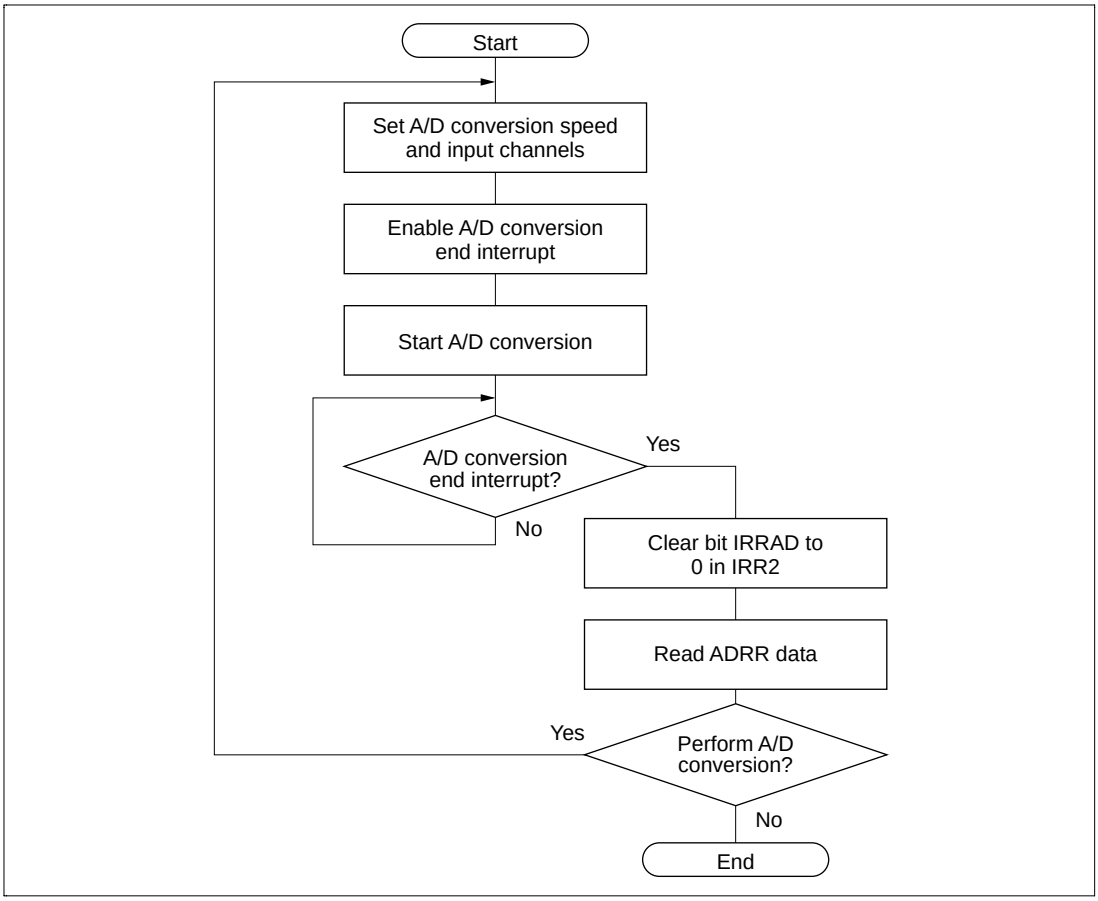



**Figure 12.5 Flow Chart of Procedure for Using A/D Converter (2) (Interrupts Used)**

## 12.6 Application Notes

- Data in the A/D result register (ADRR) should be read only when the A/D start flag (ADSF) in the A/D start register (ADSR) is cleared to 0.
- Changing the digital input signal at an adjacent pin during A/D conversion may adversely affect conversion accuracy.

# Section 13 LCD Controller/Driver

## 13.1 Overview

The H8/3834 Series has an on-chip segment-type LCD controller circuit, LCD driver, and power supply circuit, for direct driving of an LCD panel.

### 13.1.1 Features

Features of the LCD controller/driver are as follows.

- Display capacity

| | Duty | Internal Driver | External Segment Expansion Driver |
|---|---|---|---|
| On-chip driver only | — | 40 segments | 0 |
| Use with external segment expansion driver | Static | 36 segments | 476 segments |
| | 1/2 | 36 segments | 220 segments |
| | 1/3 | 36 segments | 92 segments |
| | 1/4 | 36 segments | 92 segments |

The HD66100 can be used for external expansion of the number of segments.

- LCD RAM capacity
  8 bits × 64 bytes (512 bits)
- Word access to LCD RAM
- Segment output pins can be switched to general-purpose ports in groups of 4
- Unused common output pins can be used either for boosting common output (by parallel connection) or as ports.
- Displays in all operation modes except standby mode.
- Choice of 11 frame frequencies
- Internal voltage divider for liquid crystal driver power supply

### 13.1.2 Block Diagram

Figure 13.1 shows a block diagram of the LCD controller/driver.

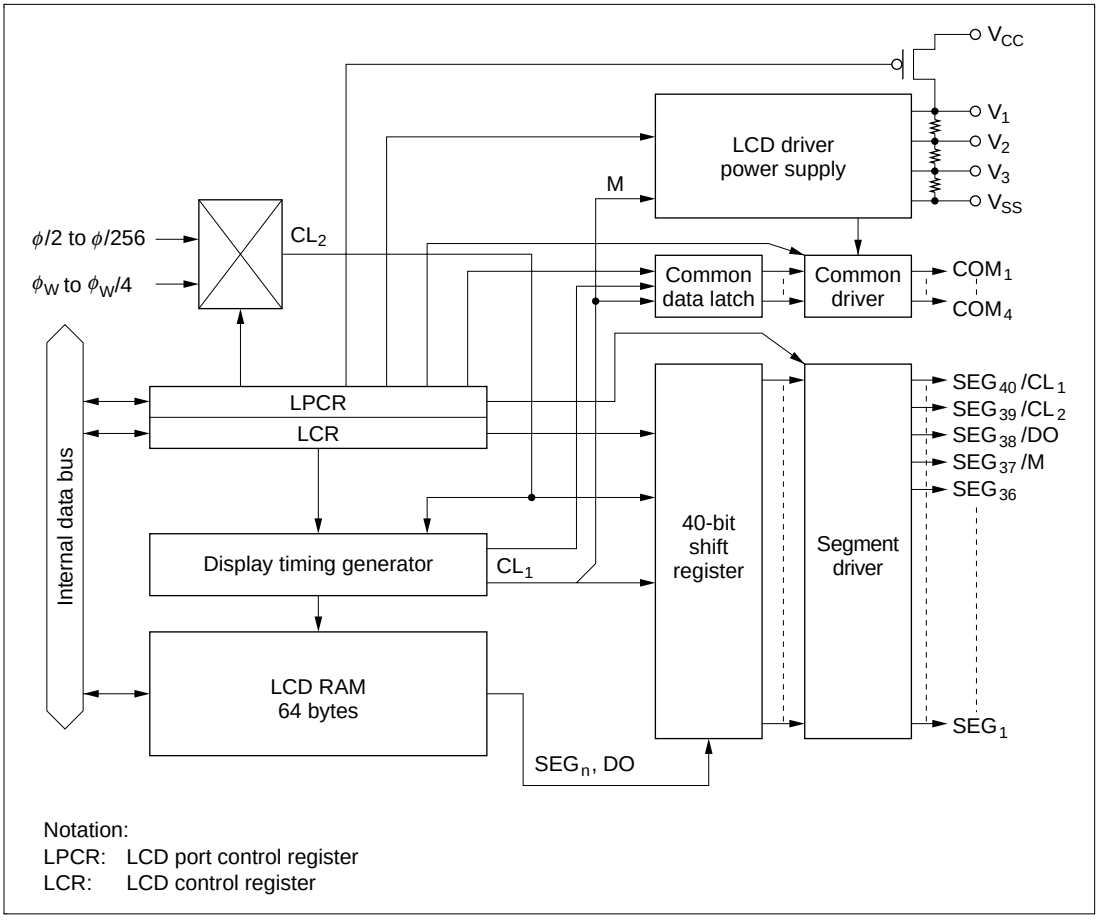



**Figure 13.1 LCD Controller/Driver Block Diagram**

### 13.1.3 Pin Configuration

Table 13.1 shows the output pins assigned to the LCD controller/driver.

**Table 13.1 Pin Configuration**

| Name | Abbrev. | I/O | Function |
|---|---|---|---|
| LCD segment output | $SEG_{40}$ to $SEG_1$ | Output | Liquid crystal segment driver pins. All pins can be programmed also as ports. |
| LCD common output | $COM_4$ to | Output | Liquid crystal common driver pins. Parallel |
| | $COM_1$ | | connection is possible at static and 1/2 duty. |
| External segment expansion signal | $CL_1$ | Output | Display data latch clock; doubles as $SEG_{40}$ |
| | $CL_2$ | Output | Display data shift clock; doubles as $SEG_{39}$ |
| | M | Output | LCD alternating signal; doubles as $SEG_{37}$ |
| | DO | Output | Serial display data; doubles as $SEG_{38}$ |
| LCD power supply | $V_1$, $V_2$, $V_3$ | Input | For external connection to bypass capacitor or for use of external power supply circuit |

### 13.1.4 Register Configuration

Table 13.2 shows the register configuration of the LCD controller/driver.

**Table 13.2 Register Configuration**

| Name | Abbrev. | R/W | Initial Value | Address |
|---|---|---|---|---|
| LCD port control register | LPCR | R/W | H'00 | H'FFC0 |
| LCD control register | LCR | R/W | H'80 | H'FFC1 |
| LCD RAM | — | R/W | Not fixed | H'F740 to H'F77F* |

Note: * Value after reset.

## 13.2 Register Descriptions

### 13.2.1 LCD Port Control Register (LPCR)

The LCD port control register is an 8-bit read/write register, used for selecting the duty cycle and the LCD driver and pin functions, etc. Upon reset, LPCR is initialized to H'00.

**Bits 7 to 5—Duty and Common Function Select (DTS1, DTS0, CMX):** Bits 7 to 6 select a driver duty of static, 1/2, 1/3, or 1/4. Bit 5 determines whether the common pins not used at a given duty are to be used as ports or, in order to increase the common driving capacity, as multiple pins outputting the same waveform.

| Bit 7: DTS1 | Bit 6: DTS0 | Bit 5: CMX | Duty | Common Driver*[1] | Other Uses |
|---|---|---|---|---|---|
| 0 | 0 | 0 | Static | $COM_1$ (initial value) | $COM_4$, $COM_3$ and $COM_2$ usable as ports |
| | | 1 | | $COM_4$ to $COM_1$ | $COM_4$, $COM_3$ and $COM_2$ output the same waveform as $COM_1$ |
| 0 | 1 | 0 | 1/2 duty | $COM_2$ to $COM_1$ | $COM_4$ and $COM_3$ usable as ports |
| | | 1 | | $COM_4$ to $COM_1$ | $COM_4$ outputs the same waveform as $COM_3$, and $COM_2$ the same waveform as $COM_1$ |
| 1 | 0 | 0 | 1/3 duty | $COM_3$ to $COM_1$ | $COM_4$ usable as port |
| | | 1 | | $COM_4$ to $COM_1$ | $COM_4$ outputs a non-select waveform*[2] |
| 1 | 1 | 0 | 1/4 duty | $COM_4$ to $COM_1$ | — |
| | | 1 | | | |

Notes: 1. Pins $COM_4$ to $COM_1$ become ports when bit SGX = 0 and bits SGS3 to SGS0 = 0000. Otherwise the common drivers are as indicated in the table above.
2. A non-select waveform is always output at pin $COM_4$, which therefore should not be used.

**Bit 4—Expansion Signal Select (SGX):** Bit 4 selects whether pins $SEG_{40}/CL_1$, $SEG_{39}/CL_2$, $SEG_{38}$/DO, and $SEG_{37}$/M are used as segment pins ($SEG_{40}$ to $SEG_{37}$) or as external segment expansion pins ($CL_1$, $CL_2$, DO, M).

| Bit 4: SGX | Description | |
|---|---|---|
| 0 | Pins $SEG_{40}$ to $SEG_{37}$* | (initial value) |
| 1 | Pins $CL_1$, $CL_2$, DO, M | |

Note: * Selected as ports when bits SGS3 to SGS0 = 0000.

**Bits 3 to 0—Segment Driver Select (SGS3 to SGS0):** Bits 3 to 0 select the pins to be used as segment drivers.

| Bit 4: SGX | Bit 3: SGS3 | Bit 2: SGS2 | Bit 1: SGS1 | Bit 0: SGS0 | Functions of Pins $SEG_{40}$ to $SEG_{21}$ | | | | | Remarks |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | $SEG_{40}$ to $SEG_{37}$ | $SEG_{36}$ to $SEG_{33}$ | $SEG_{32}$ to $SEG_{29}$ | $SEG_{28}$ to $SEG_{25}$ | $SEG_{24}$ to $SEG_{21}$ | |
| 0 | 0 | 0 | 0 | 0 | Port | Port | Port | Port | Port | (initial value) |
| | 0 | 0 | 0 | 1 | SEG | SEG | Port | Port | Port | |
| | 0 | 0 | 1 | 0 | SEG | SEG | SEG | Port | Port | |
| | 0 | 0 | 1 | 1 | SEG | SEG | SEG | SEG | Port | |
| | 0 | 1 | 0 | 0 | SEG | SEG | SEG | SEG | SEG | |
| | 0 | 1 | 0 | 1 | SEG | SEG | SEG | SEG | SEG | |
| | 0 | 1 | 1 | 0 | SEG | SEG | SEG | SEG | SEG | |
| | 0 | 1 | 1 | 1 | SEG | SEG | SEG | SEG | SEG | |
| | 1 | * | * | 0 | SEG | SEG | SEG | SEG | SEG | |
| | 1 | * | * | 1 | SEG | SEG | SEG | SEG | SEG | |
| 1 | 0 | 0 | 0 | 0 | External segment expansion | Port | Port | Port | Port | |
| | 0 | 0 | 0 | 1 | External segment expansion | SEG | Port | Port | Port | |
| | 0 | 0 | 1 | 0 | External segment expansion | SEG | SEG | Port | Port | |
| | 0 | 0 | 1 | 1 | External segment expansion | SEG | SEG | SEG | Port | |
| | 0 | 1 | 0 | 0 | External segment expansion | SEG | SEG | SEG | SEG | |
| | 0 | 1 | 0 | 1 | External segment expansion | SEG | SEG | SEG | SEG | |
| | 0 | 1 | 1 | 0 | External segment expansion | SEG | SEG | SEG | SEG | |
| | 0 | 1 | 1 | 1 | External segment expansion | SEG | SEG | SEG | SEG | |
| | 1 | * | * | 0 | External segment expansion | SEG | SEG | SEG | SEG | |
| | 1 | * | * | 1 | External segment expansion | SEG | SEG | SEG | SEG | |

Note: * Don't care

| | | | | | Functions of Pins $SEG_{20}$ to $SEG_1$ | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| **Bit 4: SGX** | **Bit 3: SGS3** | **Bit 2: SGS2** | **Bit 1: SGS1** | **Bit 0: SGS0** | **$SEG_{20}$ to $SEG_{17}$** | **$SEG_{16}$ to $SEG_{13}$** | **$SEG_{12}$ to $SEG_9$** | **$SEG_8$ to $SEG_5$** | **$SEG_4$ to $SEG_1$** | **Remarks** |
| 0 | 0 | 0 | 0 | 0 | Port | Port | Port | Port | Port | (initial value) |
| | 0 | 0 | 0 | 1 | Port | Port | Port | Port | Port | |
| | 0 | 0 | 1 | 0 | Port | Port | Port | Port | Port | |
| | 0 | 0 | 1 | 1 | Port | Port | Port | Port | Port | |
| | 0 | 1 | 0 | 0 | Port | Port | Port | Port | Port | |
| | 0 | 1 | 0 | 1 | SEG | Port | Port | Port | Port | |
| | 0 | 1 | 1 | 0 | SEG | SEG | Port | Port | Port | |
| | 0 | 1 | 1 | 1 | SEG | SEG | SEG | Port | Port | |
| | 1 | * | * | 0 | SEG | SEG | SEG | SEG | Port | |
| | 1 | * | * | 1 | SEG | SEG | SEG | SEG | SEG | |
| 1 | 0 | 0 | 0 | 0 | Port | Port | Port | Port | Port | |
| | 0 | 0 | 0 | 1 | Port | Port | Port | Port | Port | |
| | 0 | 0 | 1 | 0 | Port | Port | Port | Port | Port | |
| | 0 | 0 | 1 | 1 | Port | Port | Port | Port | Port | |
| | 0 | 1 | 0 | 0 | Port | Port | Port | Port | Port | |
| | 0 | 1 | 0 | 1 | SEG | Port | Port | Port | Port | |
| | 0 | 1 | 1 | 0 | SEG | SEG | Port | Port | Port | |
| | 0 | 1 | 1 | 1 | SEG | SEG | SEG | Port | Port | |
| | 1 | * | * | 0 | SEG | SEG | SEG | SEG | Port | |
| | 1 | * | * | 1 | SEG | SEG | SEG | SEG | SEG | |

Note: * Don't care

### 13.2.2 LCD Control Register (LCR)

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | PSW | ACT | DISP | CKS3 | CKS2 | CKS1 | CKS0 |
| Initial value | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

The LCD control register is an 8-bit read/write register for on/off control of the resistive voltage divider used as the LCD driver power supply, for display data control, and for frame frequency selection. Upon reset, LCR is initialized to H'80.

**Bit 7—Reserved Bit:** Bit 7 is reserved; it is always read as 1, and cannot be modified.

**Bit 6—Power Switch (PSW):** Bit 6 switches the resistive voltage divider provided to power the LCD driver on/off. In low-power modes when the LCD display is not used, or when an external power supply is used for the LCD, the resistive voltage divider can be switched off. When bit ACT = 0, or in standby mode, the resistive voltage divider is in the off state regardless of the bit 6 setting.

| Bit 6: PSW | Description | |
|---|---|---|
| 0 | LCD power supply resistive voltage divider off | (initial value) |
| 1 | LCD power supply resistive voltage divider on | |

**Bit 5:—Display Active (ACT):** Bit 5 selects whether the LCD controller/driver is used or not. When this bit is cleared to 0, the LCD controller/driver module halts operation, and the resistive voltage divider provided for the LCD driver power supply goes to the off state regardless of the PSW setting. However, register contents are retained.

| Bit 5: ACT | Description | |
|---|---|---|
| 0 | LCD controller/driver operation stopped | (initial value) |
| 1 | LCD controller/driver operational | |

**Bit 4—Display Data Control (DISP):** Bit 4 selects whether the LCD RAM contents are displayed or blank data is displayed regardless of the LCD RAM contents. This bit is valid also when the HD66100 is used for external segment expansion.

| Bit 4: DISP | Description | |
|---|---|---|
| 0 | Blank data displayed | (initial value) |
| 1 | LCD RAM data displayed | |

**Bits 3 to 0—Frame Frequency Select 3 to 0 (CKS3 to CKS0):** Bits 3 to 0 select the clock used by the LCD controller/driver, and the frame frequency. In subactive, watch, and subsleep modes the system clock ($\phi$) is stopped, so there will be no display in these modes if $\phi/2$ to $\phi/256$ is chosen as the clock source. For display in these modes, clock $\phi_W/2$ or $\phi_W/4$ must be selected.

| Bit 3: CKS3 | Bit 2: CKS2 | Bit 1: CKS1 | Bit 0: CKS0 | Clock | Frame Frequency*3 | |
|---|---|---|---|---|---|---|
| | | | | | $\phi$ = 5 MHz | $\phi$ = 625 kHz*1 |
| 0 | * | 0 | 0 | $\phi_W$ | 128 Hz*2 | 128 Hz*2 |
| 0 | * | 0 | 1 | $\phi_W/2$ | 64 Hz | 64 Hz |
| 0 | * | 1 | * | $\phi_W/4$ | 32 Hz | 32 Hz |
| 1 | 0 | 0 | 0 | $\phi/2$ | — | 610 Hz |
| 1 | 0 | 0 | 1 | $\phi/4$ | — | 305 Hz |
| 1 | 0 | 1 | 0 | $\phi/8$ | — | 153 Hz |
| 1 | 0 | 1 | 1 | $\phi/16$ | 610 Hz | 76.3 Hz |
| 1 | 1 | 0 | 0 | $\phi/32$ | 305 Hz | 38.1 Hz |
| 1 | 1 | 0 | 1 | $\phi/64$ | 153 Hz | — |
| 1 | 1 | 1 | 0 | $\phi/128$ | 76.3 Hz | — |
| 1 | 1 | 1 | 1 | $\phi/256$ | 38.1 Hz | — |

Notes: * Don't care

1. Frame frequency in active (medium-speed) mode when $\phi$ = 5 MHz
2. Only the upper 32 bytes of the display RAM are used.
3. When a duty cycle of 1/3 is chosen, the frame frequency will be 4/3 times the frequencies shown in the above table.

## 13.3 Operation

### 13.3.1 Settings Prior to LCD Display

Various decisions related to hardware and software must be made before using the LCD controller/driver with an LCD display. The settings are described below.

**Hardware Settings**

- Use at 1/2 duty
  To use at 1/2 duty, connect pins $V_2$ and $V_3$ as shown in figure 13.2.

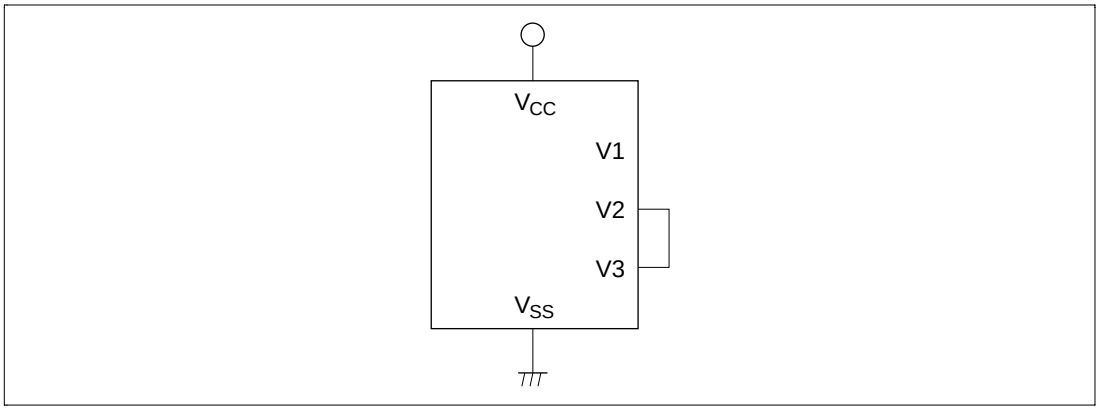



**Figure 13.2 LCD Driver Power Supply Processing at 1/2 Duty**

- Large-panel display

  Because of the large impedance of the built-in resistive voltage divider, the H8/3834 Series LCD controller/driver is not well suited to driving large-panel displays. If use of a large panel leads to an unclear display, refer to 13.3.5 Boosting the LCD Driver Power Supply. At static and 1/2 duty it is possible to boost the common output driving capacity. Set bit CMX to 1 when selecting the duty cycle. In this mode, at static duty pins $COM_4$ to $COM_1$ output the same waveform, while at 1/2 duty pins $COM_2$ and $COM_1$ output the $COM_1$ waveform and pins $COM_4$ and $COM_3$ output the $COM_2$ waveform.
- Segment expansion

  The HD66100 can be connected externally to expand the number of segments. See 13.3.3, Connection to HD66100.

**Software Settings**

- Duty cycle selection

  The duty cycle is selected in bits DTS1 and DTS0, with a choice of static, 1/2, 1/3, or 1/4 duty.
- Segment driver selection

  The segment drivers to be used are selected in bits SGS3 to SGS0.
- Frame frequency selection

  The frame frequency is selected in bits CKS3 to CKS0. The frame frequency should be selected depending on the specification of the LCD panel to be used. Refer to 13.3.4, Operation in Power-Down Modes, for information on clock selection in watch mode, subactive mode, and subsleep mode.

### 13.3.2 Relation of LCD RAM to Display

The relation of the LCD RAM to segments depends on the duty cycle. LCD RAM memory maps for each duty cycle when segments are not expanded externally are shown in figures 13.3 to 13.6. When segments are expanded externally, the LCD RAM memory maps for each duty cycle are as shown in figures 13.7 to 13.10. It is also possible to use only external segments and not use the segment pins on this chip, in which case the LCD RAM memory map is as shown in figure 13.11.

After setting the registers that control the LCD display, write data to the area corresponding to the duty cycle selected, using the same instructions as for the ordinary RAM. If the display is switched on, the data will be displayed automatically. Both word and byte access instructions can be used for writing to the LCD RAM.

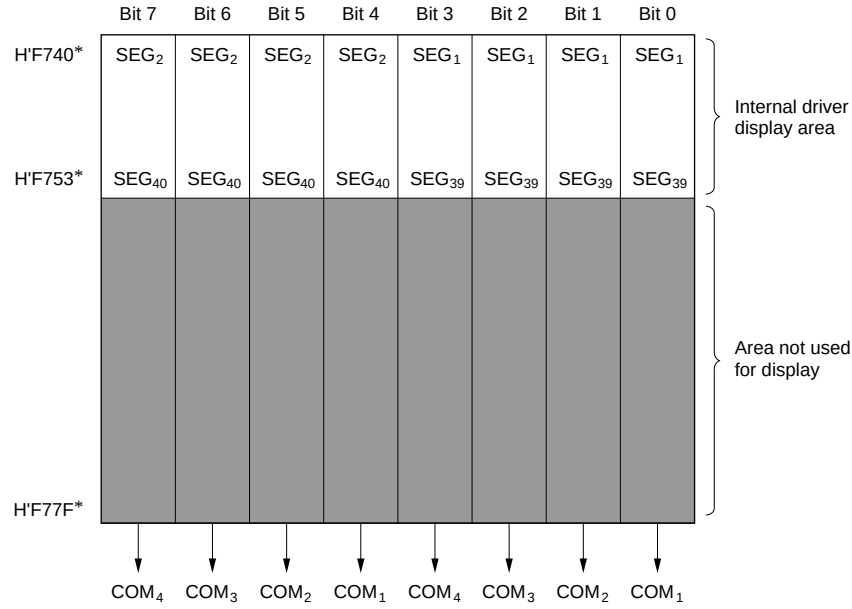
### 13.3.3 Connection to HD66100

To expand the number of segments externally, connect the H8/3834 Series to the HD66100 segment chip. The HD66100 chip provides an additional 80 segments. When external segments are used, set bit SGX in LPCR for use of pins $SEG_{40}$ to $SEG_{37}$ as external segment expansion signal pins. Data will be output starting from LCD RAM pin $SEG_{37}$. When bits SGS3 to SGS0 in LPCR are set to 0000, data will be output starting from LCD RAM pin $SEG_1$.

Figure 13.12 shows typical connections to the HD66100. The output level is determined by the combination of data pins and pin M; but that combination differs between the H8/3834 Series and the HD66100. Table 13.3 shows the output level of the LCD driver power supply.
Figure 13.13 shows the common and segment waveforms at each duty.

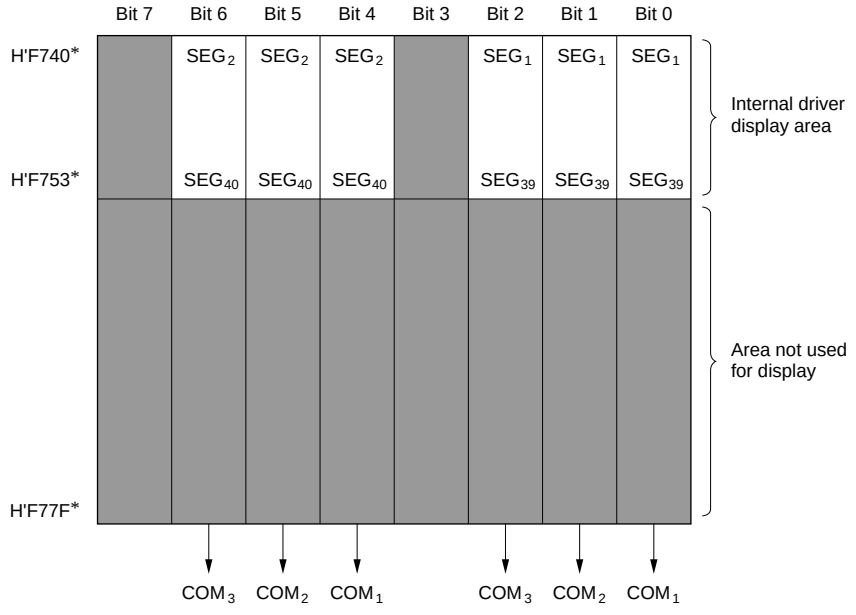
If bit ACT = 0, then if $CL_2$ = 0, $CL_1$ = 0 and M = 0, DO stops with the data output at that moment (1 or 0). In standby mode the expansion pins are in the high-impedance (floating) state.

External expansion increases the load on the LCD panel, as a result of which the internal power supply may not have sufficient capacity. In that case refer to 13.3.5, Boosting the LCD Driver Power Supply.



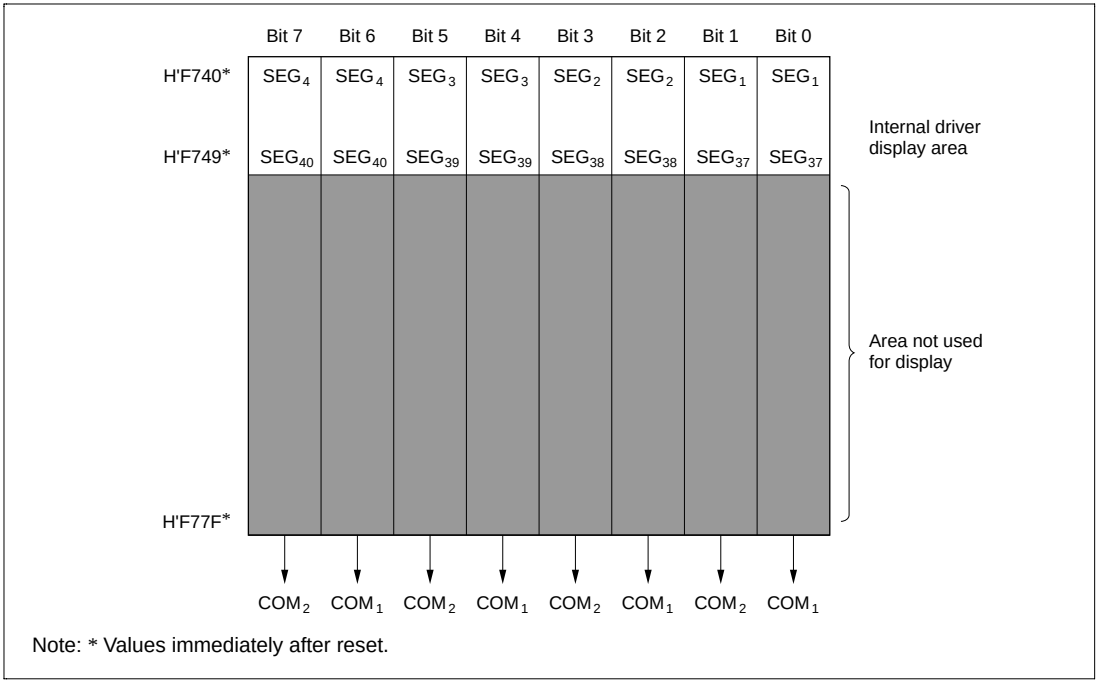

Note: * Values immediately after reset.

**Figure 13.3 LCD RAM Map: No External Segment Expansion (1/4 Duty)**



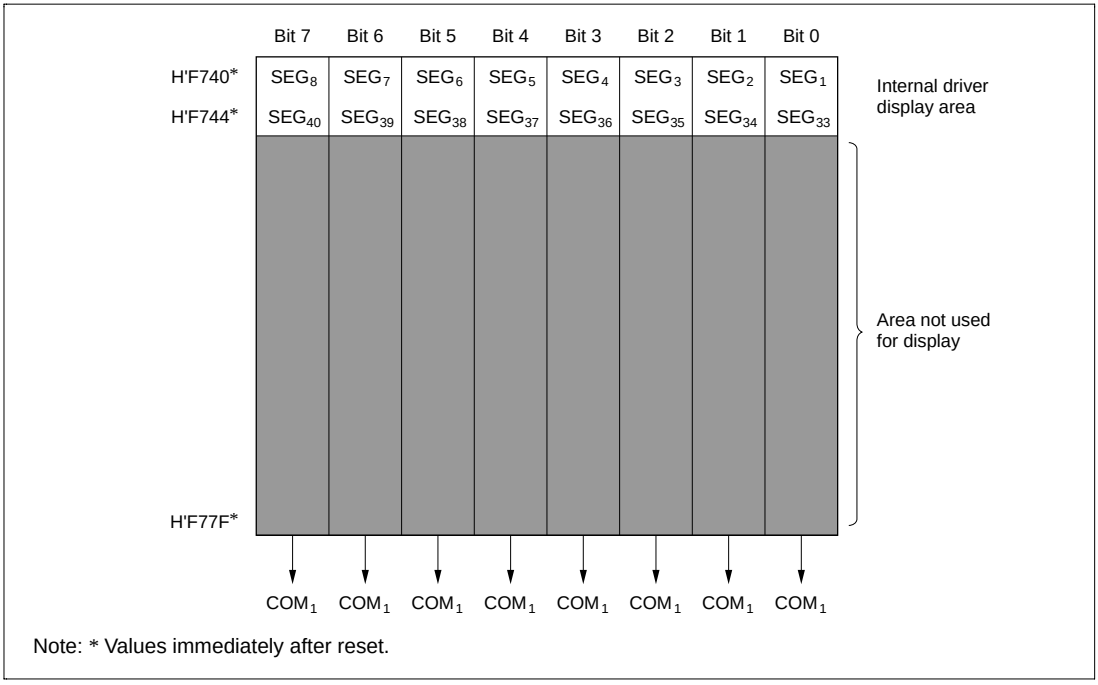

Note: * Values immediately after reset.

**Figure 13.4 LCD RAM Map: No External Segment Expansion (1/3 Duty)**



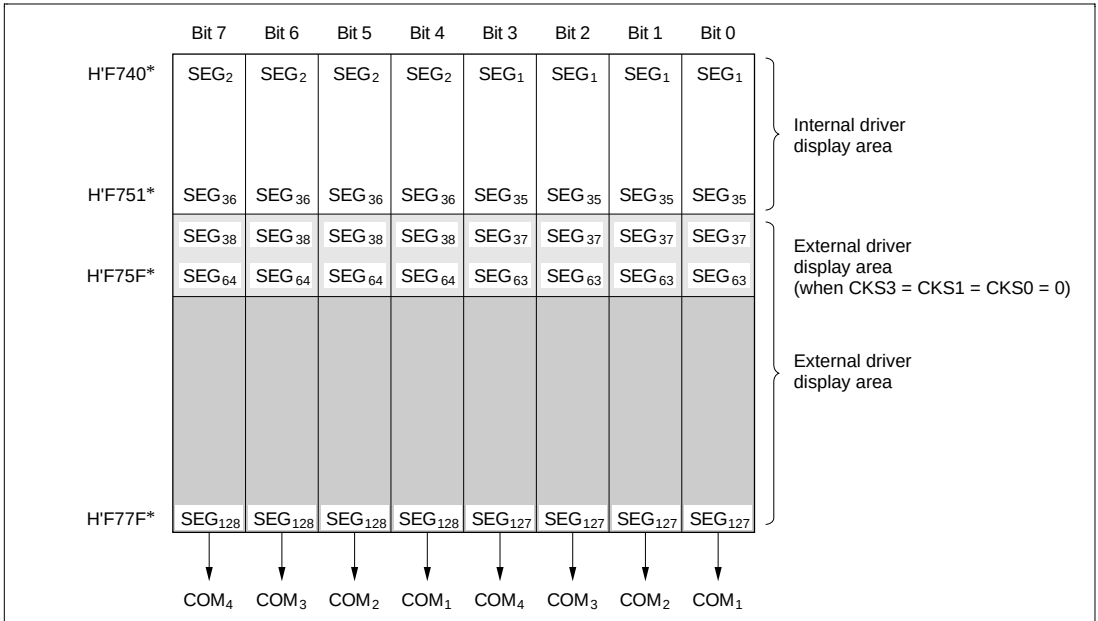

Note: * Values immediately after reset.

**Figure 13.5 LCD RAM Map: No External Segment Expansion (1/2 Duty)**



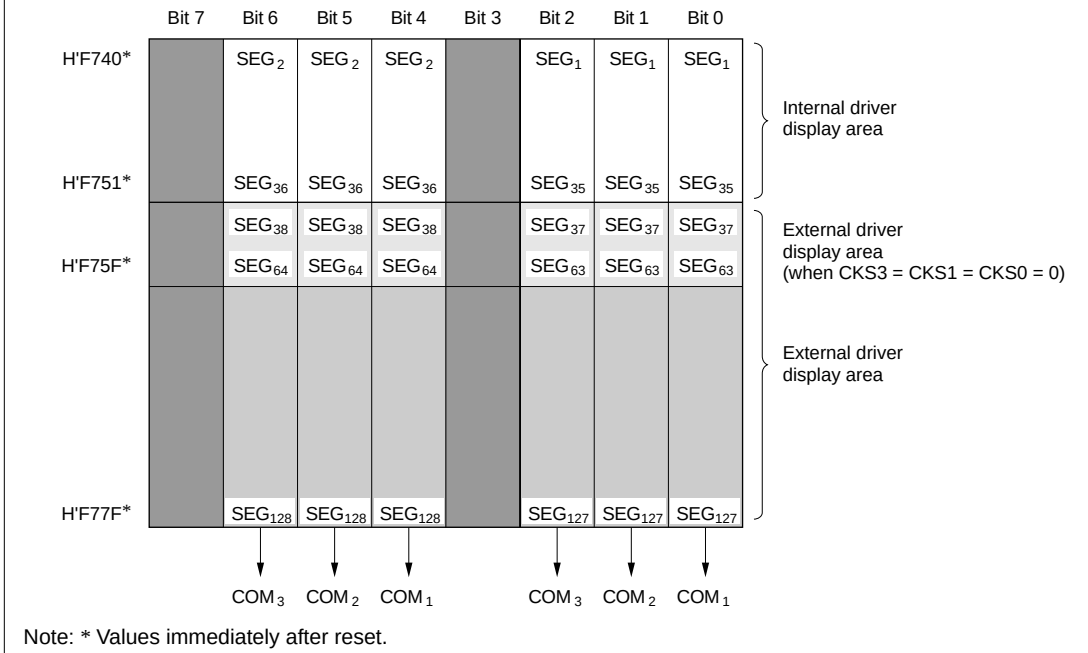

Note: * Values immediately after reset.

**Figure 13.6 LCD RAM Map: No External Segment Expansion (Static Duty)**



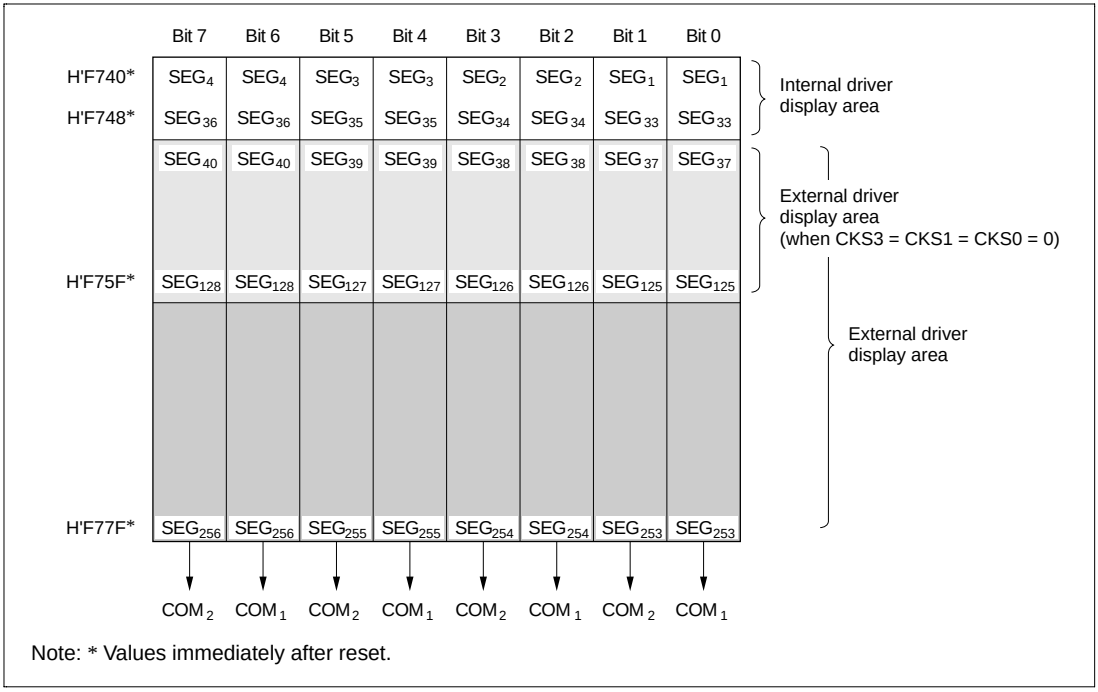

Note: * Values immediately after reset.

**Figure 13.7 LCD RAM Map: External Segment Expansion (1/4 Duty)**



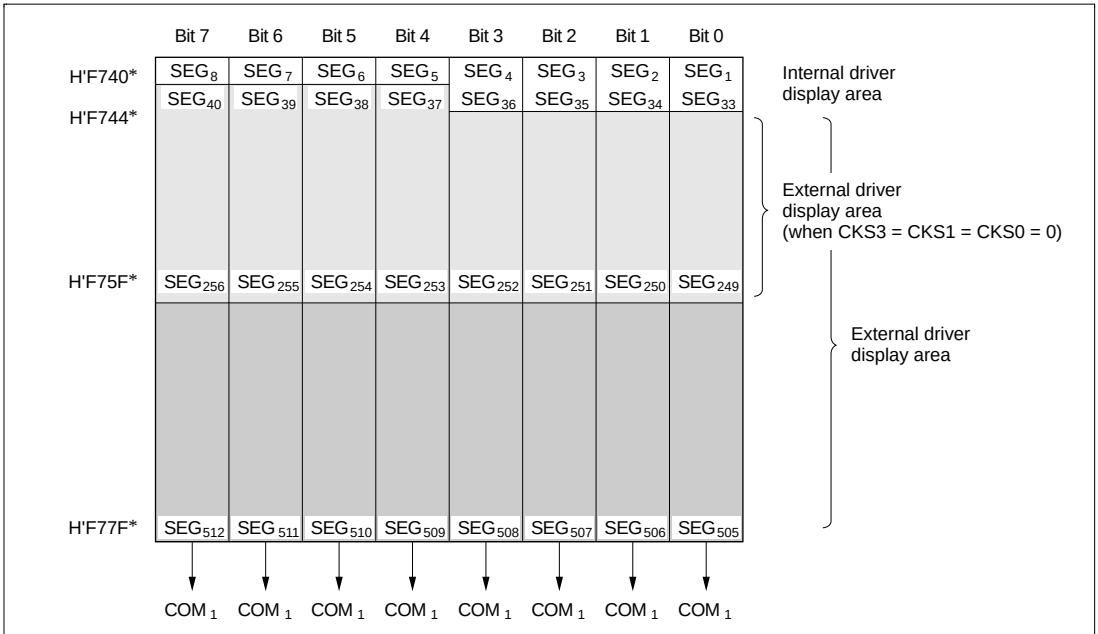

Note: * Values immediately after reset.

**Figure 13.8 LCD RAM Map: External Segment Expansion (1/3 Duty)**



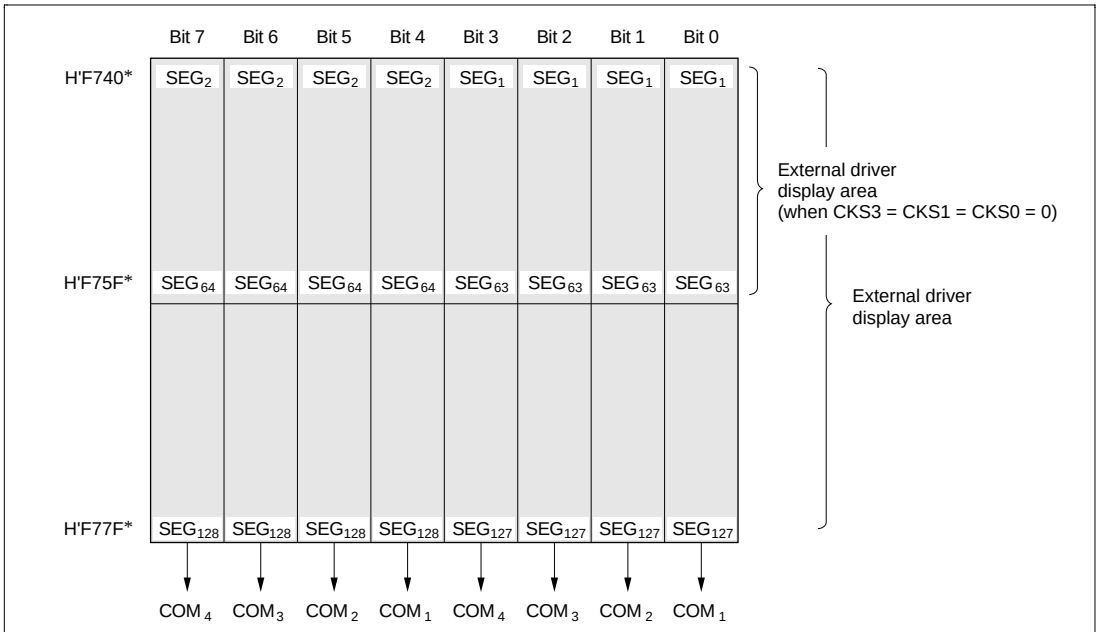

Note: * Values immediately after reset.

