To our customers,

Old Company Name in Catalogs and Other Documents

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

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

April 1st, 2010
Renesas Electronics Corporation

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

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

Notice

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

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

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

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

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

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

User's Manual

8

H8/3827R Group, H8/3827S Group, H8/38327 Group, H8/38427 Group

Hardware Manual

Renesas 8-Bit Single-Chip Microcomputer
H8 Family/H8/300L Super Low Power Series

H8/3827R Group	H8/3822R	H8/38327 Group	H8/38322
	H8/3823R		H8/38323
	H8/3824R		H8/38324
	H8/3825R		H8/38325
	H8/3826R		H8/38326
	H8/3827R		H8/38327
H8/3827S Group	H8/3824S	H8/38427 Group	H8/38422
	H8/3825S		H8/38423
	H8/3826S		H8/38424
	H8/3827S		H8/38425
			H8/38426
			H8/38427

Renesas Electronics
www.renesas.com

Rev.6.00 2006.08

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

RENESAS

an ideal configuration as a microcomputer for embedding in sophisticated control syst
(ZTAT™*1), flash memory (F-ZTAT™*2), and mask ROM are available as on-chip F
enabling users to respond quickly and flexibly to changing application specifications a
demands of the transition from initial to full-fledged volume production.

Notes: 1. ZTAT is a trademark of Renesas Technology Corp.
2. F-ZTAT is a trademark of Renesas Technology Corp.

Intended Readership: This manual is intended for users undertaking the design of an a
system using the H8/3827R Group, H8/3827S Group, H8/38327
H8/38427 Group. Readers using this manual require a basic knov
electrical circuits, logic circuits, and microcomputers.

Purpose: The purpose of this manual is to give users an understanding of t
functions and electrical characteristics of the H8/3827R Group, H
Group, H8/38327 Group, and H8/38427 Group. Details of execu
instructions can be found in the H8/300L Series Programming M
which should be read in conjunction with the present manual.

Using this Manual:

- For an overall understanding of the H8/3827R Group, H8/3827S Group, H8/38327
H8/38427 Group's functions
Follow the Table of Contents. This manual is broadly divided into sections on the
control functions, peripheral functions, and electrical characteristics.
- For a detailed understanding of CPU functions
Refer to the separate publication H8/300L Series Programming Manual.
Note on bit notation: Bits are shown in high-to-low order from left to right.

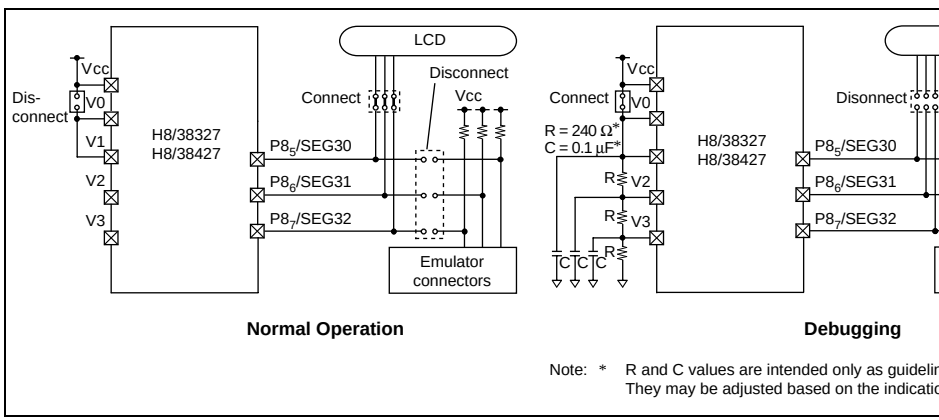
4. The address area from H'F300 to H'F6FF must not be accessed under any circumstances.
5. When the on-chip emulator is used, pin $P3_2$ functions as an I/O pin, pins $P8_5$ function as input pins, and pin $P8_7$ functions as an output pin.
6. It is necessary to change the user board for LCD display debugging. This ite apply if LCD display is not used or if the emulator is not operated in writer

   User Board Changes

   The following changes are necessary.

   - Connect pin V1 to the V_{CC} power supply. Also connect capacitors and re pins V1, V2, and V3. Connect a capacitor and a resistor to pins V1, V2, a
   - No SEG signals are output from pins $P8_5$/SEG30, $P8_6$/SEG31, and $P8_7$/SE the display is unstable. In addition, a DC voltage is applied. If damage to a concern, disconnect the above three pins from the LCD.

   A connection example is shown below. Refer to the emulator user's manual information on other settings.



RENESAS

Manual Title	Documen
H8/3827R Group, H8/3827S Group, H8/38327 Group, H8/38427 Group Hardware Manual	This man
H8/300L Series Programming Manual	REJ09B0

User's manuals for development tools:

Manual Title	Documen
C/C++ Compiler, Assembler, Optimizing Linkage Editor User's Manual	REJ10B0
H8S, H8/300 Series Simulator/Debugger User's Manual	REJ10B0
High-Performance Embedded Workshop User's Manual	ADE-702
H8S, H8/300 Series High-Performance Embedded Workshop, High-Performance Debugging Interface User's Manual	ADE-702

Application Note:

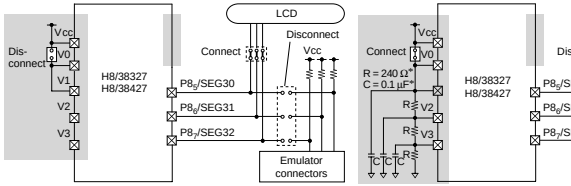
Manual Title	Documen
H8/300L Series Application Note	ADE-502

or if the emulator is not operated in writer mode.

User Board Changes

The following changes are necessary.

- Connect pin V1 to the V_{CC} power supply. Also c
  capacitors and resistors to pins V1, V2, and V3.
  capacitor and a resistor to pins V1, V2, and V3.



Normal Operation **Debug**

Note: * R and C values are intended only
They may be adjusted based on

7. During a break, the watchdog timer continues to o
   Therefore, an internal reset is generated if an over
   during the break.

1.3.2 Pin Functions	28	Table amended

Table 1.6 Pin Functions

		Pin No.			
		FP-80A			
Type	**Symbol**	**TFP-80C**	**FP-80B**	**I/O**	**Name and Functions**
System control	TEST	8	10	Input	**Test pin:** This pin is re cannot be used. It sho to V_{SS}.

8.3.1 Overview	203	Description amended

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

In the F-ZTAT version, the on-chip pull-up MOS for pi during the reset period. It turns off and normal operati after the reset is cleared. This should be considered w connections to external circuitry. Note that in the mas ZTAT versions $P3_2$ continues to operate normally.

	MOS on	
$P3_2$*3 $P3_1$/UD*2 $P3_1$/UD/EXCL*3*4 $P3_0$/PWM	High-impedance	

Notes: 1. A high-level signal is output when the MOS pull-up is in the on state.
2. Applies to H8/3827R Group and H8/3827S Group.
3. Applies to the mask ROM version of the H8/38327 Group and H8/3842
4. Applies to the F-ZTAT version of the H8/38327 Group and H8/38427 G

8.12.1 The Management of the Un-Use Terminal	236	Description amended • If an unused pin is an output pin, handle it in one of following ways: — Set the output of the unused pin to high and pul Vcc with an external resistor of approximately 1 — Set the output of the unused pin to low and pull Vss with an external resistor of approximately 1
C.2 Block Diagrams of Port 3 Figure C.2 (e-2) Port 3 Block Diagram (Pin $P3_2$ in the Mask ROM Version of the H8/38327 Group and H8/38427 Group)	593	Figure title amended
Figure C.2 (e-3) Port 3 Block Diagram (Pin $P3_2$ in the F-ZTAT Version of the H8/38327 Group and H8/38427 Group)	594	Newly added

RENESAS

RENESAS

RENESAS

RENESAS

RENESAS

Section 2 CPU

Section 4 Clock Pulse Generators

Section 5 Power-Down Modes

Section 6 ROM

Section 7 RAM

Section 8 I/O Ports

Section 9 Timers

RENESAS

Section 10 Serial Communication Interface

Section 14 Power Supply Circuit

Section 15 Electrical Characteristics

Appendix C I/O Port Block Diagrams

RENESAS

Section 8 I/O Ports

Section 9 Timers

Section 11 14-Bit PWM

Section 12 A/D Converter

Section 13 LCD Controller/Driver

Section 15 Electrical Characteristics

RENESAS

Within the H8/300L Series, the H8/3827R Group, H8/3827S Group, H8/38327 Group
H8/38427 Group comprise single-chip microcomputers equipped with an LCD (liquid
display) controller/driver. Other on-chip peripheral functions include six timers, a 14-
width modulator (PWM), two serial communication interface channels, and an A/D co
Together, these functions make the H8/3827R Group, H8/3827S Group, H8/38327 Gr
H8/38427 Group ideally suited for embedded applications in systems requiring low po
consumption and LCD display. Also available are models incorporating 16 Kbytes to
ROM and 1 Kbyte to 2 Kbytes of RAM on-chip.

The H8/3827R is also available in a ZTAT™*1 version with on-chip PROM which ca
programmed as required by the user.

The H8/38327 and H8/38427 are available in a F-ZTAT™*2 version with on-chip flas
that can be programmed on-board.

Table 1.1 summarizes the features of the H8/3827R Group, H8/3827S Group, H8/383
and H8/38427 Group.

Notes: 1. ZTAT (Zero Turn Around Time) is a trademark of Renesas Technology Co
2. F-ZTAT is a trademark of Renesas Technology Corp.

	— Max. operating speed: 8 MHz — Add/subtract: 0.25 µs (operating at 8 MHz) — Multiply/divide: 1.75 µs (operating at 8 MHz) — Can run on 32.768 kHz or 38.4 kHz subclock • Instruction set compatible with H8/300 CPU — Instruction length of 2 bytes or 4 bytes — Basic arithmetic operations between registers — MOV instruction for data transfer between memory and • Typical instructions — Multiply (8 bits × 8 bits) — Divide (16 bits ÷ 8 bits) — Bit accumulator — Register-indirect designation of bit position
Interrupts	36 interrupt sources • 13 external interrupt sources (IRQ_4 to IRQ_0, WKP_7 to WKP • 23 internal interrupt sources
Clock pulse generators	Two on-chip clock pulse generators • System clock pulse generator: — Maximum 16 MHz (H8/3827R, H8/38327, H8/38427 Gr — Maximum 10 MHz (H8/3827S Group) • Subclock pulse generator: 32.768 kHz, 38.4 kHz

	• Subactive mode • Active (medium-speed) mode
Memory	Large on-chip memory • H8/3822R, H8/38322, H8/38422: 16-Kbyte ROM, 1-Kbyte RAM • H8/3823R, H8/38323, H8/38423: 24-Kbyte ROM, 1-Kbyte RAM • H8/3824R, H8/3824S, H8/38324, H8/38424: 32-Kbyte ROM, 2-Kbyte RAM • H8/3825R, H8/3825S, H8/38325, H8/38425: 40-Kbyte ROM, 2-Kbyte RAM • H8/3826R, H8/3826S, H8/38326, H8/38426: 48-Kbyte ROM, 2-Kbyte RAM • H8/3827R, H8/3827S, H8/38327, H8/38427: 60-Kbyte ROM, 2-Kbyte RAM
I/O ports	64 pins • 55 I/O pins • 9 input pins

	independently of the MCU's internal clocks • Timer C: 8-bit timer — Count-up/down timer with selection of seven internal cl or event input from external pin — Auto-reloading • Timer F: 16-bit timer — Can be used as two independent 8-bit timers — Count-up timer with selection of four internal clock sign input from external pin — Provision for toggle output by means of compare-match • Timer G: 8-bit timer — Count-up timer with selection of four internal clock sign — Incorporates input capture function (built-in noise cance • Watchdog timer — Reset signal generated by overflow of 8-bit counter
Serial communication interface	Two serial communication interface channels on chip • SCI3-1: 8-bit synchronous/asynchronous serial interface Incorporates multiprocessor communication function • SCI3-2: 8-bit synchronous/asynchronous serial interface Incorporates multiprocessor communication function
14-bit PWM	Pulse-division PWM output for reduced ripple • Can be used as a 14-bit D/A converter by connecting to an low-pass filter.
A/D converter	Successive approximations using a resistance ladder • 8-channel analog input pins • Conversion time: 31/ϕ or 62/ϕ per channel

HD6433827R HD6433827S HD64338327 HD64338427	HD6473827R	HD64F38327 HD64F38427	FP-80B (H8/3827R only) FP-80A TFP-80C Die
HD6433826R HD6433826S HD64338326 HD64338426	—	—	FP-80B (H8/3826R only) FP-80A TFP-80C Die
HD6433825R HD6433825S HD64338325 HD64338425	—	—	FP-80B (H8/3825R only) FP-80A TFP-80C Die
HD6433824R HD6433824S HD64338324 HD64338424	—	HD64F38324 HD64F38424	FP-80B (H8/3824R only) FP-80A TFP-80C Die (Mask ROM version only)
HD6433823R HD64338323 HD64338423	—	—	FP-80B (H8/3823R only) FP-80A TFP-80C Die
HD6433822R HD64338322 HD64338422	—	—	FP-80B (H8/3822R only) FP-80A TFP-80C Die

See appendix E for a list of product codes.

Note: * See section 4, Clock Pulse Generators, for the definition of ϕ and ϕ_W.

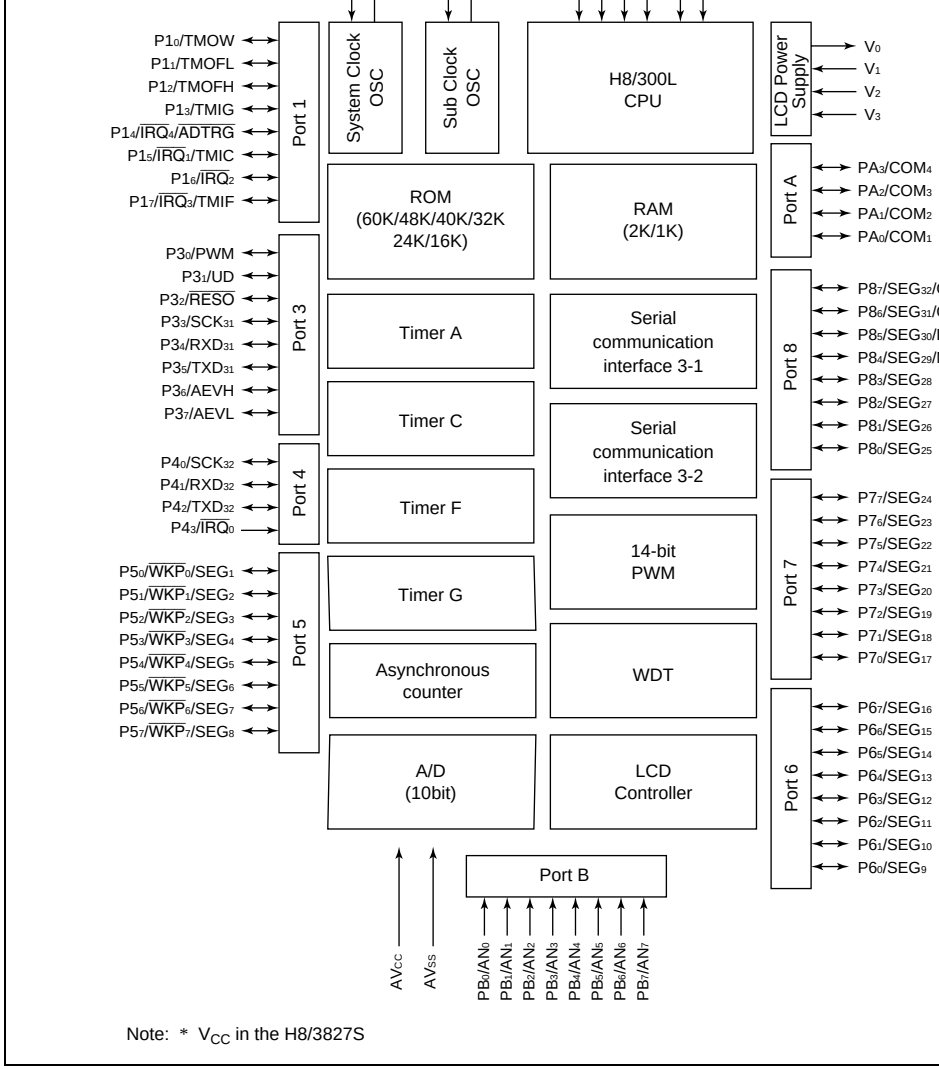
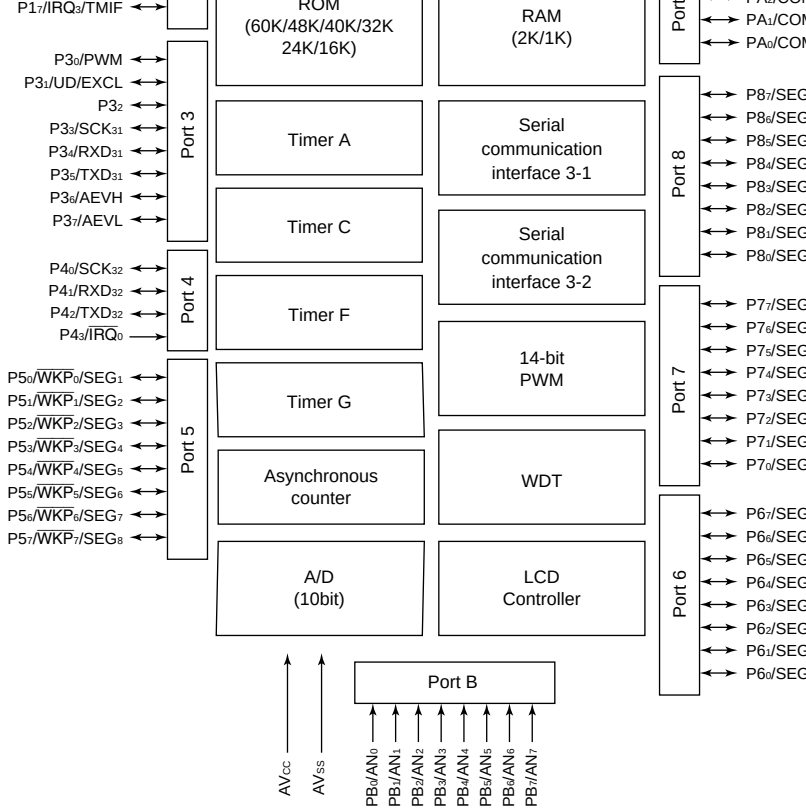


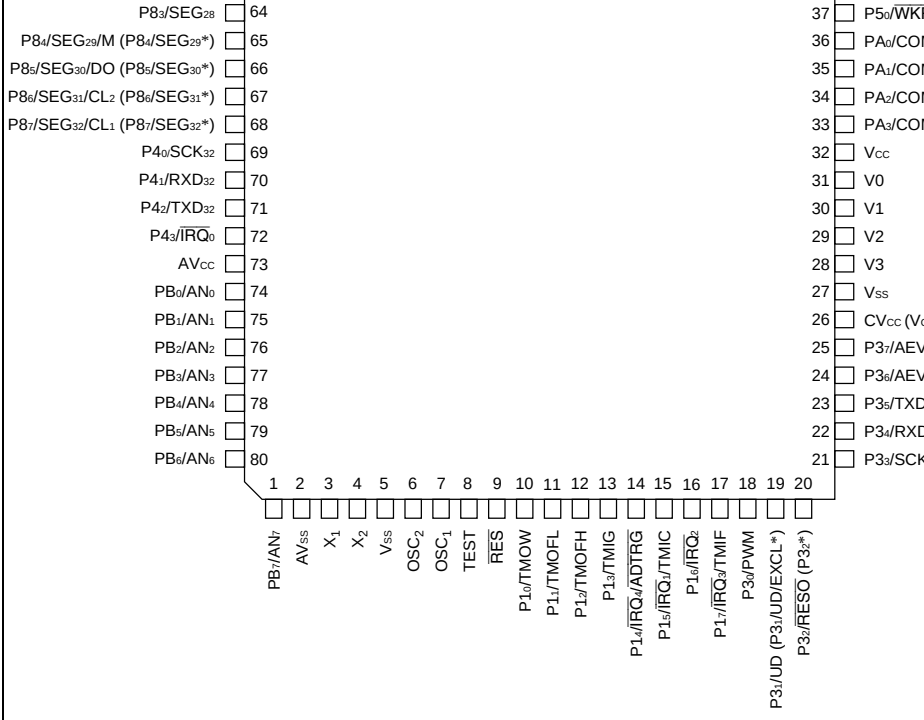
Figure 1.1(1) Block Diagram (H8/3827R Group and H8/3827S Group)



Note: When the on-chip emulator is used, pins P3₂, P8₅, P8₆, and P8₇ are reserved for use exclusiv
by the emulator and therefore cannot be accessed by the user.

Figure 1.1(2) Block Diagram (H8/38327 Group and H8/38427 Group

the H8/3827S Group (mask ROM version) is shown in figure 1.5, and the bonding pad
are given in table 1.3. The bonding pad location diagram of the HCD64F38327 and
HCD64F38427 is shown in figure 1.6, and the bonding pad coordinates are given in tab
bonding pad location diagram of the H8/38327 Group (mask ROM version) and H8/38
(mask ROM version) is shown in figure 1.7, and the bonding pad coordinates are given
1.5.