**Figure 13.9 LCD RAM Map: External Segment Expansion (1/2 Duty)**




Note: * Values immediately after reset.

**Figure 13.10 LCD RAM Map: External Segment Expansion (Static Duty)**




Note: * Values immediately after reset.

**Figure 13.11 LCD RAM Map When All External Segments are Used (Example: SGX = 1, SGS3 to SGS0 = 0000, 1/4 Duty)**

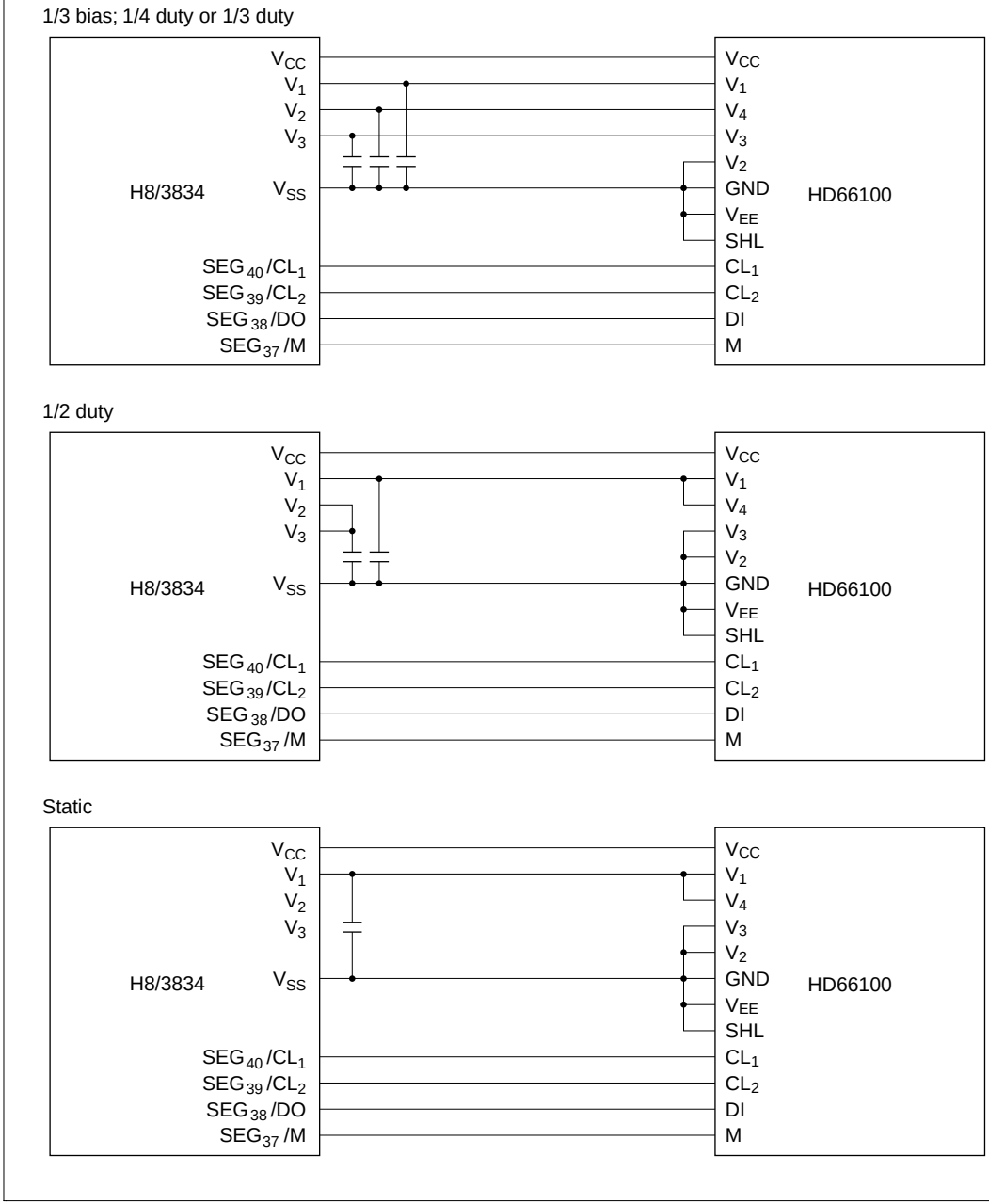



**Figure 13.12 Connection to HD66100**

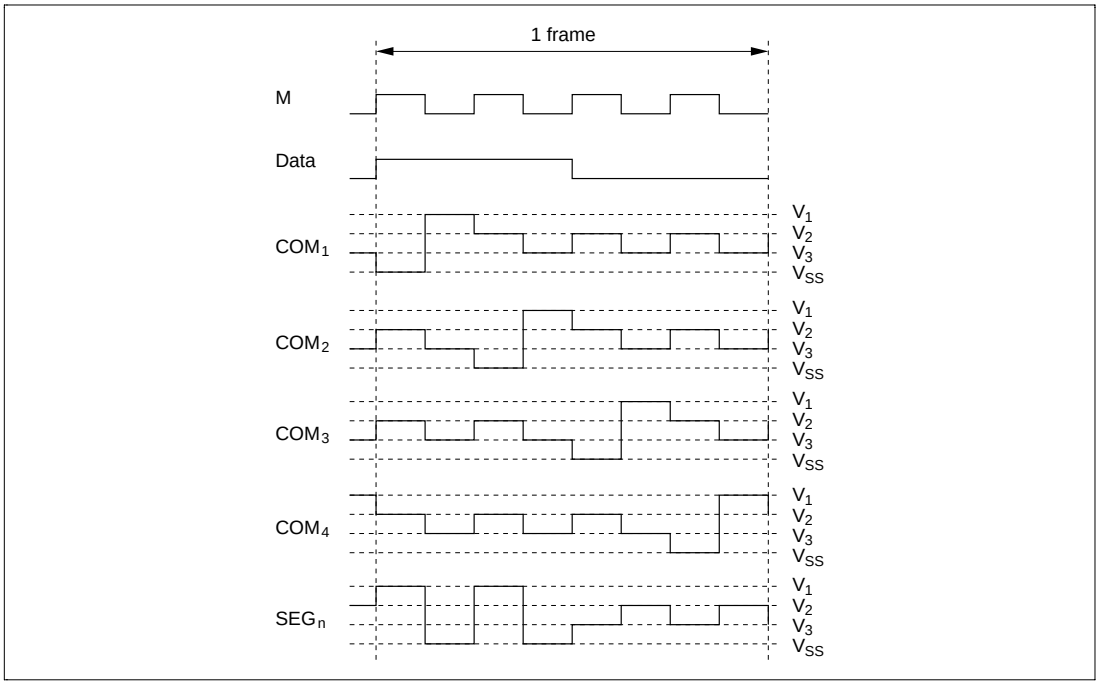



**Figure 13.13 (a) Waveforms at 1/4 Duty**

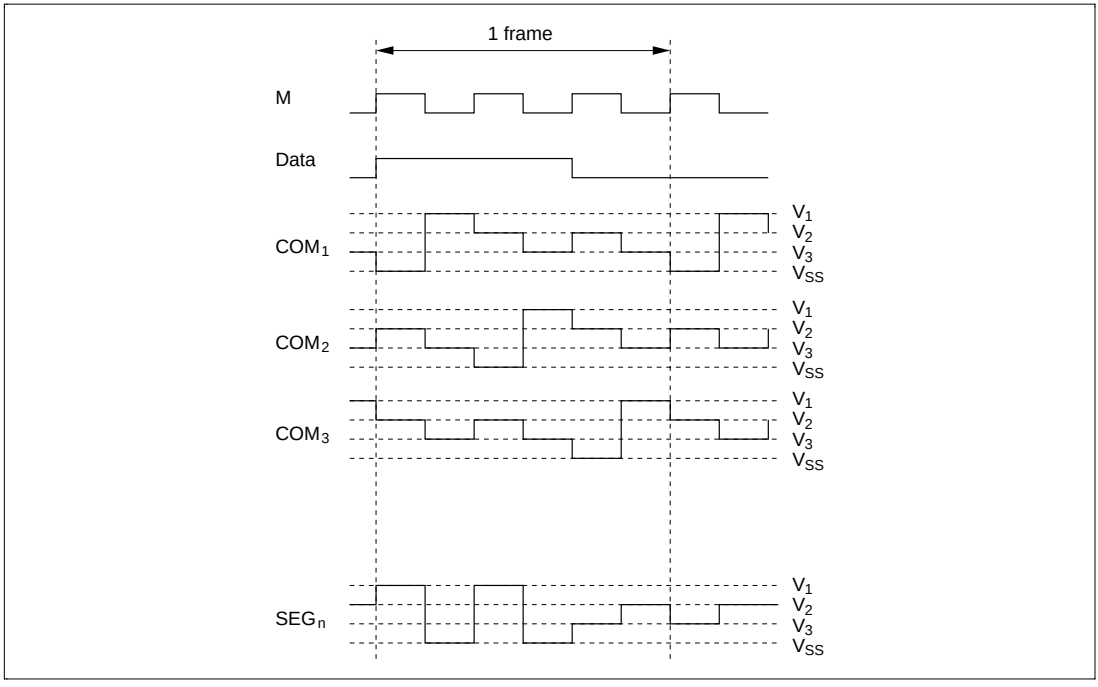



**Figure 13.13 (b) Waveforms at 1/3 Duty**

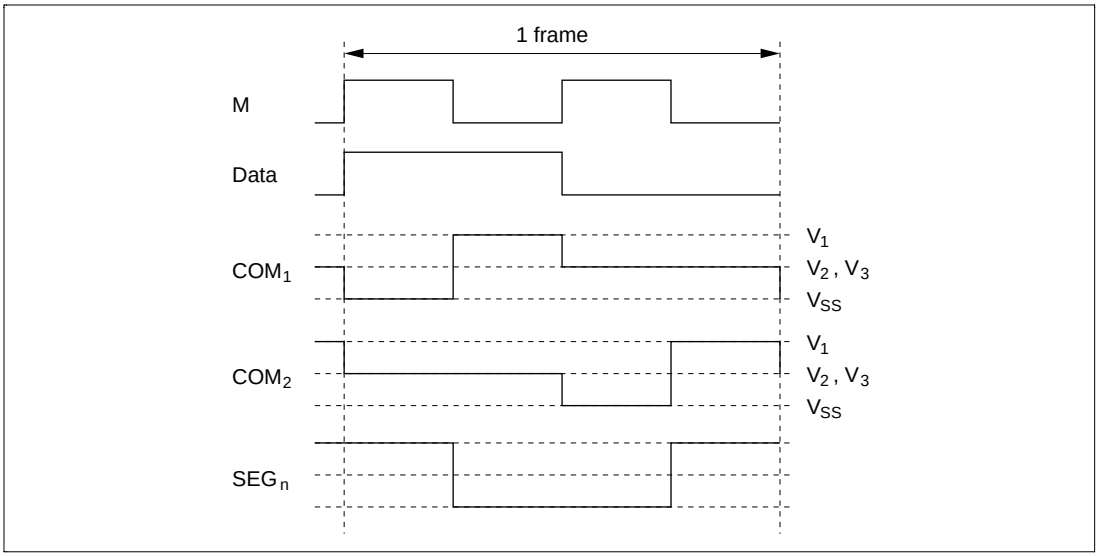



**Figure 13.13 (c) Waveforms at 1/2 Duty**

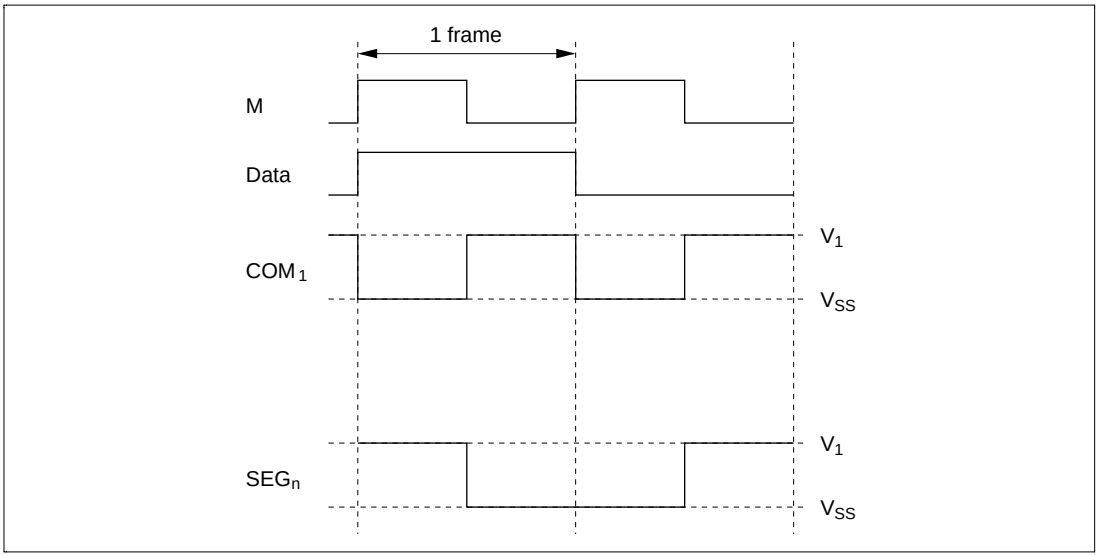



**Figure 13.13 (d) Waveforms at Static Duty**

**Table 13.3 Output Levels**

| Data | | 0 | 0 | 1 | 1 |
|---|---|---|---|---|---|
| **M** | | **0** | **1** | **0** | **1** |
| Static | Common output | $V_1$ | $V_{SS}$ | $V_1$ | $V_{SS}$ |
| | Segment output | $V_1$ | $V_{SS}$ | $V_{SS}$ | $V_1$ |
| 1/2 duty | Common output | $V_2$, $V_3$ | $V_2$, $V_3$ | $V_1$ | $V_{SS}$ |
| | Segment output | $V_1$ | $V_{SS}$ | $V_{SS}$ | $V_1$ |
| 1/3 duty | Common output | $V_3$ | $V_2$ | $V_1$ | $V_{SS}$ |
| | Segment output | $V_2$ | $V_3$ | $V_{SS}$ | $V_1$ |
| 1/4 duty | Common output | $V_3$ | $V_2$ | $V_1$ | $V_{SS}$ |
| | Segment output | $V_2$ | $V_3$ | $V_{SS}$ | $V_1$ |

### 13.3.4 Operation in Power-Down Modes

The LCD controller/driver can be operated in the low-power modes, as shown in table 13.4.

In the subactive, watch, and subsleep modes, the system clock pulse generator stops running, so no clock signal will be supplied and the display will be stopped, unless $\phi_W$ or $\phi_W/2$ was selected when setting bits CKS3 to CKS0 in LCR. Since this may result in a direct current being applied to the LCD panel, be sure to select $\phi_W$ or $\phi_W/2$ as the clock if these modes are used. In active (medium-speed) mode the system clock is changed, making it necessary to adjust the frame frequency setting (in bits CKS3 to CKS0) to avoid a change in frame frequency.

**Table 13.4 LCD Controller/Driver Operation in Power-Down Modes**

| Mode | | Reset | Active | Sleep | Watch | Subactive | Subsleep | Standby |
|---|---|---|---|---|---|---|---|---|
| Clock | $\phi$ | Running | Running | Running | Stopped | Stopped | Stopped | Stopped |
| | $\phi_W$ | Running | Running | Running | Running | Running | Running | Stopped*1 |
| Display | ACT = 0 | Stopped | Stopped | Stopped | Stopped | Stopped | Stopped | Stopped*2 |
| | ACT = 1 | Stopped | On | On | On*3 | On*3 | On*3 | Stopped*2 |

Notes: 1. The subclock pulse generator does not stop, but clock supply is stopped.
2. The LCD driver power supply resistive voltage divider is off regardless of bit PSW.
3. The display will not function unless $\phi_W$ or $\phi_W/2$ is selected as the clock.

### 13.3.5 Boosting the LCD Driver Power Supply

When a large LCD panel is driven, or if segments are expanded externally, the built-in power supply capacity may be insufficient, making it necessary to lower the power supply impedance. One method, shown in figure 13.12, is to connect a bypass capacitor of around 0.1 µF to 0.3 µF to pins $V_1$, $V_2$, and $V_3$. Another approach, shown in figure 13.14 below, is to connect a resistive voltage divider externally.

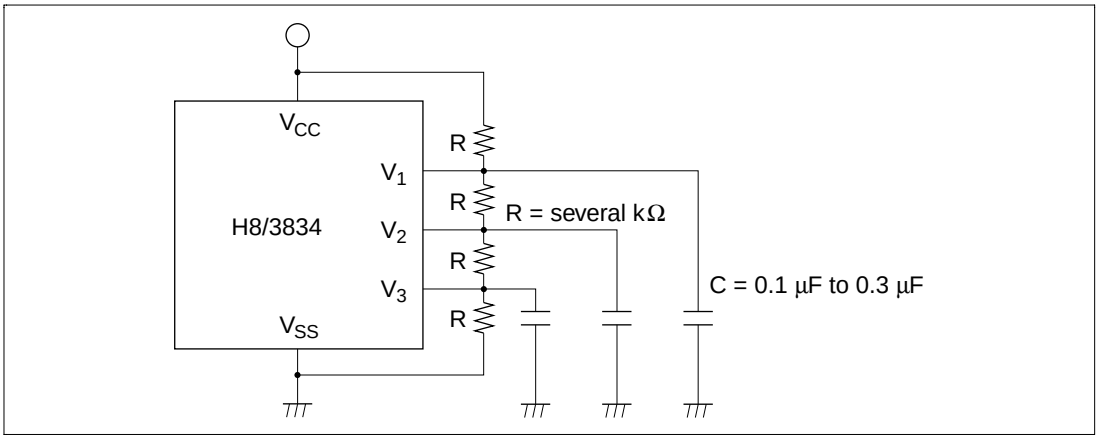



**Figure 13.14 Connecting an External Resistive Voltage Divider**

# Section 14 Electrical Characteristics

## 14.1 H8/3832S, H8/3833S, H8/3834S, H8/3835S, H8/3836S and H8/3837S Absolute Maximum Ratings (Standard Specifications)

Table 14.1 lists the absolute maximum ratings.

**Table 14.1 Absolute Maximum Ratings**

| Item | | Symbol | Value | Unit |
|---|---|---|---|---|
| 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 C | $V_{in}$ | –0.3 to $V_{CC}$ + 0.3 | V |
| | Ports B and C | $AV_{in}$ | –0.3 to $AV_{CC}$ + 0.3 | V |
| Operating temperature | | $T_{opr}$ | –20 to +75 | °C |
| Storage temperature | | $T_{stg}$ | –55 to +125 | °C |

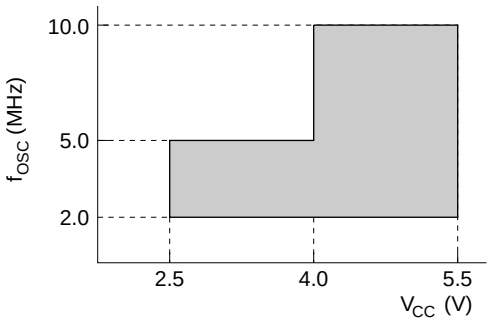
Note: Permanent damage may occur to the chip if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

## 14.2 H8/3832S, H8/3833S and H8/3834S Electrical Characteristics (Standard Specifications)

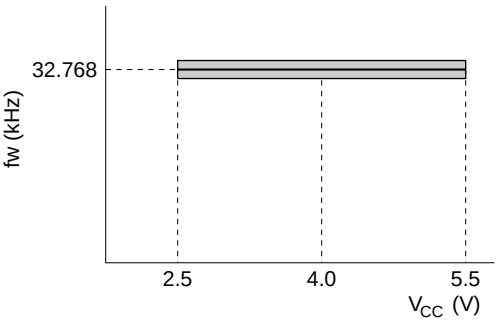
### 14.2.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3832S, H8/3833S and H8/3834S are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range of H8/3832S, H8/3833S and H8/3834S

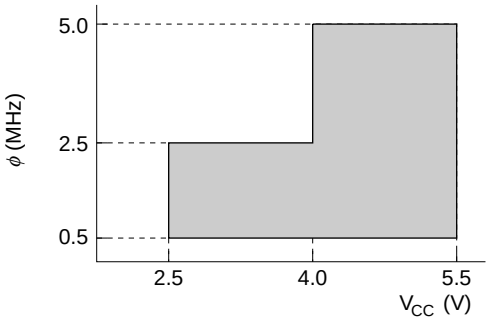



• Active mode (high speed)
• Sleep mode

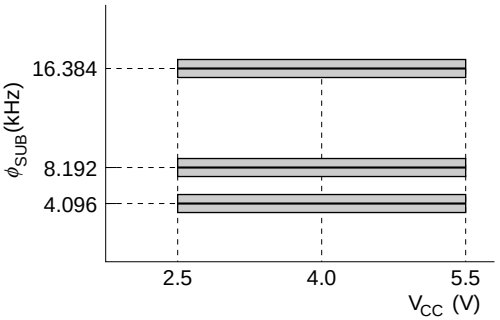



• All operating modes

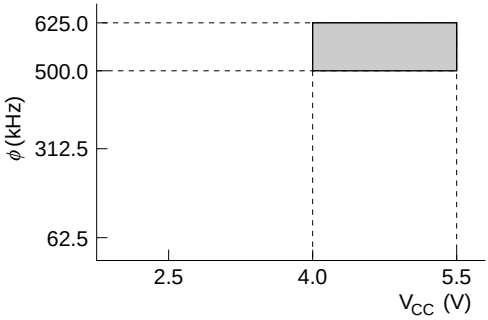
2. Power supply voltage vs. clock frequency range of H8/3832S, H8/3833S and H8/3834S




- Active mode (high speed)
- Sleep mode (except CPU)

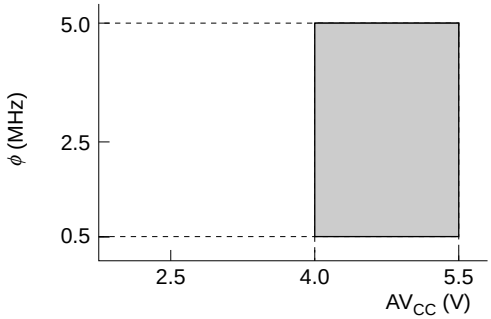



- Subactive mode
- Subsleep mode (except CPU)
- Watch mode (except CPU)

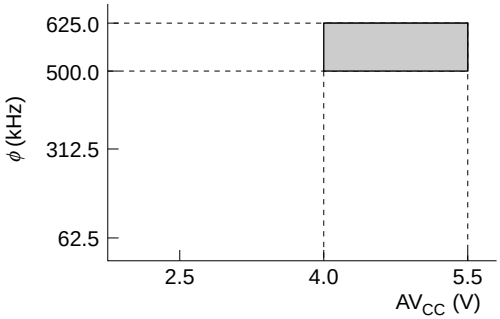



- Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range of H8/3832S, H8/3833S and H8/3834S




- Active (high speed) mode
- Sleep mode




- Active (medium speed) mode

### 14.2.2 DC Characteristics

Table 14.2 lists the DC characteristics of the H8/3832S, H8/3833S and H8/3834S.

**Table 14.2 DC Characteristics of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.9 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF, | –0.3 | — | 0.2 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.1 $V_{CC}$ | | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.2 $V_{CC}$ | | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.3 | | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.2 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.2 $V_{CC}$ | | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | $V_{CC}$ – 0.5 | — | — | | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 0.5 mA | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.1 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | — | — | 1.5 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 10 mA | |
| | | | — | — | 0.6 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |

**Table 14.2 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input leakage current | $\|I_{IL}\|$ | $\overline{RES}$<br>$OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 1.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 1.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $-I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$ | 50.0 | — | 300.0 | µA | $V_{CC}$ = 5 V, $V_{IN}$ = 0 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | — | 35.0 | — | µA | $V_{CC}$ = 2.7 V, $V_{IN}$ = 0 V | Reference value |
| Input capacitance | $C_{IN}$ | All input pins except power supply | — | — | 15.0 | pF | f = 1 MHz, $V_{IN}$ = 0 V, $T_a$ = 25°C | |

**Table 14.2 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 9.0 | 13.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 1.7 | 3.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 4.0 | 7.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 30.0 | 65.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| | | | — | 22.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 1, 2 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 20.0 | 45.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | — | 5.5 | µA | $V_{CC}$ = 2.7 V, LCD not used, 32-kHz crystal oscillator | 1, 2 |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 5.0 | µA | 32-kHz crystal oscillator not used | 1, 2 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 1, 2 |

Notes: 1. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

2. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.2 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 10.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 0.5 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 80.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 20.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 0.2 | | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 10.0 | | |

### 14.2.3 AC Characteristics

Table 14.3 lists the control signal timing, and tables 14.4 and 14.5 list the serial interface timing of the H8/3832S, H8/3833S and H8/3834S.

**Table 14.3 Control Signal Timing of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 2.0 | — | 5.0 | | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | 1 |
| | | | 200.0 | — | 1000.0 | | | Figure 14.1 |
| System clock ($\phi$) cycle time | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_w$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | — | — | 60.0 | | $V_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | 100.0 | | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system clock control register 2 (SYSCR2).

**Table 14.3 Control Signal Timing of H8/3832S, H8/3833S and H8/3834S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | $t_{cyc}$ | | Figure 14.2 |
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.4 Serial Interface (SCI1, SCI2) Timing of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 350.0 | | | |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.5 Serial Interface (SCI3) Timing of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | — | — | 1 | | | |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |

### 14.2.4 A/D Converter Characteristics

Table 14.6 shows the A/D converter characteristics of the H8/3832S, H8/3833S and H8/3834S.

**Table 14.6 A/D Converter Characteristics of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 2.5 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.5 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 5.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | ±3.0 | LSB | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | ±3.5 | LSB | | |
| Conversion time | | | 12.4 | — | 124 | µs | $AV_{CC}$ = 4.5 V to 5.5 V | |
| | | | 24.8 | — | 124 | µs | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.2.5 LCD Characteristics

Table 14.7 lists the LCD characteristics, and table 14.8 lists the AC characteristics for external segment expansion of the H8/3832S, H8/3833S and H8/3834S.

**Table 14.7 LCD Characteristics of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 100.0 | 300.0 | 600.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.8 AC Characteristics for External Segment Expansion of H8/3832S, H8/3833S and H8/3834S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

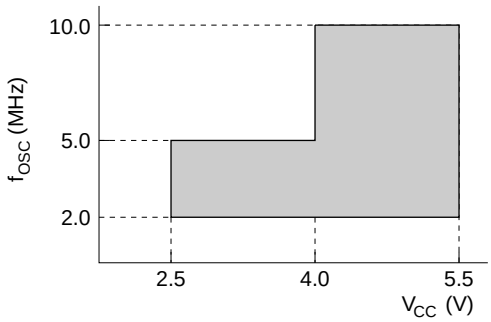
Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.3 H8/3835S, H8/3836S and H8/3837S Electrical Characteristics (Standard Specifications)

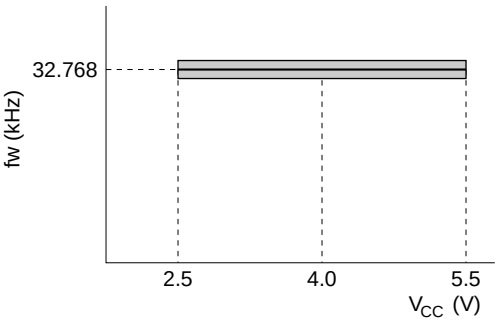
### 14.3.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3835S, H8/3836S and H8/3837S are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range of H8/3835S, H8/3836S and H8/3837S

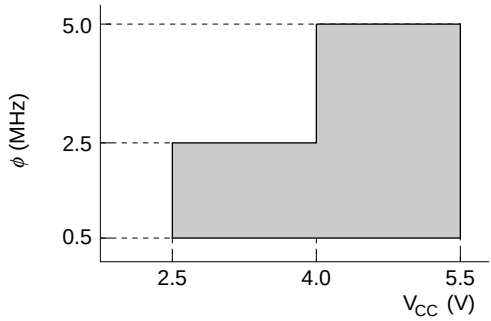



• Active mode (high speeds)
• Sleep mode

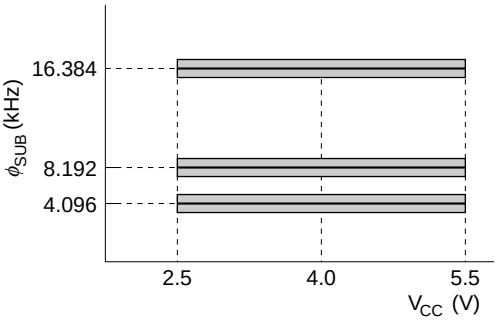



• All operating modes

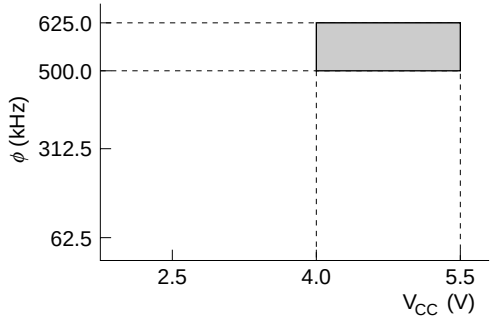
2. Power supply voltage vs. clock frequency range of H8/3835S, H8/3836S and H8/3837S




- Active mode (high speed)
- Sleep mode (except CPU)

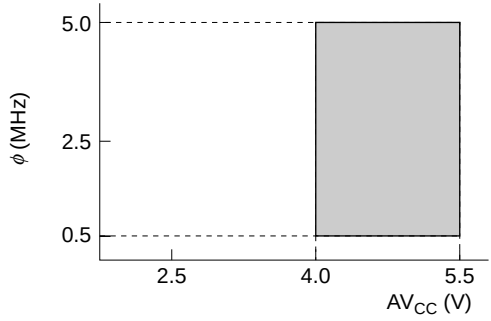



- Subactive mode
- Subsleep mode (except CPU)
- Watch mode (except CPU)

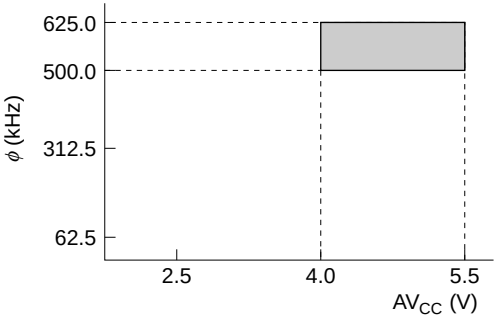



- Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range of H8/3835S, H8/3836S and H8/3837S




- Active (high speed) mode
- Sleep mode




- Active (medium speed) mode

### 14.3.2 DC Characteristics

Table 14.9 lists the DC characteristics of the H8/3835S, H8/3836S and H8/3837S.

**Table 14.9 DC Characteristics of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.9 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF, | –0.3 | — | 0.2 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.1 $V_{CC}$ | | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.2 $V_{CC}$ | | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.3 | | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.9 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.2 $V_{CC}$ | | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | $V_{CC}$ – 0.5 | — | — | | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 0.5 mA | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.1 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | — | — | 1.5 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 10 mA | |
| | | | — | — | 0.6 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.9 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input leakage current | $\|I_{IL}\|$ | $\overline{RES}$<br>$OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 1.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 1.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $–I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$ | 50.0 | — | 300.0 | µA | $V_{CC}$ = 5 V, $V_{IN}$ = 0 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | — | 35.0 | — | µA | $V_{CC}$ = 2.7 V, $V_{IN}$ = 0 V | Reference value |
| Input capacitance | $C_{IN}$ | All input pins except power supply | — | — | 15.0 | pF | f = 1 MHz, $V_{IN}$ = 0 V, $T_a$ = 25°C | |

**Table 14.9 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 9.0 | 13.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 1.7 | 3.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 4.0 | 7.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 30.0 | 65.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| | | | — | 22.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 1, 2 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 20.0 | 45.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | — | 5.5 | µA | $V_{CC}$ = 2.7 V, LCD not used, 32-kHz crystal oscillator | 1, 2 |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 5.0 | µA | 32-kHz crystal oscillator not used | 1, 2 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 1, 2 |

Notes: 1. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

2. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.9 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 10.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 0.5 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 80.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 20.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 0.2 | | |
| Allowable output high current (total) | $\Sigma-I_{OH}$ | All output pins | — | — | 15.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 10.0 | | |

### 14.3.3 AC Characteristics

Table 14.10 lists the control signal timing, and tables 14.11 and 14.12 list the serial interface timing of the H8/3835S, H8/3836S and H8/3837S.

**Table 14.10 Control Signal Timing of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 2.0 | — | 5.0 | | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | 1 |
| | | | 200.0 | — | 1000.0 | | | Figure 14.1 |
| System clock ($\phi$) cycle time | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_w$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | — | — | 60.0 | | $V_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | 100.0 | | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system clock control register 2 (SYSCR2).

**Table 14.10 Control Signal Timing of H8/3835S, H8/3836S and H8/3837S (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | $t_{cyc}$ | | Figure 14.2 |
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.11 Serial Interface (SCI1, SCI2) Timing of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 350.0 | | | |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.12 Serial Interface (SCI3) Timing of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | — | — | 1 | | | |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |

### 14.3.4 A/D Converter Characteristics

Table 14.13 shows the A/D converter characteristics of the H8/3835S, H8/3836S and H8/3837S.

**Table 14.13 A/D Converter Characteristics of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 2.5 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.5 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 5.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | ±3.0 | LSB | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | ±3.5 | LSB | | |
| Conversion time | | | 12.4 | — | 124 | µs | $AV_{CC}$ = 4.5 V to 5.5 V | |
| | | | 24.8 | — | 124 | µs | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.3.5 LCD Characteristics

Table 14.14 lists the LCD characteristics, and table 14.15 lists the AC characteristics for external segment expansion of the H8/3835S, H8/3836S and H8/3837S.

**Table 14.14 LCD Characteristics of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 100.0 | 300.0 | 600.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.15 AC Characteristics for External Segment Expansion of H8/3835S, H8/3836S and H8/3837S**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.4 H8/3832S, H8/3833S, H8/3834S, H8/3835S, H8/3836S and H8/3837S Absolute Maximum Ratings (Wide Temperature Range (I-Spec) Version)

Table 14.6 lists the absolute maximum ratings.

**Table 14.6 Absolute Maximum Ratings**

| Item | | Symbol | Value | Unit |
|---|---|---|---|---|
| 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 C | $V_{in}$ | –0.3 to $V_{CC}$ + 3.0 | V |
| | Ports B and C | $AV_{in}$ | –0.3 to $AV_{CC}$ + 3.0 | V |
| Operating temperature | | $T_{opr}$ | –40 to +85 | °C |
| Storage temperature | | $T_{stg}$ | –55 to +125 | °C |

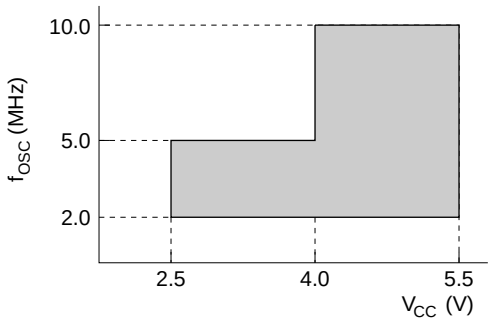
Note: Permanent damage may occur to the chip if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

## 14.5 H8/3832S, H8/3833S and H8/3834S Electrical Characteristics (Wide Temperature Range (I-Spec) Version)

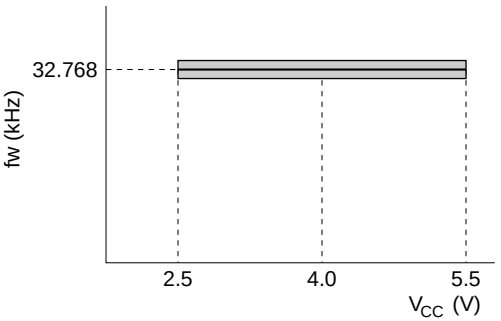
### 14.5.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3832S, H8/3833S, and H8/3834S (wide temperature range (I-spec) version) are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range

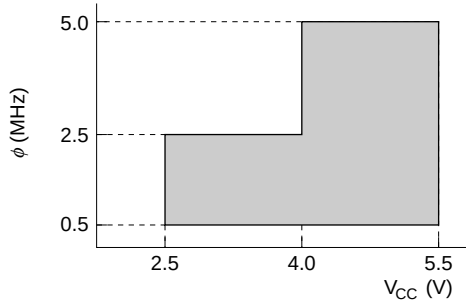



• Active mode (high speeds)
• Sleep mode

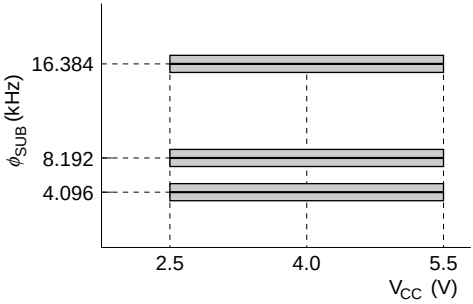



• All operating modes

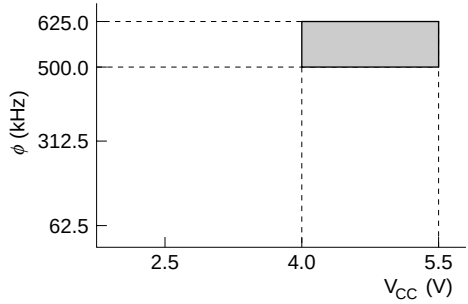
2. Power supply voltage vs. clock frequency range




• Active mode (high speed)
• Sleep mode (except CPU)

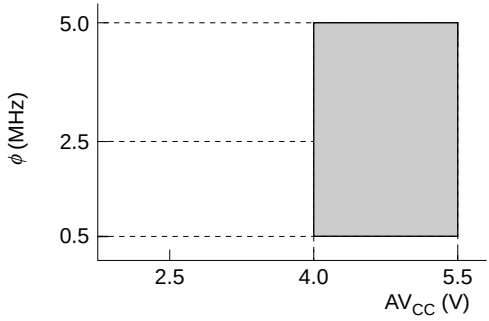



• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)

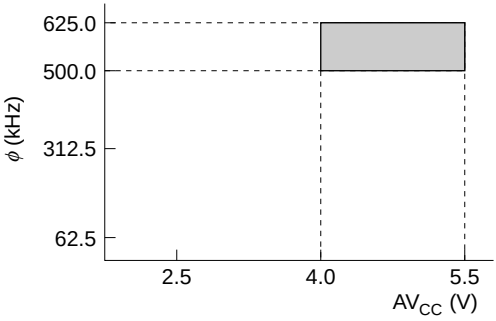



• Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range




• Active (high speed) mode
• Sleep mode




• Active (medium speed) mode

### 14.5.2 DC Characteristics

Table 14.17 lists the DC characteristics of H8/3832S, H8/3833S and H8/3834S (wide temperature range (I-spec) version).

**Table 14.17 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.9 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF, | –0.3 | — | 0.2 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.1 $V_{CC}$ | | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.2 $V_{CC}$ | | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.17 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $OSC_1$ | –0.3 | — | 0.5 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.2 $V_{CC}$ | | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | $V_{CC}$ – 0.5 | — | — | | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 0.5 mA | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.1 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |

**Table 14.17 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Output low voltage | $V_{OL}$ | $P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | — | — | 1.5 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 10 mA | |
| | | | — | — | 0.6 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| Input/ output leakage current | $\|I_{IL}\|$ | $\overline{RES}$<br>$OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 1.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 1.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $–I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$ | 50.0 | — | 300.0 | µA | $V_{CC}$ = 5 V,<br>$V_{IN}$ = 0 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | — | 35.0 | — | µA | $V_{CC}$ = 2.7 V,<br>$V_{IN}$ = 0 V | Reference value |
| Input capacitance | $C_{IN}$ | All input pins except power supply | — | — | 15.0 | pF | f = 1 MHz,<br>$V_{IN}$ = 0 V<br>$T_a$ = 25°C | |

**Table 14.17 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 9.0 | 13.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 1.7 | 3.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 4.0 | 7.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 30.0 | 65.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| | | | — | 22.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 1, 2 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 20.0 | 45.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | — | 5.5 | µA | $V_{CC}$ = 2.7 V, LCD not used, 32-kHz crystal oscillator | 1, 2 |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 5.0 | µA | 32-kHz crystal oscillator not used | 1, 2 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 1, 2 |

Notes: 1. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | Subclock oscillator: Crystal |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

2. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.17 DC Characteristics of H8/3832S, H8/3833S and H8/3834S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 10.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 0.5 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 80.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 20.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 0.2 | | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 10.0 | | |

### 14.5.3 AC Characteristics

Table 14.18 lists the control signal timing, and tables 14.19 and 14.20 list the serial interface timing of the H8/3832S, H8/3833S, and H8/3834S (wide temperature range (I-spec) version).

**Table 14.18 Control Signal Timing of H8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 2.0 | — | 5.0 | | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | 1 |
| | | | 200.0 | — | 1000.0 | | | Figure 14.1 |
| System clock ($\phi$) cycle time | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_w$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | — | — | 60.0 | | $V_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | 100.0 | | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system control register 2 (SYSCR2).

**Table 14.18 Control Signal Timing of H8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | $t_{cyc}$ | | Figure 14.2 |
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.19 Serial Interface (SCI1, SCI2) Timing of H8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 350.0 | | | |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.20 Serial Interface (SCI3) Timing of H8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | — | — | 1 | | | |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |

### 14.5.4 A/D Converter Characteristics

Table 14.21 shows the A/D converter characteristics of H8/3832S, H8/3833S, and H8/3834S (wide temperature range (I-spec) version).

**Table 14.21 A/D Converter Characteristics of H8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 2.5 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.5 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 5.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | ±3.0 | | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | | | ±3.5 | | | |
| Conversion time | | | 12.4 | — | 124 | µs | $AV_{CC}$ = 4.5 V to 5.5 V | |
| | | | 24.8 | — | 124 | | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.5.5 LCD Characteristics

Table 14.22 lists the LCD characteristics, and table 14.23 lists the AC characteristics for external segment expansion of H8/3832S, H8/3833S, and H8/3834S (wide temperature range (I-spec) version).