Notes: When the on-chip emulator is used, pins P32, P85, P86, and P87 are reserved for use exclusively by the em
therefore cannot be accessed by the user.
* H8/38327, H8/38427

Figure 1.2 Pin Arrangement (FP-80A, TFP-80C: Top View)

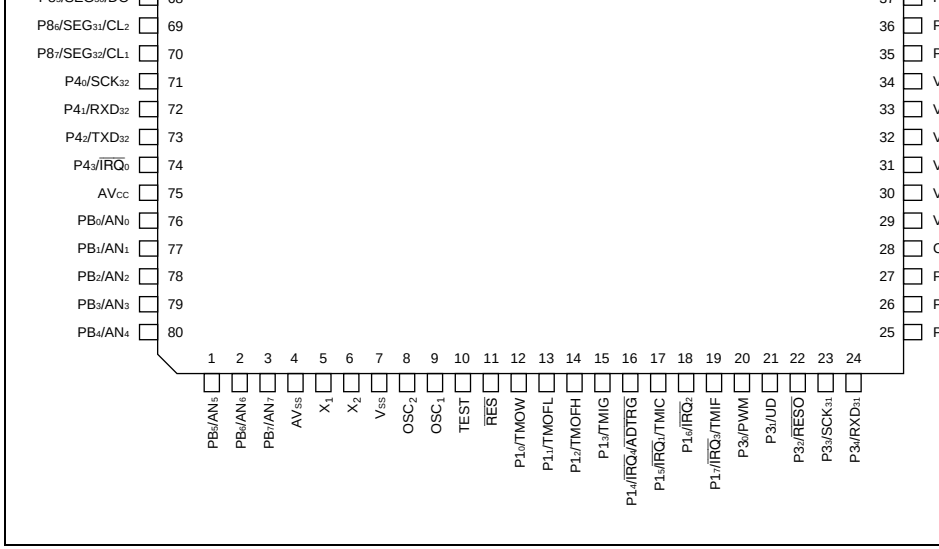


Figure 1.3 Pin Arrangement (FP-80B: Top View)

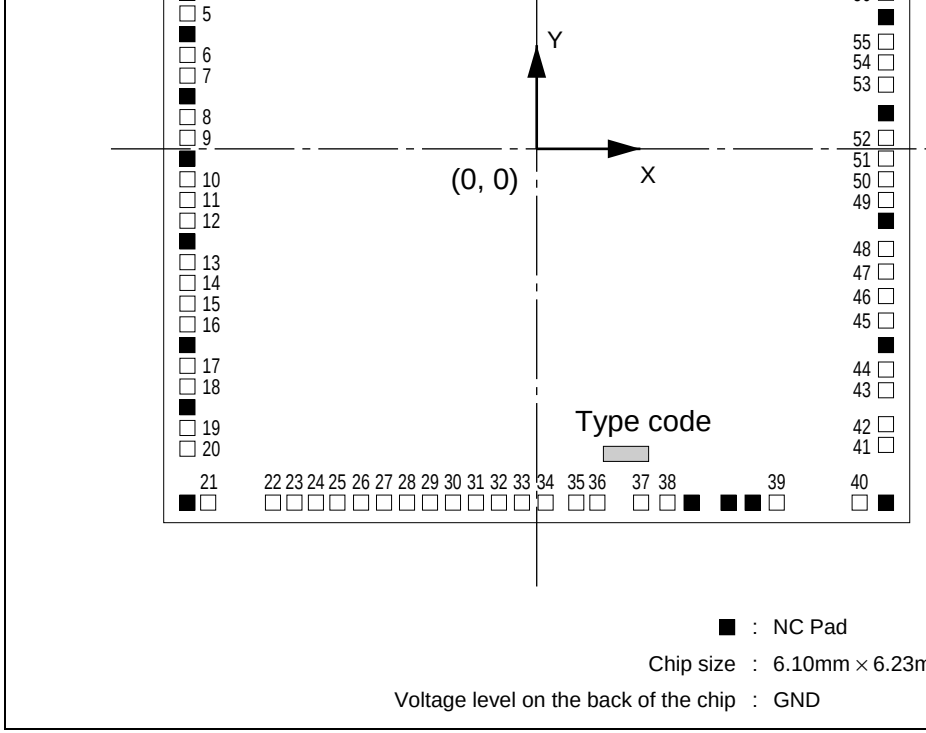


Figure 1.4 Bonding Pad Location Diagram of H8/3827R Group (Mask ROM Version) (Top View)

5	Vss	-2866	1156
6	OSC_2	-2866	810
7	OSC_1	-2866	636
8	TEST	-2866	288
9	$\overline{RES}$	-2866	116
10	$P1_0$/TMOW	-2866	-228
11	$P1_1$/TMOFL	-2866	-402
12	$P1_2$/TMOFH	-2866	-576
13	$P1_3$/TMIG	-2866	-920
14	$P1_4/\overline{IRQ_4}/\overline{ADTRG}$	-2866	-1094
15	$P1_5/\overline{IRQ_1}$/TMIC	-2866	-1266
16	$P1_6/\overline{IRQ_2}$	-2866	-1440
17	$P1_7/\overline{IRQ_3}$/TMIF	-2866	-1785
18	$P3_0$/PWM	-2866	-1969
19	$P3_1$/UD	-2866	-2327
20	$P3_2/\overline{RESO}$	-2866	-2503
21	$P3_3/SCK_{31}$	-2669	-2931
22	$P3_4/RXD_{31}$	-2142	-2931
23	$P3_5/TXD_{31}$	-1971	-2931
24	$P3_6$/AEVH	-1798	-2931
25	$P3_7$/AEVL	-1624	-2931
26	CVcc	-1413	-2931
27	Vss	-1213	-2931
28	V3	-1017	-2931
29	V2	-844	-2931

35	PA_1/COM2	320	-2931
36	PA_0/COM1	544	-2931
37	$P5_0/\overline{WKP}_0/SEG_1$	842	-2931
38	$P5_1/\overline{WKP}_1/SEG_2$	1069	-2931
39	$P5_2/\overline{WKP}_2/SEG_3$	2017	-2931
40	$P5_3/\overline{WKP}_3/SEG_4$	2648	-2931
41	$P5_4/\overline{WKP}_4/SEG_5$	2866	-2484
42	$P5_5/\overline{WKP}_5/SEG_6$	2866	-2296
43	$P5_6/\overline{WKP}_6/SEG_7$	2866	-2061
44	$P5_7/\overline{WKP}_7/SEG_8$	2866	-1846
45	$P6_0/SEG_9$	2866	-1430
46	$P6_1/SEG_{10}$	2866	-1244
47	$P6_2/SEG_{11}$	2866	-1056
48	$P6_3/SEG_{12}$	2866	-828
49	$P6_4/SEG_{13}$	2866	-452
50	$P6_5/SEG_{14}$	2866	-264
51	$P6_6/SEG_{15}$	2866	-76
52	$P6_7/SEG_{16}$	2866	112
53	$P7_0/SEG_{17}$	2866	528
54	$P7_1/SEG_{18}$	2866	756
55	$P7_2/SEG_{19}$	2866	944
56	$P7_3/SEG_{20}$	2866	1318
57	$P7_4/SEG_{21}$	2866	1506
58	$P7_5/SEG_{22}$	2866	1694
59	$P7_6/SEG_{23}$	2866	2070
60	$P7_7/SEG_{24}$	2866	2367

66	$P8_5/SEG_{30}/DO$	1396	2931
67	$P8_6/SEG_{31}/CL_2$	1209	2931
68	$P8_7/SEG_{32}/CL_1$	977	2931
69	$P4_0/SCK_{32}$	631	2931
70	$P4_1/RXD_{32}$	456	2931
71	$P4_2/TXD_{32}$	284	2931
72	$P4_3/\overline{IRQ}_0$	109	2931
73	AVcc	-64	2931
74	PB_0/AN_0	-236	2931
75	PB_1/AN_1	-409	2931
76	PB_2/AN_2	-581	2931
77	PB_3/AN_3	-925	2931
78	PB_4/AN_4	-1268	2931
79	PB_5/AN_5	-2048	2931
80	PB_6/AN_6	-2658	2931

Note: * These values show the coordinates of the centers of pads. The accuracy is ±
home-point position is the chip's center and the center is located at half the d
between the upper and lower pads and left and right pads.

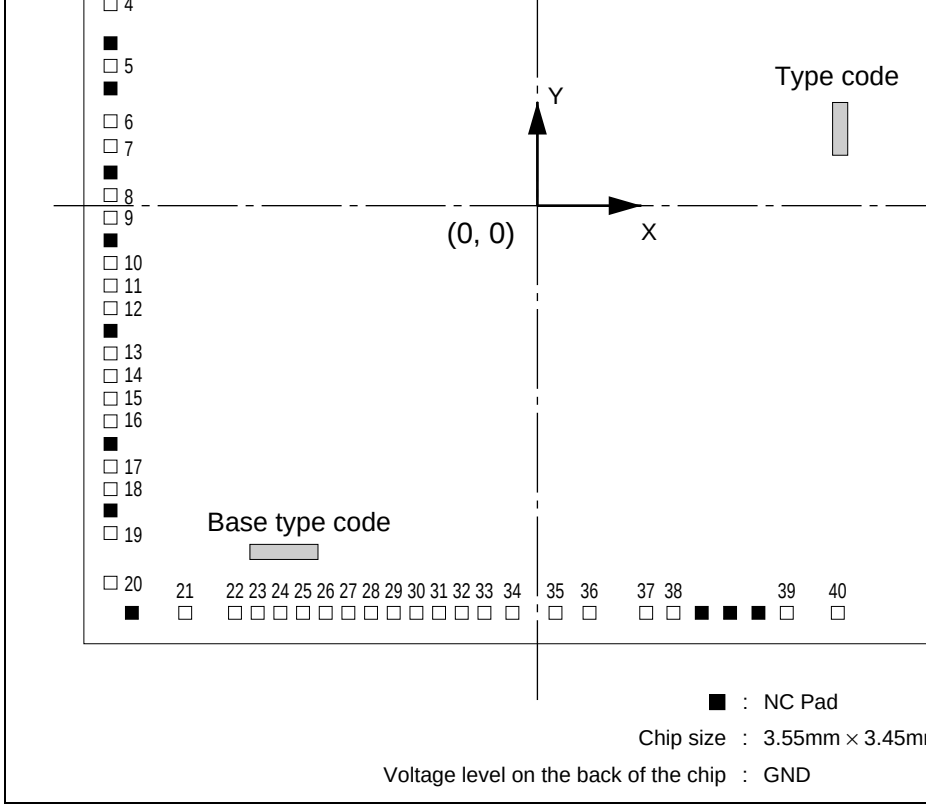


Figure 1.5 Bonding Pad Location Diagram of H8/3827S Group (Mask ROM Version) (Top View)

5	Vss	-1655	536
6	OSC_2	-1655	334
7	OSC_1	-1655	226
8	TEST	-1655	37
9	$\overline{RES}$	-1655	-48
10	$P1_0$/TMOW	-1655	-223
11	$P1_1$/TMOFL	-1655	-308
12	$P1_2$/TMOFH	-1655	-393
13	$P1_3$/TMIG	-1655	-563
14	$P1_4/\overline{IRQ_4}/\overline{ADTRG}$	-1655	-648
15	$P1_5/\overline{IRQ_1}$/TMIC	-1655	-733
16	$P1_6/\overline{IRQ_2}$	-1655	-818
17	$P1_7/\overline{IRQ_3}$/TMIF	-1655	-988
18	$P3_0$/PWM	-1655	-1073
19	$P3_1$/UD	-1655	-1243
20	$P3_2/\overline{RESO}$	-1655	-1480
21	$P3_3$/SCK31	-1357	-1605
22	$P3_4$/RXD31	-1178	-1605
23	$P3_5$/TXD31	-1093	-1605
24	$P3_6$/AEVH	-992	-1605
25	$P3_7$/AEVL	-906	-1605
26	Vcc	-821	-1605
27	Vss	-736	-1605
28	V3	-651	-1605
29	V2	-566	-1605