**Table 14.22 LCD Characteristics of H8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 100.0 | 300.0 | 600.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.23 AC Characteristics for External Segment Expansion of 8/3832S, H8/3833S, and H8/3834S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

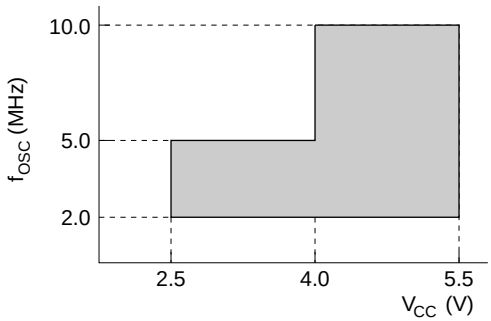
Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.6 H8/3835S, H8/3836S and H8/3837S Electrical Characteristics (Wide Temperature Range (I-Spec) Version)

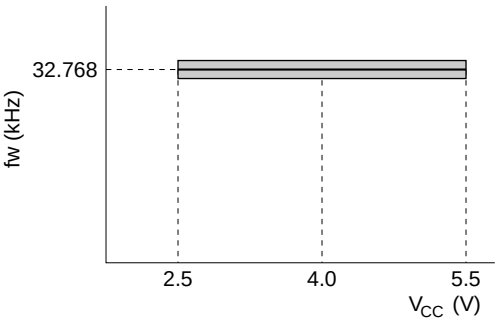
### 14.6.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3835S, H8/3836S, and H8/3837S are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range of H8/3835S, H8/3836S, and H8/3837S

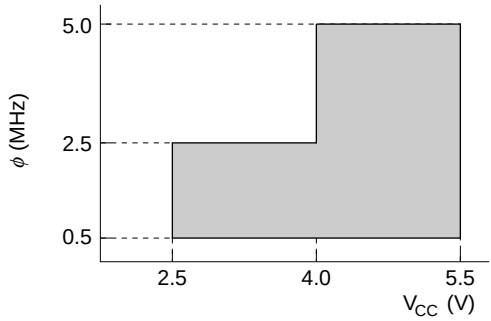



• Active mode (high speeds)
• Sleep mode

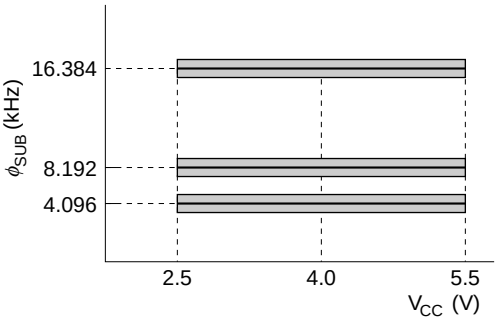



• All operating modes

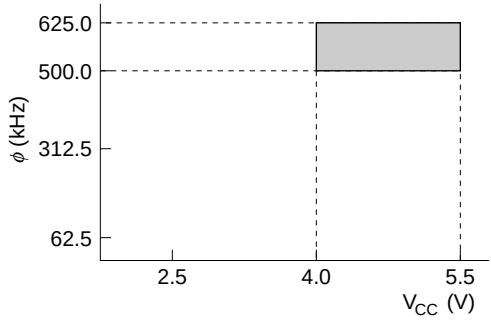
2. Power supply voltage vs. clock frequency range of H8/3835S, H8/3836S, and H8/3837S




• Active mode (high speed)
• Sleep mode (except CPU)

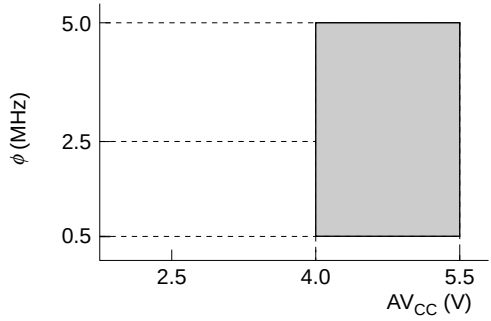



• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)

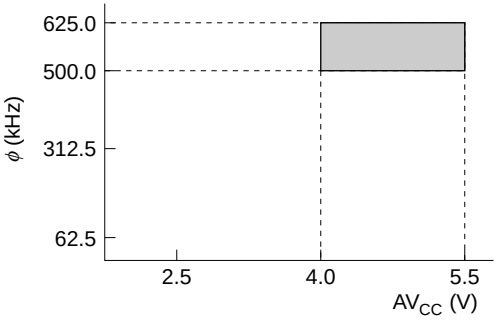



• Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range of H8/3835S, H8/3836S, and H8/3837S




• Active (high speed) mode
• Sleep mode




• Active (medium speed) mode

### 14.6.2 DC Characteristics

Table 14.24 lists the DC characteristics of the H8/3835S, H8/3836S and H8/3837S (wide temperature range (I-spec) version).

**Table 14.24 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF $\overline{CS}$, TMIG, $SCK_1$, $\overline{SCK_2}$, $\overline{SCK_3}$, $\overline{ADTRG}$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.9 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF, $\overline{CS}$, TMIG, $SCK_1$, $\overline{SCK_2}$, $\overline{SCK_3}$, $\overline{ADTRG}$ | –0.3 | — | 0.2 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.1 $V_{CC}$ | | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.24 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.2 $V_{CC}$ | | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.2 $V_{CC}$ | | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$ | $V_{CC}$ – 0.5 | — | — | | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 0.5 mA | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.1 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.24 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Output low voltage | $V_{OL}$ | $P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | — | — | 1.5 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 10 mA | |
| | | | — | — | 0.6 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| Input/output leakage current | $\|I_{IL}\|$ | $\overline{RES}$<br>$OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 1.0 | µA | $V_{IN}$ = 0.5 V to<br>$V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 1.0 | | $V_{IN}$ = 0.5 V to<br>$AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $-I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$ | 50.0 | — | 300.0 | µA | $V_{CC}$ = 5 V,<br>$V_{IN}$ = 0 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | — | 35.0 | — | | $V_{CC}$ = 2.7 V,<br>$V_{IN}$ = 0 V | Reference value |
| Input capacitance | $C_{IN}$ | All input pins except power supply | — | — | 15.0 | pF | f = 1 MHz,<br>$V_{IN}$ = 0 V<br>$T_a$ = 25°C | |

**Table 14.24 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 9.0 | 13.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 1.7 | 3.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 4.0 | 7.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 30.0 | 65.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| | | | — | 22.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 1, 2 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 20.0 | 45.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | — | 5.5 | µA | $V_{CC}$ = 2.7 V, LCD not used, 32-kHz crystal oscillator | 1, 2 |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 5.0 | µA | 32-kHz crystal oscillator not used | 1, 2 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 1, 2 |

Notes: 1. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

2. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.24 DC Characteristics of H8/3835S, H8/3836S and H8/3837S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 10.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 0.5 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 80.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 20.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 0.2 | | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 10.0 | | |

### 14.6.3 AC Characteristics

Table 14.25 lists the control signal timing, and tables 14.26 and 14.27 list the serial interface timing of the H8/3835S, H8/3836S, and H8/3837S (wide temperature range (I-spec) version).

**Table 14.25 Control Signal Timing of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 2.0 | — | 5.0 | | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | 1 Figure 14.1 |
| | | | 200.0 | — | 1000.0 | | | |
| System clock ($\phi$) cycle time | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_W$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$ | | |
| | | | | | | $t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | — | — | 60.0 | | $V_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | 100.0 | | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system control register 2 (SYSCR2).

**Table 14.25 Control Signal Timing of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | tcyc | | Figure 14.2 |
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.26 Serial Interface (SCI1, SCI2) Timing of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 350.0 | | | |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.27 Serial Interface (SCI3) Timing of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | — | — | 1 | | | |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |

### 14.6.4 A/D Converter Characteristics

Table 14.28 shows the A/D converter characteristics of the H8/3835S, H8/3836S, and H8/3837S (wide temperature range (I-spec) version).

**Table 14.28 A/D Converter Characteristics of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 2.5 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.5 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 5.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | — | — | ±3.0 | | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | $V_{CC}$ = 2.7 V to 5.5 V<br>$AV_{CC}$ = 2.7 V to 5.5 V | |
| | | | | | ±3.5 | | | |
| Conversion time | | | 12.4 | — | 124 | µs | $AV_{CC}$ = 4.5 V to 5.5 V | |
| | | | 24.8 | — | 124 | | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.6.5 LCD Characteristics

Table 14.29 lists the LCD characteristics, and table 14.30 lists the AC characteristics for external segment expansion of the H8/3835S, H8/3836S, and H8/3837S (wide temperature range (I-spec) version).

**Table 14.29 LCD Characteristics of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 100.0 | 300.0 | 600.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.30 AC Characteristics for External Segment Expansion of H8/3835S, H8/3836S, and H8/3837S (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.5 V to 5.5 V, $AV_{CC}$ = 2.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.7 H8/3833, H8/3834, H8/3835, H8/3836, and H8/3837 (Standard Specification) Absolute Maximum Ratings

Table 14.31 lists the absolute maximum ratings.

**Table 14.31 Absolute Maximum Ratings**

| Item | | Symbol | Value | Unit |
|---|---|---|---|---|
| Power supply voltage | | $V_{CC}$ | –0.3 to +7.0 | V |
| Analog power supply voltage | | $AV_{CC}$ | –0.3 to +7.0 | V |
| Programming voltage | | $V_{PP}$ | –0.3 to + 13.0 | V |
| Input voltage | Ports other than ports B and C | $V_{in}$ | –0.3 to $V_{CC}$ + 3.0 | V |
| | Ports B and C | $AV_{in}$ | –0.3 to $AV_{CC}$ + 3.0 | V |
| Operating temperature | | $T_{opr}$ | –20 to +75 | °C |
| Storage temperature | | $T_{stg}$ | –55 to +125 | °C |

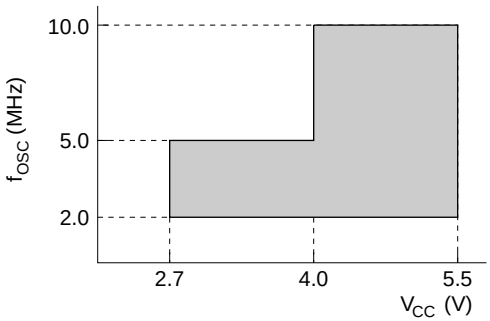
Note: Permanent damage may occur to the chip if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

## 14.8 H8/3833 and H8/3834 Electrical Characteristics (Standard Specifications)

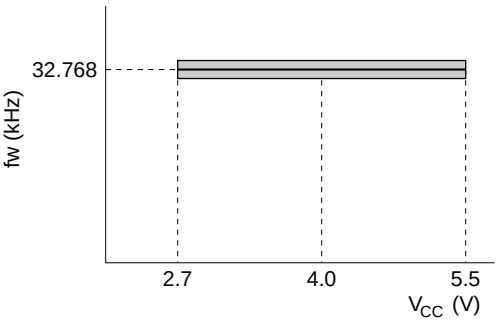
### 14.8.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3833 and H8/3834 are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range of H8/3833 and H8/3834

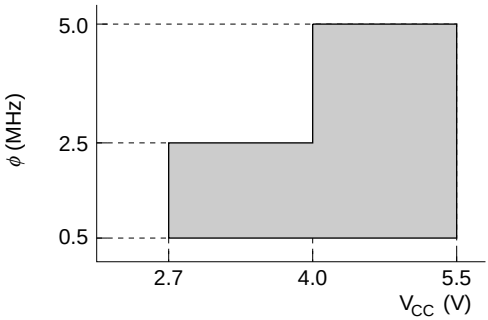



• Active mode (high speed)
• Sleep mode

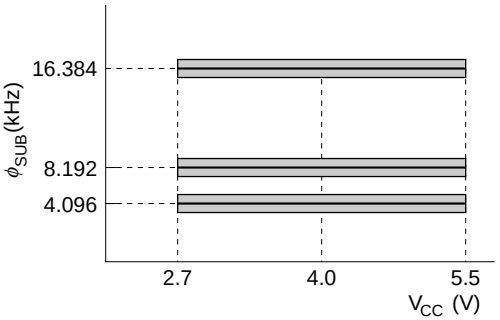



• All operating modes

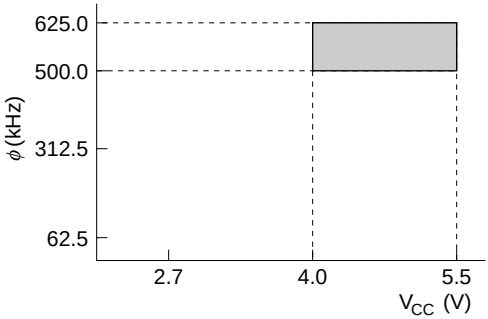
2. Power supply voltage vs. clock frequency range of H8/3833 and H8/3834




• Active mode (high speed)
• Sleep mode (except CPU)

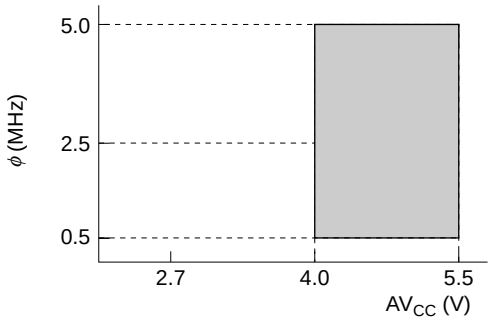



• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)

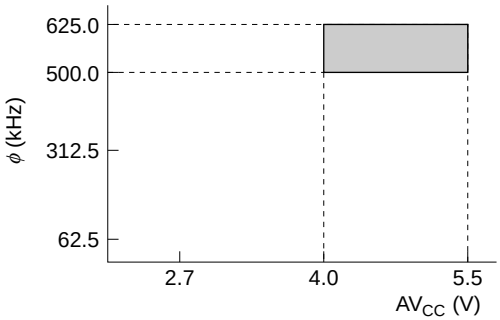



• Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range of H8/3833 and H8/3834




• Active (high speed) mode
• Sleep mode




• Active (medium speed) mode

### 14.8.2 DC Characteristics

Table 14.32 lists the DC characteristics of the H8/3833 and H8/3834.

**Table 14.32 DC Characteristics of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.9 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF, $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.2 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.1 $V_{CC}$ | | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.2 $V_{CC}$ | | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.3 | | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.32 DC Characteristics of H8/3833 and H8/3834 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$ | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.2 $V_{CC}$ | | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$-I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$ | $V_{CC}$ – 0.5 | — | — | | $V_{CC}$ = 4.0 V to 5.5 V<br>$-I_{OH}$ = 0.5 mA | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $-I_{OH}$ = 0.1 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | — | — | 1.5 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 10 mA | |
| | | | — | — | 0.6 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.32 DC Characteristics of H8/3833 and H8/3834 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input/output leakage current | $\|I_{IL}\|$ | $\overline{RES}$, $P4_3$ | — | — | 20.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | 2 |
| | | | — | — | 1.0 | | | 1 |
| | | $OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 1.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 1.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $-I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$ | 50.0 | — | 300.0 | µA | $V_{CC}$ = 5 V, $V_{IN}$ = 0 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | — | 35.0 | — | | $V_{CC}$ = 2.7 V, $V_{IN}$ = 0 V | Reference value |
| Input capacitance | $C_{IN}$ | All input pins except power supply, $\overline{RES}$, $P4_3$ pin | — | — | 15.0 | pF | f = 1 MHz, $V_{IN}$ = 0 V, $T_a$ = 25°C | |
| | | $\overline{RES}$ | — | — | 60.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |
| | | $P4_3$ | — | — | 30.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |

Notes: 1. Applies to HD6433833 and HD6433834.
2. Applies to HD6473834.

**Table 14.32 DC Characteristics of H8/3833 and H8/3834 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 12.0 | 24.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 3, 4 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 2.5 | 5.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 3, 4 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 5.0 | 10.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 3, 4 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 50.0 | 130.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | Reference value 3, 4 |
| | | | — | 40.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 3, 4 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 40.0 | 90.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | Reference value 3, 4 |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | — | 6.0 | µA | $V_{CC}$ = 2.7 V, LCD not used, 32-kHz crystal oscillator | Reference value 3, 4 |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 5.0 | µA | 32-kHz crystal oscillator not used | 3, 4 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 3, 4 |

Notes: 3. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | Subclock oscillator: Crystal |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

4. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.32 DC Characteristics of H8/3833 and H8/3834 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 10.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 0.5 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 80.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 20.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 0.2 | | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 10.0 | | |

### 14.8.3 AC Characteristics

Table 14.33 lists the control signal timing, and tables 14.34 and 14.35 list the serial interface timing of the H8/3833 and H8/3834.

**Table 14.33 Control Signal Timing of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 2.0 | — | 5.0 | | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | 1 |
| | | | 200.0 | — | 1000.0 | | | Figure 14.1 |
| System clock ($\phi$) | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| cycle time | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_W$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | — | — | 60.0 | | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | tcyc | | Figure 14.2 |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system control register 2 (SYSCR2).

**Table 14.33 Control Signal Timing of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input pin high width | $t_{IH}$ | $\overline{IRQ}_0$ to $\overline{IRQ}_4$<br>$\overline{WKP}_0$ to $\overline{WKP}_7$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ}_0$ to $\overline{IRQ}_4$<br>$\overline{WKP}_0$ to $\overline{WKP}_7$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.34 Serial Interface (SCI1, SCI2) Timing of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 350.0 | | | |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.35 Serial Interface (SCI3) Timing of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | — | — | 1 | | | |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |

### 14.8.4 A/D Converter Characteristics

Table 14.36 shows the A/D converter characteristics of the H8/3833 and H8/3834.

**Table 14.36 A/D Converter Characteristics of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 4.0 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.5 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 5.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | | |
| Conversion time | | | 12.4 | — | 124 | µs | $AV_{CC}$ = 4.5 V to 5.5 V | |
| | | | 24.8 | — | 124 | | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.8.5 LCD Characteristics

Table 14.37 lists the LCD characteristics, and table 14.38 lists the AC characteristics for external segment expansion of the H8/3833 and H8/3834.

**Table 14.37 LCD Characteristics of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 50.0 | 300.0 | 900.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.38 AC Characteristics for External Segment Expansion of H8/3833 and H8/3834**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

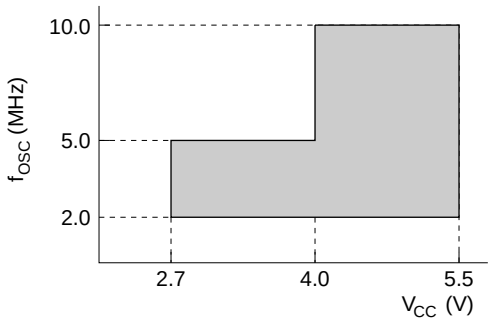
Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.9 H8/3835 and H8/3836 and H8/3837 (Standard Specifications) Electrical Characteristics

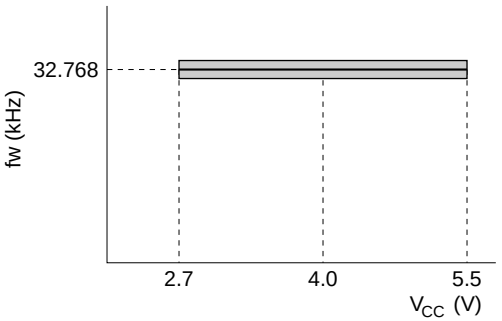
### 14.9.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3835, H8/3836 and H8/3837 are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range of H8/3835, H8/3836 and H8/3837

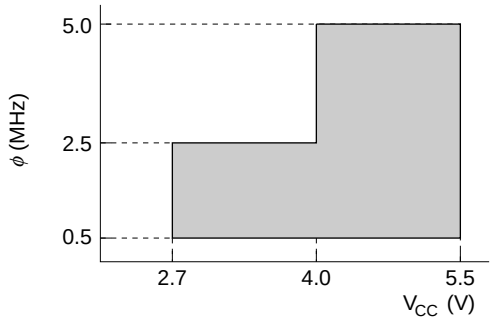



• Active mode (high speed)
• Sleep mode

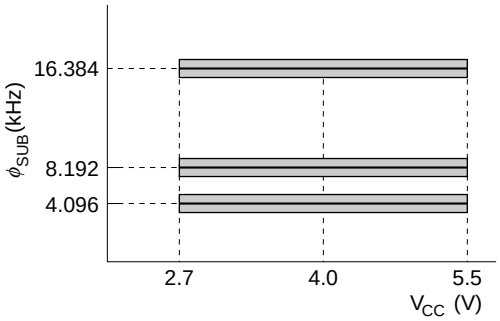



• All operating modes

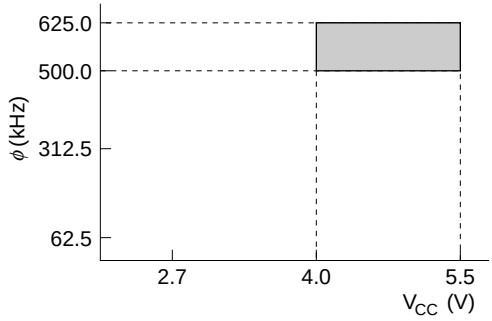
2. Power supply voltage vs. clock frequency range of H8/3835, H8/3836 and H8/3837




- Active mode (high speed)
- Sleep mode (except CPU)

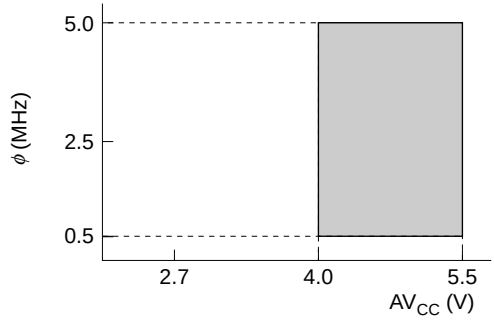



- Subactive mode
- Subsleep mode (except CPU)
- Watch mode (except CPU)

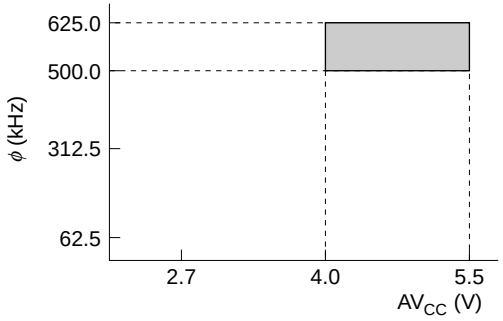



- Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range of H8/3835, H8/3836 and H8/3837




- Active (high speed) mode
- Sleep mode




- Active (medium speed) mode

### 14.9.2 DC Characteristics

Table 14.39 lists the DC characteristics of the H8/3835, H8/3836 and H8/3837.

**Table 14.39 DC Characteristics of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.9 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | $V_{CC}$ – 0.3 | — | $V_{CC}$ + 0.3 | | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 0.8 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, TMIB, TMIC, TMIF, | –0.3 | — | 0.2 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.1 $V_{CC}$ | | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.2 $V_{CC}$ | | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | –0.3 | — | 0.3 | | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.39 DC Characteristics of H8/3835, H8/3836 and H8/3837 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | –0.3 | — | 0.3 $V_{CC}$ | V | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.2 $V_{CC}$ | | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | $V_{CC}$ – 0.5 | — | — | | $V_{CC}$ = 4.0 V to 5.5 V<br>$–I_{OH}$ = 0.5 mA | |
| | | $P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.1 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | — | — | 1.5 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 10 mA | |
| | | | — | — | 0.6 | | $V_{CC}$ = 4.0 V to 5.5 V<br>$I_{OL}$ = 1.6 mA | |
| | | | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.39 DC Characteristics of H8/3835, H8/3836 and H8/3837 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input/output leakage current | $\|I_{IL}\|$ | $\overline{RES}$, $P4_3$ | — | — | 20.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | 2 |
| | | | — | — | 1.0 | | | 1 |
| | | $OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 1.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 1.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $–I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$ | 50.0 | — | 330.0 | µA | $V_{CC}$ = 5 V, $V_{IN}$ = 0 V | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | — | 35.0 | — | µA | $V_{CC}$ = 2.7 V, $V_{IN}$ = 0 V | Reference value |
| Input capacitance | $C_{IN}$ | All input pins except power supply, $\overline{RES}$, $P4_3$ pin | — | — | 15.0 | pF | f = 1 MHz, $V_{IN}$ = 0 V, $T_a$ = 25°C | |
| | | $\overline{RES}$ | — | — | 60.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |
| | | $P4_3$ | — | — | 30.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |

Notes: 1. Applies to HD6433835, HD6433836 and HD6433837.
2. Applies to HD6473837.

**Table 14.39 DC Characteristics of H8/3835, H8/3836 and H8/3837 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 13.5 | 24.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 3, 4 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 2.5 | 5.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 3, 4 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 5.0 | 10.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 3, 4 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 50.0 | 130.0 | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | Reference value 3, 4 |
| | | | — | 40.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 3, 4 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 40.0 | — | µA | $V_{CC}$ = 2.7 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | Reference value 3, 4 |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | — | 6.0 | µA | $V_{CC}$ = 2.7 V, LCD not used, 32-kHz crystal oscillator | Reference value 3, 4 |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 5.0 | µA | 32-kHz crystal oscillator not used | 3, 4 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 3, 4 |

Notes: 3. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | Subclock oscillator: Crystal |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

4. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.39 DC Characteristics of H8/3835, H8/3836 and H8/3837 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 10.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 0.5 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | Ports 2 and 3 | — | — | 80.0 | | $V_{CC}$ = 4.0 V to 5.5 V |
| | | All output pins | — | — | 20.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 0.2 | | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | $V_{CC}$ = 4.0 V to 5.5 V |
| | | | — | — | 10.0 | | |

### 14.9.3 AC Characteristics

Table 14.40 lists the control signal timing, and tables 14.41 and 14.42 list the serial interface timing of the H8/3835, H8/3836 and H8/3837.

**Table 14.40 Control Signal Timing of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 2.0 | — | 5.0 | | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | 1 |
| | | | 200.0 | — | 1000.0 | | | Figure 14.1 |
| System clock ($\phi$) cycle time | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_W$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | — | — | 60.0 | | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | 80.0 | — | — | | | |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.1 |
| | | | — | — | 20.0 | | | |
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | $t_{cyc}$ | | Figure 14.2 |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system control register 2 (SYSCR2).

**Table 14.40 Control Signal Timing of H8/3835, H8/3836 and H8/3837 (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.41 Serial Interface (SCI1, SCI2) Timing of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 80.0 | | | |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | — | — | 350.0 | | | |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.5 |
| | | | 400.0 | — | — | | | |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.42 Serial Interface (SCI3) Timing of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | — | — | 1 | | | |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | $V_{CC}$ = 4.0 V to 5.5 V | Figure 14.8 |
| | | | 400.0 | — | — | | | |

### 14.9.4 A/D Converter Characteristics

Table 14.43 shows the A/D converter characteristics of the H8/3835, H8/3836 and H8/3837.

**Table 14.43 A/D Converter Characteristics of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{SS}$ = $V_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 4.0 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | $AV_{SS}$ – 0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.5 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 5.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | | |
| Conversion time | | | 12.4 | — | 124 | µs | $V_{CC}$ = 4.0 V to 5.5 V | |
| | | | 24.8 | — | 124 | | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.9.5 LCD Characteristics

Table 14.44 lists the LCD characteristics, and table 14.45 lists the AC characteristics for external segment expansion of the H8/3835, H8/3836 and H8/3837.

**Table 14.44 LCD Characteristics of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 50.0 | 300.0 | 900.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 2.7 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.45 AC Characteristics for External Segment Expansion of H8/3835, H8/3836 and H8/3837**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –20°C to +75°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.10 H8/3833, H8/3834, H8/3835, H8/3836, and H8/3837 Absolute Maximum Ratings (Wide Temperature Range (I-Spec) Version)

Table 14.46 lists the absolute maximum ratings.

**Table 14.46 Absolute Maximum Ratings**

| Item | | Symbol | Value | Unit |
|---|---|---|---|---|
| Power supply voltage | | $V_{CC}$ | –0.3 to +7.0 | V |
| Analog power supply voltage | | $AV_{CC}$ | –0.3 to +7.0 | V |
| Programming voltage | | $V_{PP}$ | –0.3 to + 13.0 | V |
| Input voltage | Ports other than ports B and C | $V_{in}$ | –0.3 to $V_{CC}$ + 3.0 | V |
| | Ports B and C | $AV_{in}$ | –0.3 to $AV_{CC}$ + 3.0 | V |
| Operating temperature | | $T_{opr}$ | –20 to +75 | °C |
| Storage temperature | | $T_{stg}$ | –55 to +125 | °C |

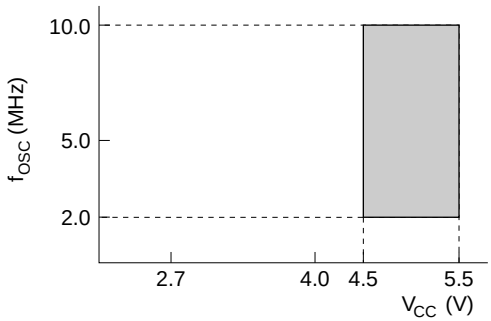
Note: Permanent damage may occur to the chip if maximum ratings are exceeded. Normal operation should be under the conditions specified in Electrical Characteristics. Exceeding these values can result in incorrect operation and reduced reliability.

## 14.11 H8/3833 and H8/3834 Electrical Characteristics (Wide Temperature Range (I-Spec) Version)

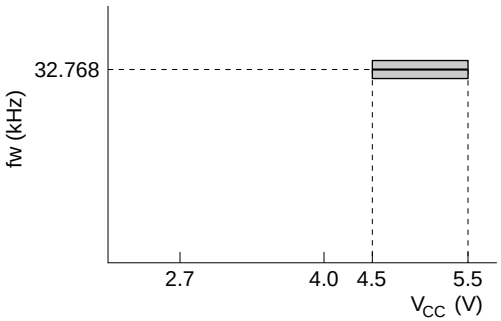
### 14.11.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3833 and H8/3834 (wide temperature range (I-spec) version) are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range

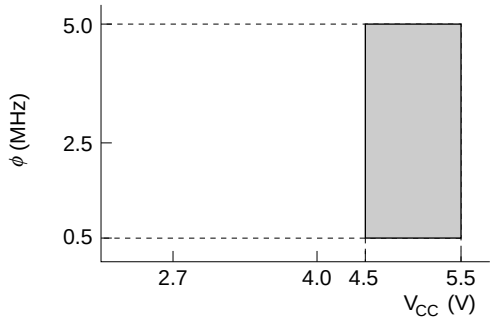



• Active mode (high speed)
• Sleep mode

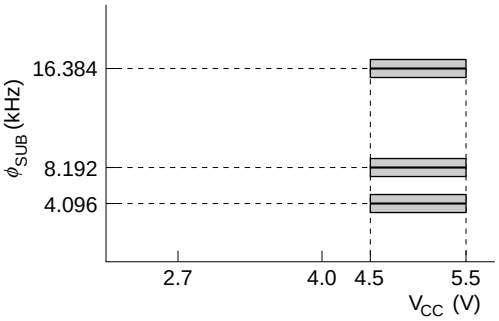



• All operating modes

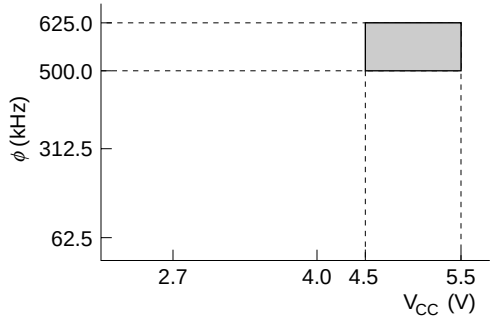
2. Power supply voltage vs. clock frequency range




• Active mode (high speed)
• Sleep mode (except CPU)

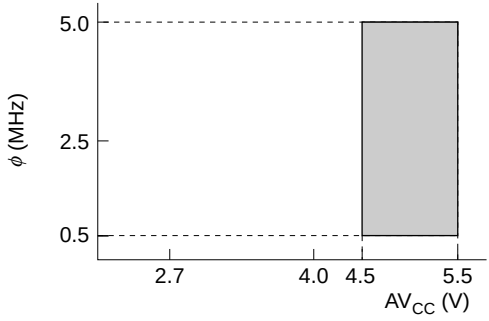



• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)

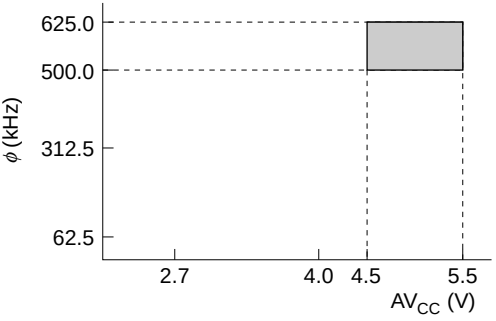



• Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range




• Active (high speed) mode
• Sleep mode




• Active (medium speed) mode

### 14.11.2 DC Characteristics

Table 14.47 lists the DC characteristics of the H8/3833 and H8/3834 (wide temperature range (I-spec) version).

**Table 14.47 DC Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF, $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.2 $V_{CC}$ | V | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.47 DC Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.3 $V_{CC}$ | V | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.5 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $I_{OL}$ = 1.6 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P2_0$ to $P2_7$ | — | — | 1.5 | | $I_{OL}$ = 10 mA | |
| | | $P3_0$ to $P3_7$ | — | — | 0.6 | | $I_{OL}$ = 1.6 mA | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.47 DC Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input/output leakage current | $\|I_{IL}\|$ | $\overline{RES}$, $P4_3$ | — | — | 24.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | 2 |
| | | | — | — | 2.0 | | | 1 |
| | | $OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 2.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 2.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $-I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | 20.0 | — | 330.0 | µA | $V_{CC}$ = 5 V,<br>$V_{IN}$ = 0 V | |
| Input capacitance | $C_{IN}$ | All input pins except power supply, $\overline{RES}$, $P4_3$ pin | — | — | 15.0 | pF | f = 1 MHz,<br>$V_{IN}$ = 0 V<br>$T_a$ = 25°C | |
| | | $\overline{RES}$ | — | — | 60.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |
| | | $P4_3$ | — | — | 30.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |

Notes: 1. Applies to HD6433833 and HD6433834 (wide temperature range version).
2. Applies to HD6473834 (wide temperature range version).

**Table 14.47 DC Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 12.0 | 30.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 2.5 | 6.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 5.0 | 12.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 100.0 | — | µA | $V_{CC}$ = 5 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 Reference value |
| | | | — | 70.0 | — | µA | $V_{CC}$ = 5 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 1, 2 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 60.0 | — | µA | $V_{CC}$ = 5 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 Reference value |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | 6.0 | — | µA | $V_{CC}$ = 5 V, LCD not used, 32-kHz crystal oscillator | 1, 2 Reference value |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 10.0 | µA | 32-kHz crystal oscillator not used | 1, 2 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 1, 2 |

Notes: 1. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal Subclock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

2. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.47 DC Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | |
| | | Ports 2 and 3 | — | — | 10.0 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | |
| | | Ports 2 and 3 | — | — | 80.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | |

### 14.11.3 AC Characteristics

Table 14.48 lists the control signal timing, and tables 14.49 and 14.50 list the serial interface timing of the H8/3833 and H8/3834 (wide temperature range (I-spec) version).

**Table 14.48 Control Signal Timing of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to + 85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | | 1<br>Figure 14.1 |
| System clock ($\phi$) | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| cycle time | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_W$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | | Figure 14.1 |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | | Figure 14.1 |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | | Figure 14.1 |
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | | Figure 14.1 |
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | tcyc | | Figure 14.2 |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system control register 2 (SYSCR2).

**Table 14.48 Control Signal Timing of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.49 Serial Interface (SCI1, SCI2) Timing of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | | Figure 14.5 |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | | Figure 14.5 |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | | Figure 14.5 |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | | Figure 14.5 |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | | Figure 14.5 |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.50 Serial Interface (SCI3) Timing of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | | Figure 14.8 |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | | Figure 14.8 |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | | Figure 14.8 |

### 14.11.4 A/D Converter Characteristics

Table 14.51 shows the A/D converter characteristics of the H8/3833S and H8/3834S (wide temperature range (I-spec) version).

**Table 14.51 A/D Converter Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS} = V_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 4.5 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.7 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 7.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | | |
| Conversion time | | | 12.4 | — | 124 | µs | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.11.5 LCD Characteristics

Table 14.52 lists the LCD characteristics, and table 14.53 lists the AC characteristics for external segment expansion of the H8/3833 and H8/3834 (wide temperature range (I-spec) version).

**Table 14.52 LCD Characteristics of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 40.0 | 300.0 | 1000.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 4.5 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.53 AC Characteristics for External Segment Expansion of H8/3833 and H8/3834 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

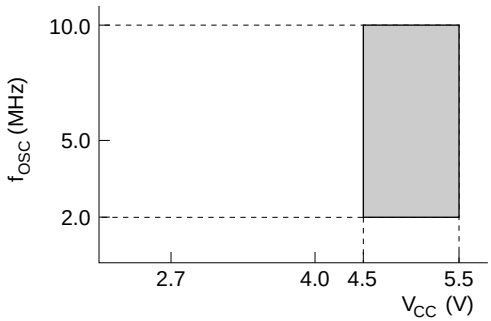
Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.12 H8/3835, H8/3836, and H8/3837 Electrical Characteristics (Wide Temperature Range (I-Spec) Version)

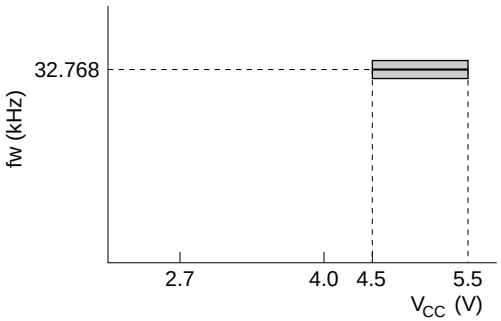
### 14.12.1 Power Supply Voltage and Operating Range

The power supply voltage and operating range of the H8/3835, H8/3836, and H8/3837 (wide temperature range (I-spec) version) are indicated by the shaded region in the figures below.

1. Power supply voltage vs. oscillator frequency range

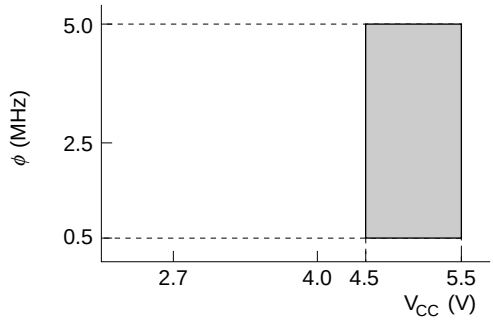



• Active mode (high speeds)
• Sleep mode

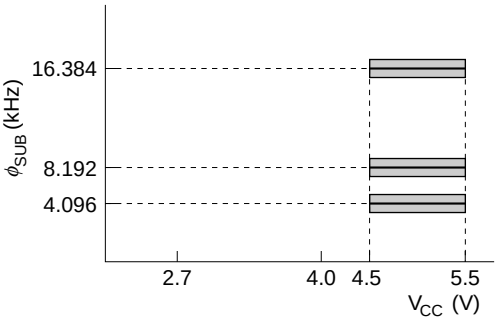



• All operating modes

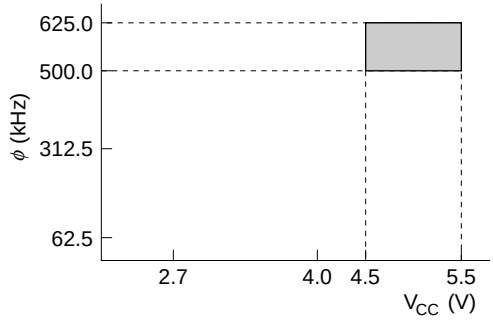
2. Power supply voltage vs. clock frequency range




• Active mode (high speed)
• Sleep mode (except CPU)

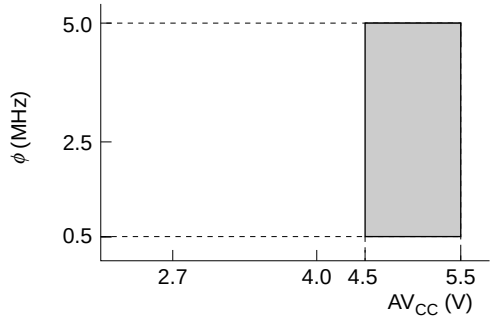



• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)

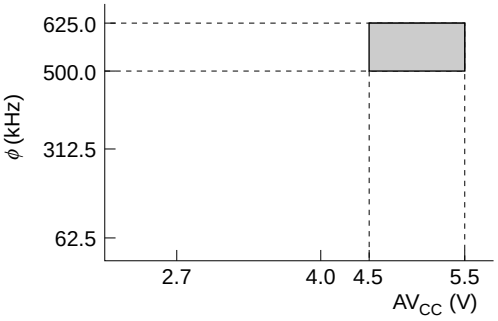



• Active mode (medium speed)

3. Analog power supply voltage vs. A/D converter operating range




• Active (high speed) mode
• Sleep mode




• Active (medium speed) mode

### 14.12.2 DC Characteristics

Table 14.54 lists the DC characteristics of the H8/3835, H8/3836, and H8/3837 (wide temperature range (I-spec) version).

**Table 14.54 DC Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input high voltage | $V_{IH}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | 0.8 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | | |
| | | UD, $SI_1$, $SI_2$, RXD | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | | |
| | | $OSC_1$ | $V_{CC}$ – 0.5 | — | $V_{CC}$ + 0.3 | V | | |
| | | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | 0.7 $V_{CC}$ | — | $V_{CC}$ + 0.3 | V | | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | 0.7 $V_{CC}$ | — | $AV_{CC}$ + 0.3 | V | | |
| Input low voltage | $V_{IL}$ | $\overline{RES}$, MD0, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, TMIB, TMIC, TMIF, $\overline{CS}$, TMIG, $SCK_1$, $SCK_2$, $SCK_3$, $\overline{ADTRG}$ | –0.3 | — | 0.2 $V_{CC}$ | V | | |
| | | UD, $SI_1$, $SI_2$, RXD | –0.3 | — | 0.3 $V_{CC}$ | V | | |
| | | $OSC_1$ | –0.3 | — | 0.5 | V | | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.54 DC Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input low voltage | $V_{IL}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_3$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$<br>$PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | –0.3 | — | 0.3 $V_{CC}$ | V | | |
| Output high voltage | $V_{OH}$ | $P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$ | $V_{CC}$ – 1.0 | — | — | V | $–I_{OH}$ = 1.0 mA | |
| | | $P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | $V_{CC}$ – 0.5 | — | — | | $–I_{OH}$ = 0.5 mA | |
| Output low voltage | $V_{OL}$ | $P1_0$ to $P1_7$<br>$P4_0$ to $P4_2$ | — | — | 0.6 | V | $I_{OL}$ = 1.6 mA | |
| | | $P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 0.5 | | $I_{OL}$ = 0.4 mA | |
| | | $P2_0$ to $P2_7$ | — | — | 1.5 | | $I_{OL}$ = 10 mA | |
| | | $P3_0$ to $P3_7$ | — | — | 0.6 | | $I_{OL}$ = 1.6 mA | |

Note: Connect pin TEST to $V_{SS}$.

**Table 14.54 DC Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Input/output leakage current | $\|I_{IL}\|$ | $\overline{RES}$, $P4_3$ | — | — | 24.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | 2 |
| | | | — | — | 2.0 | | | 1 |
| | | $OSC_1$, MD0<br>$P1_0$ to $P1_7$<br>$P2_0$ to $P2_7$<br>$P3_0$ to $P3_7$<br>$P4_0$ to $P4_2$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$<br>$P7_0$ to $P7_7$<br>$P8_0$ to $P8_7$<br>$P9_0$ to $P9_7$<br>$PA_0$ to $PA_3$ | — | — | 2.0 | µA | $V_{IN}$ = 0.5 V to $V_{CC}$ – 0.5 V | |
| | | $PB_0$ to $PB_7$<br>$PC_0$ to $PC_3$ | — | — | 2.0 | | $V_{IN}$ = 0.5 V to $AV_{CC}$ – 0.5 V | |
| Pull-up MOS current | $-I_P$ | $P1_0$ to $P1_7$<br>$P3_0$ to $P3_7$<br>$P5_0$ to $P5_7$<br>$P6_0$ to $P6_7$ | 20.0 | — | 330.0 | µA | $V_{CC}$ = 5 V,<br>$V_{IN}$ = 0 V | |
| Input capacitance | $C_{IN}$ | All input pins except power supply, $\overline{RES}$, $P4_3$ pin | — | — | 15.0 | pF | f = 1 MHz,<br>$V_{IN}$ = 0 V<br>$T_a$ = 25°C | |
| | | $\overline{RES}$ | — | — | 60.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |
| | | $P4_3$ | — | — | 30.0 | | | 2 |
| | | | — | — | 15.0 | | | 1 |

Notes: 1. Applies to HD6433835, HD6433836, and HD6433837 (wide temperature range version).

2. Applies to HD6473837 (wide temperature range version).

**Table 14.54 DC Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Active mode current dissipation | $I_{OPE1}$ | $V_{CC}$ | — | 13.5 | 30.0 | mA | Active mode (high speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| | $I_{OPE2}$ | $V_{CC}$ | — | 2.5 | 6.0 | mA | Active mode (medium speed), $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Sleep mode current dissipation | $I_{SLEEP}$ | $V_{CC}$ | — | 5.0 | 12.0 | mA | $V_{CC}$ = 5 V, $f_{osc}$ = 10 MHz | 1, 2 |
| Subactive mode current dissipation | $I_{SUB}$ | $V_{CC}$ | — | 100.0 | — | µA | $V_{CC}$ = 5 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 Reference value |
| | | | — | 70.0 | — | µA | $V_{CC}$ = 5 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/8) | Reference value 1, 2 |
| Subsleep mode current dissipation | $I_{SUBSP}$ | $V_{CC}$ | — | 60.0 | — | µA | $V_{CC}$ = 5 V, LCD on, 32-kHz crystal oscillator ($\phi_{SUB}$ = $\phi$w/2) | 1, 2 Reference value |
| Watch mode current dissipation | $I_{WATCH}$ | $V_{CC}$ | — | 6.0 | — | µA | $V_{CC}$ = 5 V, LCD not used, 32-kHz crystal oscillator | 1, 2 Reference value |
| Standby mode current dissipation | $I_{STBY}$ | $V_{CC}$ | — | — | 10.0 | µA | 32-kHz crystal oscillator not used | 1, 2 |
| RAM data retaining voltage | $V_{RAM}$ | $V_{CC}$ | 2.0 | — | — | V | | 1, 2 |

Notes: 1. Pin states during current measurement

| Mode | Internal State | Other Pins | LCD Power Supply | Oscillator Pins |
|---|---|---|---|---|
| Active mode (high and medium speed) | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |
| Sleep mode | Only timer operates | $V_{CC}$ | Open | |
| Subactive mode | Operates | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Crystal |
| Subsleep mode | Only timer operates, CPU stops | $V_{CC}$ | Open | |
| Watch mode | Only time-base clock operates, CPU stops | $V_{CC}$ | Open | |
| Standby mode | CPU and timers all stop | $V_{CC}$ | Open | System clock oscillator: Crystal<br>Subclock oscillator: Pin $X_1$ = $V_{CC}$ |

2. Excludes current in pull-up MOS transistors and output buffers.

**Table 14.54 DC Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 2.7 V to 5.5 V, $AV_{CC}$ = 2.7 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise indicated.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition |
|---|---|---|---|---|---|---|---|
| Allowable output low current (per pin) | $I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 2.0 | mA | |
| | | Ports 2 and 3 | — | — | 10.0 | | |
| Allowable output low current (total) | $\Sigma I_{OL}$ | Output pins except in ports 2 and 3 | — | — | 40.0 | mA | |
| | | Ports 2 and 3 | — | — | 80.0 | | |
| Allowable output high current (per pin) | $-I_{OH}$ | All output pins | — | — | 2.0 | mA | |
| Allowable output high current (total) | $\Sigma -I_{OH}$ | All output pins | — | — | 15.0 | mA | |

### 14.12.3 AC Characteristics

Table 14.55 lists the control signal timing, and tables 14.56 and 14.57 list the serial interface timing of the H8/3835, H8/3836, and H8/3837 (wide temperature range (I-spec) version).

**Table 14.55 Control Signal Timing of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| System clock oscillation frequency | $f_{OSC}$ | $OSC_1$, $OSC_2$ | 2.0 | — | 10.0 | MHz | | |
| OSC clock ($\phi_{OSC}$) cycle time | $t_{OSC}$ | $OSC_1$, $OSC_2$ | 100.0 | — | 1000.0 | ns | | 1<br>Figure 14.1 |
| System clock ($\phi$) | $t_{cyc}$ | | 2 | — | 16 | $t_{OSC}$ | | 1 |
| cycle time | | | — | — | 2000.0 | ns | | |
| Subclock oscillation frequency | $f_W$ | $X_1$, $X_2$ | — | 32.768 | — | kHz | | |
| Watch clock ($\phi_W$) cycle time | $t_W$ | $X_1$, $X_2$ | — | 30.5 | — | µs | | |
| Subclock ($\phi_{SUB}$) cycle time | $t_{subcyc}$ | | 2 | — | 8 | $t_W$ | | 2 |
| Instruction cycle time | | | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | |
| Oscillation stabilization time (crystal oscillator) | $t_{rc}$ | $OSC_1$, $OSC_2$ | — | — | 40.0 | ms | | |
| Oscillation stabilization time | $t_{rc}$ | $X_1$, $X_2$ | — | — | 2.0 | s | | |
| External clock high width | $t_{CPH}$ | $OSC_1$ | 40.0 | — | — | ns | | Figure 14.1 |
| External clock low width | $t_{CPL}$ | $OSC_1$ | 40.0 | — | — | ns | | Figure 14.1 |
| External clock rise time | $t_{CPr}$ | | — | — | 15.0 | ns | | Figure 14.1 |
| External clock fall time | $t_{CPf}$ | | — | — | 15.0 | ns | | Figure 14.1 |
| Pin $\overline{RES}$ low width | $t_{REL}$ | $\overline{RES}$ | 10 | — | — | tcyc | | Figure 14.2 |

Notes: 1. A frequency between 1 MHz to 10 MHz is required when an external clock is input.
2. Selected with SA1 and SA0 of system control register 2 (SYSCR2).

**Table 14.55 Control Signal Timing of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version) (cont)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input pin high width | $t_{IH}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Input pin low width | $t_{IL}$ | $\overline{IRQ_0}$ to $\overline{IRQ_4}$<br>$\overline{WKP_0}$ to $\overline{WKP_7}$<br>$\overline{ADTRG}$<br>TMIB, TMIC<br>TMIF, TMIG | 2 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.3 |
| Pin UD minimum modulation width | $t_{UDH}$<br>$t_{UDL}$ | UD | 4 | — | — | $t_{cyc}$<br>$t_{subcyc}$ | | Figure 14.4 |

**Table 14.56 Serial Interface (SCI1, SCI2) Timing of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input serial clock cycle time | $t_{scyc}$ | $SCK_1$, $SCK_2$ | 2 | — | — | $t_{cyc}$ | | Figure 14.5 |
| Input serial clock high width | $t_{SCKH}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock low width | $t_{SCKL}$ | $SCK_1$, $SCK_2$ | 0.4 | — | — | $t_{scyc}$ | | Figure 14.5 |
| Input serial clock rise time | $t_{SCKr}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | | Figure 14.5 |
| Input serial clock fall time | $t_{SCKf}$ | $SCK_1$, $SCK_2$ | — | — | 60.0 | ns | | Figure 14.5 |
| Serial output data delay time | $t_{SOD}$ | $SO_1$, $SO_2$ | — | — | 200.0 | ns | | Figure 14.5 |
| Serial input data setup time | $t_{SIS}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | | Figure 14.5 |
| Serial input data hold time | $t_{SIH}$ | $SI_1$, $SI_2$ | 200.0 | — | — | ns | | Figure 14.5 |
| $\overline{CS}$ setup time | $t_{CSS}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |
| $\overline{CS}$ hold time | $t_{CSH}$ | $\overline{CS}$ | 2 | — | — | $t_{cyc}$ | | Figure 14.6 |

**Table 14.57 Serial Interface (SCI3) Timing of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | | Symbol | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Input clock cycle | Asynchronous | $t_{scyc}$ | 4 | — | — | $t_{cyc}$ | | Figure 14.7 |
| | Synchronous | | 6 | — | — | | | |
| Input clock pulse width | | $t_{SCKW}$ | 0.4 | — | 0.6 | $t_{scyc}$ | | Figure 14.7 |
| Transmit data delay time (synchronous mode) | | $t_{TXD}$ | — | — | 1 | $t_{cyc}$ | | Figure 14.8 |
| Receive data setup time (synchronous mode) | | $t_{RXS}$ | 200.0 | — | — | ns | | Figure 14.8 |
| Receive data hold time (synchronous mode) | | $t_{RXH}$ | 200.0 | — | — | ns | | Figure 14.8 |

### 14.12.4 A/D Converter Characteristics

Table 14.58 shows the A/D converter characteristics of the H8/3835, H8/3836, and H8/3837 (wide temperature range (I-spec) version).

**Table 14.58 A/D Converter Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{SS}$ = $V_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Analog power supply voltage | $AV_{CC}$ | $AV_{CC}$ | 4.5 | — | 5.5 | V | | 1 |
| Analog input voltage | $AV_{IN}$ | $AN_0$ to $AN_{11}$ | $AV_{SS}$ –0.3 | — | $AV_{CC}$ + 0.3 | V | | |
| Analog power supply current | $AI_{OPE}$ | $AV_{CC}$ | — | — | 1.7 | mA | $AV_{CC}$ = 5.0 V | |
| | $AI_{STOP1}$ | $AV_{CC}$ | — | 150.0 | — | µA | | 2 Reference value |
| | $AI_{STOP2}$ | $AV_{CC}$ | — | — | 7.0 | µA | | 3 |
| Analog input capacitance | $C_{AIN}$ | $AN_0$ to $AN_{11}$ | — | — | 30.0 | pF | | |
| Allowable signal source impedance | $R_{AIN}$ | | — | — | 10.0 | kΩ | | |
| Resolution (data length) | | | — | — | 8 | bit | | |
| Non-linearity error | | | — | — | ±2.0 | LSB | | |
| Quantization error | | | — | — | ±0.5 | LSB | | |
| Absolute accuracy | | | — | — | ±2.5 | LSB | | |
| Conversion time | | | 12.4 | — | 124 | µs | | |

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, watch, subactive, and subsleep modes while the A/D converter is idle.

### 14.12.5 LCD Characteristics

Table 14.59 lists the LCD characteristics, and table 14.60 lists the AC characteristics for external segment expansion of the H8/3835, H8/3836, and H8/3837 (wide temperature range (I-spec) version).

**Table 14.59 LCD Characteristics of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Note |
|---|---|---|---|---|---|---|---|---|
| Segment driver voltage drop | $V_{DS}$ | $SEG_1$ to $SEG_{40}$ | — | — | 0.6 | V | $I_D$ = 2 µA | 1 |
| Common driver voltage drop | $V_{DC}$ | $COM_1$ to $COM_4$ | — | — | 0.3 | V | $I_D$ = 2 µA | 1 |
| LCD power supply voltage divider resistance | $R_{LCD}$ | | 40.0 | 300.0 | 100.0 | kΩ | Between $V_1$ and $V_{SS}$ | |
| LCD power supply voltage | $V_{LCD}$ | $V_1$ | 4.5 | — | $V_{CC}$ | V | | 2 |

Notes: 1. These are the voltage drops between the voltage supply pins $V_1$, $V_2$, $V_3$, and $V_{SS}$, and the segment pins or common pins.

2. When $V_{LCD}$ is supplied from an external source, the following relation must hold: $V_{CC} \geq V_1 \geq V_2 \geq V_3 \geq V_{SS}$

**Table 14.60 AC Characteristics for External Segment Expansion of H8/3835, H8/3836, and H8/3837 (Wide Temperature Range (I-Spec) Version)**

$V_{CC}$ = 4.5 V to 5.5 V, $AV_{CC}$ = 4.5 V to 5.5 V, $V_{SS}$ = $AV_{SS}$ = 0.0 V, $T_a$ = –40°C to +85°C, including subactive mode, unless otherwise specified.

| Item | Symbol | Applicable Pins | Min | Typ | Max | Unit | Test Condition | Reference Figure |
|---|---|---|---|---|---|---|---|---|
| Clock high width | $t_{CWH}$ | $CL_1$, $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock low width | $t_{CWL}$ | $CL_2$ | 800.0 | — | — | ns | * | Figure 14.9 |
| Clock setup time | $t_{CSU}$ | $CL_1$, $CL_2$ | 500.0 | — | — | ns | * | Figure 14.9 |
| Data setup time | $t_{SU}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| Data hold time | $t_{DH}$ | DO | 300.0 | — | — | ns | * | Figure 14.9 |
| M delay time | $t_{DM}$ | M | –1000 | — | 1000.0 | ns | | Figure 14.9 |
| Clock rise and fall times | $t_{CT}$ | $CL_1$, $CL_2$ | — | — | 100.0 | ns | | Figure 14.9 |

Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

## 14.13 Operation Timing

Figures 14.1 to 14.9 show timing diagrams.

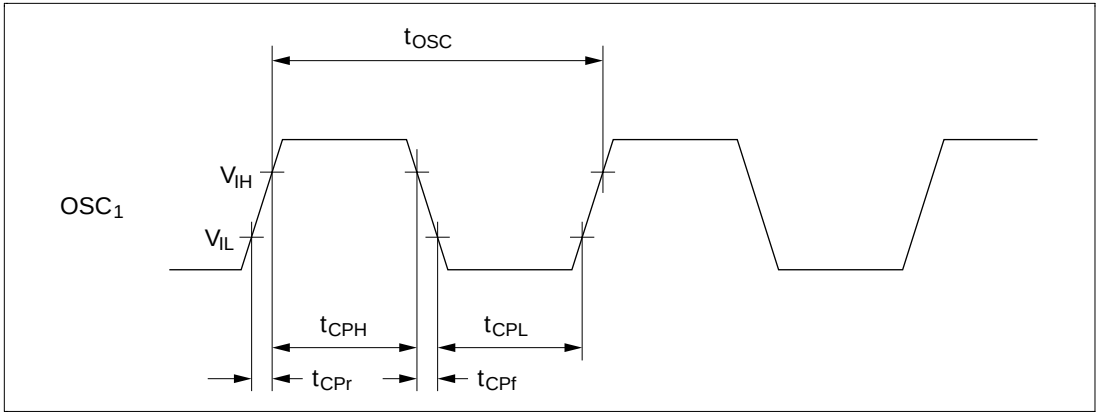



**Figure 14.1 System Clock Input Timing**

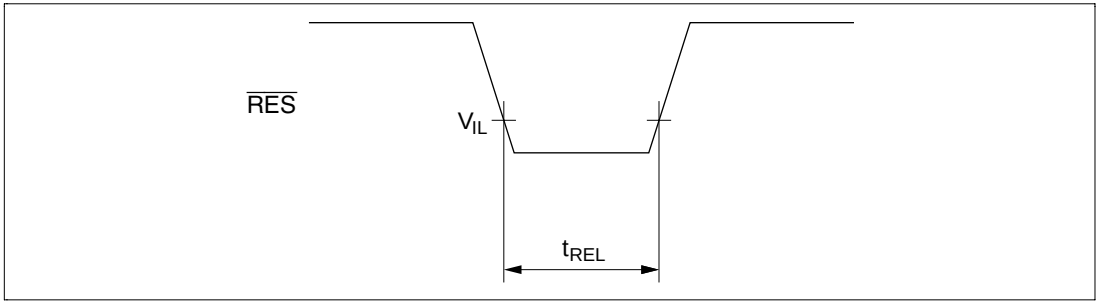



**Figure 14.2 $\overline{RES}$ Low Width Timing**

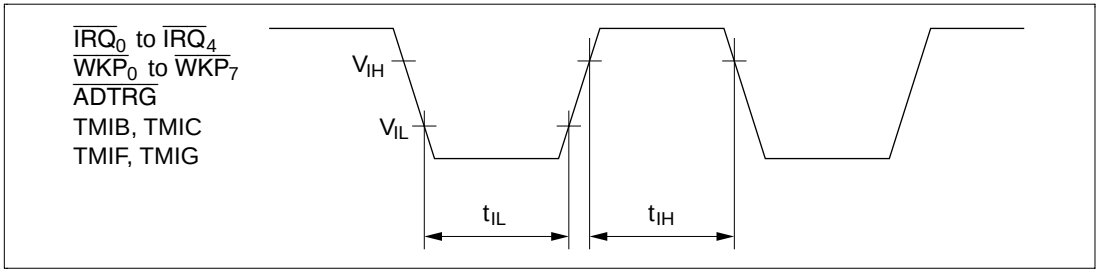



**Figure 14.3 Input Timing**

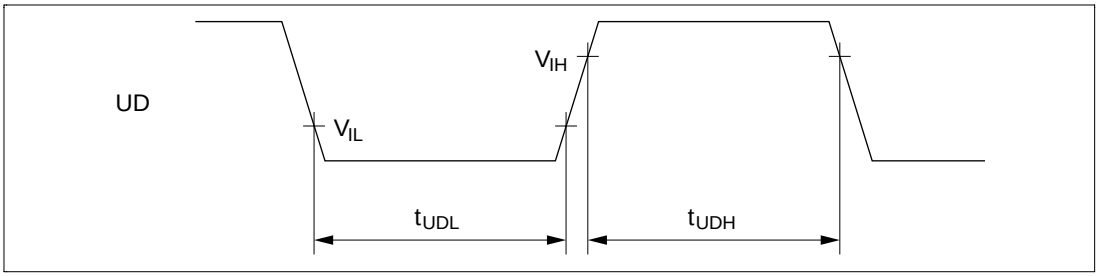
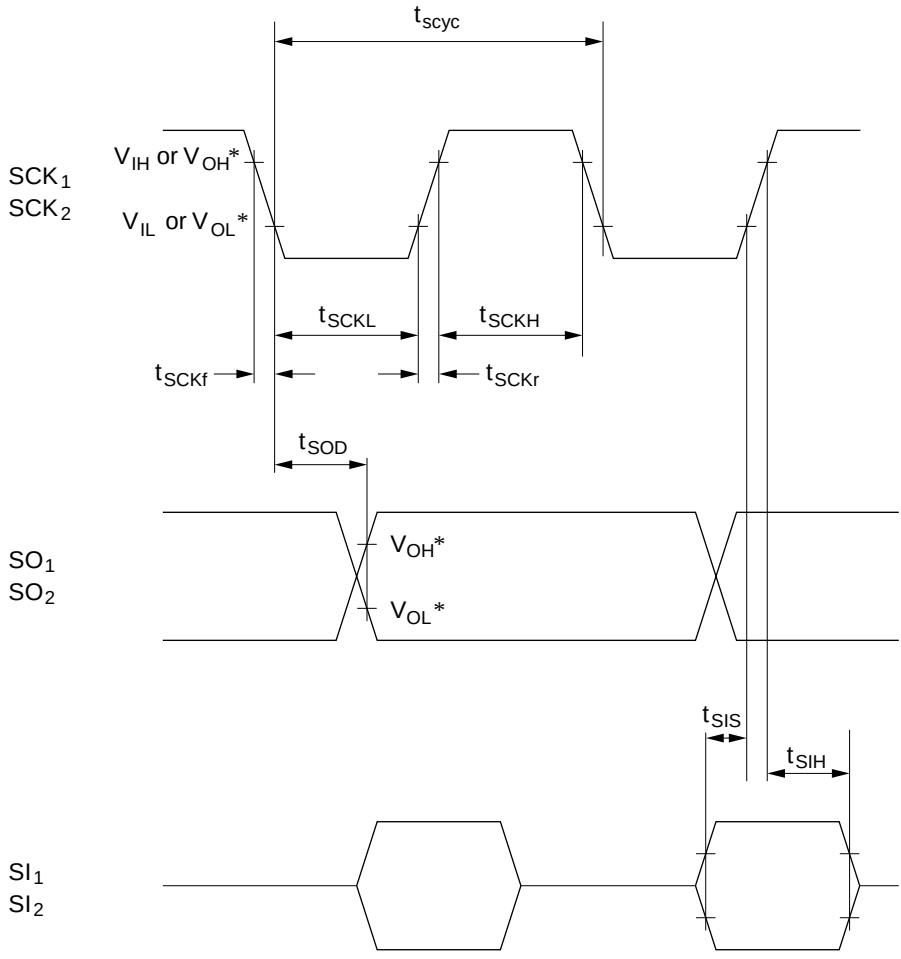



**Figure 14.4 UD Pin Minimum Transition Width Timing**




Notes: * Output timing reference levels
Output high: $V_{OH}$ = 1.8 V ($V_{CC}$= 2.5 V to 5.5 V)/2.0 V ($V_{CC}$= 2.7 V to 5.5 V)
Output low: $V_{OL}$ = 0.8 V
Load conditions are shown in figure 14-10.

**Figure 14.5 Serial Interface 1 and 2 Input/Output Timing**

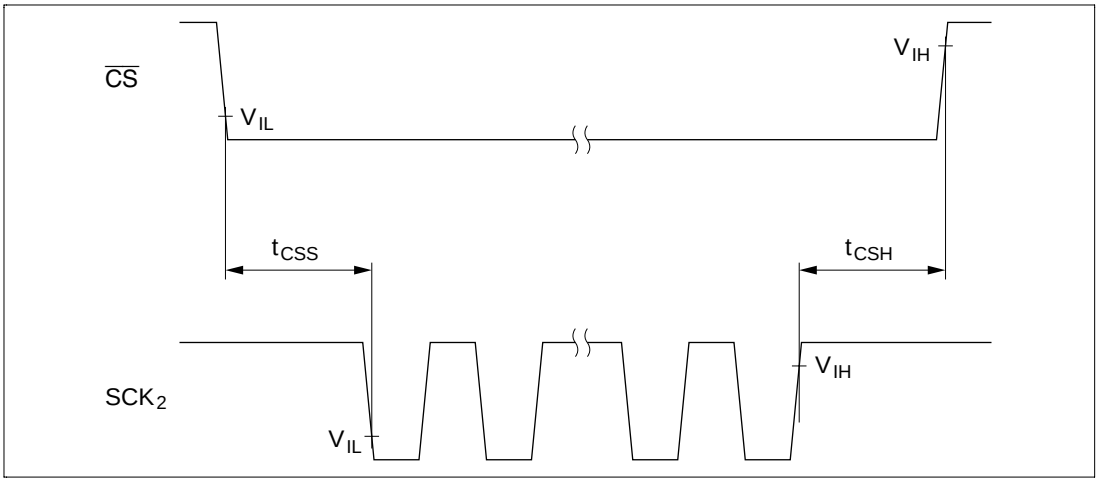



**Figure 14.6 Serial Interface 2 Chip Select Timing**

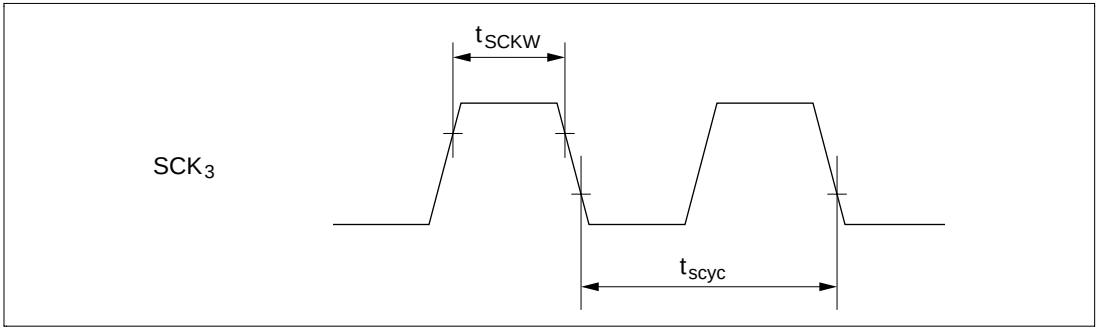
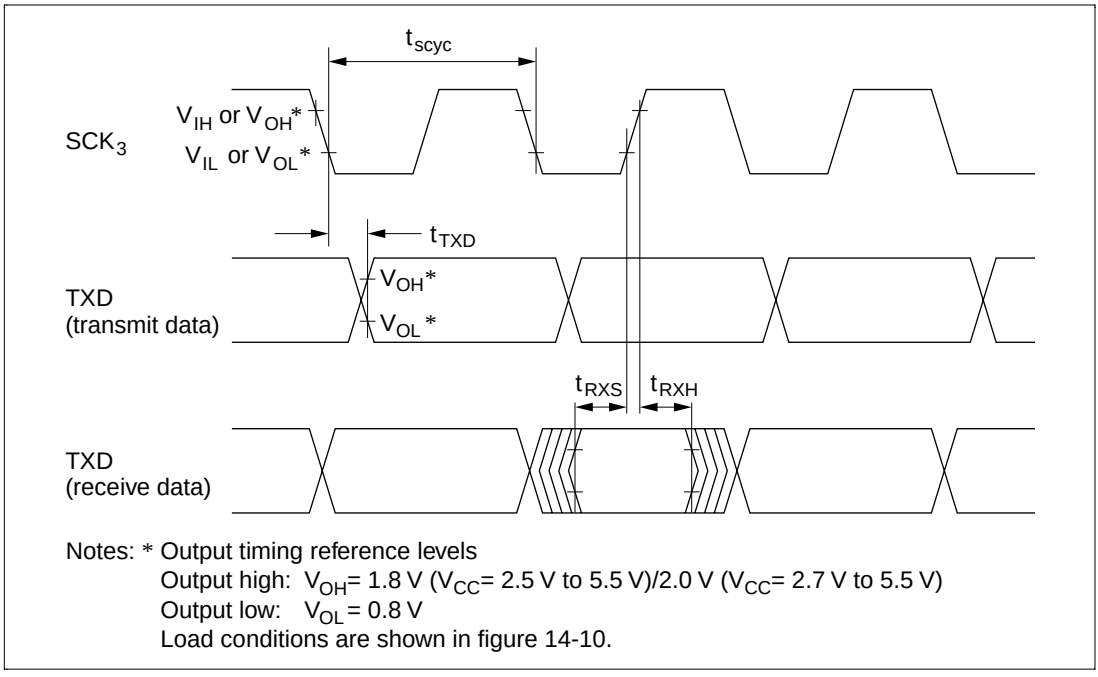



**Figure 14.7 $SCK_3$ Input Clock Timing**




Notes: * Output timing reference levels
Output high: $V_{OH}$= 1.8 V ($V_{CC}$= 2.5 V to 5.5 V)/2.0 V ($V_{CC}$= 2.7 V to 5.5 V)
Output low: $V_{OL}$ = 0.8 V
Load conditions are shown in figure 14-10.

**Figure 14.8 Input/Output Timing of Serial Interface 3 in Synchronous Mode**

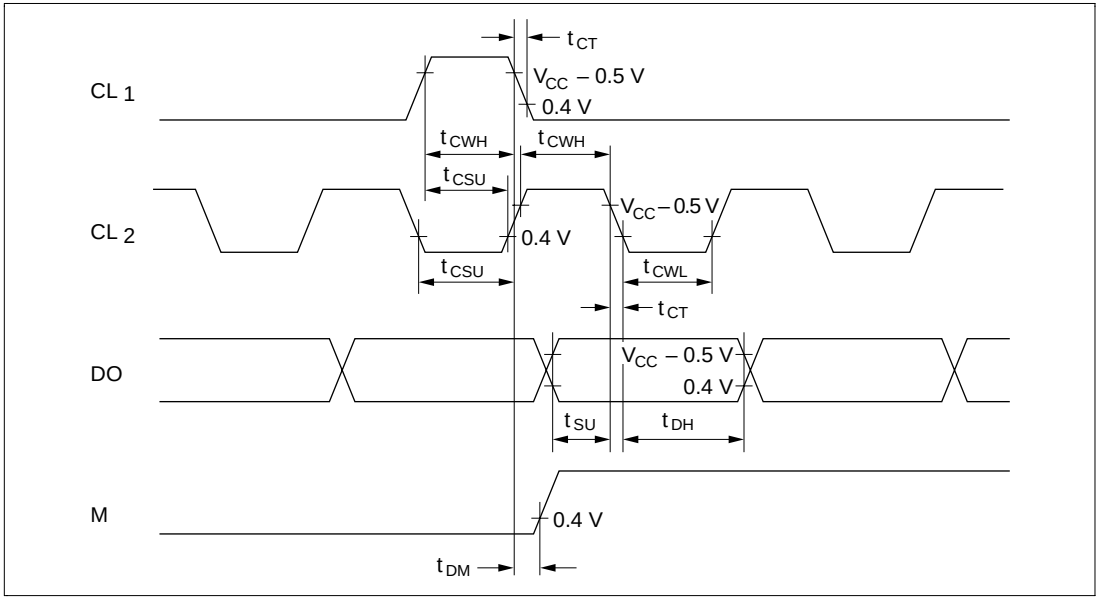



**Figure 14.9 Segment Expansion Signal Timing**

## 14.14 Output Load Circuit

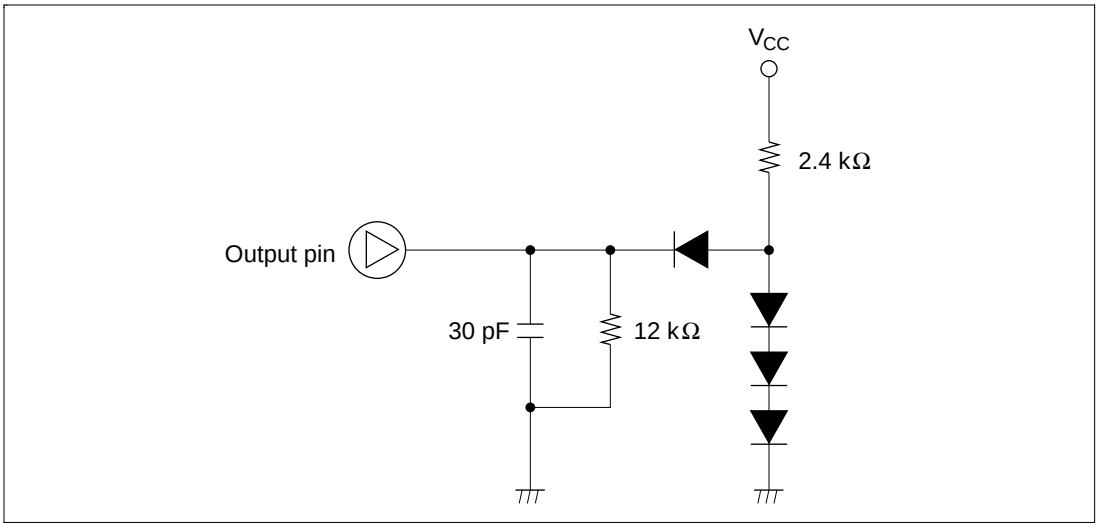



**Figure 14.10 Output Load Condition**

# Appendix A CPU Instruction Set

## A.1 Instructions

### Operation Notation

| | |
|---|---|
| Rd8/16 | General register (destination) (8 or 16 bits) |
| Rs8/16 | General register (source) (8 or 16 bits) |
| Rn8/16 | General register (8 or 16 bits) |
| 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 |
| #xx: 3/8/16 | Immediate data (3, 8, or 16 bits) |
| d: 8/16 | Displacement (8 or 16 bits) |
| @aa: 8/16 | Absolute address (8 or 16 bits) |
| + | Addition |
| – | Subtraction |
| × | Multiplication |
| ÷ | Division |
| ∧ | Logical AND |
| ∨ | Logical OR |
| ⊕ | Exclusive logical OR |
| → | Move |
| ¯ | Logical complement |

### Condition Code Notation

| Symbol | |
|---|---|
| ↕ | Modified according to the instruction result |
| * | Not fixed (value not guaranteed) |
| 0 | Always cleared to 0 |
| — | Not affected by the instruction execution result |

Table A.1 lists the H8/300L CPU instruction set.

**Table A.1 Instruction Set**

| Mnemonic | Operand Size | Operation | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| MOV.B #xx:8, Rd | B | #xx:8 → Rd8 | 2 | | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| MOV.B Rs, Rd | B | Rs8 → Rd8 | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| MOV.B @Rs, Rd | B | @Rs16 → Rd8 | | | 2 | | | | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.B @(d:16, Rs), Rd | B | @(d:16, Rs16)→ Rd8 | | | | 4 | | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.B @Rs+, Rd | B | @Rs16 → Rd8<br>Rs16+1 → Rs16 | | | | | 2 | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.B @aa:8, Rd | B | @aa:8 → Rd8 | | | | | | 2 | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.B @aa:16, Rd | B | @aa:16 → Rd8 | | | | | | 4 | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.B Rs, @Rd | B | Rs8 → @Rd16 | | | 2 | | | | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.B Rs, @(d:16, Rd) | B | Rs8 → @(d:16, Rd16) | | | | 4 | | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.B Rs, @–Rd | B | Rd16–1 → Rd16<br>Rs8 → @Rd16 | | | | | 2 | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.B Rs, @aa:8 | B | Rs8 → @aa:8 | | | | | | 2 | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.B Rs, @aa:16 | B | Rs8 → @aa:16 | | | | | | 4 | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.W #xx:16, Rd | W | #xx:16 → Rd | 4 | | | | | | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.W Rs, Rd | W | Rs16 → Rd16 | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| MOV.W @Rs, Rd | W | @Rs16 → Rd16 | | | 2 | | | | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.W @(d:16, Rs), Rd | W | @(d:16, Rs16) → Rd16 | | | | 4 | | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.W @Rs+, Rd | W | @Rs16 → Rd16<br>Rs16+2 → Rs16 | | | | | 2 | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.W @aa:16, Rd | W | @aa:16 → Rd16 | | | | | | 4 | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.W Rs, @Rd | W | Rs16 → @Rd16 | | | 2 | | | | | | | — | — | ↕ | ↕ | 0 | — | 4 |
| MOV.W Rs, @(d:16, Rd) | W | Rs16 → @(d:16, Rd16) | | | | 4 | | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.W Rs, @–Rd | W | Rd16–2 → Rd16<br>Rs16 → @Rd16 | | | | | 2 | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| MOV.W Rs, @aa:16 | W | Rs16 → @aa:16 | | | | | | 4 | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| POP Rd | W | @SP → Rd16<br>SP+2 → SP | | | | | 2 | | | | | — | — | ↕ | ↕ | 0 | — | 6 |

**Table A.1 Instruction Set (cont)**

| Mnemonic | Operand Size | Operation | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| PUSH Rs | W | SP–2 → SP<br>Rs16 → @SP | | | | | 2 | | | | | — | — | ↕ | ↕ | 0 | — | 6 |
| EEPMOV | — | if R4L≠0 then<br>Repeat @R5 → @R6<br>R5+1 → R5<br>R6+1 → R6<br>R4L–1 → R4L<br>Until R4L=0<br>else next; | | | | | | | | | 4 | — | — | — | — | — | — | (4) |
| ADD.B #xx:8, Rd | B | Rd8+#xx:8 → Rd8 | 2 | | | | | | | | | — | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| ADD.B Rs, Rd | B | Rd8+Rs8 → Rd8 | | 2 | | | | | | | | — | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| ADD.W Rs, Rd | W | Rd16+Rs16 → Rd16 | | 2 | | | | | | | | — | (1) | ↕ | ↕ | ↕ | ↕ | 2 |
| ADDX.B #xx:8, Rd | B | Rd8+#xx:8 +C → Rd8 | 2 | | | | | | | | | — | ↕ | ↕ | (2) | ↕ | ↕ | 2 |
| ADDX.B Rs, Rd | B | Rd8+Rs8 +C → Rd8 | | 2 | | | | | | | | — | ↕ | ↕ | (2) | ↕ | ↕ | 2 |
| ADDS.W #1, Rd | W | Rd16+1 → Rd16 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| ADDS.W #2, Rd | W | Rd16+2 → Rd16 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| INC.B Rd | B | Rd8+1 → Rd8 | | 2 | | | | | | | | — | — | ↕ | ↕ | ↕ | — | 2 |
| DAA.B Rd | B | Rd8 decimal adjust → Rd8 | | 2 | | | | | | | | — | * | ↕ | ↕ | * | (3) | 2 |
| SUB.B Rs, Rd | B | Rd8–Rs8 → Rd8 | | 2 | | | | | | | | — | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| SUB.W Rs, Rd | W | Rd16–Rs16 → Rd16 | | 2 | | | | | | | | — | (1) | ↕ | ↕ | ↕ | ↕ | 2 |
| SUBX.B #xx:8, Rd | B | Rd8–#xx:8 –C → Rd8 | 2 | | | | | | | | | — | ↕ | ↕ | (2) | ↕ | ↕ | 2 |
| SUBX.B Rs, Rd | B | Rd8–Rs8 –C → Rd8 | | 2 | | | | | | | | — | ↕ | ↕ | (2) | ↕ | ↕ | 2 |
| SUBS.W #1, Rd | W | Rd16–1 → Rd16 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| SUBS.W #2, Rd | W | Rd16–2 → Rd16 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| DEC.B Rd | B | Rd8–1 → Rd8 | | 2 | | | | | | | | — | — | ↕ | ↕ | ↕ | — | 2 |
| DAS.B Rd | B | Rd8 decimal adjust → Rd8 | | 2 | | | | | | | | — | * | ↕ | ↕ | * | — | 2 |
| NEG.B Rd | B | 0–Rd → Rd | | 2 | | | | | | | | — | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| CMP.B #xx:8, Rd | B | Rd8–#xx:8 | 2 | | | | | | | | | — | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| CMP.B Rs, Rd | B | Rd8–Rs8 | | 2 | | | | | | | | — | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| CMP.W Rs, Rd | W | Rd16–Rs16 | | 2 | | | | | | | | — | (1) | ↕ | ↕ | ↕ | ↕ | 2 |
| MULXU.B Rs, Rd | B | Rd8 × Rs8 → Rd16 | | 2 | | | | | | | | — | — | — | — | — | — | 14 |

**Table A.1 Instruction Set (cont)**

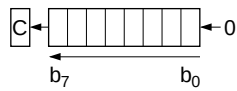
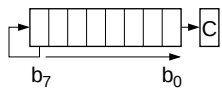
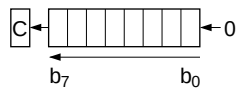
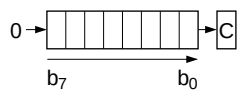
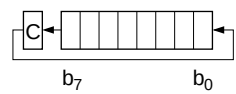
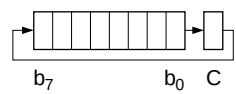
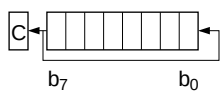
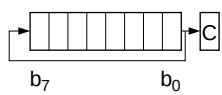
| Mnemonic | Operand Size | Operation | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| DIVXU.B Rs, Rd | B | Rd16÷Rs8 → Rd16 (RdH: remainder, RdL: quotient) | | 2 | | | | | | | | — | — | (5) | (6) | — | — | 14 |
| AND.B #xx:8, Rd | B | Rd8∧#xx:8 → Rd8 | 2 | | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| AND.B Rs, Rd | B | Rd8∧Rs8 → Rd8 | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| OR.B #xx:8, Rd | B | Rd8∨#xx:8 → Rd8 | 2 | | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| OR.B Rs, Rd | B | Rd8∨Rs8 → Rd8 | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| XOR.B #xx:8, Rd | B | Rd8⊕#xx:8 → Rd8 | 2 | | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| XOR.B Rs, Rd | B | Rd8⊕Rs8 → Rd8 | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| NOT.B Rd | B | $\overline{Rd} \rightarrow Rd$ | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | — | 2 |
| SHAL.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | ↕ | ↕ | 2 |
| SHAR.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | ↕ | 2 |
| SHLL.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | ↕ | 2 |
| SHLR.B Rd | B |  | | 2 | | | | | | | | — | — | 0 | ↕ | 0 | ↕ | 2 |
| ROTXL.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | ↕ | 2 |
| ROTXR.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | ↕ | 2 |
| ROTL.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | ↕ | 2 |
| ROTR.B Rd | B |  | | 2 | | | | | | | | — | — | ↕ | ↕ | 0 | ↕ | 2 |

**Table A.1 Instruction Set (cont)**

| Mnemonic | Operand Size | Operation | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| BSET #xx:3, Rd | B | (#xx:3 of Rd8) ← 1 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BSET #xx:3, @Rd | B | (#xx:3 of @Rd16) ← 1 | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BSET #xx:3, @aa:8 | B | (#xx:3 of @aa:8) ← 1 | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BSET Rn, Rd | B | (Rn8 of Rd8) ← 1 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BSET Rn, @Rd | B | (Rn8 of @Rd16) ← 1 | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BSET Rn, @aa:8 | B | (Rn8 of @aa:8) ← 1 | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BCLR #xx:3, Rd | B | (#xx:3 of Rd8) ← 0 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BCLR #xx:3, @Rd | B | (#xx:3 of @Rd16) ← 0 | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BCLR #xx:3, @aa:8 | B | (#xx:3 of @aa:8) ← 0 | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BCLR Rn, Rd | B | (Rn8 of Rd8) ← 0 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BCLR Rn, @Rd | B | (Rn8 of @Rd16) ← 0 | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BCLR Rn, @aa:8 | B | (Rn8 of @aa:8) ← 0 | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BNOT #xx:3, Rd | B | (#xx:3 of Rd8) ← ($\overline{\text{#xx:3 of Rd8}}$) | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BNOT #xx:3, @Rd | B | (#xx:3 of @Rd16) ← ($\overline{\text{#xx:3 of @Rd16}}$) | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BNOT #xx:3, @aa:8 | B | (#xx:3 of @aa:8) ← ($\overline{\text{#xx:3 of @aa:8}}$) | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BNOT Rn, Rd | B | (Rn8 of Rd8) ← ($\overline{\text{Rn8 of Rd8}}$) | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BNOT Rn, @Rd | B | (Rn8 of @Rd16) ← ($\overline{\text{Rn8 of @Rd16}}$) | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BNOT Rn, @aa:8 | B | (Rn8 of @aa:8) ← ($\overline{\text{Rn8 of @aa:8}}$) | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BTST #xx:3, Rd | B | ($\overline{\text{#xx:3 of Rd8}}$) → Z | | 2 | | | | | | | | — | — | — | ↕ | — | — | 2 |
| BTST #xx:3, @Rd | B | ($\overline{\text{#xx:3 of @Rd16}}$) → Z | | | 4 | | | | | | | — | — | — | ↕ | — | — | 6 |
| BTST #xx:3, @aa:8 | B | ($\overline{\text{#xx:3 of @aa:8}}$) → Z | | | | | | 4 | | | | — | — | — | ↕ | — | — | 6 |
| BTST Rn, Rd | B | ($\overline{\text{Rn8 of Rd8}}$) → Z | | 2 | | | | | | | | — | — | — | ↕ | — | — | 2 |
| BTST Rn, @Rd | B | ($\overline{\text{Rn8 of @Rd16}}$) → Z | | | 4 | | | | | | | — | — | — | ↕ | — | — | 6 |
| BTST Rn, @aa:8 | B | ($\overline{\text{Rn8 of @aa:8}}$) → Z | | | | | | 4 | | | | — | — | — | ↕ | — | — | 6 |

**Table A.1 Instruction Set (cont)**

| Mnemonic | Operand Size | Operation | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| BLD #xx:3, Rd | B | (#xx:3 of Rd8) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BLD #xx:3, @Rd | B | (#xx:3 of @Rd16) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BLD #xx:3, @aa:8 | B | (#xx:3 of @aa:8) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BILD #xx:3, Rd | B | ($\overline{\text{#xx:3 of Rd8}}$) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BILD #xx:3, @Rd | B | ($\overline{\text{#xx:3 of @Rd16}}$) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BILD #xx:3, @aa:8 | B | ($\overline{\text{#xx:3 of @aa:8}}$) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BST #xx:3, Rd | B | C → (#xx:3 of Rd8) | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BST #xx:3, @Rd | B | C → (#xx:3 of @Rd16) | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BST #xx:3, @aa:8 | B | C → (#xx:3 of @aa:8) | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BIST #xx:3, Rd | B | $\overline{\text{C}}$ → (#xx:3 of Rd8) | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| BIST #xx:3, @Rd | B | $\overline{\text{C}}$ → (#xx:3 of @Rd16) | | | 4 | | | | | | | — | — | — | — | — | — | 8 |
| BIST #xx:3, @aa:8 | B | $\overline{\text{C}}$ → (#xx:3 of @aa:8) | | | | | | 4 | | | | — | — | — | — | — | — | 8 |
| BAND #xx:3, Rd | B | C∧(#xx:3 of Rd8) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BAND #xx:3, @Rd | B | C∧(#xx:3 of @Rd16) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BAND #xx:3, @aa:8 | B | C∧(#xx:3 of @aa:8) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BIAND #xx:3, Rd | B | C∧($\overline{\text{#xx:3 of Rd8}}$) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BIAND #xx:3, @Rd | B | C∧($\overline{\text{#xx:3 of @Rd16}}$) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BIAND #xx:3, @aa:8 | B | C∧($\overline{\text{#xx:3 of @aa:8}}$) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BOR #xx:3, Rd | B | C∨(#xx:3 of Rd8) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BOR #xx:3, @Rd | B | C∨(#xx:3 of @Rd16) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BOR #xx:3, @aa:8 | B | C∨(#xx:3 of @aa:8) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BIOR #xx:3, Rd | B | C∨($\overline{\text{#xx:3 of Rd8}}$) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BIOR #xx:3, @Rd | B | C∨($\overline{\text{#xx:3 of @Rd16}}$) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BIOR #xx:3, @aa:8 | B | C∨($\overline{\text{#xx:3 of @aa:8}}$) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BXOR #xx:3, Rd | B | C⊕(#xx:3 of Rd8) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |
| BXOR #xx:3, @Rd | B | C⊕(#xx:3 of @Rd16) → C | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BXOR #xx:3, @aa:8 | B | C⊕(#xx:3 of @aa:8) → C | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BIXOR #xx:3, Rd | B | C⊕($\overline{\text{#xx:3 of Rd8}}$) → C | | 2 | | | | | | | | — | — | — | — | — | ↕ | 2 |

**Table A.1 Instruction Set (cont)**

| Mnemonic | Operand Size | Operation | Branching Condition | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| BIXOR #xx:3, @Rd | B | C⊕(#xx:3 of @Rd16) → C | | | | 4 | | | | | | | — | — | — | — | — | ↕ | 6 |
| BIXOR #xx:3, @aa:8 | B | C⊕(#xx:3 of @aa:8) → C | | | | | | | 4 | | | | — | — | — | — | — | ↕ | 6 |
| BRA d:8 (BT d:8) | — | PC ← PC+d:8 | | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BRN d:8 (BF d:8) | — | PC ← PC+2 | | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BHI d:8 | — | If | C ∨ Z = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BLS d:8 | — | condition | C ∨ Z = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BCC d:8 (BHS d:8) | — | is true | C = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BCS d:8 (BLO d:8) | — | then | C = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BNE d:8 | — | PC ← | Z = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BEQ d:8 | — | PC+d:8 | Z = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BVC d:8 | — | else next; | V = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BVS d:8 | — | | V = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BPL d:8 | — | | N = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BMI d:8 | — | | N = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BGE d:8 | — | | N⊕V = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BLT d:8 | — | | N⊕V = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BGT d:8 | — | | Z ∨ (N⊕V) = 0 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| BLE d:8 | — | | Z ∨ (N⊕V) = 1 | | | | | | | 2 | | | — | — | — | — | — | — | 4 |
| JMP @Rn | — | PC ← Rn16 | | | | 2 | | | | | | | — | — | — | — | — | — | 4 |
| JMP @aa:16 | — | PC ← aa:16 | | | | | | | 4 | | | | — | — | — | — | — | — | 6 |
| JMP @@aa:8 | — | PC ← @aa:8 | | | | | | | | | 2 | | — | — | — | — | — | — | 8 |
| BSR d:8 | — | SP–2 → SP<br>PC → @SP<br>PC ← PC+d:8 | | | | | | | | 2 | | | — | — | — | — | — | — | 6 |
| JSR @Rn | — | SP–2 → SP<br>PC → @SP<br>PC ← Rn16 | | | | 2 | | | | | | | — | — | — | — | — | — | 6 |
| JSR @aa:16 | — | SP–2 → SP<br>PC → @SP<br>PC ← aa:16 | | | | | | | 4 | | | | — | — | — | — | — | — | 8 |

**Table A.1 Instruction Set (cont)**

| Mnemonic | Operand Size | Operation | Addressing Mode/ Instruction Length (Bytes) | | | | | | | | | Condition Code | | | | | | No. of States |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | #xx: 8/16 | Rn | @Rn | @(d:16, Rn) | @–Rn/@Rn+ | @aa: 8/16 | @(d:8, PC) | @@aa | Implied | I | H | N | Z | V | C | |
| JSR @@aa:8 | | SP–2 → SP<br>PC → @SP<br>PC ← @aa:8 | | | | | | | | 2 | | — | — | — | — | — | — | 8 |
| RTS | — | PC ← @SP<br>SP+2 → SP | | | | | | | | | 2 | — | — | — | — | — | — | 8 |
| RTE | — | CCR ← @SP<br>SP+2 → SP<br>PC ← @SP<br>SP+2 → SP | | | | | | | | | 2 | ↕ | ↕ | ↕ | ↕ | ↕ | ↕ | 10 |
| SLEEP | — | Transit to power-down state | | | | | | | | | 2 | — | — | — | — | — | — | 2 |
| LDC #xx:8, CCR | B | #xx:8 → CCR | 2 | | | | | | | | | ↕ | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| LDC Rs, CCR | B | Rs8 → CCR | | 2 | | | | | | | | ↕ | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| STC CCR, Rd | B | CCR → Rd8 | | 2 | | | | | | | | — | — | — | — | — | — | 2 |
| ANDC #xx:8, CCR | B | CCR∧#xx:8 → CCR | 2 | | | | | | | | | ↕ | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| ORC #xx:8, CCR | B | CCR∨#xx:8 → CCR | 2 | | | | | | | | | ↕ | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| XORC #xx:8, CCR | B | CCR⊕#xx:8 → CCR | 2 | | | | | | | | | ↕ | ↕ | ↕ | ↕ | ↕ | ↕ | 2 |
| NOP | — | PC ← PC+2 | | | | | | | | | 2 | — | — | — | — | — | — | 2 |

Notes: (1) Set to 1 when there is a carry or borrow from bit 11; otherwise cleared to 0.