35	PA_1/COM2	64	-1605
36	PA_0/COM1	197	-1605
37	$P5_0/\overline{WKP}_0/SEG_1$	421	-1605
38	$P5_1/\overline{WKP}_1/SEG_2$	528	-1605
39	$P5_2/\overline{WKP}_2/SEG_3$	957	-1605
40	$P5_3/\overline{WKP}_3/SEG_4$	1154	-1605
41	$P5_4/\overline{WKP}_4/SEG_5$	1655	-1527
42	$P5_5/\overline{WKP}_5/SEG_6$	1655	-1294
43	$P5_6/\overline{WKP}_6/SEG_7$	1655	-1209
44	$P5_7/\overline{WKP}_7/SEG_8$	1655	-1117
45	$P6_0/SEG_9$	1655	-903
46	$P6_1/SEG_{10}$	1655	-796
47	$P6_2/SEG_{11}$	1655	-689
48	$P6_3/SEG_{12}$	1655	-559
49	$P6_4/SEG_{13}$	1655	-345
50	$P6_5/SEG_{14}$	1655	-237
51	$P6_6/SEG_{15}$	1655	-130
52	$P6_7/SEG_{16}$	1655	-23
53	$P7_0/SEG_{17}$	1655	191
54	$P7_1/SEG_{18}$	1655	317
55	$P7_2/SEG_{19}$	1655	424
56	$P7_3/SEG_{20}$	1655	639
57	$P7_4/SEG_{21}$	1655	746
58	$P7_5/SEG_{22}$	1655	853
59	$P7_6/SEG_{23}$	1655	1067
60	$P7_7/SEG_{24}$	1655	1527

66	$P8_5/SEG_{30}/DO$	854	1605
67	$P8_6/SEG_{31}/CL_2$	747	1605
68	$P8_7/SEG_{32}/CL_1$	640	1605
69	$P4_0/SLK_{32}$	524	1605
70	$P4_1/RXD_{32}$	439	1605
71	$P4_2/TXD_{32}$	354	1605
72	$P4_3/\overline{IRQ_0}$	269	1605
73	AVcc	101	1605
74	PB_0/AN_0	16	1605
75	PB_1/AN_1	-92	1605
76	PB_2/AN_2	-207	1605
77	PB_3/AN_3	-431	1605
78	PB_4/AN_4	-655	1605
79	PB_5/AN_5	-1103	1605
80	PB_6/AN_6	-1290	1605

Note: * These values show the coordinates of the centers of pads. The accuracy is ±
home-point position is the chip's center and the center is located at half the d
between the upper and lower pads and left and right pads.

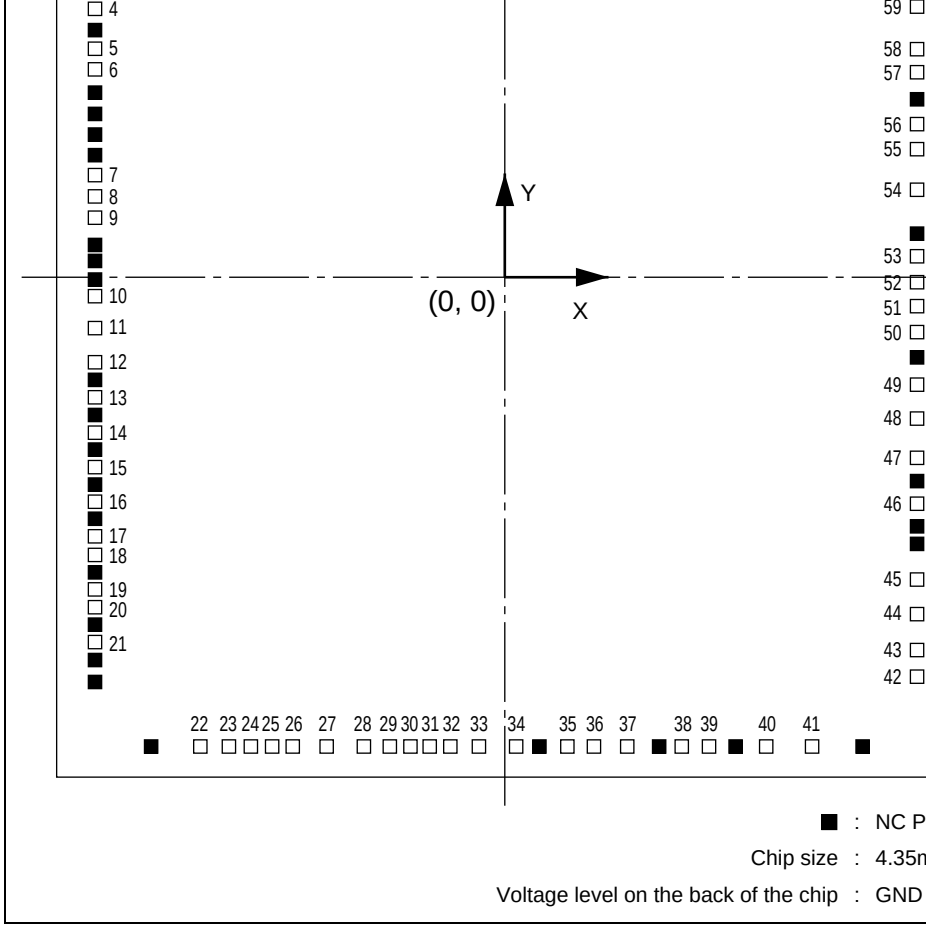


Figure 1.6 Bonding Pad Location Diagram of HCD64F38327 and HCD64I (Top View)

5	Vss	-2056	1158
6	Vss	-2056	1062
7	OSC_2	-2056	533
8	OSC_1	-2056	431
9	TEST	-2056	329
10	$\overline{RES}$	-2056	-66
11	$P1_0$/TMOW	-2056	-244
12	$P1_1$/TMOFL	-2056	-402
13	$P1_2$/TMOFH	-2056	-574
14	$P1_3$/TMIG	-2056	-747
15	$P1_4/\overline{IRQ_4}/\overline{ADTRG}$	-2056	-919
16	$P1_5/\overline{IRQ_1}$/TMIC	-2056	-1091
17	$P1_6/\overline{IRQ_2}$	-2056	-1263
18	$P1_7/\overline{IRQ_3}$/TMIF	-2056	-1349
19	$P3_0$/PWM	-2056	-1521
20	$P3_1$/UD/EXCL	-2056	-1607
21	$P3_2$	-2056	-1779
22	$P3_3/SCK_{31}$	-1530	-2295
23	$P3_4/RXD_{31}$	-1382	-2295
24	$P3_5/TXD_{31}$	-1280	-2295
25	$P3_6$/AEVH	-1178	-2295
26	$P3_7$/AEVL	-1076	-2295
27	CVcc	-896	-2295
28	Vss	-710	-2295
29	V3	-584	-2295
30	V2	-483	-2295

36	PA_1/COM2	441	-2295
37	PA_0/COM1	604	-2295
38	$P5_0/\overline{WKP}_0/SEG_1$	883	-2295
39	$P5_1/\overline{WKP}_1/SEG_2$	1022	-2295
40	$P5_2/\overline{WKP}_2/SEG_3$	1302	-2295
41	$P5_3/\overline{WKP}_3/SEG_4$	1530	-2295
42	$P5_4/\overline{WKP}_4/SEG_5$	2056	-1955
43	$P5_5/\overline{WKP}_5/SEG_6$	2056	-1830
44	$P5_6/\overline{WKP}_6/SEG_7$	2056	-1651
45	$P5_7/\overline{WKP}_7/SEG_8$	2056	-1481
46	$P6_0/SEG_9$	2056	-1111
47	$P6_1/SEG_{10}$	2056	-879
48	$P6_2/SEG_{11}$	2056	-671
49	$P6_3/SEG_{12}$	2056	-505
50	$P6_4/SEG_{13}$	2056	-255
51	$P6_5/SEG_{14}$	2056	-130
52	$P6_6/SEG_{15}$	2056	-6
53	$P6_7/SEG_{16}$	2056	119
54	$P7_0/SEG_{17}$	2056	457
55	$P7_1/SEG_{18}$	2056	660
56	$P7_2/SEG_{19}$	2056	784
57	$P7_3/SEG_{20}$	2056	1034
58	$P7_4/SEG_{21}$	2056	1159
59	$P7_5/SEG_{22}$	2056	1378
60	$P7_6/SEG_{23}$	2056	1627
61	$P7_7/SEG_{24}$	2056	1840

67	$P8_5/SEG_{30}$	901	2295
68	$P8_6/SEG_{31}$	728	2295
69	$P8_7/SEG_{32}$	603	2295
70	$P4_0/SCK_{32}$	451	2295
71	$P4_1/RXD_{32}$	350	2295
72	$P4_2/TXD_{32}$	175	2295
73	$P4_3/\overline{IRQ}_0$	73	2295
74	AVcc	-155	2295
75	PB_0/AN_0	-290	2295
76	PB_1/AN_1	-440	2295
77	PB_2/AN_2	-695	2295
78	PB_3/AN_3	-801	2295
79	PB_4/AN_4	-996	2295
80	PB_5/AN_5	-1419	2295
81	PB_6/AN_6	-1530	2295

Note: * These values show the coordinates of the centers of pads. The accuracy is ±
home-point position is the chip's center and the center is located at half the d
between the upper and lower pads and left and right pads. Pad numbers 5, 6
are power supply (Vss) pads and must be connected. They should not be lef
number 9 (TEST) must be connected to the Vss position. The device will not
properly if the pads are not connected as indicated.

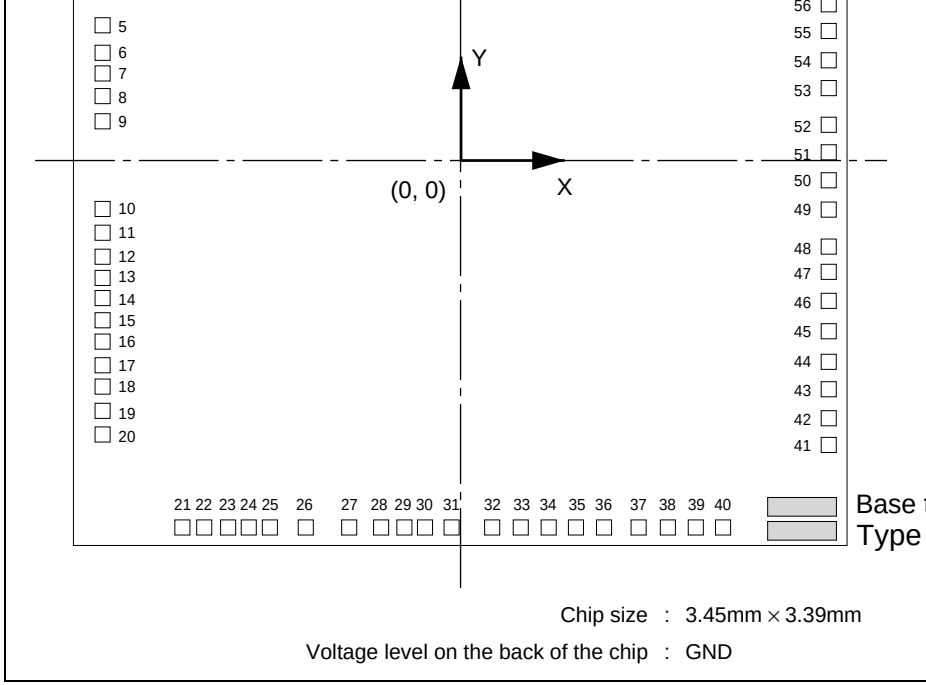


Figure 1.7 Bonding Pad Location Diagram of H8/38327 Group (Mask ROM V
H8/38427 Group (Mask ROM Version) (Top View)