(2) If the result is zero, the previous value of the flag is retained; otherwise the flag is cleared to 0.

(3) Set to 1 if decimal adjustment produces a carry; otherwise retains value prior to arithmetic operation.

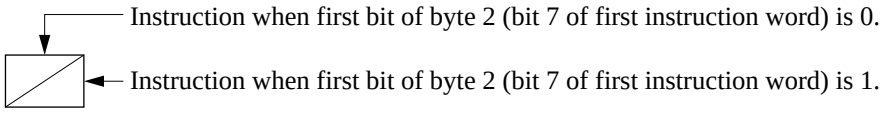
(4) The number of states required for execution is 4n + 9 (n = value of R4L).

(5) Set to 1 if the divisor is negative; otherwise cleared to 0.

(6) Set to 1 if the divisor is zero; otherwise cleared to 0.

## A.2 Operation Code Map

Table A.2 is an operation code map. It shows the operation codes contained in the first byte of the instruction code (bits 15 to 8 of the first instruction word).



## Table A.2 Operation Code Map

| High \ Low | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NOP | SLEEP | STC | LDC | ORC | XORC | ANDC | LDC | ADD | | INC | ADDS | MOV | | ADDX | DAA |
| 1 | SHLL / SHAL | SHLR / SHAR | ROTXL / ROTL | ROTXR / ROTR | OR | XOR | AND | NOT / NEG | SUB | | DEC | SUBS | CMP | | SUBX | DAS |
| 2 | MOV | | | | | | | | | | | | | | | |
| 3 | | | | | | | | | | | | | | | | |
| 4 | BRA | BRN | BHI | BLS | BCC | BCS | BNE | BEQ | BVC | BVS | BPL | BMI | BGE | BLT | BGT | BLE |
| 5 | MULXU | DIVXU | | | RTS | BSR | RTE | | | JMP | | | | JSR | | |
| 6 | BSET | BNOT | BCLR | BTST | | | | BST / BIST | MOV* | | | | | | | |
| 7 | | | | | BOR / BIOR | BXOR / BIXOR | BAND / BIAND | BLD / BILD | | MOV | | EEPMOV | Bit-manipulation instructions | | | |
| 8 | ADD | | | | | | | | | | | | | | | |
| 9 | ADDX | | | | | | | | | | | | | | | |
| A | CMP | | | | | | | | | | | | | | | |
| B | SUBX | | | | | | | | | | | | | | | |
| C | OR | | | | | | | | | | | | | | | |
| D | XOR | | | | | | | | | | | | | | | |
| E | AND | | | | | | | | | | | | | | | |
| F | MOV | | | | | | | | | | | | | | | |

Note: * The PUSH and POP instructions are identical in machine language to MOV instructions.

## A.3 Number of Execution States

The tables here can be used to calculate the number of states required for instruction execution. Table A-3 indicates the number of states required for each cycle (instruction fetch, data read/write, etc.) in instruction execution, and table A-4 indicates the number of cycles of each type occurring in each instruction. The total number of states required for execution of an instruction can be calculated from these two tables as follows:

$$\text{Execution states} = I \times S_I + J \times S_J + K \times S_K + L \times S_L + M \times S_M + N \times S_N$$

**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 | $S_M$ | | — |
| Internal operation | $S_N$ | 1 | |

Note: * Depends on which on-chip module is accessed. See 2.9.1, Notes on Data Access for details.

**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 Rs, Rd | 1 | | | | | |
| ADDS | ADDS.W #1, Rd | 1 | | | | | |
| | ADDS.W #2, Rd | 1 | | | | | |
| ADDX | ADDX.B #xx:8, Rd | 1 | | | | | |
| | ADDX.B Rs, Rd | 1 | | | | | |
| AND | AND.B #xx:8, Rd | 1 | | | | | |
| | AND.B Rs, Rd | 1 | | | | | |
| ANDC | ANDC #xx:8, CCR | 1 | | | | | |
| BAND | BAND #xx:3, Rd | 1 | | | | | |
| | BAND #xx:3, @Rd | 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 | | | | | |
| | BLT d:8 | 2 | | | | | |
| | BGT d:8 | 2 | | | | | |
| | BLE d:8 | 2 | | | | | |
| BCLR | BCLR #xx:3, Rd | 1 | | | | | |
| | BCLR #xx:3, @Rd | 2 | | | 2 | | |
| | BCLR #xx:3, @aa:8 | 2 | | | 2 | | |
| | BCLR Rn, Rd | 1 | | | | | |

**Table A.4 Number of Cycles in Each Instruction (cont)**

| Instruction | Mnemonic | Instruction Fetch I | Branch Addr. Read J | Stack Operation K | Byte Data Access L | Word Data Access M | Internal Operation N |
|---|---|---|---|---|---|---|---|
| BCLR | BCLR Rn, @Rd | 2 | | | 2 | | |
| | BCLR Rn, @aa:8 | 2 | | | 2 | | |
| BIAND | BIAND #xx:3, Rd | 1 | | | | | |
| | BIAND #xx:3, @Rd | 2 | | | 1 | | |
| | BIAND #xx:3, @aa:8 | 2 | | | 1 | | |
| BILD | BILD #xx:3, Rd | 1 | | | | | |
| | BILD #xx:3, @Rd | 2 | | | 1 | | |
| | BILD #xx:3, @aa:8 | 2 | | | 1 | | |
| BIOR | BIOR #xx:3, Rd | 1 | | | | | |
| | BIOR #xx:3, @Rd | 2 | | | 1 | | |
| | BIOR #xx:3, @aa:8 | 2 | | | 1 | | |
| BIST | BIST #xx:3, Rd | 1 | | | | | |
| | BIST #xx:3, @Rd | 2 | | | 2 | | |
| | BIST #xx:3, @aa:8 | 2 | | | 2 | | |
| BIXOR | BIXOR #xx:3, Rd | 1 | | | | | |
| | BIXOR #xx:3, @Rd | 2 | | | 1 | | |
| | BIXOR #xx:3, @aa:8 | 2 | | | 1 | | |
| BLD | BLD #xx:3, Rd | 1 | | | | | |
| | BLD #xx:3, @Rd | 2 | | | 1 | | |
| | BLD #xx:3, @aa:8 | 2 | | | 1 | | |
| BNOT | BNOT #xx:3, Rd | 1 | | | | | |
| | BNOT #xx:3, @Rd | 2 | | | 2 | | |
| | BNOT #xx:3, @aa:8 | 2 | | | 2 | | |
| | BNOT Rn, Rd | 1 | | | | | |
| | BNOT Rn, @Rd | 2 | | | 2 | | |
| | BNOT Rn, @aa:8 | 2 | | | 2 | | |
| BOR | BOR #xx:3, Rd | 1 | | | | | |
| | BOR #xx:3, @Rd | 2 | | | 1 | | |
| | BOR #xx:3, @aa:8 | 2 | | | 1 | | |
| BSET | BSET #xx:3, Rd | 1 | | | | | |
| | BSET #xx:3, @Rd | 2 | | | 2 | | |
| | BSET #xx:3, @aa:8 | 2 | | | 2 | | |
| | BSET Rn, Rd | 1 | | | | | |
| | BSET Rn, @Rd | 2 | | | 2 | | |

**Table A.4 Number of Cycles in Each Instruction (cont)**

| Instruction | Mnemonic | Instruction Fetch I | Branch Addr. Read J | Stack Operation K | Byte Data Access L | Word Data Access M | Internal Operation N |
|---|---|---|---|---|---|---|---|
| BSET | BSET Rn, @aa:8 | 2 | | | 2 | | |
| BSR | BSR d:8 | 2 | | 1 | | | |
| BST | BST #xx:3, Rd | 1 | | | | | |
| | BST #xx:3, @Rd | 2 | | | 2 | | |
| | BST #xx:3, @aa:8 | 2 | | | 2 | | |
| BTST | BTST #xx:3, Rd | 1 | | | | | |
| | BTST #xx:3, @Rd | 2 | | | 1 | | |
| | BTST #xx:3, @aa:8 | 2 | | | 1 | | |
| | BTST Rn, Rd | 1 | | | | | |
| | BTST Rn, @Rd | 2 | | | 1 | | |
| | BTST Rn, @aa:8 | 2 | | | 1 | | |
| BXOR | BXOR #xx:3, Rd | 1 | | | | | |
| | BXOR #xx:3, @Rd | 2 | | | 1 | | |
| | BXOR #xx:3, @aa:8 | 2 | | | 1 | | |
| CMP | CMP. B #xx:8, Rd | 1 | | | | | |
| | CMP. B Rs, Rd | 1 | | | | | |
| | CMP.W Rs, Rd | 1 | | | | | |
| DAA | DAA.B Rd | 1 | | | | | |
| DAS | DAS.B Rd | 1 | | | | | |
| DEC | DEC.B Rd | 1 | | | | | |
| DIVXU | DIVXU.B Rs, Rd | 1 | | | | | 12 |
| EEPMOV | EEPMOV | 2 | | | 2n+2* | | 1 |
| INC | INC.B Rd | 1 | | | | | |
| JMP | JMP @Rn | 2 | | | | | |
| | JMP @aa:16 | 2 | | | | | 2 |
| | JMP @@aa:8 | 2 | 1 | | | | 2 |
| JSR | JSR @Rn | 2 | | 1 | | | |
| | JSR @aa:16 | 2 | | 1 | | | 2 |
| | JSR @@aa:8 | 2 | 1 | 1 | | | |
| LDC | LDC #xx:8, CCR | 1 | | | | | |
| | LDC Rs, CCR | 1 | | | | | |
| MOV | MOV.B #xx:8, Rd | 1 | | | | | |
| | MOV.B Rs, Rd | 1 | | | | | |

Note: n: Initial value in R4L. The source and destination operands are accessed n + 1 times each.

**Table A.4 Number of Cycles in Each Instruction (cont)**

| 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, Rd | 1 | | | 1 | | |
| | MOV.B @(d:16, Rs), Rd | 2 | | | 1 | | |
| | MOV.B @Rs+, Rd | 1 | | | 1 | | 2 |
| | MOV.B @aa:8, Rd | 1 | | | 1 | | |
| | MOV.B @aa:16, Rd | 2 | | | 1 | | |
| | MOV.B Rs, @Rd | 1 | | | 1 | | |
| | MOV.B Rs, @(d:16, Rd) | 2 | | | 1 | | |
| | MOV.B Rs, @–Rd | 1 | | | 1 | | 2 |
| | MOV.B Rs, @aa:8 | 1 | | | 1 | | |
| | MOV.B Rs, @aa:16 | 2 | | | 1 | | |
| | MOV.W #xx:16, Rd | 2 | | | | | |
| | MOV.W Rs, Rd | 1 | | | | | |
| | MOV.W @Rs, Rd | 1 | | | | 1 | |
| | MOV.W @(d:16, Rs), Rd | 2 | | | | 1 | |
| | MOV.W @Rs+, Rd | 1 | | | | 1 | 2 |
| | MOV.W @aa:16, Rd | 2 | | | | 1 | |
| | MOV.W Rs, @Rd | 1 | | | | 1 | |
| | MOV.W Rs, @(d:16, Rd) | 2 | | | | 1 | |
| | MOV.W Rs, @–Rd | 1 | | | | 1 | 2 |
| | MOV.W Rs, @aa:16 | 2 | | | | 1 | |
| MULXU | MULXU.B Rs, Rd | 1 | | | | | 12 |
| NEG | NEG.B Rd | 1 | | | | | |
| NOP | NOP | 1 | | | | | |
| NOT | NOT.B Rd | 1 | | | | | |
| OR | OR.B #xx:8, Rd | 1 | | | | | |
| | OR.B Rs, Rd | 1 | | | | | |
| ORC | ORC #xx:8, CCR | 1 | | | | | |
| POP | POP Rd | 1 | | 1 | | | 2 |
| PUSH | PUSH Rs | 1 | | 1 | | | 2 |
| ROTL | ROTL.B Rd | 1 | | | | | |
| ROTR | ROTR.B Rd | 1 | | | | | |

**Table A.4 Number of Cycles in Each Instruction (cont)**

| Instruction | Mnemonic | Instruction Fetch I | Branch Addr. Read J | Stack Operation K | Byte Data Access L | Word Data Access M | Internal Operation N |
|---|---|---|---|---|---|---|---|
| ROTXL | ROTXL.B Rd | 1 | | | | | |
| ROTXR | ROTXR.B Rd | 1 | | | | | |
| RTE | RTE | 2 | | 2 | | | 2 |
| RTS | RTS | 2 | | 1 | | | 2 |
| SHLL | SHLL.B Rd | 1 | | | | | |
| SHAL | SHAL.B Rd | 1 | | | | | |
| SHAR | SHAR.B Rd | 1 | | | | | |
| SHLR | SHLR.B Rd | 1 | | | | | |
| SLEEP | SLEEP | 1 | | | | | |
| STC | STC CCR, Rd | 1 | | | | | |
| SUB | SUB.B Rs, Rd | 1 | | | | | |
| | SUB.W Rs, Rd | 1 | | | | | |
| SUBS | SUBS.W #1, Rd | 1 | | | | | |
| | SUBS.W #2, Rd | 1 | | | | | |
| SUBX | SUBX.B #xx:8, Rd | 1 | | | | | |
| | SUBX.B Rs, Rd | 1 | | | | | |
| XOR | XOR.B #xx:8, Rd | 1 | | | | | |
| | XOR.B Rs, Rd | 1 | | | | | |
| XORC | XORC #xx:8, CCR | 1 | | | | | |

## B.1 I/O Registers (1)

| Address (low) | Register Name | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module Name |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Bit Names | | | | | | | | |
| H'A0 | SCR1 | SNC1 | SNC0 | — | — | CKS3 | CKS2 | CKS1 | CKS0 | SCI1 |
| H'A1 | SCSR1 | — | SOL | ORER | — | — | — | — | STF | |
| H'A2 | SDRU | SDRU7 | SDRU6 | SDRU5 | SDRU4 | SDRU3 | SDRU2 | SDRU1 | SDRU0 | |
| H'A3 | SDRL | SDRL7 | SDRL6 | SDRL5 | SDRL4 | SDRL3 | SDRL2 | SDRL1 | SDRL0 | |
| H'A4 | STAR | — | — | — | STA4 | STA3 | STA2 | STA1 | STA0 | SCI2 |
| H'A5 | EDAR | — | — | — | EDA4 | EDA3 | EDA2 | EDA1 | EDA0 | |
| H'A6 | SCR2 | — | — | — | GAP1 | GAP0 | CKS2 | CKS1 | CKS0 | |
| H'A7 | SCSR2 | — | — | — | SOL | ORER | WT | ABT | STF | |
| H'A8 | SMR | COM | CHR | PE | PM | STOP | MP | CKS1 | CKS0 | SCI3 |
| H'A9 | BRR | BRR7 | BRR6 | BRR5 | BRR4 | BRR3 | BRR2 | BRR1 | BRR0 | |
| H'AA | SCR3 | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 | |
| H'AB | TDR | TDR7 | TDR6 | TDR5 | TDR4 | TDR3 | TDR2 | TDR1 | TDR0 | |
| H'AC | SSR | TDRE | RDRF | OER | FER | PER | TEND | MPBR | MPBT | |
| H'AD | RDR | RDR7 | RDR6 | RDR5 | RDR4 | RDR3 | RDR2 | RDR1 | RDR0 | |
| H'AE | | | | | | | | | | |
| H'AF | | | | | | | | | | |
| H'B0 | TMA | TMA7 | TMA6 | TMA5 | — | TMA3 | TMA2 | TMA1 | TMA0 | Timer A |
| H'B1 | TCA | TCA7 | TCA6 | TCA5 | TCA4 | TCA3 | TCA2 | TCA1 | TCA0 | |
| H'B2 | TMB | TMB7 | — | — | — | — | TMB2 | TMB1 | TMB0 | Timer B |
| H'B3 | TCB/TLB | TCB7/ TLB7 | TCB6/ TLB6 | TCB5/ TLB5 | TCB4/ TLB4 | TCB3/ TLB3 | TCB2/ TLB2 | TCB1/ TLB1 | TCB0/ TLB0 | |
| H'B4 | TMC | TMC7 | TMC6 | TMC5 | — | — | TMC2 | TMC1 | TMC0 | Timer C |
| H'B5 | TCC/TLC | TCC7/ TLC7 | TCC6/ TLC6 | TCC5/ TLC5 | TCC4/ TLC4 | TCC3/ TLC3 | TCC2/ TLC2 | TCC1/ TLC1 | TCC0/ TLC0 | |
| H'B6 | TCRF | TOLH | CKSH2 | CKSH1 | CKSH0 | TOLL | CKSL2 | CKSL1 | CKSL0 | Timer F |
| H'B7 | TCSRF | OVFH | CMFH | OVIEH | CCLRH | OVFL | CMFL | OVIEL | CCLRL | |
| H'B8 | TCFH | TCFH7 | TCFH6 | TCFH5 | TCFH4 | TCFH3 | TCFH2 | TCFH1 | TCFH0 | |

Notation:

SCI1: Serial communication interface 1

SCI2: Serial communication interface 2

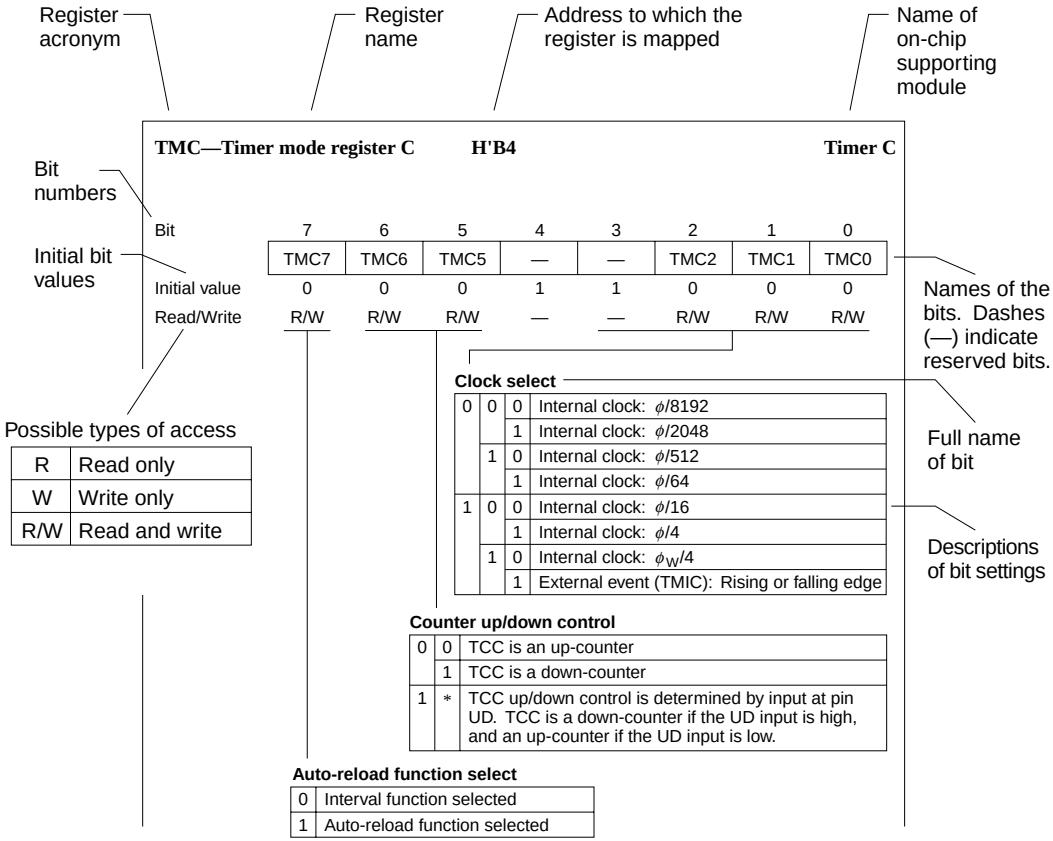
SCI3: Serial communication interface 3

| Address (low) | Register Name | Bit Names Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module Name |
|---|---|---|---|---|---|---|---|---|---|---|
| H'B9 | TCFL | TCFL7 | TCFL6 | TCFL5 | TCFL4 | TCFL3 | TCFL2 | TCFL1 | TCFL0 | Timer F |
| H'BA | OCRFH | OCRFH7 | OCRFH6 | OCRFH5 | OCRFH4 | OCRFH3 | OCRFH2 | OCRFH1 | OCRFH0 | |
| H'BB | OCRFL | OCRFL7 | OCRFL6 | OCRFL5 | OCRFL4 | OCRFL3 | OCRFL2 | OCRFL1 | OCRFL0 | |
| H'BC | TMG | OVFH | OVFL | OVIE | IIEGS | CCLR1 | CCLR0 | CKS1 | CKS0 | Timer G |
| H'BD | ICRGF | ICRGF7 | ICRGF6 | ICRGF5 | ICRGF4 | ICRGF3 | ICRGF2 | ICRGF1 | ICRGF0 | |
| H'BE | ICRGR | ICRGR7 | ICRGR6 | ICRGR5 | ICRGR4 | ICRGR3 | ICRGR2 | ICRGR1 | ICRGR0 | |
| H'BF | | | | | | | | | | |
| H'C0 | LPCR | DTS1 | DTS0 | CMX | SGX | SGS3 | SGS2 | SGS1 | SGS0 | LCD controller/ driver |
| H'C1 | LCR | — | PSW | ACT | DISP | CKS3 | CKS2 | CKS1 | CKS0 | |
| H'C2 | | | | | | | | | | |
| H'C3 | | | | | | | | | | |
| H'C4 | AMR | CKS | TRGE | — | — | CH3 | CH2 | CH1 | CH0 | A/D converter |
| H'C5 | ADRR | ADR7 | ADR6 | ADR5 | ADR4 | ADR3 | ADR2 | ADR1 | ADR0 | |
| H'C6 | ADSR | ADSF | — | — | — | — | — | — | — | |
| H'C7 | | | | | | | | | | |
| H'C8 | PMR1 | IRQ3 | IRQ2 | IRQ1 | PWM | TMIG | TMOFH | TMOFL | TMOW | I/O ports |
| H'C9 | PMR2 | — | — | POF2 | NCS | IRQ0 | POF1 | UD | IRQ4 | |
| H'CA | PMR3 | CS | STRB | SO2 | SI2 | SCK2 | SO1 | SI1 | SCK1 | |
| H'CB | PMR4 | NMOD7 | NMOD6 | NMOD5 | NMOD4 | NMOD3 | NMOD2 | NMOD1 | NMOD0 | |
| H'CC | PMR5 | $WKP_7$ | $WKP_6$ | $WKP_5$ | $WKP_4$ | $WKP_3$ | $WKP_2$ | $WKP_1$ | $WKP_0$ | |
| H'CD | | | | | | | | | | |
| H'CE | | | | | | | | | | |
| H'CF | RLCTR | — | — | — | — | — | — | RLCT1 | RLCT0 | |
| H'D0 | PWCR | — | — | — | — | — | — | — | PWCR0 | 14-bit PWM |
| H'D1 | PWDRU | — | — | PWDRU5 | PWDRU4 | PWDRU3 | PWDRU2 | PWDRU1 | PWDRU0 | |
| H'D2 | PWDRL | PWDRL7 | PWDRL6 | PWDRL5 | PWDRL4 | PWDRL3 | PWDRL2 | PWDRL1 | PWDRL0 | |
| H'D3 | | | | | | | | | | |
| H'D4 | PDR1 | $P1_7$ | $P1_6$ | $P1_5$ | $P1_4$ | $P1_3$ | $P1_2$ | $P1_1$ | $P1_0$ | I/O ports |
| H'D5 | PDR2 | $P2_7$ | $P2_6$ | $P2_5$ | $P2_4$ | $P2_3$ | $P2_2$ | $P2_1$ | $P2_0$ | |
| H'D6 | PDR3 | $P3_7$ | $P3_6$ | $P3_5$ | $P3_4$ | $P3_3$ | $P3_2$ | $P3_1$ | $P3_0$ | |
| H'D7 | PDR4 | — | — | — | — | $P4_3$ | $P4_2$ | $P4_1$ | $P4_0$ | |
| H'D8 | PDR5 | $P5_7$ | $P5_6$ | $P5_5$ | $P5_4$ | $P5_3$ | $P5_2$ | $P5_1$ | $P5_0$ | |
| H'D9 | PDR6 | $P6_7$ | $P6_6$ | $P6_5$ | $P6_4$ | $P6_3$ | $P6_2$ | $P6_1$ | $P6_0$ | |

| Address (low) | Register Name | Bit Names Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module Name |
|---|---|---|---|---|---|---|---|---|---|---|
| H'DA | PDR7 | $P7_7$ | $P7_6$ | $P7_5$ | $P7_4$ | $P7_3$ | $P7_2$ | $P7_1$ | $P7_0$ | I/O ports |
| H'DB | PDR8 | $P8_7$ | $P8_6$ | $P8_5$ | $P8_4$ | $P8_3$ | $P8_2$ | $P8_1$ | $P8_0$ | |
| H'DC | PDR9 | $P9_7$ | $P9_6$ | $P9_5$ | $P9_4$ | $P9_3$ | $P9_2$ | $P9_1$ | $P9_0$ | |
| H'DD | PDRA | — | — | — | — | $PA_3$ | $PA_2$ | $PA_1$ | $PA_0$ | |
| H'DE | PDRB | $PB_7$ | $PB_6$ | $PB_5$ | $PB_4$ | $PB_3$ | $PB_2$ | $PB_1$ | $PB_0$ | |
| H'DF | PDRC | — | — | — | — | $PC_3$ | $PC_2$ | $PC_1$ | $PC_0$ | |
| H'E0 | PUCR1 | $PUCR1_7$ | $PUCR1_6$ | $PUCR1_5$ | $PUCR1_4$ | $PUCR1_3$ | $PUCR1_2$ | $PUCR1_1$ | $PUCR1_0$ | |
| H'E1 | PUCR3 | $PUCR3_7$ | $PUCR3_6$ | $PUCR3_5$ | $PUCR3_4$ | $PUCR3_3$ | $PUCR3_2$ | $PUCR3_1$ | $PUCR3_0$ | |
| H'E2 | PUCR5 | $PUCR5_7$ | $PUCR5_6$ | $PUCR5_5$ | $PUCR5_4$ | $PUCR5_3$ | $PUCR5_2$ | $PUCR5_1$ | $PUCR5_0$ | |
| H'E3 | PUCR6 | $PUCR6_7$ | $PUCR6_6$ | $PUCR6_5$ | $PUCR6_4$ | $PUCR6_3$ | $PUCR6_2$ | $PUCR6_1$ | $PUCR6_0$ | |
| H'E4 | PCR1 | $PCR1_7$ | $PCR1_6$ | $PCR1_5$ | $PCR1_4$ | $PCR1_3$ | $PCR1_2$ | $PCR1_1$ | $PCR1_0$ | |
| H'E5 | PCR2 | $PCR2_7$ | $PCR2_6$ | $PCR2_5$ | $PCR2_4$ | $PCR2_3$ | $PCR2_2$ | $PCR2_1$ | $PCR2_0$ | |
| H'E6 | PCR3 | $PCR3_7$ | $PCR3_6$ | $PCR3_5$ | $PCR3_4$ | $PCR3_3$ | $PCR3_2$ | $PCR3_1$ | $PCR3_0$ | |
| H'E7 | PCR4 | — | — | — | — | — | $PCR4_2$ | $PCR4_1$ | $PCR4_0$ | |
| H'E8 | PCR5 | $PCR5_7$ | $PCR5_6$ | $PCR5_5$ | $PCR5_4$ | $PCR5_3$ | $PCR5_2$ | $PCR5_1$ | $PCR5_0$ | |
| H'E9 | PCR6 | $PCR6_7$ | $PCR6_6$ | $PCR6_5$ | $PCR6_4$ | $PCR6_3$ | $PCR6_2$ | $PCR6_1$ | $PCR6_0$ | |
| H'EA | PCR7 | $PCR7_7$ | $PCR7_6$ | $PCR7_5$ | $PCR7_4$ | $PCR7_3$ | $PCR7_2$ | $PCR7_1$ | $PCR7_0$ | |
| H'EB | PCR8 | $PCR8_7$ | $PCR8_6$ | $PCR8_5$ | $PCR8_4$ | $PCR8_3$ | $PCR8_2$ | $PCR8_1$ | $PCR8_0$ | |
| H'EC | PCR9 | $PCR9_7$ | $PCR9_6$ | $PCR9_5$ | $PCR9_4$ | $PCR9_3$ | $PCR9_2$ | $PCR9_1$ | $PCR9_0$ | |
| H'ED | PCRA | — | — | — | — | $PCRA_3$ | $PCRA_2$ | $PCRA_1$ | $PCRA_0$ | |
| H'EE | | | | | | | | | | |
| H'EF | | | | | | | | | | |
| H'F0 | SYSCR1 | SSBY | STS2 | STS1 | STS0 | LSON | — | — | — | System control |
| H'F1 | SYSCR2 | — | — | — | NESEL | DTON | MSON | SA1 | SA0 | |
| H'F2 | IEGR | — | — | — | IEG4 | IEG3 | IEG2 | IEG1 | IEG0 | |
| H'F3 | IENR1 | IENTA | IENS1 | IENWP | IEN4 | IEN3 | IEN2 | IEN1 | IEN0 | |
| H'F4 | IENR2 | IENDT | IENAD | IENS2 | IENTG | IENTFH | IENTFL | IENTC | IENTB | |
| H'F5 | | | | | | | | | | |
| H'F6 | IRR1 | IRRTA | IRRS1 | — | IRRI4 | IRRI3 | IRRI2 | IRRI1 | IRRI0 | System control |
| H'F7 | IRR2 | IRRDT | IRRAD | IRRS2 | IRRTG | IRRTFH | IRRTFL | IRRTC | IRRTB | |
| H'F8 | | | | | | | | | | |
| H'F9 | IWPR | IWPF7 | IWPF6 | IWPF5 | IWPF4 | IWPF3 | IWPF2 | IWPF1 | IWPF0 | System control |

| Address (low) | Register Name | Bit Names Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Module Name |
|---|---|---|---|---|---|---|---|---|---|---|
| H'FA | | | | | | | | | | |
| H'FB | | | | | | | | | | |
| H'FC | | | | | | | | | | |
| H'FD | | | | | | | | | | |
| H'FE | | | | | | | | | | |
| H'FF | | | | | | | | | | |
| H'FF | | | | | | | | | | |

# B.2 I/O Registers (2)



| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SNC1 | SNC0 | — | — | CKS3 | CKS2 | CKS1 | CKS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Clock Select (CKS2 to CKS0)**

| Bit 2 | Bit 1 | Bit 0 | Prescaler Division | Serial Clock Cycle | |
|---|---|---|---|---|---|
| | | | | Synchronous | |
| CKS2 | CKS1 | CKS0 | | $\phi$ = 5 MHz | $\phi$ = 2.5 MHz |
| 0 | 0 | 0 | $\phi$/1024 | 204.8 µs | 409.6 µs |
| | | 1 | $\phi$/256 | 51.2 µs | 102.4 µs |
| | 1 | 0 | $\phi$/64 | 12.8 µs | 25.6 µs |
| | | 1 | $\phi$/32 | 6.4 µs | 12.8 µs |
| 1 | 0 | 0 | $\phi$/16 | 3.2 µs | 6.4 µs |
| | | 1 | $\phi$/8 | 1.6 µs | 3.2 µs |
| | 1 | 0 | $\phi$/4 | 0.8 µs | 1.6 µs |
| | | 1 | $\phi$/2 | — | 0.8 µs |

**Clock source select**

| | |
|---|---|
| 0 | Clock source is prescaler S, and pin $SCK_1$ is output pin |
| 1 | Clock source is external clock, and pin $SCK_1$ is input pin |

**Operation mode select**

| | | |
|---|---|---|
| 0 | 0 | 8-bit synchronous transfer mode |
| | 1 | 16-bit synchronous transfer mode |
| 1 | 0 | Continuous clock output mode |
| | 1 | Reserved |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | SOL | ORER | — | — | — | — | STF |
| Initial value | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | R/W | R/(W)* | — | — | — | — | R/W |

**Start flag**

| | | |
|---|---|---|
| 0 | Read | Indicates that transfer is stopped |
| | Write | Invalid |
| 1 | Read | Indicates transfer in progress |
| | Write | Starts a transfer operation |

**Overrun error flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading 1, cleared by writing 0 |
| 1 | [Setting condition]<br>Set if a clock pulse is input after transfer is complete, when an external clock is used |

**Extended data bit**

| | | |
|---|---|---|
| 0 | Read | $SO_1$ pin output level is low |
| | Write | $SO_1$ pin output level changes to low |
| 1 | Read | $SO_1$ pin output level is high |
| | Write | $SO_1$ pin output level changes to high |

Note: * Only a write of 0 for flag clearing is possible.

**SDRU—Serial data register U** **H'A2** **SCI1**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SDRU7 | SDRU6 | SDRU5 | SDRU4 | SDRU3 | SDRU2 | SDRU1 | SDRU0 |
| Initial value | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Stores transmit and receive data
8-bit transfer mode: Not used
16-bit transfer mode: Upper 8 bits of data

**SDRL—Serial data register L** **H'A3** **SCI1**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SDRL7 | SDRL6 | SDRL5 | SDRL4 | SDRL3 | SDRL2 | SDRL1 | SDRL0 |
| Initial value | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Stores transmit and receive data
8-bit transfer mode: 8-bit data
16-bit transfer mode: Lower 8 bits of data

**STAR—Start address register** **H'A4** **SCI2**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | STA4 | STA3 | STA2 | STA1 | STA0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

Transfer start address in range from H'FF80 to H'FF9F

**EDAR—End address register** **H'A5** **SCI2**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | EDA4 | EDA3 | EDA2 | EDA1 | EDA0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

Transfer end address in range from H'FF80 to H'FF9F

**SCR2—Serial control register 2** **H'A6** **SCI2**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | GAP1 | GAP0 | CKS2 | CKS1 | CKS0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

**Clock Select (CKS2 to CKS0)**

| Bit 2 | Bit 1 | Bit 0 | | | Prescaler Division | Serial Clock Cycle | |
|---|---|---|---|---|---|---|---|
| CKS2 | CKS1 | CKS0 | Pin $SCK_2$ | Clock Source | | $\phi$ = 5 MHz | $\phi$ = 2.5 MHz |
| 0 | 0 | 0 | $SCK_2$ output | Prescaler S | $\phi$/256 | 51.2 µs | 102.4 µs |
| | | 1 | | | $\phi$/64 | 12.8 µs | 25.6 µs |
| | 1 | 0 | | | $\phi$/32 | 6.4 µs | 12.8 µs |
| | | 1 | | | $\phi$/16 | 3.2 µs | 6.4 µs |
| 1 | 0 | 0 | | | $\phi$/8 | 1.6 µs | 3.2 µs |
| | | 1 | | | $\phi$/4 | 0.8 µs | 1.6 µs |
| | 1 | 0 | | | $\phi$/2 | — | 0.8 µs |
| | | 1 | $SCK_2$ input | External clock | — | — | — |

**Gap select**

| | | |
|---|---|---|
| 0 | 0 | No gaps between bytes |
| | 1 | A gap of 8 clock cycles is inserted between bytes |
| 1 | 0 | A gap of 24 clock cycles is inserted between bytes |
| | 1 | A gap of 56 clock cycles is inserted between bytes |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | SOL | ORER | WT | ABT | STF |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/(W)* | R/(W)* | R/(W)* | R/W |

**Start flag**

| | | |
|---|---|---|
| 0 | Read | Indicates that transfer is stopped |
| | Write | Stops a transfer operation |
| 1 | Read | Indicates transfer in progress or waiting for $\overline{CS}$ input |
| | Write | Starts a transfer operation |

**Abort flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading 1, cleared by writing 0 |
| 1 | [Setting condition]<br>When $\overline{CS}$ goes high during a transfer |

**Wait flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading 1, cleared by writing 0 |
| 1 | [Setting condition]<br>An attempt was made to read or write the (32-byte) serial data buffer during a transfer or while waiting for $\overline{CS}$ input |

**Overrun error flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading 1, cleared by writing 0 |
| 1 | [Setting condition]<br>Set if a clock pulse is input after transfer is complete, when an external clock is used |

**Extended data bit**

| | | |
|---|---|---|
| 0 | Read | $SO_2$ pin output level is low |
| | Write | $SO_2$ pin output level changes to low |
| 1 | Read | $SO_2$ pin output level is high |
| | Write | $SO_2$ pin output level changes to high |

Note: * Only a write of 0 for flag clearing is possible.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | COM | CHR | PE | PM | STOP | MP | CKS1 | CKS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Clock select 0, 1**

| | | |
|---|---|---|
| 0 | 0 | $\phi$ clock |
| | 1 | $\phi$/4 clock |
| 1 | 0 | $\phi$/16 clock |
| | 1 | $\phi$/64 clock |

**Multiprocessor mode**

| | |
|---|---|
| 0 | Multiprocessor communication function disabled |
| 1 | Multiprocessor communication function enabled |

**Stop bit length**

| | |
|---|---|
| 0 | 1 stop bit |
| 1 | 2 stop bits |

**Parity mode**

| | |
|---|---|
| 0 | Even parity |
| 1 | Odd parity |

**Parity enable**

| | |
|---|---|
| 0 | Parity bit adding and checking disabled |
| 1 | Parity bit adding and checking enabled |

**Character length**

| | |
|---|---|
| 0 | 8-bit data |
| 1 | 7-bit data |

**Communication mode**

| | |
|---|---|
| 0 | Asynchronous mode |
| 1 | Synchronous mode |



| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | BRR7 | BRR6 | BRR5 | BRR4 | BRR3 | BRR2 | BRR1 | BRR0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TIE | RIE | TE | RE | MPIE | TEIE | CKE1 | CKE0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Clock enable**

| Bit 1 | Bit 0 | Description | | |
|---|---|---|---|---|
| CKE1 | CKE0 | Communication Mode | Clock Source | $SCK_3$ Pin Function |
| 0 | 0 | Asynchronous | Internal clock | I/O port |
| | | Synchronous | Internal clock | Serial clock output |
| | 1 | Asynchronous | Internal clock | Clock output |
| | | Synchronous | Reserved | Reserved (Do not set this combination) |
| 1 | 0 | Asynchronous | External clock | Clock input |
| | | Synchronous | External clock | Serial clock input |
| | 1 | Asynchronous | Reserved | Reserved |
| | | Synchronous | Reserved (Do not set this combination) | Reserved (Do not set this combination) |

**Transmit end interrupt enable**

| | |
|---|---|
| 0 | Transmit end interrupt (TEI) disabled |
| 1 | Transmit end interrupt (TEI) enabled |

**Multiprocessor interrupt enable**

| | |
|---|---|
| 0 | Multiprocessor interrupt request disabled (ordinary receive operation)<br>[Clearing condition]<br>Multiprocessor bit receives a data value of 1 |
| 1 | Multiprocessor interrupt request enabled<br>Until a multiprocessor bit value of 1 is received, the receive data full interrupt (RXI) and receive error interrupt (ERI) are disabled, and serial status register (SSR) flags RDRF, FER, and OER are not set. |

**Receive enable**

| | |
|---|---|
| 0 | Receive operation disabled (RXD is a general I/O port) |
| 1 | Receive operation enabled (RXD is the receive data pin) |

**Transmit enable**

| | |
|---|---|
| 0 | Transmit operation disabled (TXD is a general I/O port) |
| 1 | Transmit operation enabled (TXD is the transmit data pin) |

**Receive interrupt enable**

| | |
|---|---|
| 0 | Receive data full interrupt request (RXI) and receive error interrupt request (ERI) disabled |
| 1 | Receive data full interrupt request (RXI) and receive error interrupt request (ERI) enabled |

**Transmit interrupt enable**

| | |
|---|---|
| 0 | Transmit data empty interrupt request (TXI) disabled |
| 1 | Transmit data empty interrupt request (TXI) enabled |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TDR7 | TDR6 | TDR5 | TDR4 | TDR3 | TDR2 | TDR1 | TDR0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Data to be transferred to TSR

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TDRE | RDRF | OER | FER | PER | TEND | MPBR | MPBT |
| Initial value | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Read/Write | R/(W)* | R/(W)* | R/(W)* | R/(W)* | R/(W)* | R | R | R/W |

**Multiprocessor bit receive**

| | |
|---|---|
| 0 | Indicates reception of data in which the multiprocessor bit is 0 |
| 1 | Indicates reception of data in which the multiprocessor bit is 1 |

**Multiprocessor bit transmit**

| | |
|---|---|
| 0 | The multiprocessor bit in transmit data is 0 |
| 1 | The multiprocessor bit in transmit data is 1 |

**Transmit end**

| | |
|---|---|
| 0 | Indicates that transmission is in progress<br>[Clearing conditions] After reading TDRE = 1, cleared by writing 0 to TDRE.<br>When data is written to TDR by an instruction. |
| 1 | Indicates that a transmission has ended<br>[Setting conditions] When bit TE in serial control register 3 (SCR3) is 0.<br>If TDRE is set to 1 when the last bit of a transmitted character is sent. |

**Parity error**

| | |
|---|---|
| 0 | Indicates that data receiving is in progress or has been completed<br>[Clearing conditions] After reading PER = 1, cleared by writing 0 |
| 1 | Indicates that a parity error occurred in data receiving<br>[Setting conditions] When the sum of 1s in received data plus the parity bit does not match the parity mode bit (PM) setting in the serial mode register (SMR) |

**Framing error**

| | |
|---|---|
| 0 | Indicates that data receiving is in progress or has been completed<br>[Clearing conditions] After reading FER = 1, cleared by writing 0 |
| 1 | Indicates that a framing error occurred in data receiving<br>[Setting conditions] The stop bit at the end of receive data is checked and found to be 0 |

**Overrun error**

| | |
|---|---|
| 0 | Indicates that data receiving is in progress or has been completed<br>[Clearing conditions] After reading OER = 1, cleared by writing 0 |
| 1 | Indicates that an overrun error occurred in data receiving<br>[Setting conditions] When data receiving is completed while RDRF is set to 1 |

**Receive data register full**

| | |
|---|---|
| 0 | Indicates there is no receive data in RDR<br>[Clearing conditions] After reading RDRF = 1, cleared by writing 0.<br>When data is read from RDR by an instruction. |
| 1 | Indicates that there is receive data in RDR<br>[Setting conditions] When receiving ends normally, with receive data transferred from RSR to RDR |