4	X_2	-1605	843
5	Vss	-1605	619
6	OSC_2	-1605	503
7	OSC_1	-1605	405
8	TEST	-1605	299
9	$\overline{RES}$	-1605	201
10	$P1_0$/TMOW	-1605	-185
11	$P1_1$/TMOFL	-1605	-283
12	$P1_2$/TMOFH	-1605	-382
13	$P1_3$/TMIG	-1605	-480
14	$P1_4/\overline{IRQ_4}/\overline{ADTRG}$	-1605	-578
15	$P1_5/\overline{IRQ_1}$/TMIC	-1605	-676
16	$P1_6/\overline{IRQ_2}$	-1605	-775
17	$P1_7/\overline{IRQ_3}$/TMIF	-1605	-873
18	$P3_0$/PWM	-1605	-971
19	$P3_1$/UD/EXCL	-1605	-1070
20	$P3_2$	-1605	-1168
21	$P3_3/SCK_{31}$	-1262	-1577
22	$P3_4/RXD_{31}$	-1164	-1577
23	$P3_5/TXD_{31}$	-1066	-1577
24	$P3_6$/AEVH	-967	-1577
25	$P3_7$/AEVL	-869	-1577
26	CVcc	-704	-1577
27	Vss	-518	-1577
28	V3	-368	-1577
29	V2	-276	-1577
30	V1	-184	-1577

36	PA_0/COM1	611	-1577
37	$P5_0/\overline{WKP}_0/SEG_1$	767	-1577
38	$P5_1/\overline{WKP}_1/SEG_2$	892	-1577
39	$P5_2/\overline{WKP}_2/SEG_3$	1017	-1577
40	$P5_3/\overline{WKP}_3/SEG_4$	1141	-1577
41	$P5_4/\overline{WKP}_4/SEG_5$	1605	-1224
42	$P5_5/\overline{WKP}_5/SEG_6$	1605	-1100
43	$P5_6/\overline{WKP}_6/SEG_7$	1605	-975
44	$P5_7/\overline{WKP}_7/SEG_8$	1605	-850
45	$P6_0/SEG_9$	1605	-723
46	$P6_1/SEG_{10}$	1605	-598
47	$P6_2/SEG_{11}$	1605	-473
48	$P6_3/SEG_{12}$	1605	-349
49	$P6_4/SEG_{13}$	1605	-195
50	$P6_5/SEG_{14}$	1605	-70
51	$P6_6/SEG_{15}$	1605	55
52	$P6_7/SEG_{16}$	1605	179
53	$P7_0/SEG_{17}$	1605	336
54	$P7_1/SEG_{18}$	1605	460
55	$P7_2/SEG_{19}$	1605	585
56	$P7_3/SEG_{20}$	1605	710
57	$P7_4/SEG_{21}$	1605	835
58	$P7_5/SEG_{22}$	1605	959
59	$P7_6/SEG_{23}$	1605	1084
60	$P7_7/SEG_{24}$	1605	1209
61	$P8_0/SEG_{25}$	1130	1577

67	$P8_6/SEG_{31}$	382	1577
68	$P8_7/SEG_{32}$	257	1577
69	$P4_0/SCK_{32}$	-4	1577
70	$P4_1/RXD_{32}$	-97	1577
71	$P4_2/TXD_{32}$	-196	1577
72	$P4_3/\overline{IRQ_0}$	-294	1577
73	AVcc	-470	1577
74	PB_0/AN_0	-598	1577
75	PB_1/AN_1	-704	1577
76	PB_2/AN_2	-810	1577
77	PB_3/AN_3	-916	1577
78	PB_4/AN_4	-1022	1577
79	PB_5/AN_5	-1128	1577
80	PB_6/AN_6	-1233	1577

Note: * These values show the coordinates of the centers of pads. The accuracy is ±
home-point position is the chip's center and the center is located at half the d
between the upper and lower pads and left and right pads. Pad numbers 2, 5
are power supply (V_{SS}) pads and must be connected. They should not be lef
number 8 (TEST) must be connected to the Vss position. The device will not
properly if the pads are not connected as indicated.

Power source pins	V_{CC} CV_{CC}	32 26	34 28	Input	**Power supply:** All V_{CC} pins connected to the system pow See section 14, Power Supp a CV_{CC} pin (V_{CC} pin in the H Group).
	V_{SS}	5 27	7 29	Input	**Ground:** All V_{SS} pins should connected to the system pow (0 V).
	AV_{CC}	73	75	Input	**Analog power supply:** This power supply pin for the A/D When the A/D converter is n connect this pin to the syste supply.
	AV_{SS}	2	4	Input	**Analog ground:** This is the converter ground pin. It sho connected to the system pow (0V).
	V_0	31	33	Output	**LCD power supply:** These power supply pins for the LC controller/driver. They incor power supply split-resistance normally used with V_0 and V
	V_1 V_2 V_3	30 29 28	32 31 30	Input	

	X_1	3	5	Input	These pins connect to a 32.7
	X_2	4	6	Output	38.4-kHz crystal oscillator. See section 4, Clock Pulse Generators, for a typical conn diagram.
	EXCL	19	—	Input	This pin connects to a 32.768 38.4-kHz external clock. See Clock Pulse Generators, for ty connection diagram. This func available on the H8/38327 Gr H8/38427 Group.
System control	$\overline{RES}$	9	11	Input	**Reset:** When this pin is driver chip is reset
	$\overline{RESO}$	20	22	Output	**Reset output:** Outputs the Cl reset signal. This function is n implemented in the H8/38327 H8/38427 Group.
	TEST	8	10	Input	**Test pin:** This pin is reserved cannot be used. It should be to V_{SS}.
Interrupt pins	$\overline{IRQ}_0$ $\overline{IRQ}_1$ $\overline{IRQ}_2$ $\overline{IRQ}_3$ $\overline{IRQ}_4$	72 15 16 17 14	74 17 18 19 16	Input	**IRQ interrupt request 0 to 4** input pins for edge-sensitive e interrupts, with a selection of falling edge
	$\overline{WKP}_7$ to $\overline{WKP}_0$	44 to 37	46 to 39	Input	**Wakeup interrupt request 0** These are input pins for rising edge-sensitive external interru
Timer pins	TMOW	10	12	Output	**Clock output:** This is an outp waveforms generated by the t output circuit.
	AEVL AEVH	25 24	27 26	Input	**Asynchronous event counte input:** This is an event input p to the asynchronous event co

					and as an up-counter when
	TMIF	17	19	Input	**Timer F event input:** This is input pin for input to the time
	TMOFL	11	13	Output	**Timer FL output:** This is an for waveforms generated by output compare function.
	TMOFH	12	14	Output	**Timer FH output:** This is an for waveforms generated by output compare function.
	TMIG	13	15	Input	**Timer G capture input:** This pin for timer G input capture
14-bit PWM pin	PWM	18	20	Output	**14-bit PWM output:** This is pin for waveforms generated bit PWM
I/O ports	PB_7 to PB_0	1, 80 to 74	3 to 1, 80 to 76	Input	**Port B:** This is an 8-bit input
	$P4_3$	72	74	Input	**Port 4 (bit 3):** This is a 1-bit
	$P4_2$ to $P4_0$	71 to 69	73 to 71	I/O	**Port 4 (bits 2 to 0):** This is a port. Input or output can be for each bit by means of port register 4 (PCR4).
	PA_3 to PA_0	33 to 36	35 to 38	I/O	**Port A:** This is a 4-bit I/O po output can be designated for means of port control registe
	$P1_7$ to $P1_0$	17 to 10	19 to 12	I/O	**Port 1:** This is an 8-bit I/O p output can be designated for means of port control registe

					accessed by the user. With th version, pull up pin $P3_2$ to hig cancel a reset in the in the us
	$P5_7$ to $P5_0$	44 to 37	46 to 39	I/O	**Port 5:** This is an 8-bit I/O por output can be designated for means of port control register
	$P6_7$ to $P6_0$	52 to 45	54 to 47	I/O	**Port 6:** This is an 8-bit I/O por output can be designated for means of port control register
	$P7_7$ to $P7_0$	60 to 53	62 to 55	I/O	**Port 7:** This is an 8-bit I/O por output can be designated for means of port control register
	$P8_7$ to $P8_0$	68 to 61	70 to 63	I/O	**Port 8:** This is an 8-bit I/O por output can be designated for means of port control register When the on-chip emulator is $P8_5$, $P8_6$, and $P8_7$ are reserve exclusively by the emulator ar cannot be accessed by the us
Serial communi-cation interface (SCI)	RXD_{31}	22	24	Input	**SCI3-1 receive data input:** This is the SCI31 data input p
	TXD_{31}	23	25	Output	**SCI3-1 transmit data output** This is the SCI31 data output
	SCK_{31}	21	23	I/O	**SCI3-1 clock I/O:** This is the SCI31 clock I/O pi
	RXD_{32}	70	72	Input	**SCI3-2 receive data input:** This is the SCI32 data input p
	TXD_{32}	71	73	Output	**SCI3-2 transmit data output** This is the SCI32 data output
	SCK_{32}	69	71	I/O	**SCI3-2 clock I/O:** This is the SCI32 clock I/O pi

					the A/D converter.
LCD controller/ driver	COM_4 to COM_1	33 to 36	35 to 38	Output	**LCD common output:** Thes LCD common output pins.
	SEG_{32} to SEG_1	68 to 37	70 to 39	Output	**LCD segment output:** Thes LCD segment output pins.
	CL_1	68	70	Output	**LCD latch clock:** This is the for the segment external exp display data latch clock. This not implemented in the H8/3 and H8/38427 Group.
	CL_2	67	69	Output	**LCD shift clock:** This is the for the segment external exp display data shift clock. This not implemented in the H8/3 and H8/38427 Group.
	DO	66	68	Output	**LCD serial data output:** Thi output pin for segment exter expansion serial display data function is not implemented H8/38327 Group and H8/384
	M	65	67	Output	**LCD alternation signal:** Thi output pin for the segment e expansion LCD alternation s function is not implemented H8/38327 Group and H8/384

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: 0.25 µs*
  — 8 × 8-bit multiply: 1.75 µs*
  — 16 ÷ 8-bit divide: 1.75 µs*
  Note: * These values are at ϕ = 8 MHz.
- Low-power operation modes
  SLEEP instruction for transfer to low-power operation