**Transmit data register empty**

| | |
|---|---|
| 0 | Indicates that transmit data written to TDR has not been transferred to TSR<br>[Clearing conditions] After reading TDRE = 1, cleared by writing 0.<br>When data is written to TDR by an instruction. |
| 1 | Indicates that no transmit data has been written to TDR, or the transmit data written to TDR has been transferred to TSR<br>[Setting conditions] When bit TE in serial control register 3 (SCR3) is 0.<br>When data is transferred from TDR to TSR. |

Note: * Only a write of 0 for flag clearing is possible.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | RDR7 | RDR6 | RDR5 | RDR4 | RDR3 | RDR2 | RDR1 | RDR0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

**TMA—Timer mode register A** **H'B0** **Timer A**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TMA7 | TMA6 | TMA5 | — | TMA3 | TMA2 | TMA1 | TMA0 |
| Initial value | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | — | R/W | R/W | R/W | R/W |

**Clock output select**

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | $\phi$/32 |
| | | 1 | $\phi$/16 |
| | 1 | 0 | $\phi$/8 |
| | | 1 | $\phi$/4 |
| 1 | 0 | 0 | $\phi_W$/32 |
| | | 1 | $\phi_W$/16 |
| | 1 | 0 | $\phi_W$/8 |
| | | 1 | $\phi_W$/4 |

**Internal clock select**

| TMA3 | TMA2 | TMA1 | TMA0 | Prescaler and Divider Ratio or Overflow Period | | Function |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | PSS | $\phi$/8192 | Interval timer |
| | | | 1 | PSS | $\phi$/4096 | |
| | | 1 | 0 | PSS | $\phi$/2048 | |
| | | | 1 | PSS | $\phi$/512 | |
| | 1 | 0 | 0 | PSS | $\phi$/256 | |
| | | | 1 | PSS | $\phi$/128 | |
| | | 1 | 0 | PSS | $\phi$/32 | |
| | | | 1 | PSS | $\phi$/8 | |
| 1 | 0 | 0 | 0 | PSW | 1 s | Time base |
| | | | 1 | PSW | 0.5 s | |
| | | 1 | 0 | PSW | 0.25 s | |
| | | | 1 | PSW | 0.03125 s | |
| | 1 | 0 | 0 | PSW and TCA are reset | | |
| | | | 1 | | | |
| | | 1 | 0 | | | |
| | | | 1 | | | |

**TCA—Timer counter A** **H'B1** **Timer A**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCA7 | TCA6 | TCA5 | TCA4 | TCA3 | TCA2 | TCA1 | TCA0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

Count value

**TMB—Timer mode register B** **H'B2** **Timer B**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TMB7 | — | — | — | — | TMB2 | TMB1 | TMB0 |
| Initial value | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | R/W | — | — | — | — | R/W | R/W | R/W |

**Auto-reload function select**

| | |
|---|---|
| 0 | Interval timer function selected |
| 1 | Auto-reload function selected |

**Clock select**

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock: $\phi$/8192 |
| | | 1 | Internal clock: $\phi$/2048 |
| | 1 | 0 | Internal clock: $\phi$/512 |
| | | 1 | Internal clock: $\phi$/256 |
| 1 | 0 | 0 | Internal clock: $\phi$/64 |
| | | 1 | Internal clock: $\phi$/16 |
| | 1 | 0 | Internal clock: $\phi$/4 |
| | | 1 | External event (TMIB): Rising or falling edge |

**TCB—Timer counter B** **H'B3** **Timer B**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCB7 | TCB6 | TCB5 | TCB4 | TCB3 | TCB2 | TCB1 | TCB0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

Count value

**TLB—Timer load register B** **H'B3** **Timer B**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TLB7 | TLB6 | TLB5 | TLB4 | TLB3 | TLB2 | TLB1 | TLB0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

Reload value

## TMC—Timer mode register C

H'B4

Timer C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TMC7 | TMC6 | TMC5 | — | — | TMC2 | TMC1 | TMC0 |
| Initial value | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | — | — | R/W | R/W | R/W |

**Auto-reload function select**

| | |
|---|---|
| 0 | Interval timer function selected |
| 1 | Auto-reload function selected |

**Clock select**

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | Internal clock: $\phi$/8192 |
| | | 1 | Internal clock: $\phi$/2048 |
| | 1 | 0 | Internal clock: $\phi$/512 |
| | | 1 | Internal clock: $\phi$/64 |
| 1 | 0 | 0 | Internal clock: $\phi$/16 |
| | | 1 | Internal clock: $\phi$/4 |
| | 1 | 0 | Internal clock: $\phi_W$/4 |
| | | 1 | External event (TMIC): Rising or falling edge |

**Counter up/down control**

| | | |
|---|---|---|
| 0 | 0 | TCC is an up-counter |
| | 1 | TCC is a down-counter |
| 1 | * | TCC up/down operation is hardware-controlled by input at the UD pin. TCC is a down-counter if the UD input is high, and an up-counter if the UD input is low. |

Note: * Don't care

## TCC—Timer counter C

H'B5

Timer C

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCC7 | TCC6 | TCC5 | TCC4 | TCC3 | TCC2 | TCC1 | TCC0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

Count value

**TLC—Timer load register C** **H'B5** **Timer C**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TLC7 | TLC6 | TLC5 | TLC4 | TLC3 | TLC2 | TLC1 | TLC0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

Reload value

**TCRF—Timer control register F** **H'B6** **Timer F**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TOLH | CKSH2 | CKSH1 | CKSH0 | TOLL | CKSL2 | CKSL1 | CKSL0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Toggle output level H**

| | |
|---|---|
| 0 | Low level |
| 1 | High level |

**Clock select L**

| | | | |
|---|---|---|---|
| 0 | * | * | External event (TMIF): Rising or falling edge |
| 1 | 0 | 0 | Internal clock: $\phi$/32 |
| | | 1 | Internal clock: $\phi$/16 |
| | 1 | 0 | Internal clock: $\phi$/4 |
| | | 1 | Internal clock: $\phi$/2 |

**Toggle output level L**

| | |
|---|---|
| 0 | Low level |
| 1 | High level |

**Clock select H**

| | | | |
|---|---|---|---|
| 0 | * | * | 16-bit mode selected. TCFL overflow signals are counted. |
| 1 | 0 | 0 | Internal clock: $\phi$/32 |
| | | 1 | Internal clock: $\phi$/16 |
| | 1 | 0 | Internal clock: $\phi$/4 |
| | | 1 | Internal clock: $\phi$/2 |

Note: * Don't care

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVFH | CMFH | OVIEH | CCLRH | OVFL | CMFL | OVIEL | CCLRL |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/(W)* | R/(W)* | R/W | R/W | R/(W)* | R/(W)* | R/W | R/W |

**Timer overflow interrupt enable L**

| | |
|---|---|
| 0 | TCFL overflow interrupt disabled |
| 1 | TCFL overflow interrupt enabled |

**Compare match flag L**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading CMFL = 1, cleared by writing 0 to CMFL |
| 1 | [Setting condition]<br>When the TCFL value matches the OCRFL value |

**Timer overflow flag L**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading OVFL = 1, cleared by writing 0 to OVFL |
| 1 | [Setting condition]<br>When the value of TCFL goes from H'FF to H'00 |

**Counter clear H**

| | |
|---|---|
| 0 | 16-bit mode: TCF clearing by compare match disabled<br>8-bit mode: TCFH clearing by compare match disabled |
| 1 | 16-bit mode: TCF clearing by compare match enabled<br>8-bit mode: TCFH clearing by compare match enabled |

**Timer overflow interrupt enable H**

| | |
|---|---|
| 0 | TCFH overflow interrupt disabled |
| 1 | TCFH overflow interrupt enabled |

**Counter clear L**

| | |
|---|---|
| 0 | TCFL clearing by compare match disabled |
| 1 | TCFL clearing by compare match enabled |

**Compare match flag H**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading CMFH = 1, cleared by writing 0 to CMFH |
| 1 | [Setting condition]<br>When the TCFH value matches the OCRFH value |

**Timer overflow flag H**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading OVFH = 1, cleared by writing 0 to OVFH |
| 1 | [Setting condition]<br>When the value of TCFH goes from H'FF to H'00 |

Note: * Only a write of 0 for flag clearing is possible.

**TCFH—8-bit timer counter FH** **H'B8** **Timer F**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCFH7 | TCFH6 | TCFH5 | TCFH4 | TCFH3 | TCFH2 | TCFH1 | TCFH0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Count value

**TCFL—8-bit timer counter FL** **H'B9** **Timer F**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | TCFL7 | TCFL6 | TCFL5 | TCFL4 | TCFL3 | TCFL2 | TCFL1 | TCFL0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

Count value

**OCRFH—Output compare register FH** **H'BA** **Timer F**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OCRFH7 | OCRFH6 | OCRFH5 | OCRFH4 | OCRFH3 | OCRFH2 | OCRFH1 | OCRFH0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**OCRFL—Output compare register FL** **H'BB** **Timer F**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OCRFL7 | OCRFL6 | OCRFL5 | OCRFL4 | OCRFL3 | OCRFL2 | OCRFL1 | OCRFL0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | OVFH | OVFL | OVIE | IIEGS | CCLR1 | CCLR0 | CKS1 | CKS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/(W)* | R/(W)* | R/W | R/W | R/W | R/W | R/W | R/W |

**Clock select**

| | | |
|---|---|---|
| 0 | 0 | Internal clock: $\phi$/64 |
| | 1 | Internal clock: $\phi$/32 |
| 1 | 0 | Internal clock: $\phi$/2 |
| | 1 | Internal clock: $\phi_W$/2 |

**Counter clear**

| | | |
|---|---|---|
| 0 | 0 | TCG is not cleared |
| | 1 | TCG is cleared at the falling edge of the input capture signal |
| 1 | 0 | TCG is cleared at the rising edge of the input capture signal |
| | 1 | TCG is cleared at both edges of the input capture signal |

**Input capture interrupt edge select**

| | |
|---|---|
| 0 | Interrupts are requested at the rising edge of the input capture signal |
| 1 | Interrupts are requested at the falling edge of the input capture signal |

**Timer overflow interrupt enable**

| | |
|---|---|
| 0 | TCG overflow interrupt disabled |
| 1 | TCG overflow interrupt enabled |

**Timer overflow flag L**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading OVFL = 1, cleared by writing 0 to OVFL |
| 1 | [Setting condition]<br>When the value of TCG goes from H'FF to H'00 |

**Timer overflow flag H**

| | |
|---|---|
| 0 | [Clearing condition]<br>After reading OVFH = 1, cleared by writing 0 to OVFH |
| 1 | [Setting condition]<br>When the value of TCG goes from H'FF to H'00 |

Note: * Only a write of 0 for flag clearing is possible.

**ICRGF—Input capture register GF** **H'BD** **Timer G**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ICRGF7 | ICRGF6 | ICRGF5 | ICRGF4 | ICRGF3 | ICRGF2 | ICRGF1 | ICRGF0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

**ICRGR—Input capture register GR** **H'BE** **Timer G**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ICRGR7 | ICRGR6 | ICRGR5 | ICRGR4 | ICRGR3 | ICRGR2 | ICRGR1 | ICRGR0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R | R | R | R | R | R | R | R |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | DTS1 | DTS0 | CMX | SGX | SGS3 | SGS2 | SGS1 | SGS0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Segment driver select**

| Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | Functions of Pins $SEG_{40}$ to $SEG_1$ | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SGX | SGS3 | SGS2 | SGS1 | SGS0 | $SEG_{40}$ to $SEG_{37}$ | $SEG_{36}$ to $SEG_{33}$ | $SEG_{32}$ to $SEG_{29}$ | $SEG_{28}$ to $SEG_{25}$ | $SEG_{24}$ to $SEG_{21}$ | $SEG_{20}$ to $SEG_{17}$ | $SEG_{16}$ to $SEG_{13}$ | $SEG_{12}$ to $SEG_9$ | $SEG_8$ to $SEG_5$ | $SEG_4$ to $SEG_1$ | Remarks |
| 0 | 0 | 0 | 0 | 0 | Port | Port | Port | Port | Port | Port | Port | Port | Port | Port | (initial value) |
| | | | | 1 | SEG | SEG | Port | Port | Port | Port | Port | Port | Port | Port | |
| | | | 1 | 0 | SEG | SEG | SEG | Port | Port | Port | Port | Port | Port | Port | |
| | | | | 1 | SEG | SEG | SEG | SEG | Port | Port | Port | Port | Port | Port | |
| | | 1 | 0 | 0 | SEG | SEG | SEG | SEG | SEG | Port | Port | Port | Port | Port | |
| | | | | 1 | SEG | SEG | SEG | SEG | SEG | SEG | Port | Port | Port | Port | |
| | | | 1 | 0 | SEG | SEG | SEG | SEG | SEG | SEG | SEG | Port | Port | Port | |
| | | | | 1 | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | Port | Port | |
| | 1 | * | * | 0 | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | Port | |
| | | | | 1 | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | |
| 1 | 0 | 0 | 0 | 0 | External segment expansion | Port | Port | Port | Port | Port | Port | Port | Port | Port | |
| | | | | 1 | External segment expansion | SEG | Port | Port | Port | Port | Port | Port | Port | Port | |
| | | | 1 | 0 | External segment expansion | SEG | SEG | Port | Port | Port | Port | Port | Port | Port | |
| | | | | 1 | External segment expansion | SEG | SEG | SEG | Port | Port | Port | Port | Port | Port | |
| | | 1 | 0 | 0 | External segment expansion | SEG | SEG | SEG | SEG | Port | Port | Port | Port | Port | |
| | | | | 1 | External segment expansion | SEG | SEG | SEG | SEG | SEG | Port | Port | Port | Port | |
| | | | 1 | 0 | External segment expansion | SEG | SEG | SEG | SEG | SEG | SEG | Port | Port | Port | |
| | | | | 1 | External segment expansion | SEG | SEG | SEG | SEG | SEG | SEG | SEG | Port | Port | |
| | 1 | * | * | 0 | External segment expansion | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | Port | |
| | | | | 1 | External segment expansion | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | SEG | |

**Expansion signal select**

| | |
|---|---|
| 0 | Pins $SEG_{40}$ to $SEG_{37}$ |
| 1 | Pins $CL_1$, $CL_2$, DO, and M |

**Duty and common function select**

| Bit 7 | Bit 6 | Bit 5 | | | |
|---|---|---|---|---|---|
| DTS1 | DTS0 | CMX | Duty | Common Driver | Other Uses |
| 0 | 0 | 0 | Static | $COM_1$ | $COM_3$, $COM_2$, and $COM_1$ usable as ports |
| | | 1 | | $COM_4$ to $COM_1$ | $COM_4$, $COM_3$, and $COM_2$ output the same waveform as $COM_1$ |
| 0 | 1 | 0 | 1/2 duty | $COM_2$, $COM_1$ | $COM_4$ and $COM_3$ usable as ports |
| | | 1 | | $COM_4$ to $COM_1$ | $COM_4$ outputs the same waveform as $COM_3$, and $COM_2$ the same waveform as $COM_1$ |
| 1 | 0 | 0 | 1/3 duty | $COM_3$ to $COM_1$ | $COM_4$ usable as port |
| | | 1 | | $COM_4$ to $COM_1$ | $COM_4$ outputs a non-select waveform |
| 1 | 1 | 0 | 1/4 duty | $COM_4$ to $COM_1$ | — |
| | | 1 | | | |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | PSW | ACT | DISP | CKS3 | CKS2 | CKS1 | CKS0 |
| Initial value | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Frame frequency select**

| Bit 3 | Bit 2 | Bit 1 | Bit 0 | | Frame Frequency | |
|---|---|---|---|---|---|---|
| CKS3 | CKS2 | CKS1 | CKS0 | Clock | $\phi$ = 5 MHz | $\phi$ = 625 Hz |
| 0 | * | 0 | 0 | $\phi_W$ | 128 Hz (initial value) | |
| | | | 1 | $\phi_W$ | 64 Hz | |
| | | 1 | * | $\phi_W/2$ | 32 Hz | |
| 1 | 0 | 0 | 0 | $\phi/2$ | — | 610 Hz |
| | | | 1 | $\phi/4$ | — | 305 Hz |
| | | 1 | 0 | $\phi/8$ | — | 153 Hz |
| | | | 1 | $\phi/16$ | 610 Hz | 76.3 Hz |
| | 1 | 0 | 0 | $\phi/32$ | 305 Hz | 38.1 Hz |
| | | | 1 | $\phi/64$ | 153 Hz | — |
| | | 1 | 0 | $\phi/128$ | 76.3 Hz | — |
| | | | 1 | $\phi/256$ | 38.1 Hz | — |

**Display data control**

| | |
|---|---|
| 0 | Blank data displayed |
| 1 | LCD RAM data displayed |

**Display active**

| | |
|---|---|
| 0 | LCD controller/driver operation stopped |
| 1 | LCD controller/driver operational |

**Power switch**

| | |
|---|---|
| 0 | LCD power supply resistive voltage divider off |
| 1 | LCD power supply resistive voltage divider on |

Note: *Don’t care

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CKS | TRGE | — | — | CH3 | CH2 | CH1 | CH0 |
| Initial value | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | — | — | R/W | R/W | R/W | R/W |

**Channel select**

| Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|---|---|---|---|---|
| CH3 | CH2 | CH1 | CH0 | Analog input channel |
| 0 | 0 | * | * | No channel selected |
| | 1 | 0 | 0 | $AN_0$ |
| | | | 1 | $AN_1$ |
| | | 1 | 0 | $AN_2$ |
| | | | 1 | $AN_3$ |
| 1 | 0 | 0 | 0 | $AN_4$ |
| | | | 1 | $AN_5$ |
| | | 1 | 0 | $AN_6$ |
| | | | 1 | $AN_7$ |
| | 1 | 0 | 0 | $AN_8$ |
| | | | 1 | $AN_9$ |
| | | 1 | 0 | $AN_{10}$ |
| | | | 1 | $AN_{11}$ |

**External trigger select**

| | |
|---|---|
| 0 | Disables start of A/D conversion by external trigger |
| 1 | Enables start of A/D conversion by rising or falling edge of external trigger at pin $\overline{\text{ADTRG}}$ |

**Clock select**

| Bit 7 | | Conversion Time | |
|---|---|---|---|
| CKS | Conversion Period | $\phi$ = 2 MHz | $\phi$ = 5 MHz |
| 0 | 62/$\phi$ | 31 µs | 12.4 µs |
| 1 | 31/$\phi$ | 15.5 µs | —*1 |

Notes: * Don't care
1. Operation is not guaranteed if the conversion time is less than 12.4 µs. Set bit 7 for a value of at least 12.4 µs.

**ADRR—A/D result register** **H'C5** **A/D converter**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADR7 | ADR6 | ADR5 | ADR4 | ADR3 | ADR2 | ADR1 | ADR0 |
| Initial value | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed | Not fixed |
| Read/Write | R | R | R | R | R | R | R | R |

A/D conversion result

**ADSR—A/D start register** **H'C6** **A/D converter**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | ADSF | — | — | — | — | — | — | — |
| Initial value | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Read/Write | R/W | — | — | — | — | — | — | — |

**A/D status flag**

| | | |
|---|---|---|
| 0 | Read | Indicates the completion of A/D conversion |
| | Write | Stops A/D conversion |
| 1 | Read | Indicates A/D conversion in progress |
| | Write | Starts A/D conversion |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IRQ3 | IRQ2 | IRQ1 | PWM | TMIG | TMOFH | TMOFL | TMOW |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**$P1_0$/TMOW pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_0$ I/O pin |
| 1 | Functions as TMOW output pin |

**$P1_1$/TMOFL pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_1$ I/O pin |
| 1 | Functions as TMOFL output pin |

**$P1_2$/TMOFH pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_2$ I/O pin |
| 1 | Functions as TMOFH output pin |

**$P1_3$/TMIG pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_3$ I/O pin |
| 1 | Functions as TMIG input pin |

**$P1_4$/PWM pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_4$ I/O pin |
| 1 | Functions as PWM output pin |

**$P1_5$/$\overline{IRQ_1}$/TMIB pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_5$ I/O pin |
| 1 | Functions as $\overline{IRQ_1}$/TMIB input pin |

**$P1_6$/$\overline{IRQ_2}$/TMIC pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_6$ I/O pin |
| 1 | Functions as $\overline{IRQ_2}$/TMIC input pin |

**$P1_7$/$\overline{IRQ_3}$/TMIF pin function switch**

| | |
|---|---|
| 0 | Functions as $P1_7$ I/O pin |
| 1 | Functions as $\overline{IRQ_3}$/TMIF input pin |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | POF2 | NCS | IRQ0 | POF1 | UD | IRQ4 |
| Initial value | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | R/W | R/W | R/W | R/W | R/W | R/W |

**$P2_0$/$\overline{IRQ_4}$/$\overline{ADTRG}$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P2_0$ I/O pin |
| 1 | Functions as $\overline{IRQ_4}$/$\overline{ADTRG}$ input pin |

**$P2_1$/UD pin function switch**

| | |
|---|---|
| 0 | Functions as $P2_1$ I/O pin |
| 1 | Functions as UD input pin |

**$P3_2$/$SO_1$ pin PMOS control**

| | |
|---|---|
| 0 | CMOS output |
| 1 | NMOS open-drain output |

**$P4_3$/$\overline{IRQ_0}$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P4_3$ I/O pin |
| 1 | Functions as $\overline{IRQ_0}$ input pin |

**TMIG noise canceller select**

| | |
|---|---|
| 0 | Noise canceller function not selected |
| 1 | Noise canceller function selected |

**$P3_5$/$SO_2$ pin PMOS control**

| | |
|---|---|
| 0 | CMOS output |
| 1 | NMOS open-drain output |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | CS | STRB | SO2 | SI2 | SCK2 | SO1 | SI1 | $SCK_1$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**$P3_0/SCK_1$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_0$ I/O pin |
| 1 | Functions as $SCK_1$ I/O pin |

**$P3_1/SI_1$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_1$ I/O pin |
| 1 | Functions as $SI_1$ input pin |

**$P3_2/SO_1$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_2$ I/O pin |
| 1 | Functions as $SO_1$ output pin |

**$P3_3/SCK_2$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_3$ I/O pin |
| 1 | Functions as $SCK_2$ I/O pin |

**$P3_4/SI_2$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_4$ I/O pin |
| 1 | Functions as $SI_2$ input pin |

**$P3_5/SO_2$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_5$ I/O pin |
| 1 | Functions as $SO_2$ output pin |

**$P3_6$/STRB pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_6$ I/O pin |
| 1 | Functions as STRB output pin |

**$P3_7/\overline{CS}$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P3_7$ I/O pin |
| 1 | Functions as $\overline{CS}$ input pin |

**PMR4—Port mode register 4** **H'CB** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | NMOD7 | NMOD6 | NMOD5 | NMOD4 | NMOD3 | NMOD2 | NMOD1 | NMOD0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

| | |
|---|---|
| 0 | $P2_n$ has CMOS output |
| 1 | $P2_n$ has NMOS open-drain output |

**PMR5—Port mode register 5** **H'CC** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | WKP7 | WKP6 | WKP5 | WKP4 | WKP3 | WKP2 | WKP1 | WKP0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**$P5_n/\overline{WKP_n}/SEG_{n+1}$ pin function switch**

| | |
|---|---|
| 0 | Functions as $P5_n$ I/O pin |
| 1 | Functions as $\overline{WKP_n}$ input pin |

**RLCTR—LCD RAM relocation register** **H'CF**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | — | — | RLCT1 | RLCT0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 |
| Read/Write | — | — | — | — | — | — | R/W | R/W |

**PWCR—PWM control register** **H'D0** **14-bit PWM**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | — | — | — | PWCR0 |
| Initial value | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| Read/Write | — | — | — | — | — | — | — | W |

**Clock select**

| | |
|---|---|
| 0 | The input clock is $\phi/2$ (t$\phi$* = 2/$\phi$). The conversion period is 16,384/$\phi$, with a minimum modulation width of 1/$\phi$ |
| 1 | The input clock is $\phi/4$ (t$\phi$* = 4/$\phi$). The conversion period is 32,768/$\phi$, with a minimum modulation width of 2/$\phi$ |

Note: *tø: Period of PWM input clock

**PWDRU—PWM data register U** **H'D1** **14-bit PWM**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | PWDRU5 | PWDRU4 | PWDRU3 | PWDRU2 | PWDUR1 | PWDRU0 |
| Initial value | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | W | W | W | W | W | W |

Upper 6 bits of data for generating PWM waveform

**PWDRL—PWM data register L** **H'D2** **14-bit PWM**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | PWDRL7 | PWDRL6 | PWDRL5 | PWDRL4 | PWDRL3 | PWDRL2 | PWDRL1 | PWDRL0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

Lower 8 bits of data for generating PWM waveform

**PDR1—Port data register 1** **H'D4** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P1_7$ | $P1_6$ | $P1_5$ | $P1_4$ | $P1_3$ | $P1_2$ | $P1_1$ | $P1_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR2—Port data register 2** **H'D5** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P2_7$ | $P2_6$ | $P2_5$ | $P2_4$ | $P2_3$ | $P2_2$ | $P2_1$ | $P2_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR3—Port data register 3** **H'D6** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P3_7$ | $P3_6$ | $P3_5$ | $P3_4$ | $P3_3$ | $P3_2$ | $P3_1$ | $P3_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR4—Port data register 4** **H'D7** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $P4_3$ | $P4_2$ | $P4_1$ | $P4_0$ |
| Initial value | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | R | R/W | R/W | R/W |

**PDR5—Port data register 5** **H'D8** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P5_7$ | $P5_6$ | $P5_5$ | $P5_4$ | $P5_3$ | $P5_2$ | $P5_1$ | $P5_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR6—Port data register 6** **H'D9** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P6_7$ | $P6_6$ | $P6_5$ | $P6_4$ | $P6_3$ | $P6_2$ | $P6_1$ | $P6_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR7—Port data register 7** **H'DA** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P7_7$ | $P7_6$ | $P7_5$ | $P7_4$ | $P7_3$ | $P7_2$ | $P7_1$ | $P7_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR8—Port data register 8** **H'DB** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P8_7$ | $P8_6$ | $P8_5$ | $P8_4$ | $P8_3$ | $P8_2$ | $P8_1$ | $P8_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDR9—Port data register 9** **H'DC** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $P9_7$ | $P9_6$ | $P9_5$ | $P9_4$ | $P9_3$ | $P9_2$ | $P9_1$ | $P9_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PDRA—Port data register A** **H'DD** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $PA_3$ | $PA_2$ | $PA_1$ | $PA_0$ |
| Initial value | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | R/W | R/W | R/W | R/W |

**PDRB—Port data register B** **H'DE** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PB_7$ | $PB_6$ | $PB_5$ | $PB_4$ | $PB_3$ | $PB_2$ | $PB_1$ | $PB_0$ |
| Initial value | | | | | | | | |
| Read/Write | R | R | R | R | R | R | R | R |

**PDRC—Port data register C** **H'DF** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $PC_3$ | $PC_2$ | $PC_1$ | $PC_0$ |
| Initial value | | | | | | | | |
| Read/Write | — | — | — | — | R | R | R | R |

**PUCR1—Port pull-up control register 1** **H'E0** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR1_7$ | $PUCR1_6$ | $PUCR1_5$ | $PUCR1_4$ | $PUCR1_3$ | $PUCR1_2$ | $PUCR1_1$ | $PUCR1_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PUCR3—Port pull-up control register 3** **H'E1** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR3_7$ | $PUCR3_6$ | $PUCR3_5$ | $PUCR3_4$ | $PUCR3_3$ | $PUCR3_2$ | $PUCR3_1$ | $PUCR3_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PUCR5—Port pull-up control register 5** **H'E2** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR5_7$ | $PUCR5_6$ | $PUCR5_5$ | $PUCR5_4$ | $PUCR5_3$ | $PUCR5_2$ | $PUCR5_1$ | $PUCR5_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PUCR6—Port pull-up control register 6** **H'E3** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PUCR6_7$ | $PUCR6_6$ | $PUCR6_5$ | $PUCR6_4$ | $PUCR6_3$ | $PUCR6_2$ | $PUCR6_1$ | $PUCR6_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**PCR1—Port control register 1** **H'E4** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR1_7$ | $PCR1_6$ | $PCR1_5$ | $PCR1_4$ | $PCR1_3$ | $PCR1_2$ | $PCR1_1$ | $PCR1_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 1 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR2—Port control register 2** **H'E5** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR2_7$ | $PCR2_6$ | $PCR2_5$ | $PCR2_4$ | $PCR2_3$ | $PCR2_2$ | $PCR2_1$ | $PCR2_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 2 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR3—Port control register 3** **H'E6** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR3_7$ | $PCR3_6$ | $PCR3_5$ | $PCR3_4$ | $PCR3_3$ | $PCR3_2$ | $PCR3_1$ | $PCR3_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 3 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR4—Port control register 4** **H'E7** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | — | $PCR4_2$ | $PCR4_1$ | $PCR4_0$ |
| Initial value | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | — | W | W | W |

**Port 4 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR5—Port control register 5** **H'E8** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR5_7$ | $PCR5_6$ | $PCR5_5$ | $PCR5_4$ | $PCR5_3$ | $PCR5_2$ | $PCR5_1$ | $PCR5_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 5 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR6—Port control register 6** **H'E9** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR6_7$ | $PCR6_6$ | $PCR6_5$ | $PCR6_4$ | $PCR6_3$ | $PCR6_2$ | $PCR6_1$ | $PCR6_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 6 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR7—Port control register 7** **H'EA** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR7_7$ | $PCR7_6$ | $PCR7_5$ | $PCR7_4$ | $PCR7_3$ | $PCR7_2$ | $PCR7_1$ | $PCR7_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 7 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR8—Port control register 8** **H'EB** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR8_7$ | $PCR8_6$ | $PCR8_5$ | $PCR8_4$ | $PCR8_3$ | $PCR8_2$ | $PCR8_1$ | $PCR8_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 8 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCR9—Port control register 9** **H'EC** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | $PCR9_7$ | $PCR9_6$ | $PCR9_5$ | $PCR9_4$ | $PCR9_3$ | $PCR9_2$ | $PCR9_1$ | $PCR9_0$ |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | W | W | W | W | W | W | W | W |

**Port 9 input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

**PCRA—Port control register A** **H'ED** **I/O ports**

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | — | $PCRA_3$ | $PCRA_2$ | $PCRA_1$ | $PCRA_0$ |
| Initial value | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | — | W | W | W | W |

**Port A input/output select**

| | |
|---|---|
| 0 | Input pin |
| 1 | Output pin |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | SSBY | STS2 | STS1 | STS0 | LSON | — | — | — |
| Initial value | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | — | — | — |

**Low speed on flag**

| | |
|---|---|
| 0 | The CPU operates on the system clock ($\phi$) |
| 1 | The CPU operates on the subclock ($\phi_{SUB}$) |

**Standby timer select 2 to 0**

| | | | |
|---|---|---|---|
| 0 | 0 | 0 | Wait time = 8,192 states |
| | | 1 | Wait time = 16,384 states |
| | 1 | 0 | Wait time = 32,768 states |
| | | 1 | Wait time = 65,536 states |
| 1 | * | * | Wait time = 131,072 states |

**Software standby**

| | |
|---|---|
| 0 | When a SLEEP instruction is executed in active mode, a transition is made to sleep mode.<br>When a SLEEP instruction is executed in subactive mode, a transition is made to subsleep mode. |
| 1 | When a SLEEP instruction is executed in active mode, a transition is made to standby mode or watch mode.<br>When a SLEEP instruction is executed in subactive mode, a transition is made to watch mode. |

Note: * Don’t care

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | NESEL | DTON | MSON | SA1 | SA0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

**Medium speed on flag**

| | |
|---|---|
| 0 | Operates in active (high-speed) mode |
| 1 | Operates in active (medium-speed) mode |

**Subactive mode clock select**

| | | |
|---|---|---|
| 0 | 0 | $\phi_W/8$ |
| | 1 | $\phi_W/4$ |
| 1 | * | $\phi_W/2$ |

**Direct transfer on flag**

| | |
|---|---|
| 0 | When a SLEEP instruction is executed in active mode, a transition is made to standby mode, watch mode, or sleep mode.<br>When a SLEEP instruction is executed in subactive mode, a transition is made to watch mode or subsleep mode. |
| 1 | When a SLEEP instruction is executed in active (high-speed) mode, a direct transition is made to active (medium-speed) mode if SSBY = 0, MSON = 1, and LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = 1.<br>When a SLEEP instruction is executed in active (medium-speed) mode, a direct transition is made to active (high-speed) mode if SSBY = 0, MSON = 0, and LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = 1.<br>When a SLEEP instruction is executed in subactive mode, a direct transition is made to active (high-speed) mode if SSBY = 1, TMA3 = 1, LSON = 0, and MSON = 0, or to active (medium-speed) mode if SSBY = 1, TMA3 = 1, LSON = 0, and MSON = 1. |

**Noise elimination sampling frequency select**

| | |
|---|---|
| 0 | Sampling rate is $\phi_{OSC}/16$ |
| 1 | Sampling rate is $\phi_{OSC}/4$ |

Note: * Don't care

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | — | — | — | IEG4 | IEG3 | IEG2 | IEG1 | IEG0 |
| Initial value | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | — | — | — | R/W | R/W | R/W | R/W | R/W |

**$\overline{IRQ_0}$ edge select**

| | |
|---|---|
| 0 | Falling edge of $\overline{IRQ_0}$ pin input is detected |
| 1 | Rising edge of $\overline{IRQ_0}$ pin input is detected |

**$\overline{IRQ_1}$ edge select**

| | |
|---|---|
| 0 | Falling edge of $\overline{IRQ_1}$/TMIB pin input is detected |
| 1 | Rising edge of $\overline{IRQ_1}$/TMIB pin input is detected |

**$\overline{IRQ_2}$ edge select**

| | |
|---|---|
| 0 | Falling edge of $\overline{IRQ_2}$/TMIC pin input is detected |
| 1 | Rising edge of $\overline{IRQ_2}$/TMIC pin input is detected |

**$\overline{IRQ_3}$ edge select**

| | |
|---|---|
| 0 | Falling edge of $\overline{IRQ_3}$/TMIF pin input is detected |
| 1 | Rising edge of $\overline{IRQ_3}$/TMIF pin input is detected |

**$\overline{IRQ_4}$ edge select**

| | |
|---|---|
| 0 | Falling edge of $\overline{IRQ_4}$/$\overline{ADTRG}$ pin input is detected |
| 1 | Rising edge of $\overline{IRQ_4}$/$\overline{ADTRG}$ pin input is detected |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IENTA | IENS1 | IENWP | IEN4 | IEN3 | IEN2 | IEN1 | IEN0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**$IRQ_4$ to $IRQ_0$ interrupt enable**

| | |
|---|---|
| 0 | Disables interrupt request $\overline{IRQ}_n$ |
| 1 | Enables interrupt request $\overline{IRQ}_n$ |

(n = 4 to 0)

**Wakeup interrupt enable**

| | |
|---|---|
| 0 | Disables interrupt requests from $\overline{WKP}_7$ to $\overline{WKP}_0$ |
| 1 | Enables interrupt requests from $\overline{WKP}_7$ to $\overline{WKP}_0$ |

**SCI1 interrupt enable**

| | |
|---|---|
| 0 | Disables SCI1 interrupts |
| 1 | Enables SCI1 interrupts |

**Timer A interrupt enable**

| | |
|---|---|
| 0 | Disables timer A interrupts |
| 1 | Enables timer A interrupts |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IENDT | IENAD | IENS2 | IENTG | IENTFH | IENTFL | IENTC | IENTB |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W |

**Timer B interrupt enable**

| | |
|---|---|
| 0 | Disables timer B interrupts |
| 1 | Enables timer B interrupts |

**Timer C interrupt enable**

| | |
|---|---|
| 0 | Disables timer C interrupts |
| 1 | Enables timer C interrupts |

**Timer FL interrupt enable**

| | |
|---|---|
| 0 | Disables timer FL interrupts |
| 1 | Enables timer FL interrupts |

**Timer FH interrupt enable**

| | |
|---|---|
| 0 | Disables timer FH interrupts |
| 1 | Enables timer FH interrupts |

**Timer G interrupt enable**

| | |
|---|---|
| 0 | Disables timer G interrupts |
| 1 | Enables timer G interrupts |

**SCI2 interrupt enable**

| | |
|---|---|
| 0 | Disables SCI2 interrupts |
| 1 | Enables SCI2 interrupts |

**A/D converter interrupt enable**

| | |
|---|---|
| 0 | Disables A/D converter interrupt requests |
| 1 | Enables A/D converter interrupt requests |

**Direct transfer interrupt enable**

| | |
|---|---|
| 0 | Disables direct transfer interrupt requests |
| 1 | Enables direct transfer interrupt requests |

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IRRTA | IRRS1 | — | IRRI4 | IRRI3 | IRRI2 | IRRI1 | IRRI0 |
| Initial value | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | — | R/W* | R/W* | R/W* | R/W* | R/W* |

**$IRQ_4$ to $IRQ_0$ interrupt request flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>When IRRIn = 1, it is cleared by writing 0 |
| 1 | [Setting condition]<br>When pin $\overline{IRQ}_n$ is set to interrupt input and the designated signal edge is detected |

(n = 4 to 0)

**SCI1 interrupt request flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>When IRRS1 = 1, it is cleared by writing 0 |
| 1 | [Setting condition]<br>When an SCI1 transfer is completed |

**Timer A interrupt request flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>When IRRTA = 1, it is cleared by writing 0 |
| 1 | [Setting condition]<br>When the timer A counter overflows from H'FF to H'00 |

Note: * Only a write of 0 for flag clearing is possible.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IRRDT | IRRAD | IRRS2 | IRRTG | IRRTFH | IRRTFL | IRRTC | IRRTB |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

**Timer B interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRTB = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When the timer B counter overflows from H'FF to H'00 |

**Timer C interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRTC = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When the timer C counter overflows from H'FF to H'00 or underflows from H'00 to H'FF |

**Timer FL interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRTFL = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When counter FL matches output compare register FL in 8-bit mode |

**Timer FH interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRTFH = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When counter FH matches output compare register FH in 8-bit mode, or when 16-bit counter F (TCFL, TCFH) matches 16-bit output compare register F (OCRFL, OCRFH) in 16-bit mode |

**Timer G interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRTG = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When pin TMIG is set to TMIG input and the designated signal edge is detected |

**SCI2 interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRS2 = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When an SCI2 transfer is completed or aborted |

**A/D converter interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRAD = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | When A/D conversion is completed and ADSF is reset |

**Direct transfer interrupt request flag**

| | | |
|---|---|---|
| 0 | [Clearing condition] | When IRRDT = 1, it is cleared by writing 0 |
| 1 | [Setting condition] | A SLEEP instruction is executed when DTON = 1 and a direct transfer is made |

Note: * Only a write of 0 for flag clearing is possible.

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| | IWPF7 | IWPF6 | IWPF5 | IWPF4 | IWPF3 | IWPF2 | IWPF1 | IWPF0 |
| Initial value | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Read/Write | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* | R/W* |

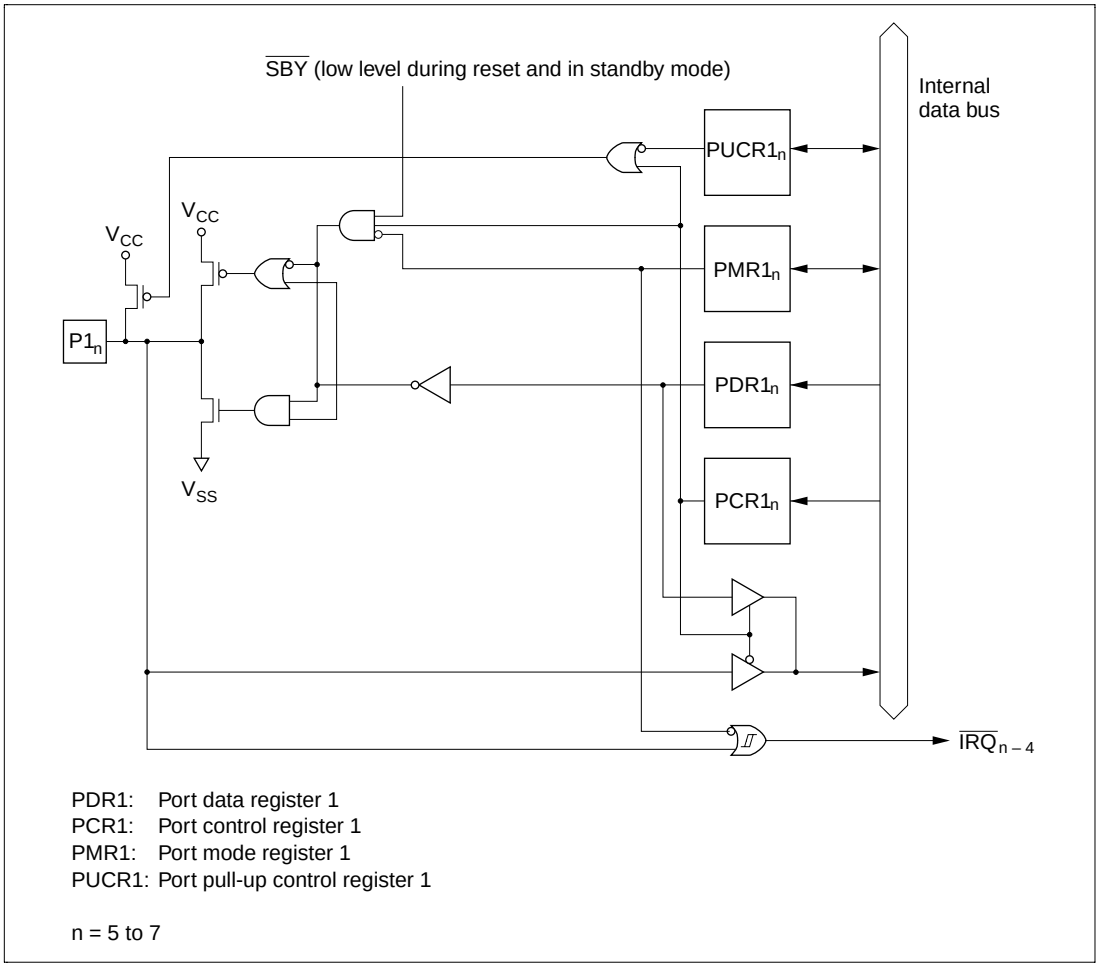
**Wakeup interrupt request flag**

| | |
|---|---|
| 0 | [Clearing condition]<br>When IWPFn = 1, it is cleared by writing 0 |
| 1 | [Setting condition]<br>When pin $\overline{WKP_n}$ is set to interrupt input and a falling signal edge is detected |

(n = 7 to 0)

Note: * Only a write of 0 for flag clearing is possible.

## C.1 Block Diagram of Port 1




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

n = 5 to 7

**Figure C.1 (a) Port 1 Block Diagram (Pins $P1_7$ to $P1_5$)**

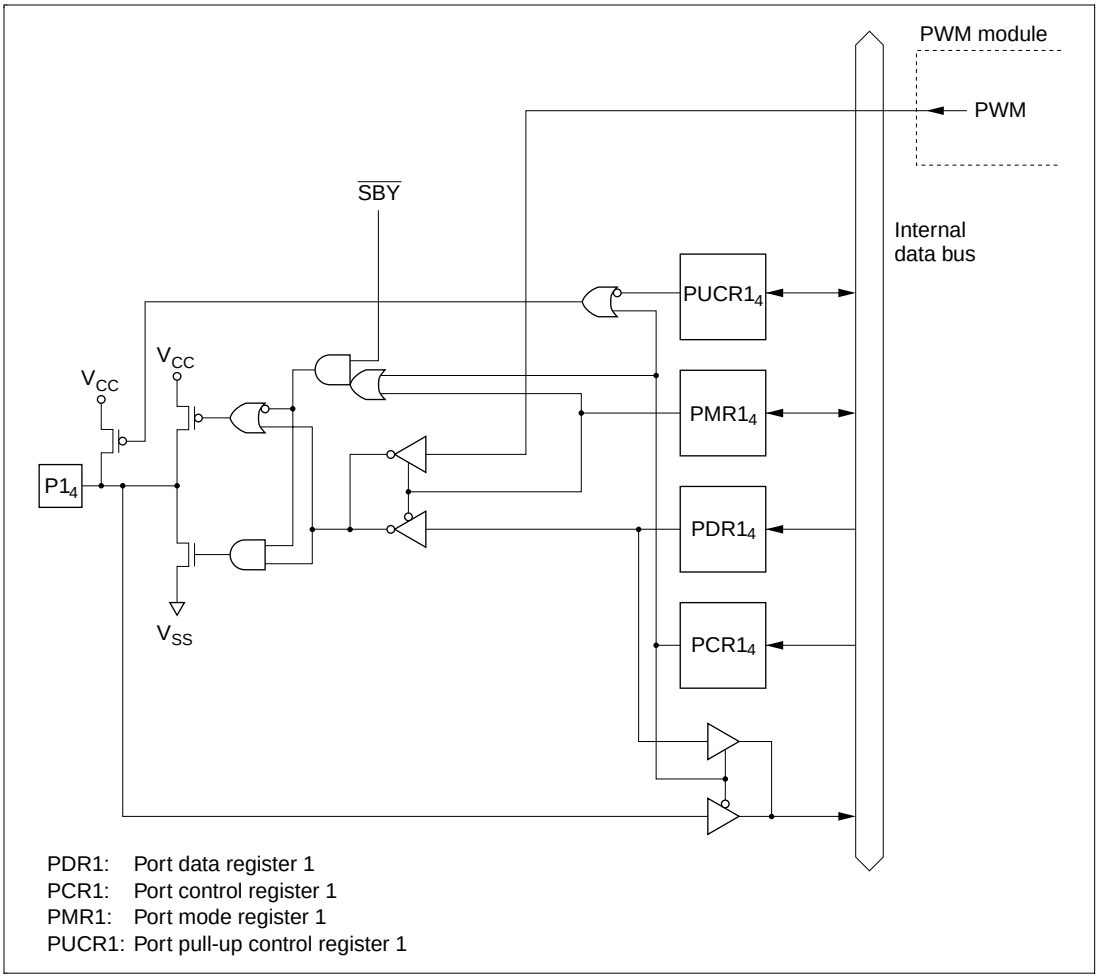



**Figure C.1 (b) Port 1 Block Diagram (Pin $P1_4$)**

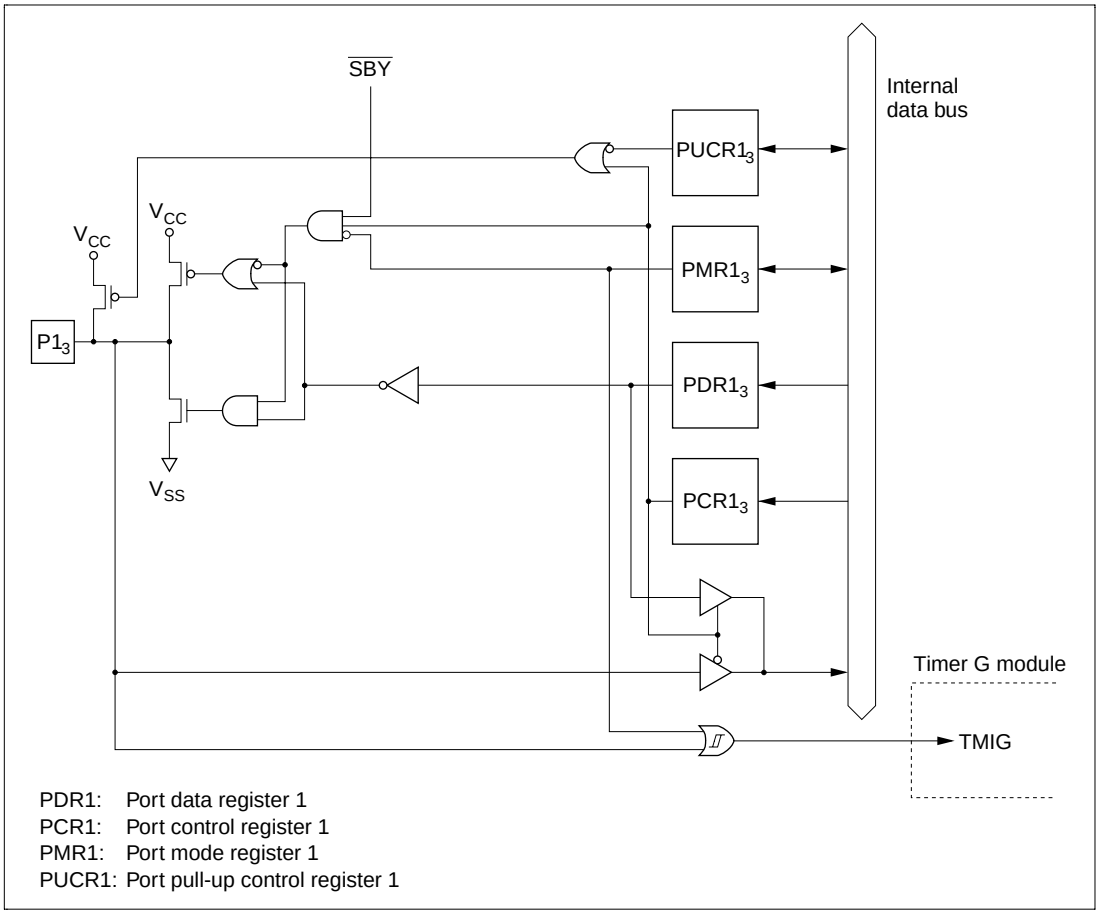



**Figure C.1 (c) Port 1 Block Diagram (Pin $P1_3$)**

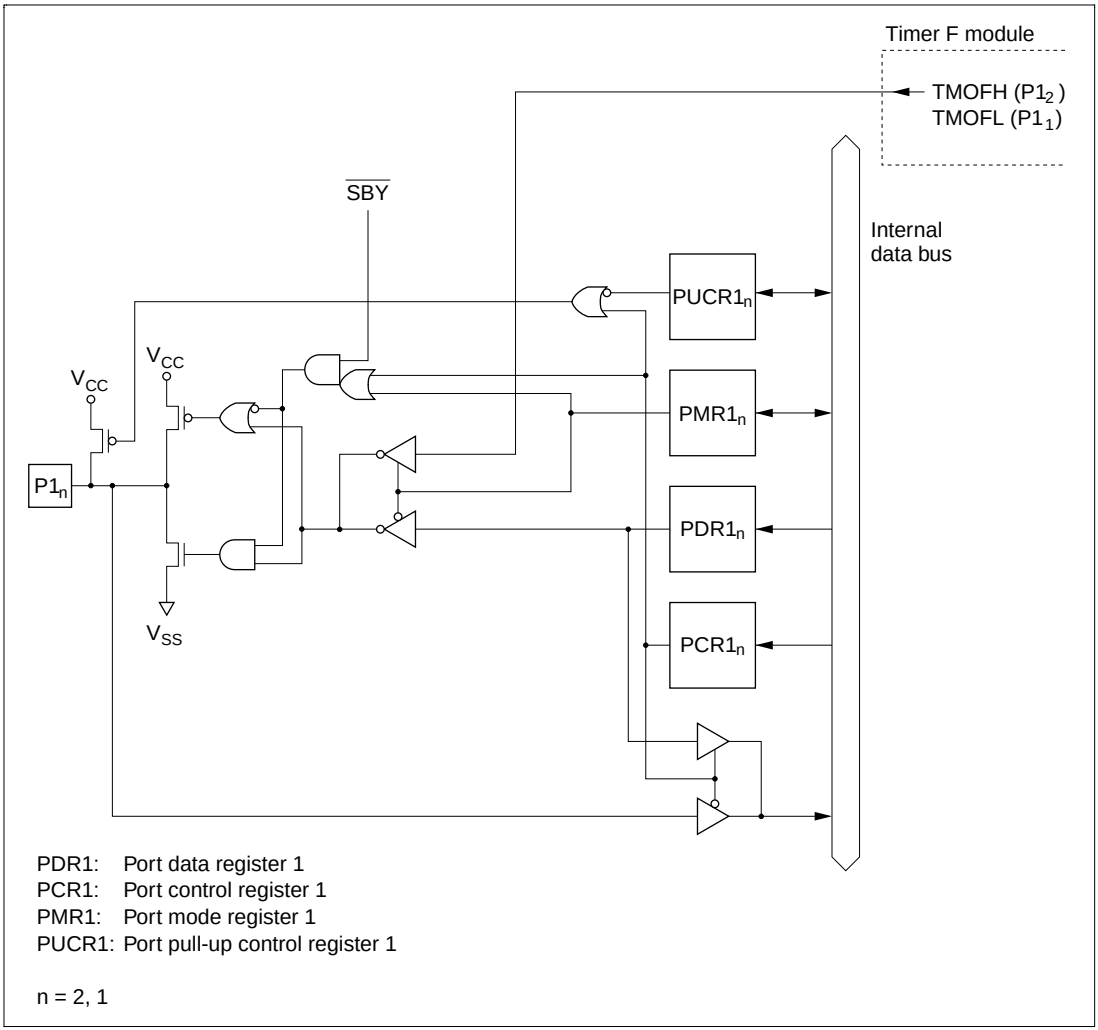



**Figure C.1 (d) Port 1 Block Diagram (Pins $P1_2$ and $P1_1$)**

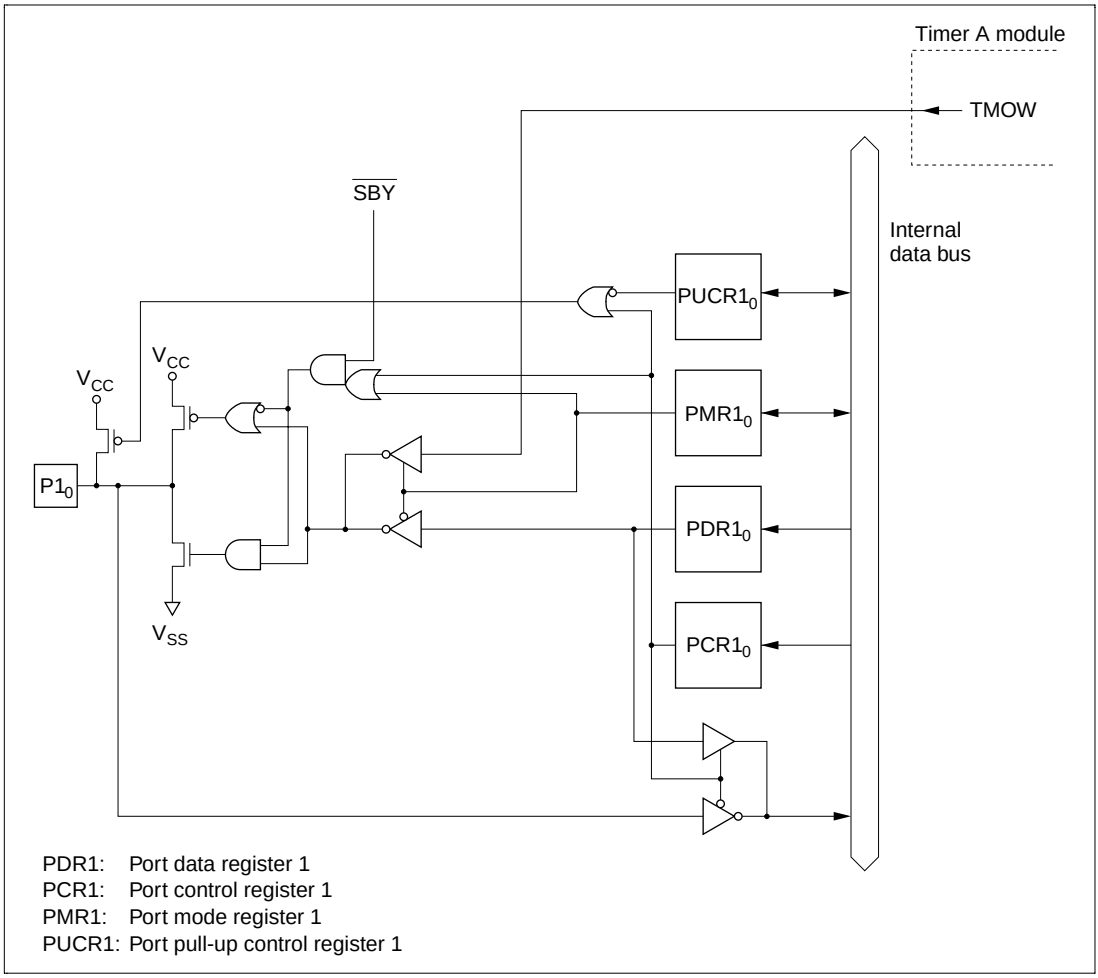



**Figure C.1 (e) Port 1 Block Diagram (Pin $P1_0$)**

## C.2 Block Diagram of Port 2

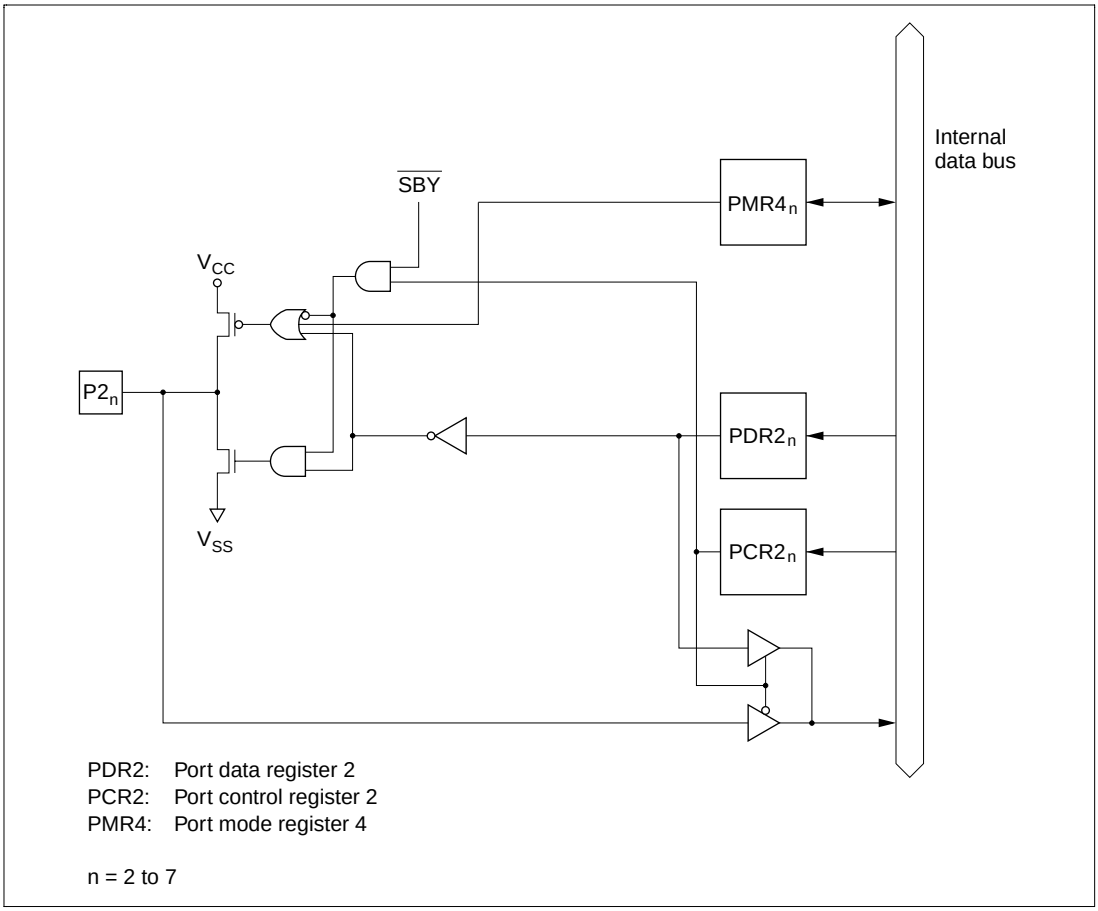



**Figure C.2 (a) Port 2 Block Diagram (Pins $P2_7$ to $P2_2$)**

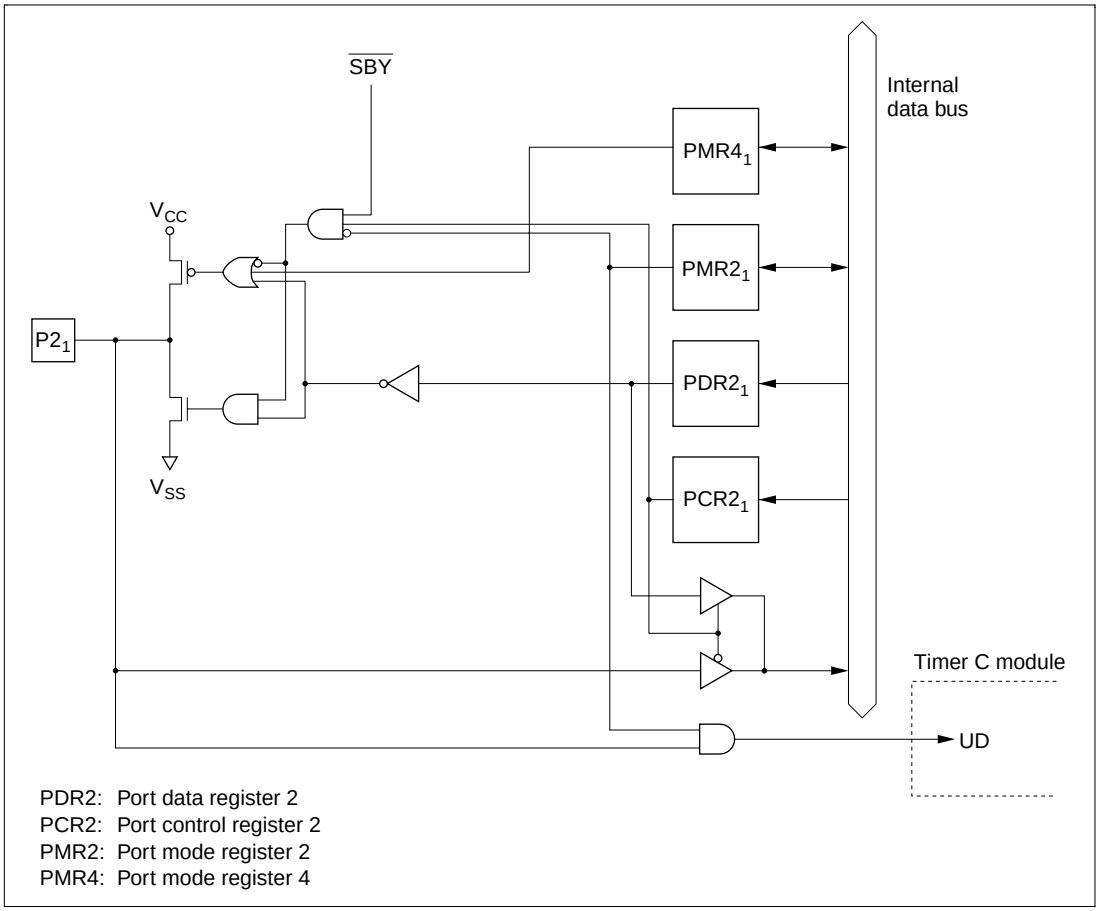
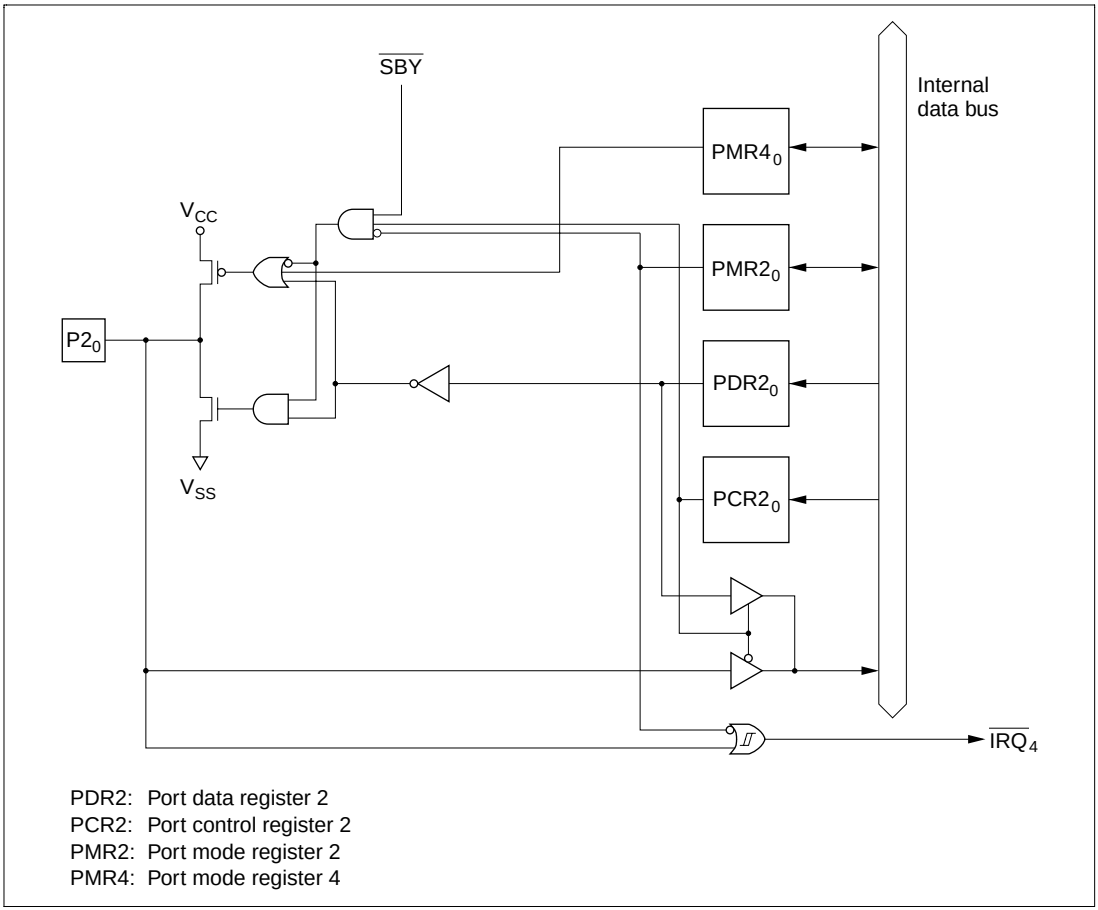



**Figure C.2 (b) Port 2 Block Diagram (Pin $P2_1$)**




PDR2: Port data register 2
PCR2: Port control register 2
PMR2: Port mode register 2
PMR4: Port mode register 4

**Figure C.2 (c) Port 2 Block Diagram (Pin $P2_0$)**

## C.3 Block Diagram of Port 3

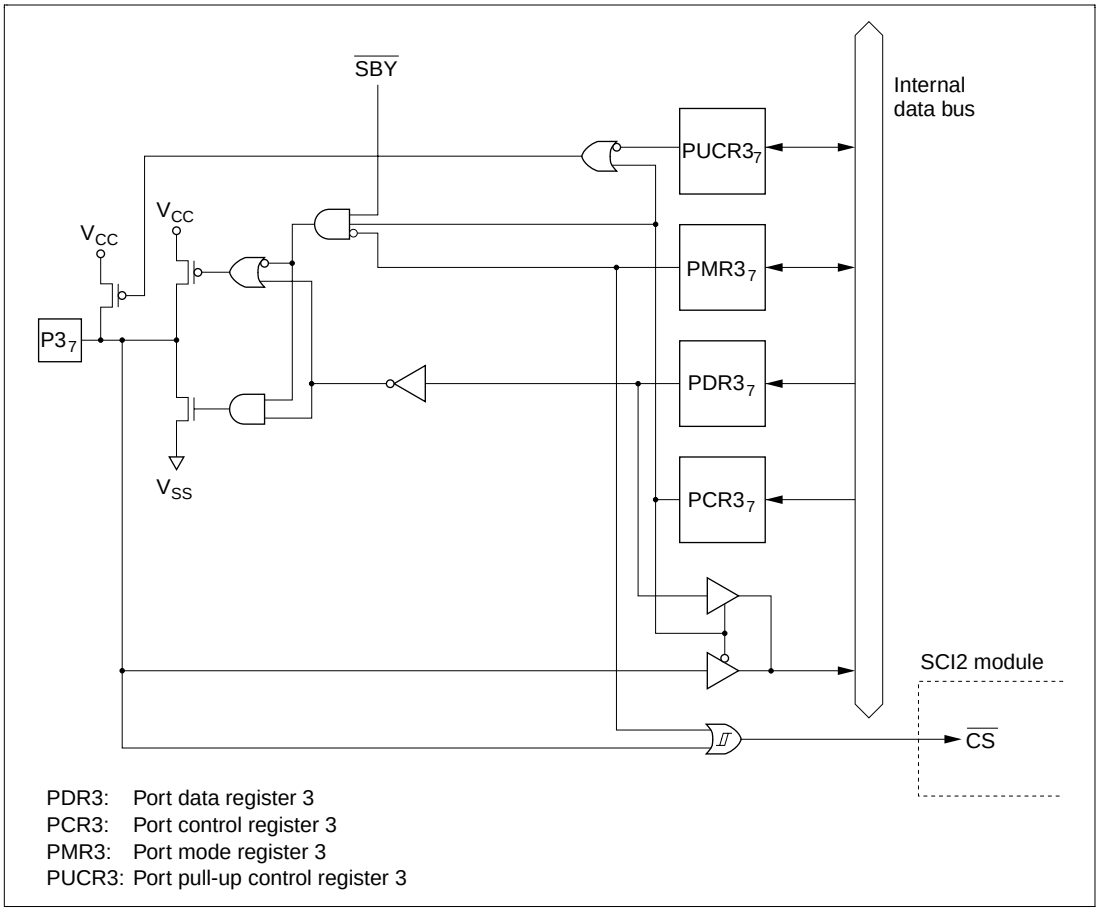



**Figure C.3 (a) Port 3 Block Diagram (Pin P3$_7$)**

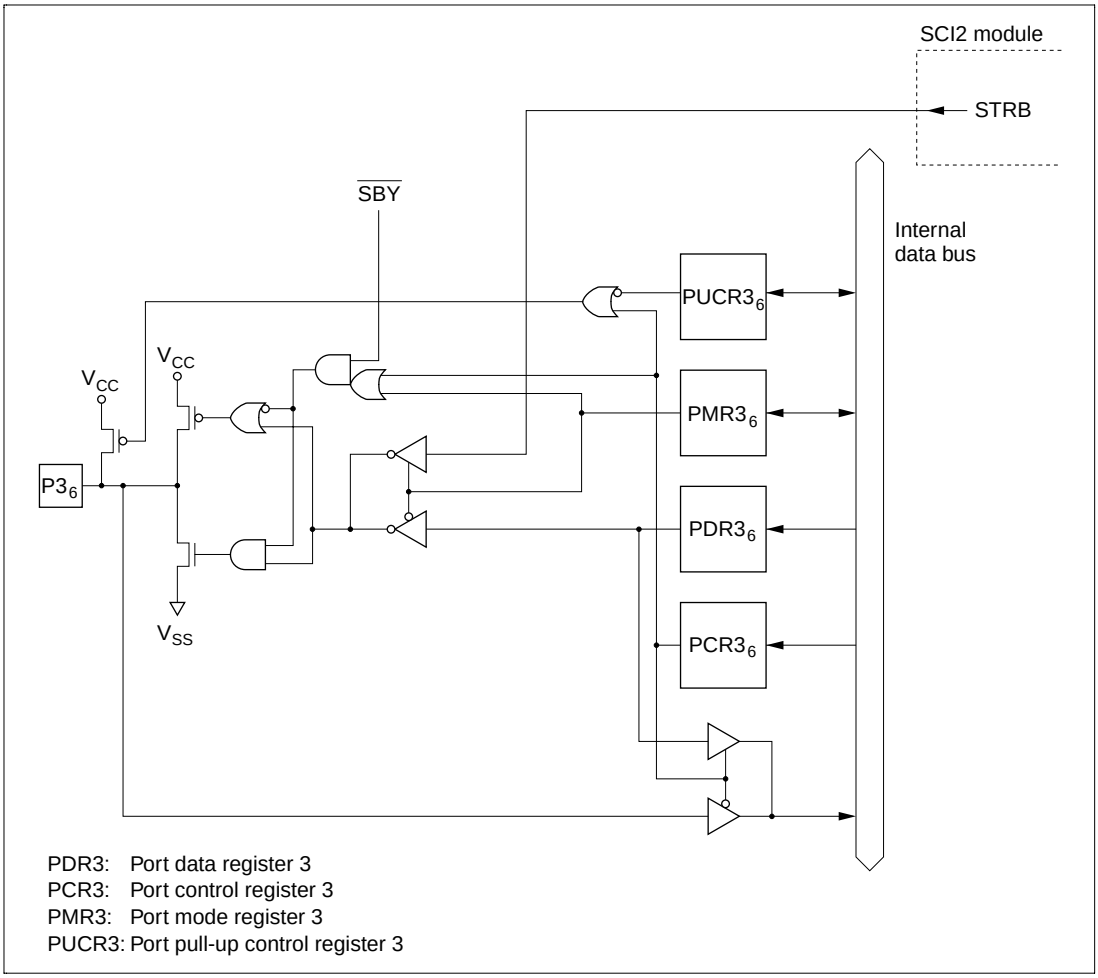



**Figure C.3 (b) Port 3 Block Diagram (Pin $P3_6$)**

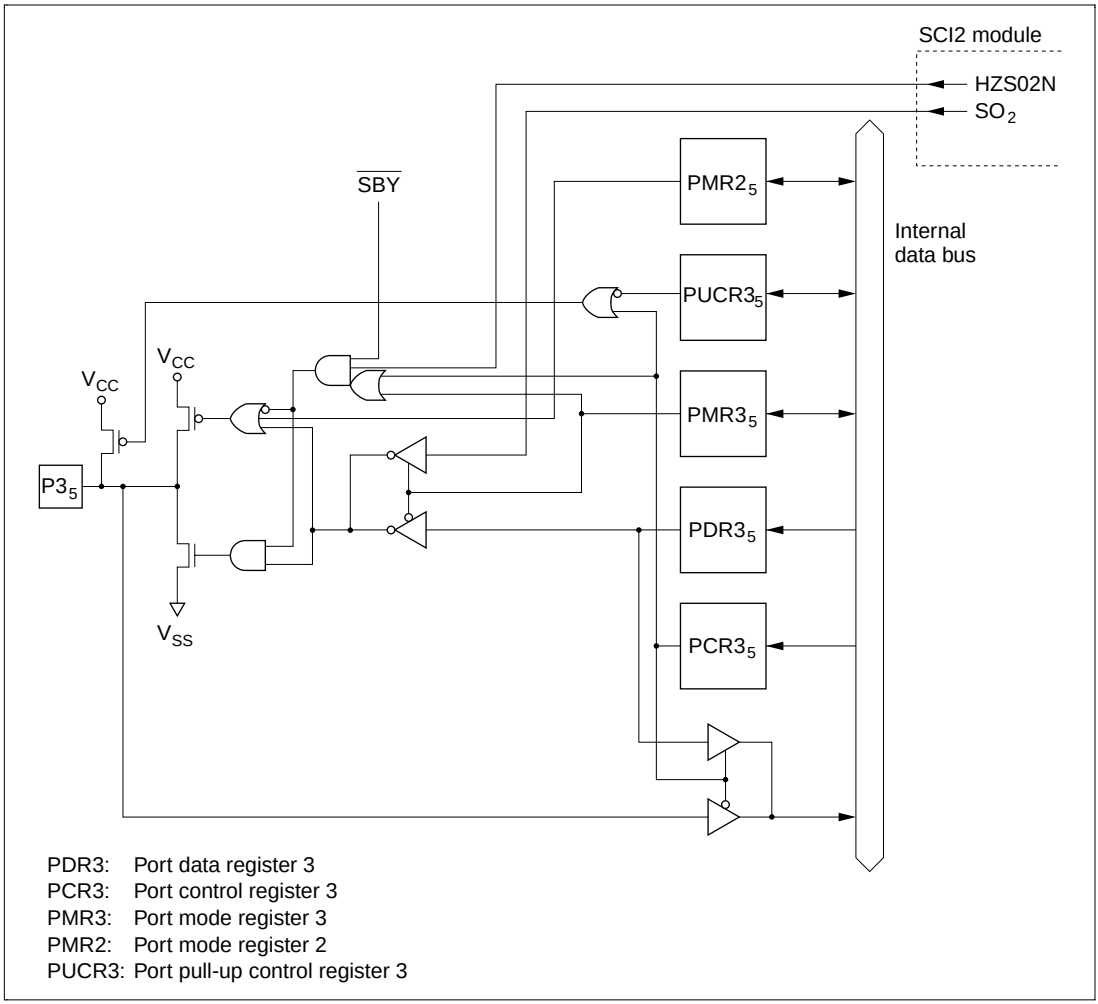



**Figure C.3 (c) Port 3 Block Diagram (Pin $P3_5$)**

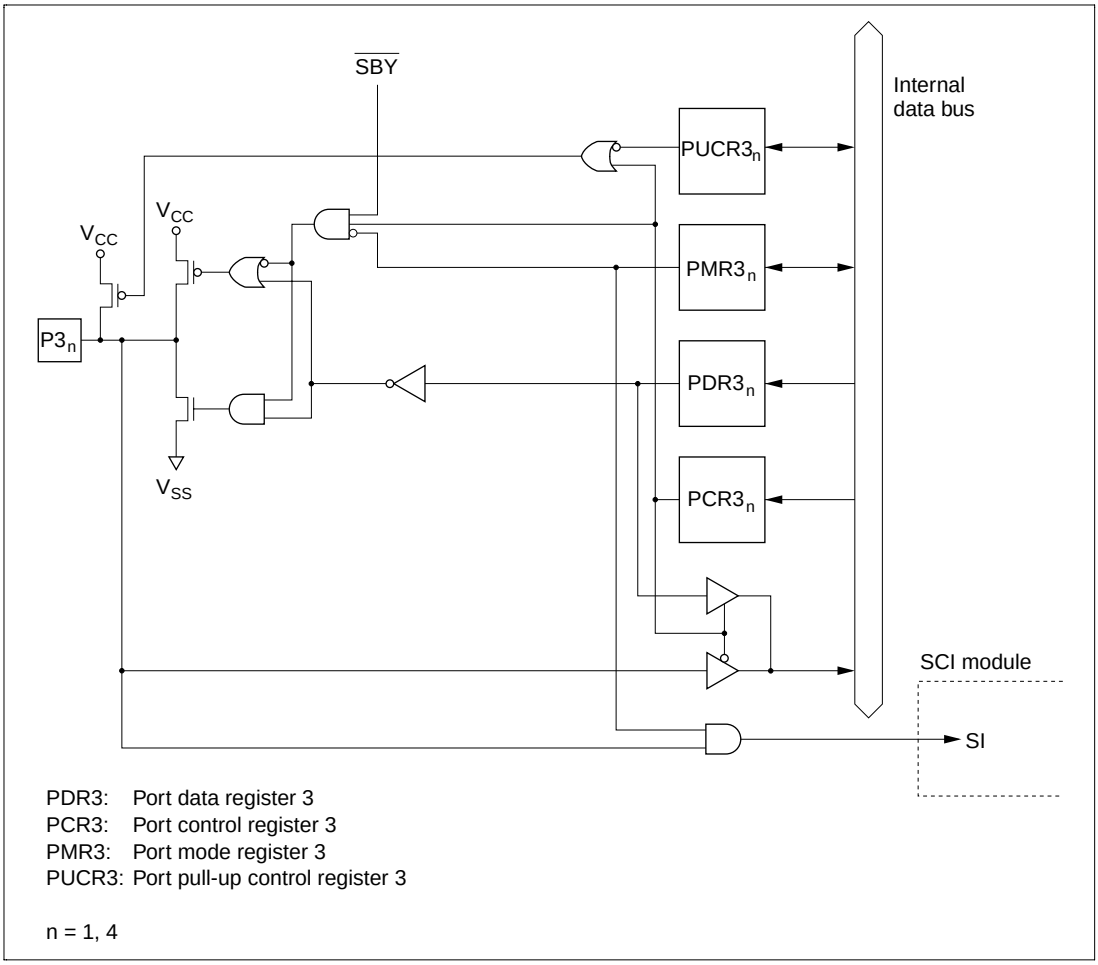
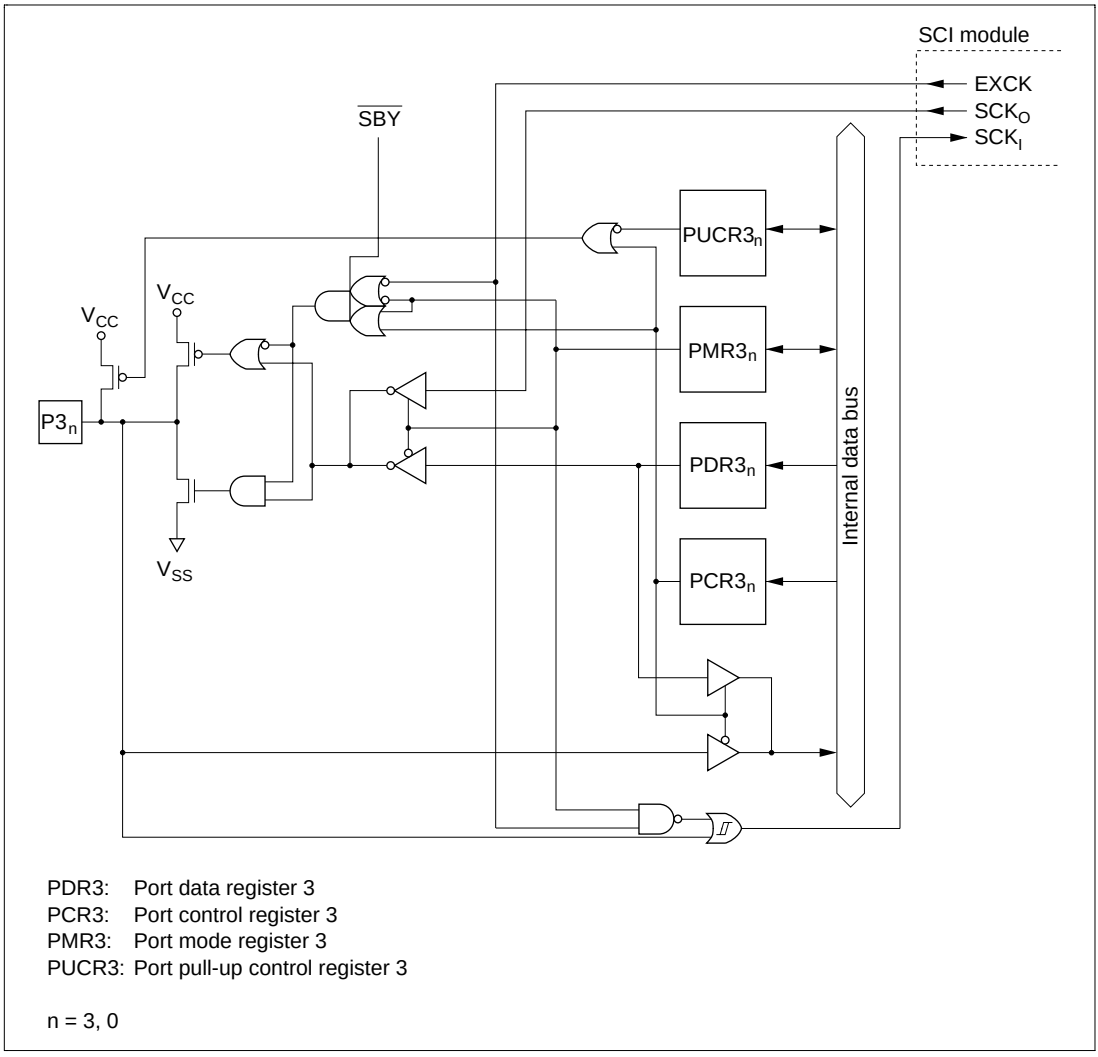