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

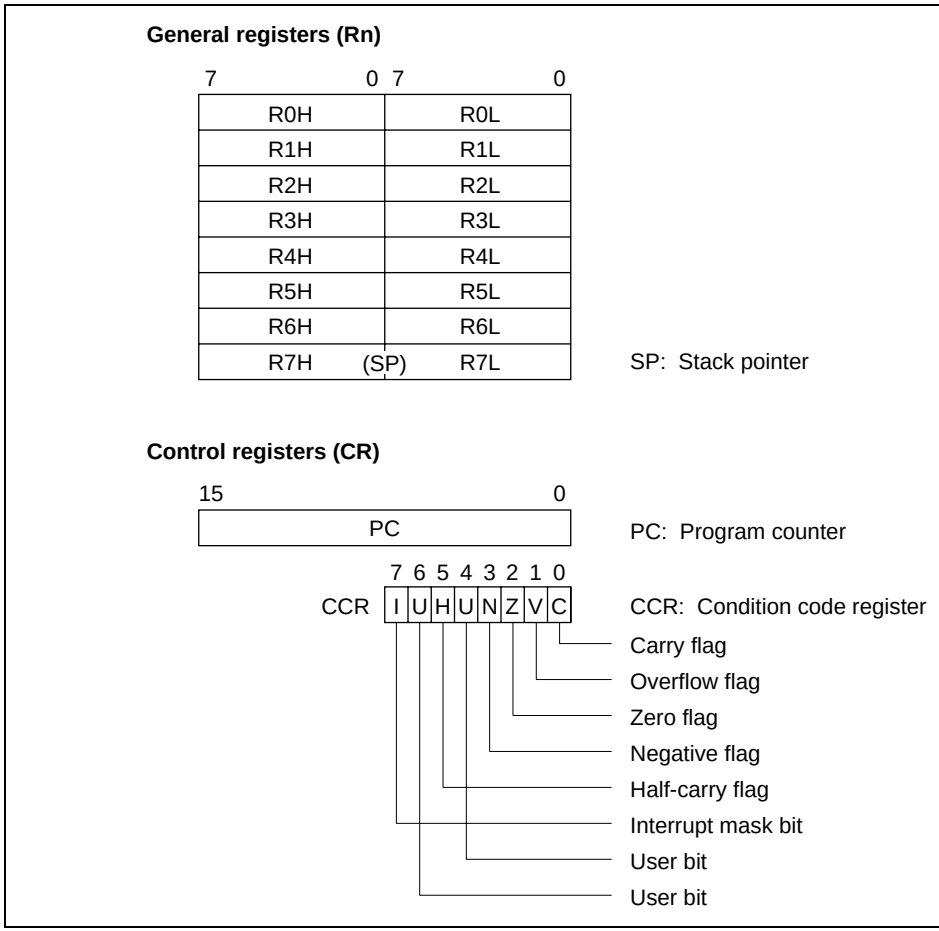


Figure 2.1 CPU Registers

When used as address registers, the general registers are accessed as 16-bit registers (

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

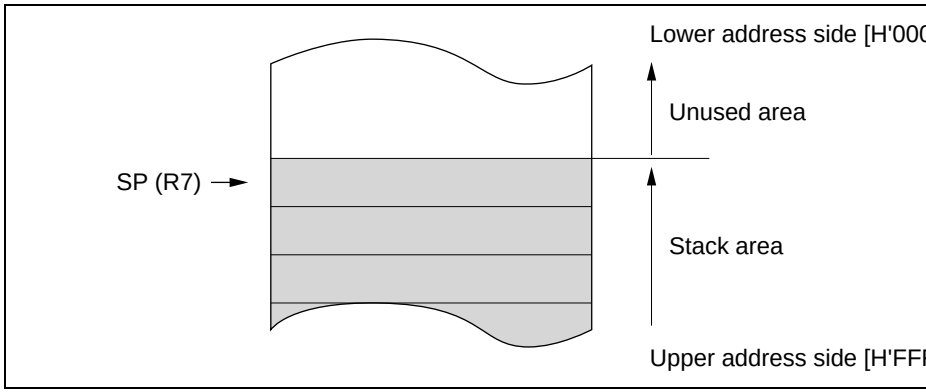


Figure 2.2 Stack Pointer

2.2.2 Control Registers

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

Program Counter (PC)

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

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, instruction is executed, this flag is set to 1 if there is a carry or borrow at bit 3, and is cl 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 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 a instruction.

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

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

Bit 0—Carry Flag (C): Set to 1 when a carry occurs, and cleared to 0 otherwise. Used

- Add instructions, to indicate a carry
- Subtract instructions, to indicate a borrow
- Shift and rotate instructions, to store the value shifted out of the end bit

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

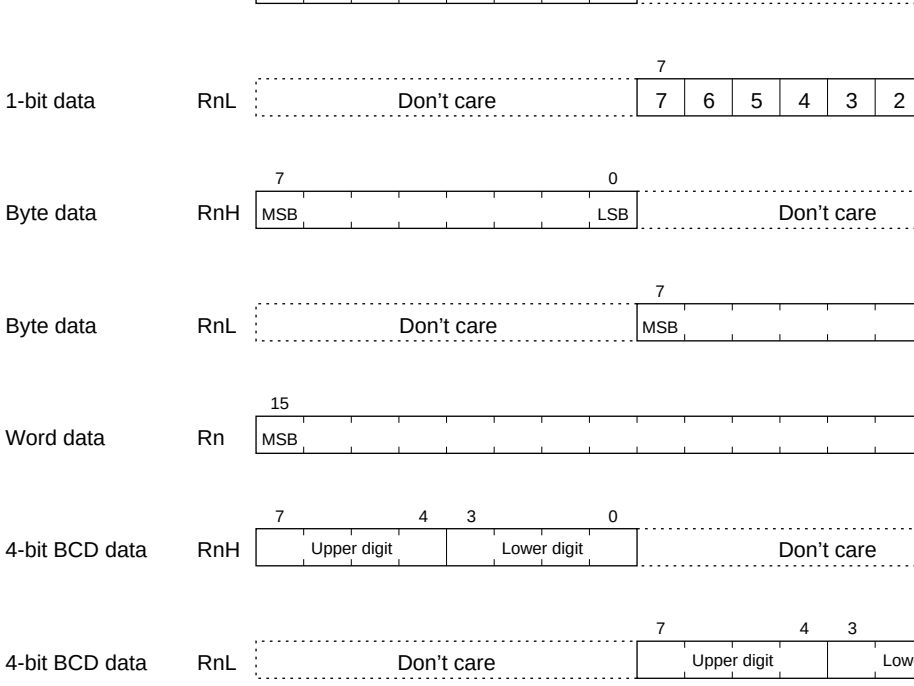
Some instructions leave some or all of the flag bits unchanged.

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-
data.

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



Legend:
RnH: Upper byte of general register
RnL: Lower byte of general register
MSB: Most significant bit
LSB: Least significant bit

Figure 2.3 Register Data Formats

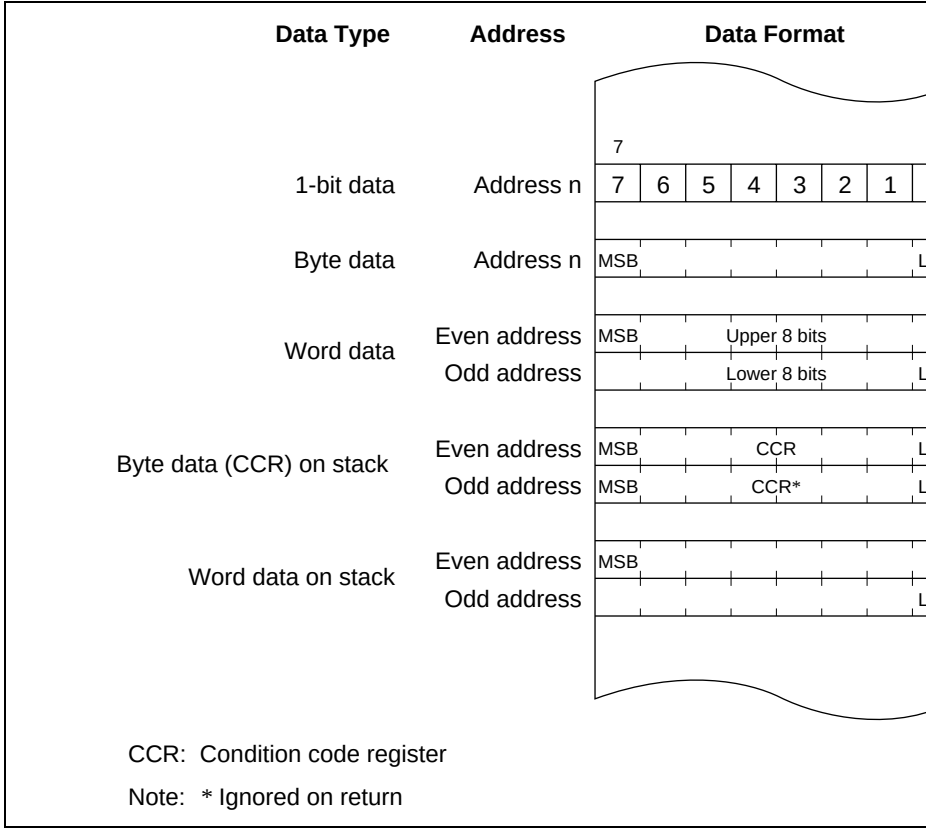


Figure 2.4 Memory Data Formats

When the stack is accessed using R7 as an address register, word access should alway performed. When the CCR is pushed on the stack, two identical copies of the CCR are make a complete word. When they are restored, the lower byte is ignored.

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 Register indirect with pre-decrement	@Rn+ @–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 g register containing the operand.

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

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

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

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

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

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

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

   The absolute address may be 8 bits long (@aa:8) or 16 bits long (@aa:16). The M
manipulation instructions can use 8-bit absolute addresses. The MOV.B, MOV.W,
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 add
H'FF00 to H'FFFF (65280 to 65535).

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

   The ADDS and SUBS instructions implicitly contain the value 1 or 2 as immediat
bit manipulation instructions contain 3-bit immediate data in the second or fourth b
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-extend
and added to the program counter contents to generate a branch destination addres
possible branching range is –126 to +128 bytes (–63 to +64 words) from the curren
The displacement should be an even number.

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

information.

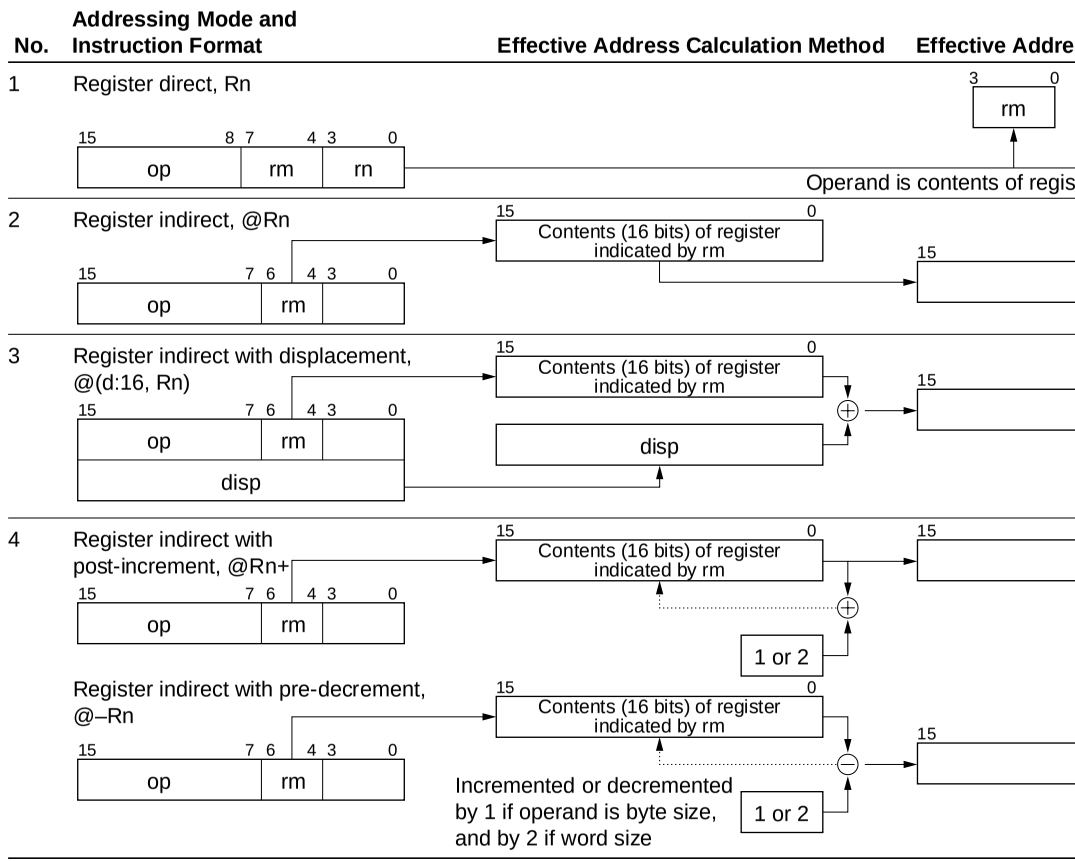
2.4.2 Effective Address Calculation

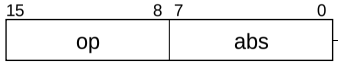
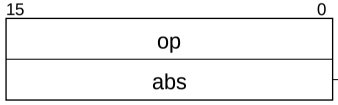
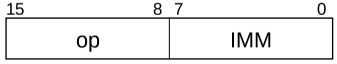
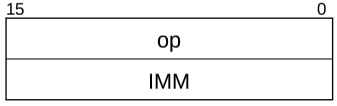
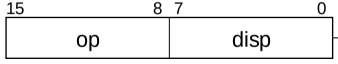
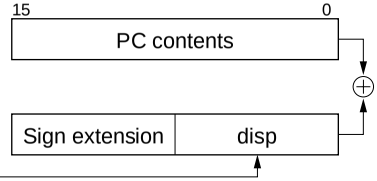
Table 2.2 shows how effective addresses are calculated in each of the addressing mode