**Figure C.3 (d) Port 3 Block Diagram (Pins $P3_4$ and $P3_1$)**




PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3

n = 3, 0

**Figure C.3 (e) Port 3 Block Diagram (Pins $P3_3$ and $P3_0$)**

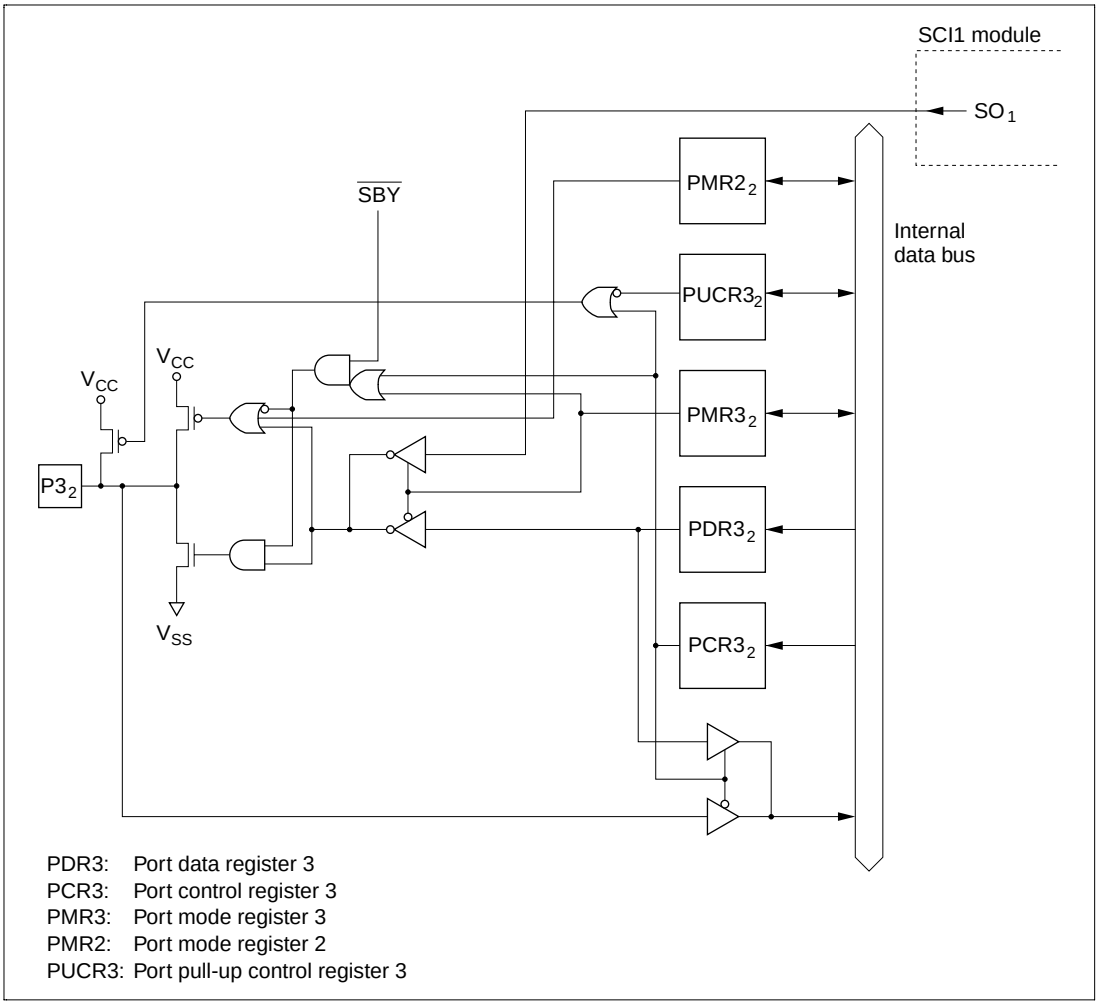



**Figure C.3 (f) Port 3 Block Diagram (Pin $P3_2$)**

## C.4 Block Diagram of Port 4

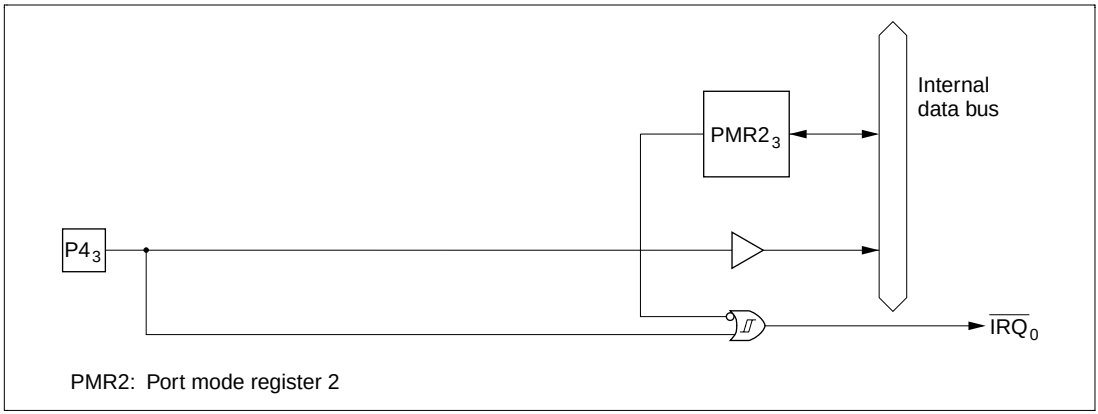



**Figure C.4 (a) Port 4 Block Diagram (Pin P4$_3$)**

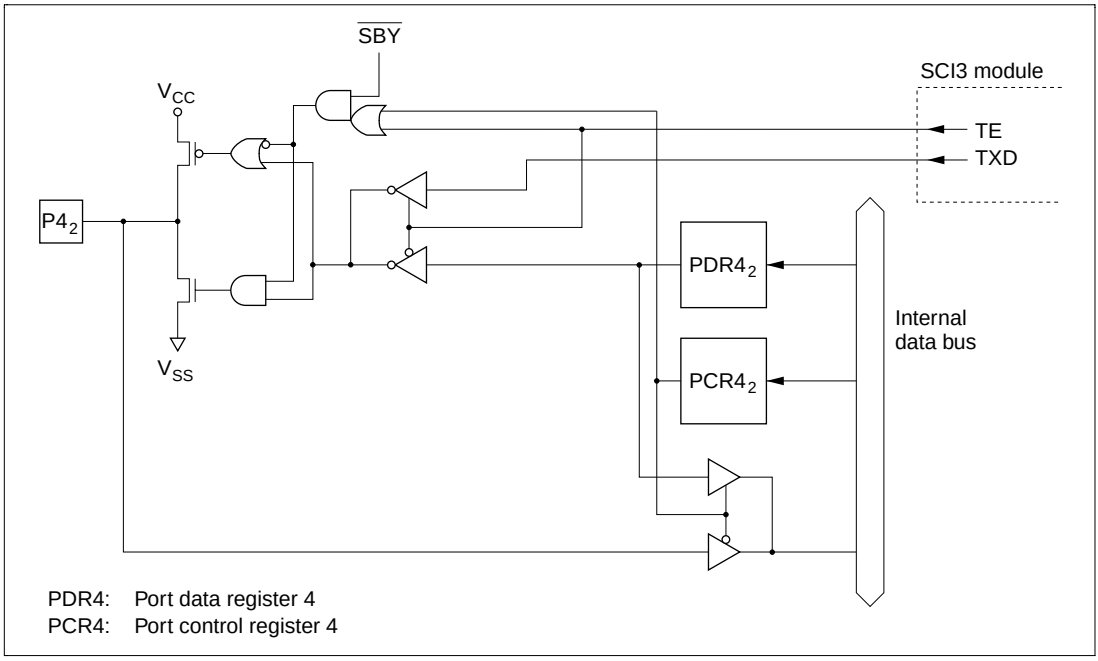



**Figure C.4 (b) Port 4 Block Diagram (Pin P4$_2$)**

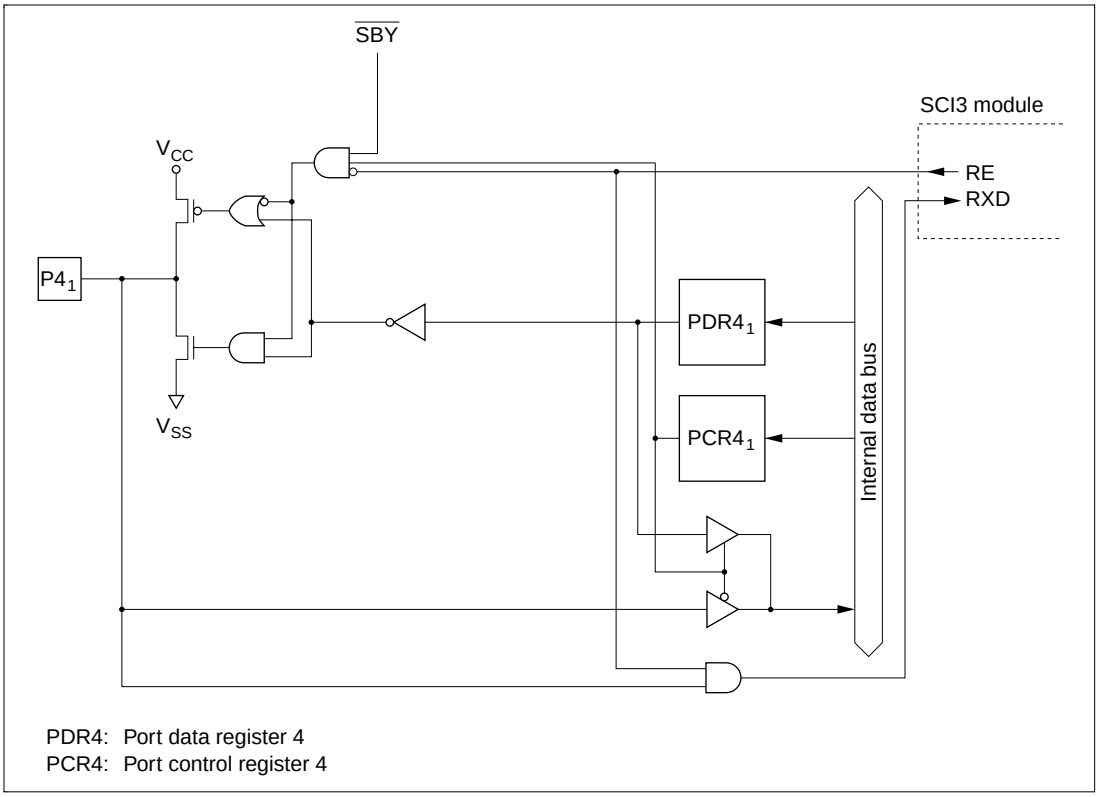
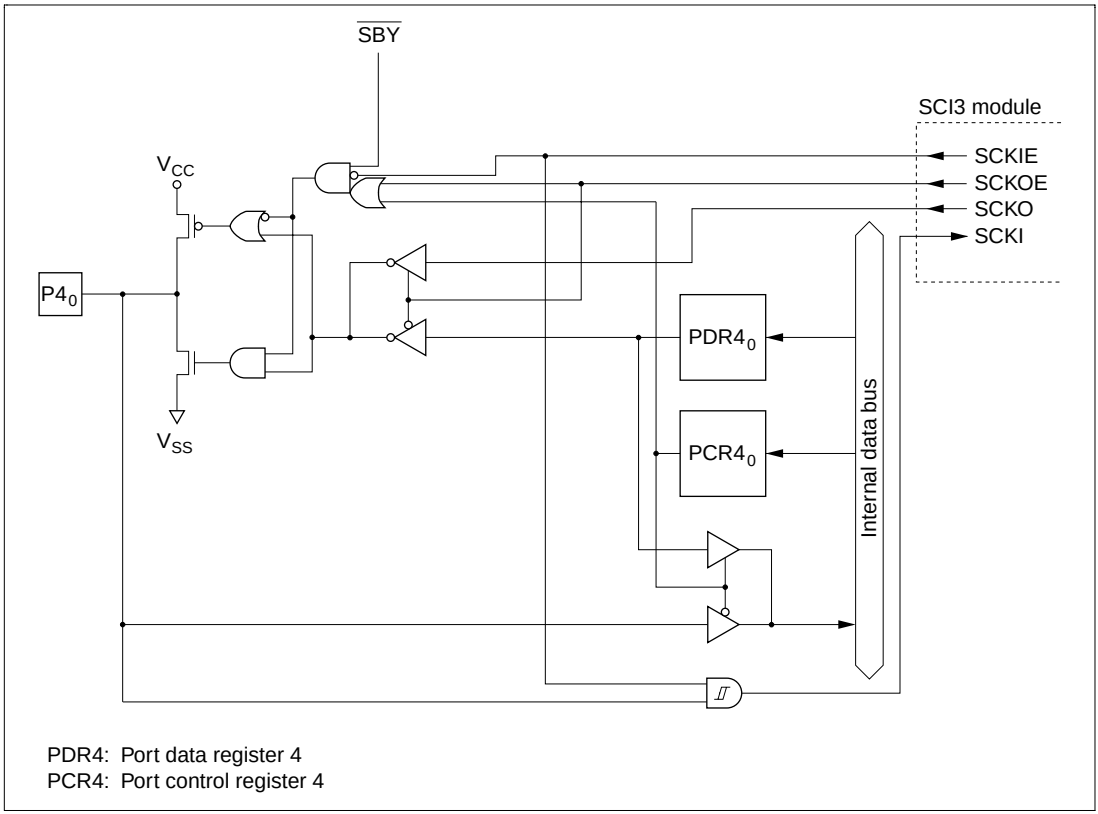


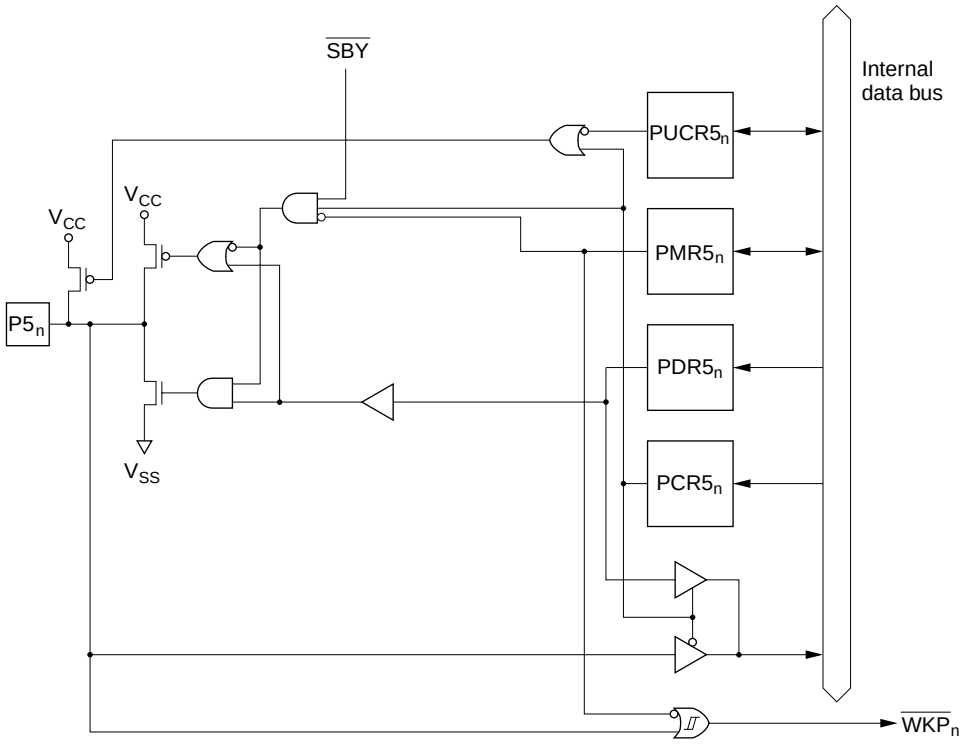

**Figure C.4 (c) Port 4 Block Diagram (Pin $P4_1$)**




PDR4: Port data register 4
PCR4: Port control register 4

**Figure C.4 (d) Port 4 Block Diagram (Pin $P4_0$)**

## C.5 Block Diagram of Port 5




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

n = 0 to 7

**Figure C.5 Port 5 Block Diagram**

## C.6 Block Diagram of Port 6

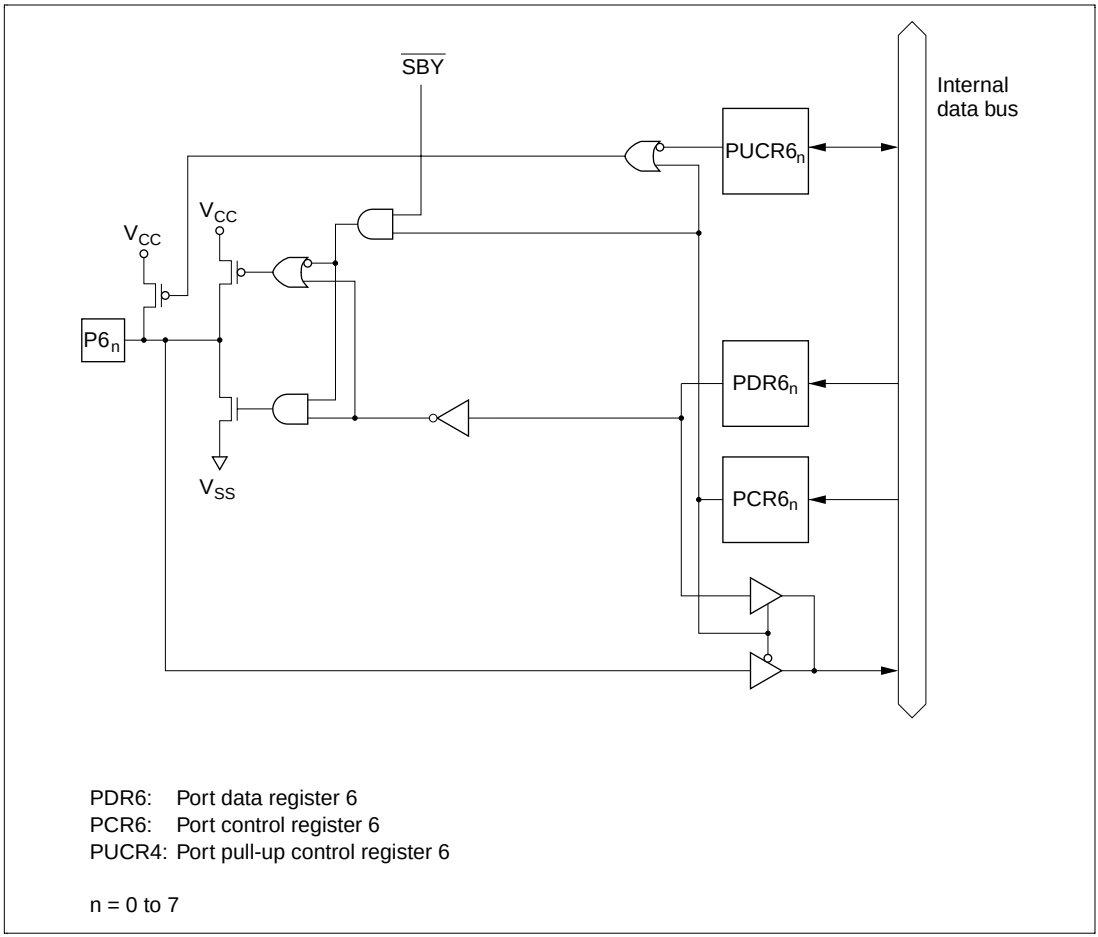



**Figure C.6 Port 6 Block Diagram**

## C.7 Block Diagram of Port 7

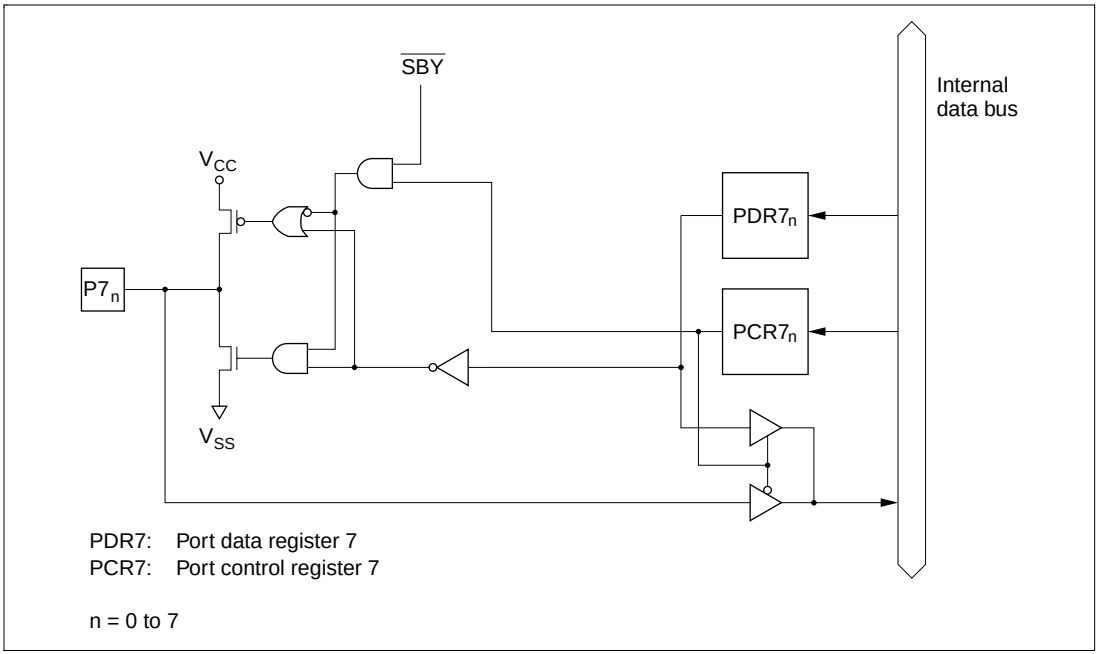



**Figure C.7 Port 7 Block Diagram**

## C.8 Block Diagram of Port 8

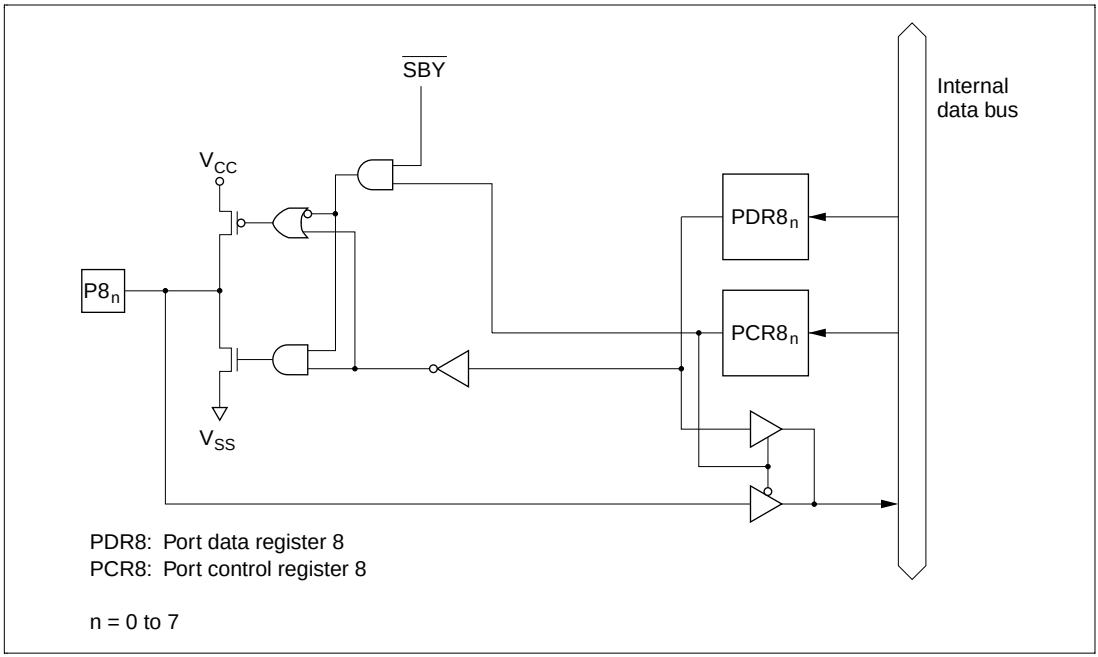



**Figure C.8 Port 8 Block Diagram**

## C.9 Block Diagram of Port 9

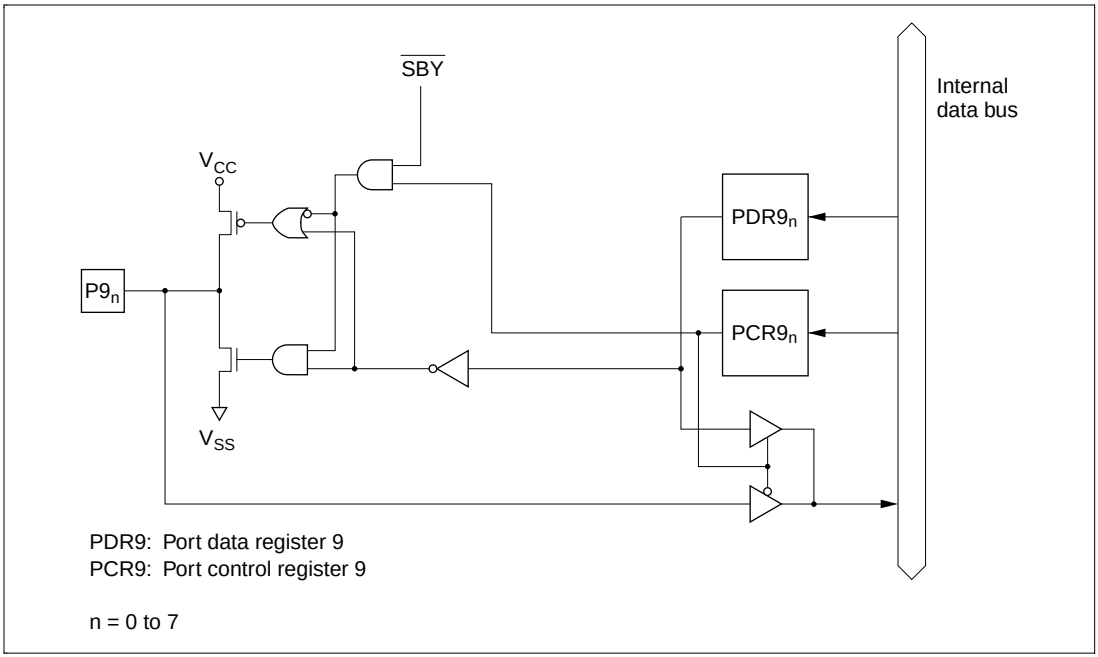



**Figure C.9 Port 9 Block Diagram**

## C.10 Block Diagram of Port A

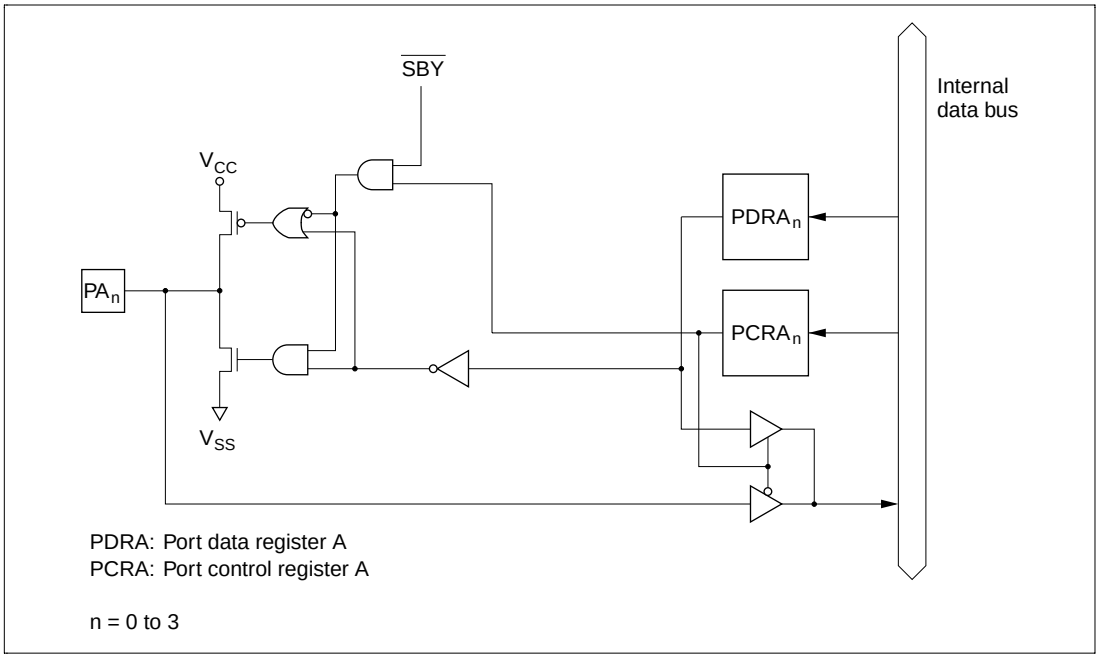



**Figure C.10 Port A Block Diagram**

## C.11 Block Diagram of Port B

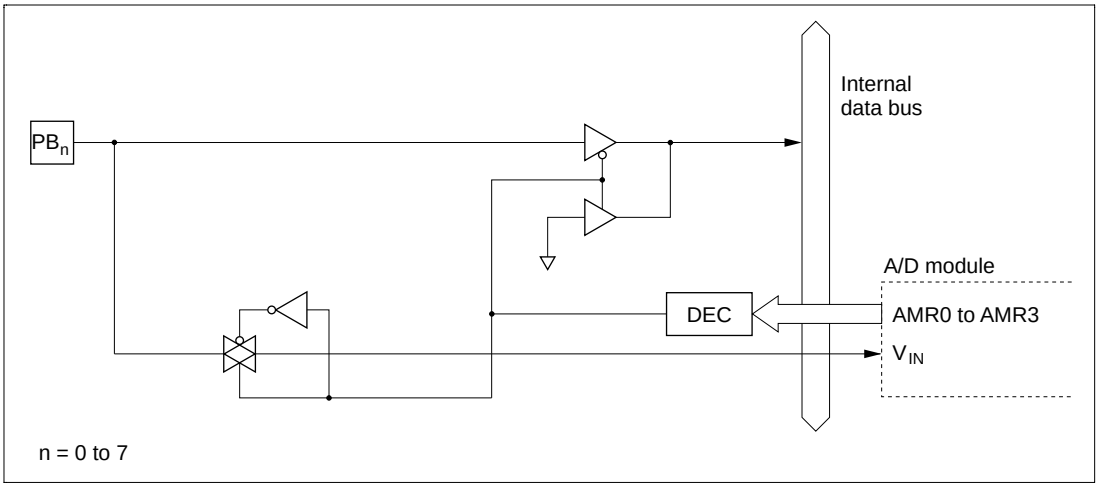



**Figure C.11 Port B Block Diagram**

## C.12 Block Diagram of Port C

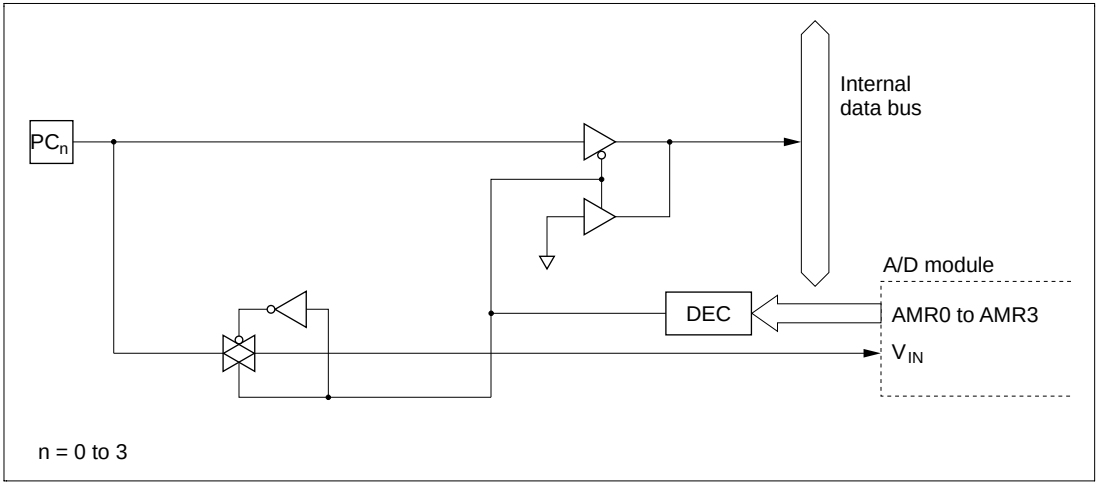



**Figure C-12 Port C Block Diagram**

# Appendix D Port States in the Different Processing States

**Table D.1 Port States Overview**

| Port | Reset | Sleep | Subsleep | Standby | Watch | Subactive | Active |
|---|---|---|---|---|---|---|---|
| $P1_7$ to $P1_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P2_7$ to $P2_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P3_7$ to $P3_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P4_3$ to $P4_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P5_7$ to $P5_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P6_7$ to $P6_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P7_7$ to $P7_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P8_7$ to $P8_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $P9_7$ to $P9_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $PA_3$ to $PA_0$ | High impedance | Retained | Retained | High impedance* | Retained | Functions | Functions |
| $PB_7$ to $PB_0$ | High impedance | High impedance | High impedance | High impedance* | High impedance | High impedance | High impedance |
| $PC_3$ to $PC_0$ | High impedance | High impedance | High impedance | High impedance* | High impedance | High impedance | High impedance |

Note: * High level output when MOS pull-up is in on state.

# Appendix E List of Product Codes

**Table E.1 H8/3834 Series Product Code Lineup**

| Product Type | | | Product Code | Mask Code | Package (Hitachi Package Code) |
|---|---|---|---|---|---|
| H8/3837 | PROM versions | Standard models | HD6473837H | HD6473837H | 100 pin QFP (FP-100B) |
| | | | HD6473837F | HD6473837F | 100 pin QFP (FP-100A) |
| | | | HD6473837X | HD6473837X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6473837D | HD6473837HI | 100 pin QFP (FP-100B) |
| | | | HD6473837E | HD6473837FI | 100 pin QFP (FP-100A) |
| | Mask ROM versions | Standard models | HD6433837H | HD6433837(***)H | 100 pin QFP (FP-100B) |
| | | | HD6433837F | HD6433837(***)F | 100 pin QFP (FP-100A) |
| | | | HD6433837X | HD6433837(***)X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433837D | HD6433837(***)HI | 100 pin QFP (FP-100B) |
| | | | HD6433837E | HD6433837(***)FI | 100 pin QFP (FP-100A) |
| | | | HD6433837L | HD6433837(***)XI | 100 pin TQFP (TFP-100B) |
| H8/3836 | Mask ROM versions | Standard models | HD6433836H | HD6433836(***)H | 100 pin QFP (FP-100B) |
| | | | HD6433836F | HD6433836(***)F | 100 pin QFP (FP-100A) |
| | | | HD6433836X | HD6433836(***)X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433836D | HD6433836(***)HI | 100 pin QFP (FP-100B) |
| | | | HD6433836E | HD6433836(***)FI | 100 pin QFP (FP-100A) |
| | | | HD6433836L | HD6433836(***)XI | 100 pin TQFP (TFP-100B) |
| H8/3835 | Mask ROM versions | Standard models | HD6433835H | HD6433835(***)H | 100 pin QFP (FP-100B) |
| | | | HD6433835F | HD6433835(***)F | 100 pin QFP (FP-100A) |
| | | | HD6433835X | HD6433835(***)X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433835D | HD6433835(***)HI | 100 pin QFP (FP-100B) |
| | | | HD6433835E | HD6433835(***)FI | 100 pin QFP (FP-100A) |
| | | | HD6433835L | HD6433835(***)XI | 100 pin TQFP (TFP-100B) |

Note: For mask ROM versions, (***) is the ROM code.

**Table E.1 H8/3834 Series Product Code Lineup (cont)**

| Product Type | | | Product Code | Mask Code | Package (Hitachi Package Code) |
|---|---|---|---|---|---|
| H8/3834 | PROM versions | Standard models | HD6473834H | HD6473834H | 100 pin QFP (FP-100B) |
| | | | HD6473834F | HD6473834F | 100 pin QFP (FP-100A) |
| | | | HD6473834X | HD6473834X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6473834D | HD6473834HI | 100 pin QFP (FP-100B) |
| | | | HD6473834E | HD6473834FI | 100 pin QFP (FP-100A) |
| | Mask ROM versions | Standard models | HD6433834H | HD6433834(***)H | 100 pin QFP (FP-100B) |
| | | | HD6433834F | HD6433834(***)F | 100 pin QFP (FP-100A) |
| | | | HD6433834X | HD6433834(***)X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433834D | HD6433834(***)HI | 100 pin QFP (FP-100B) |
| | | | HD6433834E | HD6433834(***)FI | 100 pin QFP (FP-100A) |
| | | | HD6433834L | HD6433834(***)XI | 100 pin TQFP (TFP-100B) |
| H8/3833 | Mask ROM versions | Standard models | HD6433833H | HD6433833(***)H | 100 pin QFP (FP-100B) |
| | | | HD6433833F | HD6433833(***)F | 100 pin QFP (FP-100A) |
| | | | HD6433833X | HD6433833(***)X | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433833D | HD6433833(***)HI | 100 pin QFP (FP-100B) |
| | | | HD6433833E | HD6433833(***)FI | 100 pin QFP (FP-100A) |
| | | | HD6433833L | HD6433833(***)XI | 100 pin TQFP (TFP-100B) |

Note: For mask ROM versions, (***) is the ROM code.

**Table E.2 H8/3834S Series Product Code Lineup**

| Product Type | | | Product Code | Mask Code | Package (Hitachi Package Code) |
|---|---|---|---|---|---|
| H8/3837S | Mask ROM versions | Standard models | HD6433837SH | HD6433837(***)SH | 100 pin QFP (FP-100B) |
| | | | HD6433837SF | HD6433837(***)SF | 100 pin QFP (FP-100A) |
| | | | HD6433837SX | HD6433837(***)SX | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433837SD | HD6433837(***)SHI | 100 pin QFP (FP-100B) |
| | | | HD6433837SE | HD6433837(***)SFI | 100 pin QFP (FP-100A) |
| | | | HD6433837SL | HD6433837(***)SXI | 100 pin TQFP (TFP-100B) |
| H8/3836S | Mask ROM versions | Standard models | HD6433836SH | HD6433836(***)SH | 100 pin QFP (FP-100B) |
| | | | HD6433836SF | HD6433836(***)SF | 100 pin QFP (FP-100A) |
| | | | HD6433836SX | HD6433836(***)SX | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433836SD | HD6433836(***)SHI | 100 pin QFP (FP-100B) |
| | | | HD6433836SE | HD6433836(***)SFI | 100 pin QFP (FP-100A) |
| | | | HD6433836SL | HD6433836(***)SXI | 100 pin TQFP (TFP-100B) |
| H8/3835S | Mask ROM versions | Standard models | HD6433835SH | HD6433835(***)SH | 100 pin QFP (FP-100B) |
| | | | HD6433835SF | HD6433835(***)SF | 100 pin QFP (FP-100A) |
| | | | HD6433835SX | HD6433835(***)SX | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433835SD | HD6433835(***)SHI | 100 pin QFP (FP-100B) |
| | | | HD6433835SE | HD6433835(***)SFI | 100 pin QFP (FP-100A) |
| | | | HD6433835SL | HD6433835(***)SXI | 100 pin TQFP (TFP-100B) |
| H8/3834S | Mask ROM versions | Standard models | HD6433834SH | HD6433834(***)SH | 100 pin QFP (FP-100B) |
| | | | HD6433834SF | HD6433834(***)SF | 100 pin QFP (FP-100A) |
| | | | HD6433834SX | HD6433834(***)SX | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433834SD | HD6433834(***)SHI | 100 pin QFP (FP-100B) |
| | | | HD6433834SE | HD6433834(***)SFI | 100 pin QFP (FP-100A) |
| | | | HD6433834SL | HD6433834(***)SXI | 100 pin TQFP (TFP-100B) |

Note: For mask ROM versions, (***) is the ROM code.

**Table E.2 H8/3834S Series Product Code Lineup (cont)**

| Product Type | | | Product Code | Mask Code | Package (Hitachi Package Code) |
|---|---|---|---|---|---|
| H8/3833S | Mask ROM versions | Standard models | HD6433833SH | HD6433833(***)SH | 100 pin QFP (FP-100B) |
| | | | HD6433833SF | HD6433833(***)SF | 100 pin QFP (FP-100A) |
| | | | HD6433833SX | HD6433833(***)SX | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433833SD | HD6433833(***)SHI | 100 pin QFP (FP-100B) |
| | | | HD6433833SE | HD6433833(***)SFI | 100 pin QFP (FP-100A) |
| | | | HD6433833SL | HD6433833(***)SXI | 100 pin TQFP (TFP-100B) |
| H8/3832S | Mask ROM versions | Standard models | HD6433832SH | HD6433832(***)SH | 100 pin QFP (FP-100B) |
| | | | HD6433832SF | HD6433832(***)SF | 100 pin QFP (FP-100A) |
| | | | HD6433832SX | HD6433832(***)SX | 100 pin TQFP (TFP-100B) |
| | | I-Spec models | HD6433832SD | HD6433832(***)SHI | 100 pin QFP (FP-100B) |
| | | | HD6433832SE | HD6433832(***)SFI | 100 pin QFP (FP-100A) |
| | | | HD6433832SL | HD6433832(***)SXI | 100 pin TQFP (TFP-100B) |

Note: For mask ROM versions, (***) is the ROM code.

# Appendix F Package Dimensions

Dimensional drawings of H8/3834 Series packages FP-100B, FP-100A, and TFP-100B are shown in figures F.1, F.2, and F.3 below.

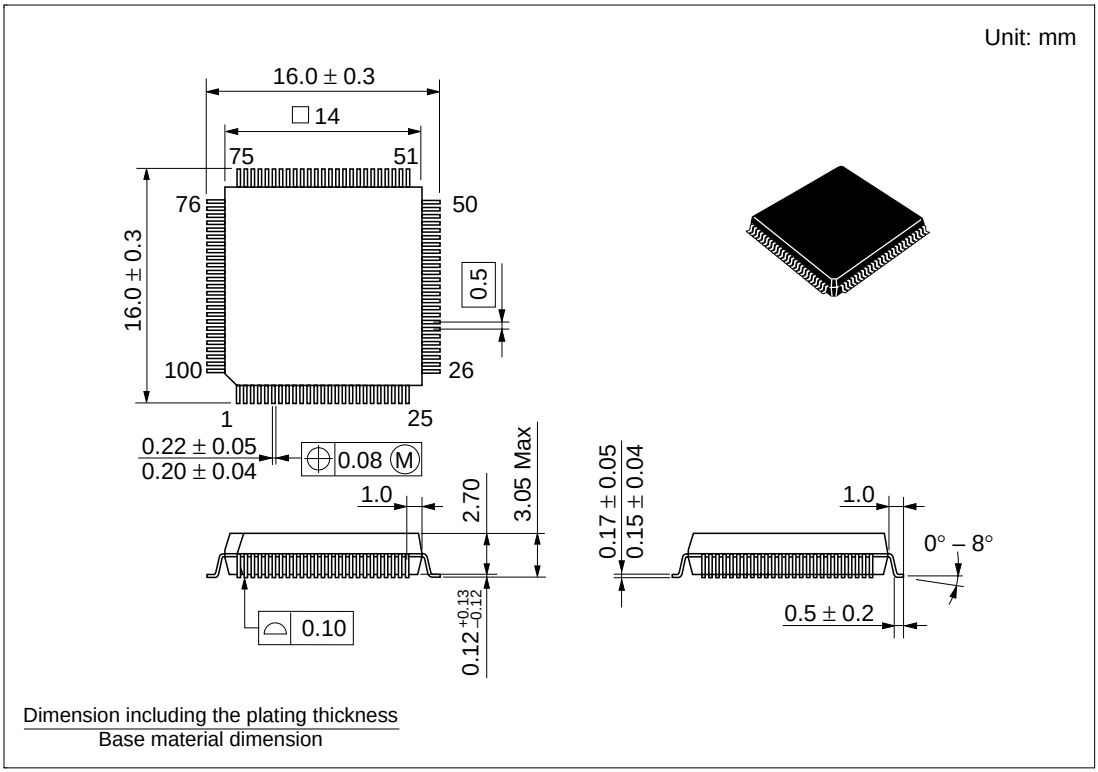



**Figure F.1 FP-100B Package Dimensions**

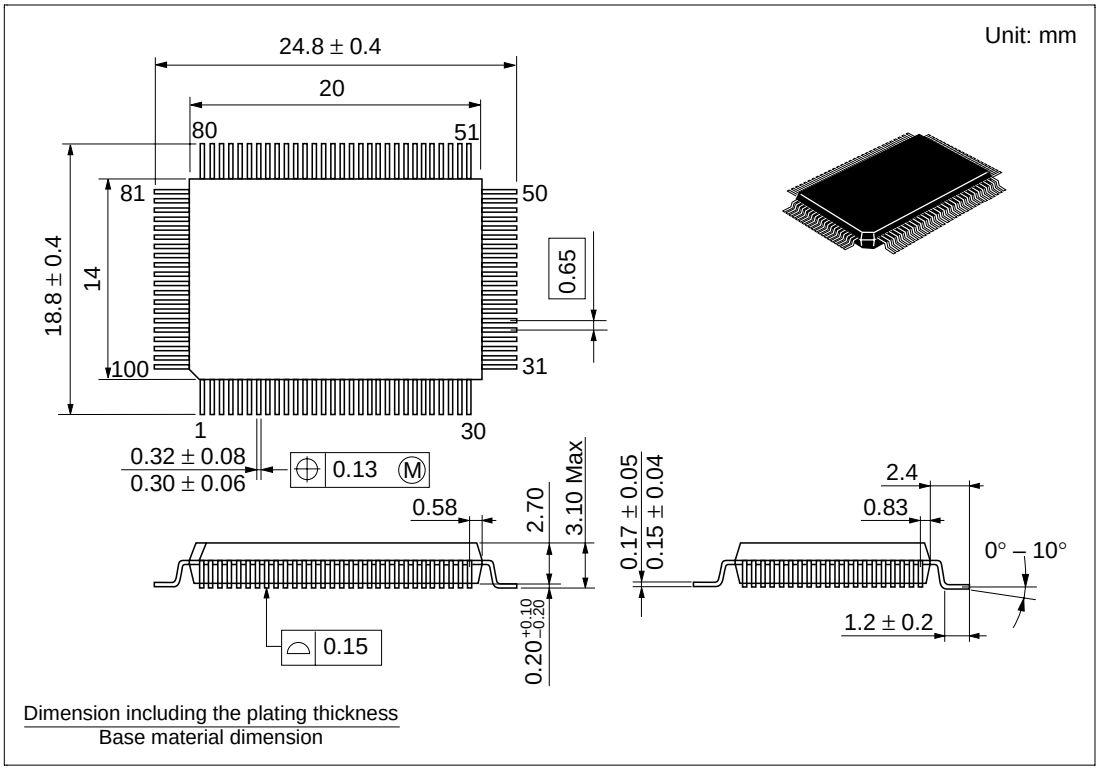



**Figure F.2 FP-100A Package Dimensions**

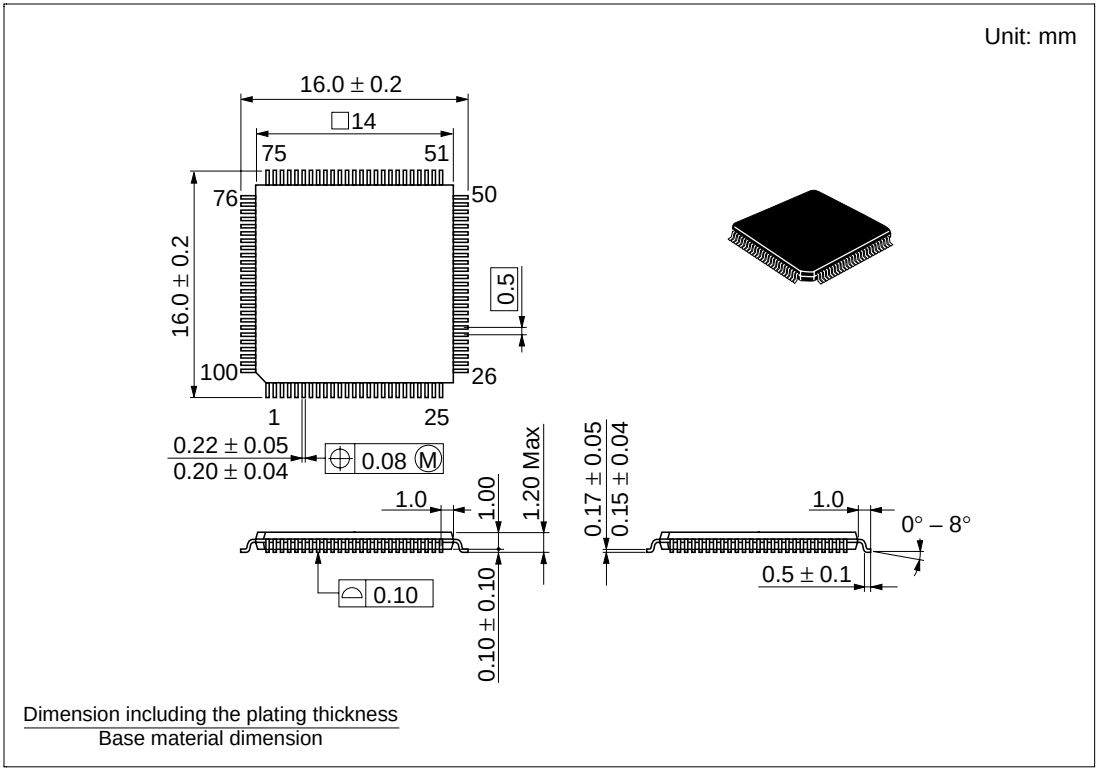



**Figure F.3 TFP-100B Package Dimensions**

Note: In case of inconsistencies arising within figures, dimensional drawings listed in the Hitachi Semiconductor Packages Manual take precedence and are considered correct.

**H8S/3834 Series Hardware Manual**

Publication Date: 1st Edition, March 1993
5th Edition, September 1997
Published by: Semiconductor and IC Div.
Hitachi, Ltd.
Edited by: Technical Documentation Center
Hitachi Microcomputer System Ltd.
Copyright © Hitachi, Ltd., 1993. All rights reserved. Printed in Japan.