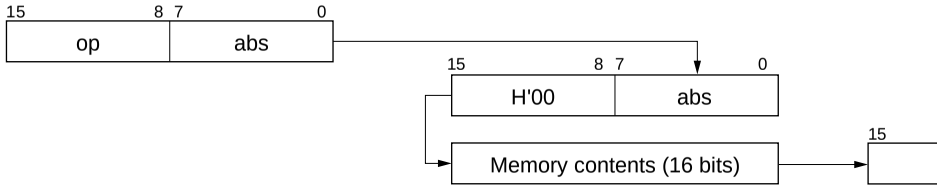
Arithmetic and logic instructions use register direct addressing (1). The ADD.B, ADDX CMP.B, AND, OR, and XOR instructions can also use immediate addressing (6).

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

Bit manipulation instructions can use register direct (1), register indirect (2), or 8-bit ab addressing (5) to specify the operand. Register indirect (1) (BSET, BCLR, BNOT, and instructions) or 3-bit immediate addressing (6) can be used independently to specify a b in the operand.


No.
Addressing Mode and Instruction Format
Effective Address Calculation Method
Effective Addre
1
Register direct, Rn
15
8 7
4 3
0
op
rm
rn
3
0
rm
Operand is contents of regis
2
Register indirect, @Rn
15
7 6
4 3
0
op
rm
15
0
Contents (16 bits) of register indicated by rm
15
3
Register indirect with displacement, @(d:16, Rn)
15
7 6
4 3
0
op
rm
disp
15
0
Contents (16 bits) of register indicated by rm
disp
15
4
Register indirect with post-increment, @Rn+
15
7 6
4 3
0
op
rm
15
0
Contents (16 bits) of register indicated by rm
1 or 2
15
Register indirect with pre-decrement, @–Rn
15
7 6
4 3
0
op
rm
15
0
Contents (16 bits) of register indicated by rm
15
Incremented or decremented by 1 if operand is byte size, and by 2 if word size
1 or 2

No.	Addressing Mode and Instruction Format	Effective Address Calculation Method	Effecti
5	Absolute address @aa:8  @aa:16  		15 H 15
6	Immediate #xx:8  #xx:16  		Operand is
7	Program-counter relative @(d:8, PC)  	 	15

No.	Addressing Mode and Instruction Format	Effective Address Calculation Method	Effecti
8	Memory indirect, @@aa:8		

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

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

Notes: 1. PUSH Rn is equivalent to MOV.W Rn, @–SP.
POP Rn is equivalent to MOV.W @SP+, Rn. The same applies to the machi
language.
2. Bcc is a conditional branch instruction in which cc represents a condition coc

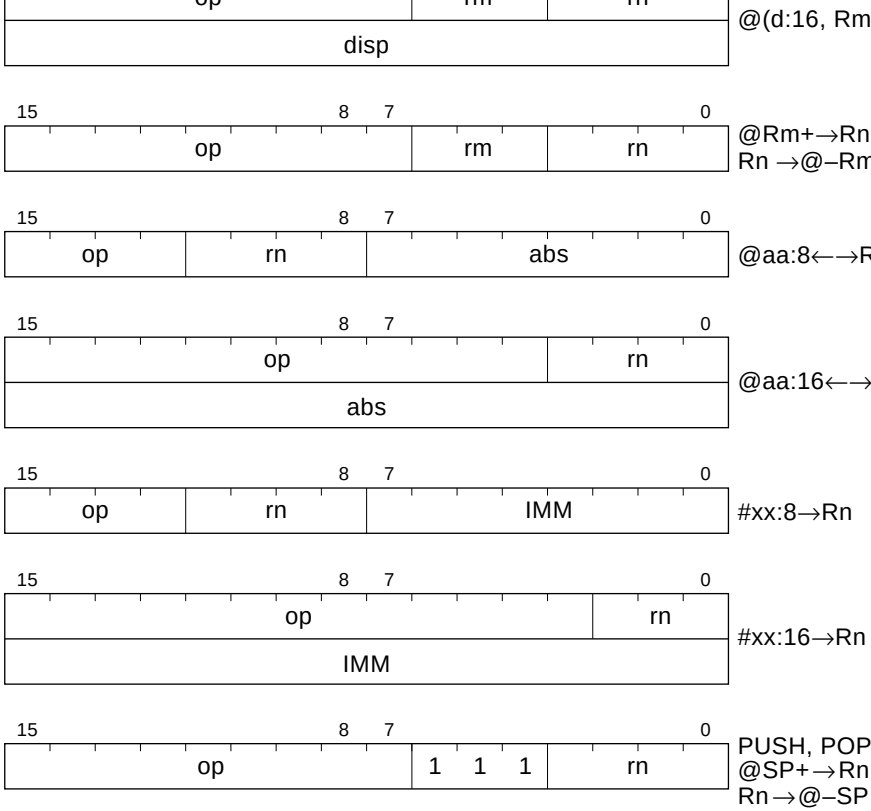
The following sections give a concise summary of the instructions in each category, an
the bit patterns of their object code. The notation used is defined next.

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

		Moves data between two general registers or between a register and memory, or moves immediate data to a ger register. The Rn, @Rn, @(d:16, Rn), @aa:16, #xx:16, @–Rn, ar addressing modes are available for word data. The @a addressing mode is available for byte data only. The @–R7 and @R7+ modes require word operands. I specify byte size for these two modes.
POP	W	@SP+ → Rn Pops a 16-bit general register from the stack. Equivaler MOV.W @SP+, Rn.
PUSH	W	Rn → @–SP Pushes a 16-bit general register onto the stack. Equiva MOV.W Rn, @–SP.

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

Certain precautions are required in data access. See section 2.9.1, Notes on Data Acces details.



Legend:

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

Figure 2.5 Data Transfer Instruction Codes

		or addition on immediate data and data in a general reg Immediate data cannot be subtracted from data in a ger register. Word data can be added or subtracted only wh words are in general registers.
ADDX SUBX	B	Rd ± Rs ± C → Rd, Rd ± #IMM ± C → Rd Performs addition or subtraction with carry or borrow on in two general registers, or addition or subtraction on im data and data in a general register.
INC DEC	B	Rd ± 1 → Rd Increments or decrements a general register by 1.
ADDS SUBS	W	Rd ± 1 → Rd, Rd ± 2 → Rd Adds or subtracts 1 or 2 to or from a general register
DAA DAS	B	Rd decimal adjust → Rd Decimal-adjusts (adjusts to 4-bit BCD) an addition or su 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 general registers, providing a 16-bit result
DIVXU	B	Rd ÷ Rs → Rd Performs 16-bit ÷ 8-bit unsigned division on data in two registers, providing an 8-bit quotient and 8-bit remainde
CMP	B/W	Rd – Rs, Rd – #IMM Compares data in a general register with data in anothe register or with immediate data, and indicates the result CCR. Word data can be compared only between two g registers.
NEG	B	0 – Rd → Rd Obtains the two's complement (arithmetic complement) general register

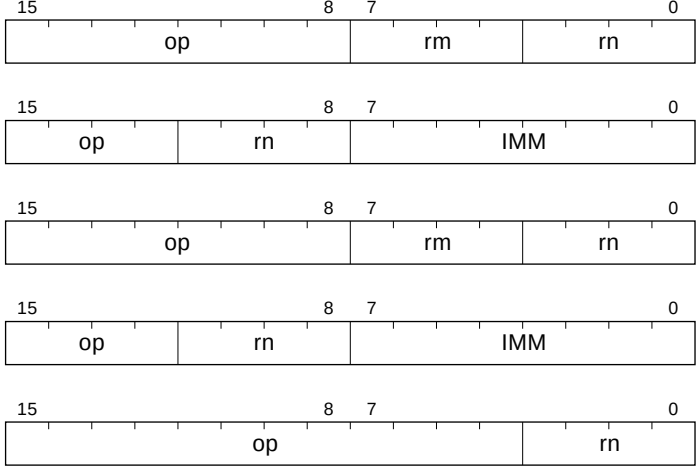
Note: * Size: Operand size
B: Byte
W: Word

		another general register or immediate data
OR	B	Rd ∨ Rs → Rd, Rd ∨ #IMM → Rd Performs a logical OR operation on a general register general register or immediate data
XOR	B	Rd ⊕ Rs → Rd, Rd ⊕ #IMM → Rd Performs a logical exclusive OR operation on a genera and another general register or immediate data
NOT	B	~ Rd → Rd Obtains the one's complement (logical complement) of register contents

Note: * Size: Operand size
B: Byte

SHAR		Performs an arithmetic shift operation on general registe
SHLL SHLR	B	Rd shift → Rd Performs a logical shift operation on general register co
ROTL ROTR	B	Rd rotate → Rd Rotates general register contents
ROTXL ROTXR	B	Rd rotate through carry → Rd Rotates general register contents through the C (carry)

Note: * Size: Operand size
B: Byte



Legend:
op: Operation field
rm, rn: Register field
IMM: Immediate data

Figure 2.6 Arithmetic, Logic, and Shift Instruction Codes

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

BLD	B	(<bit-No.> of <EAd>) → C Copies a specified bit in a general register or memory
BILD	B	~ (<bit-No.> of <EAd>) → C Copies the inverse of a specified bit in a general regist memory to the C flag. The bit number is specified by 3-bit immediate data.
BST	B	C → (<bit-No.> of <EAd>) Copies the C flag to a specified bit in a general registe
BIST	B	~ C → (<bit-No.> of <EAd>) Copies the inverse of the C flag to a specified bit in a g register or memory. The bit number is specified by 3-bit immediate data.

Note: * Size: Operand size
B: Byte

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

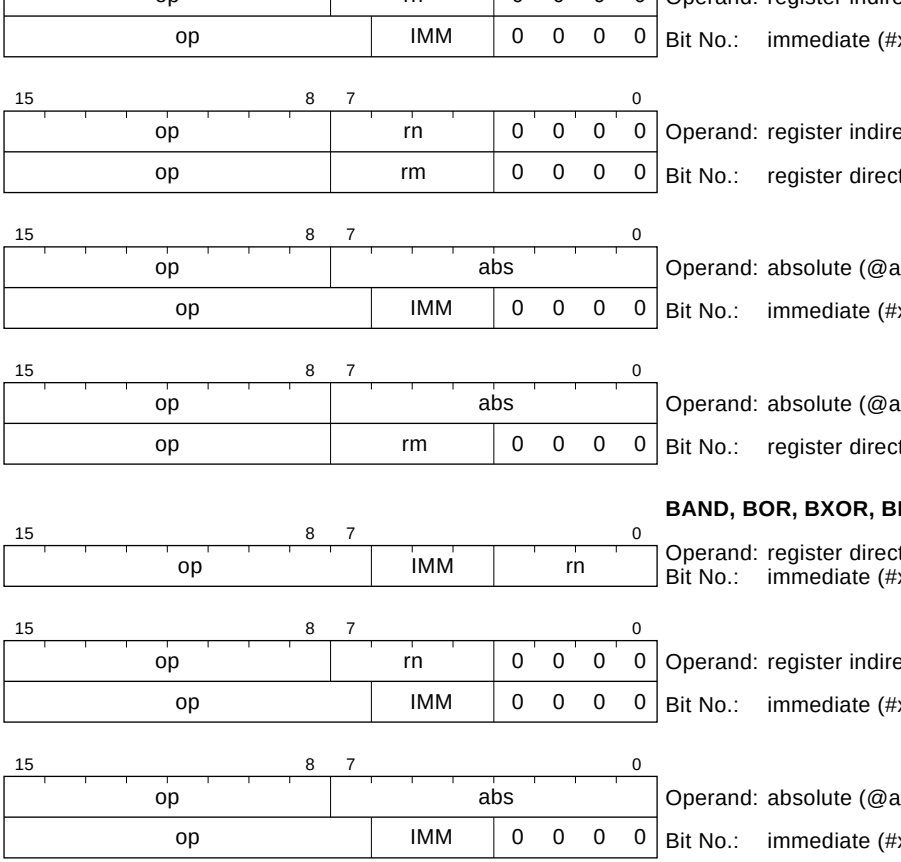
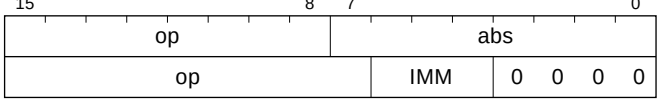


Figure 2.7 Bit Manipulation Instruction Codes



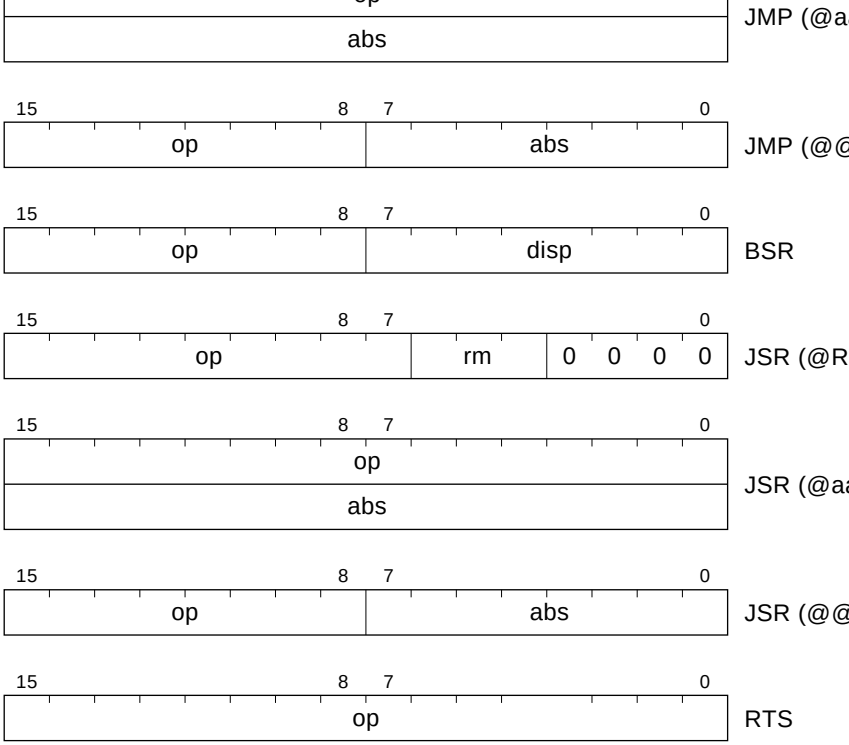
Legend:
op: Operation field
rm, rn: Register field
abs: Absolute address
IMM: Immediate data

Figure 2.7 Bit Manipulation Instruction Codes (cont)

branching conditions are given below.

Mnemonic	Description	Cond
BRA (BT)	Always (true)	Alway
BRN (BF)	Never (false)	Never
BHI	High	C ∨ Z
BLS	Low or same	C ∨ Z
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 ⊕ V
BLT	Less than	N ⊕ V
BGT	Greater than	Z ∨ (N
BLE	Less or equal	Z ∨ (N

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



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

Figure 2.8 Branching Instruction Codes

SLEEP	—	Causes a transition from active mode to a power-down section 5, Power-Down Modes, for details.
LDC	B	Rs → CCR, #IMM → CCR Moves immediate data or general register contents to th code register
STC	B	CCR → Rd Copies the condition code register to a specified genera
ANDC	B	CCR ∧ #IMM → CCR Logically ANDs the condition code register with immedia
ORC	B	CCR ∨ #IMM → CCR Logically ORs the condition code register with immediat
XORC	B	CCR ⊕ #IMM → CCR Logically exclusive-ORs the condition code register with data
NOP	—	PC + 2 → PC Only increments the program counter

Note: * Size: Operand size
B: Byte

		XORC, LDC

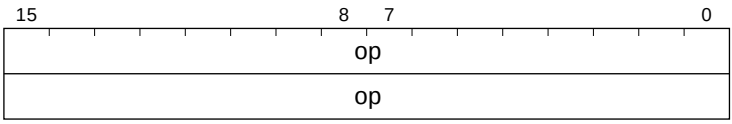
Legend:
op: Operation field
rn: Register field
IMM: Immediate data

Figure 2.9 System Control Instruction Codes

```
        repeat  @R5+ → @R6+
                R4L –1 → R4L
        until   R4L = 0
else next;
```

Block transfer instruction. Transfers the number of data specified by R4L from locations starting at the address i R5 to locations starting at the address indicated by R6. transfer, the next instruction is executed.

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



Legend:
op: Operation field

Figure 2.10 Block Data Transfer Instruction Code

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

Access to on-chip memory takes place in two states. The data bus width is 16 bits, all access in byte or word size. Figure 2.11 shows the on-chip memory access cycle.

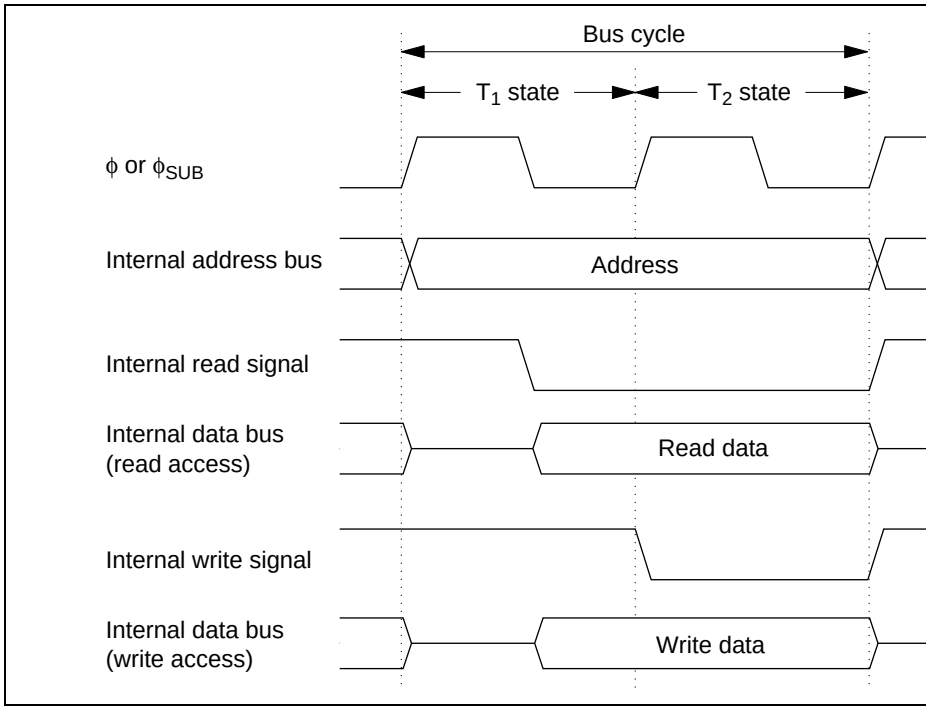


Figure 2.11 On-Chip Memory Access Cycle

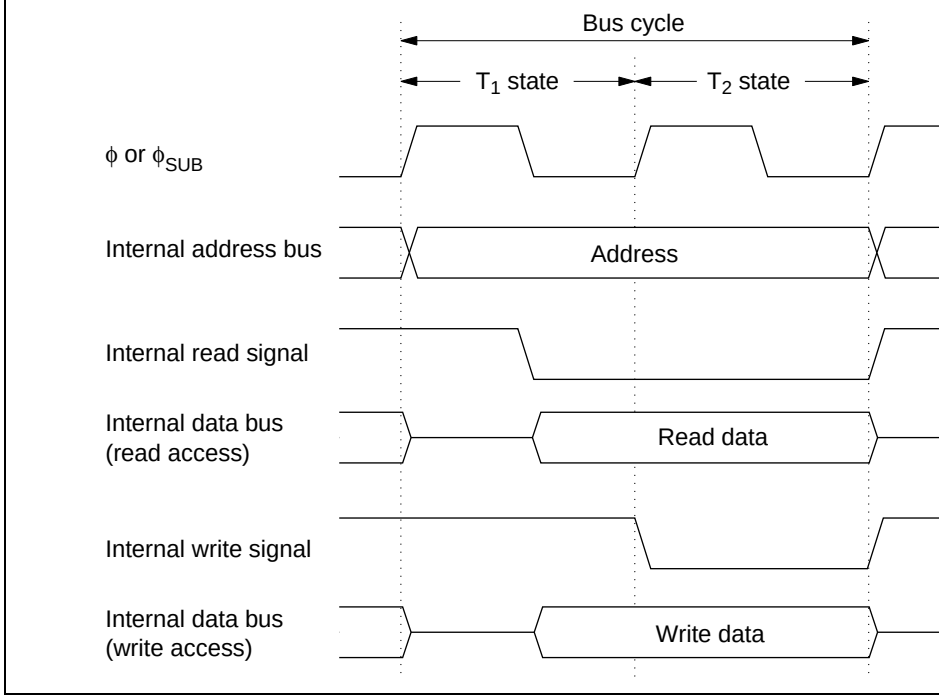


Figure 2.12 On-Chip Peripheral Module Access Cycle (2-State Access)

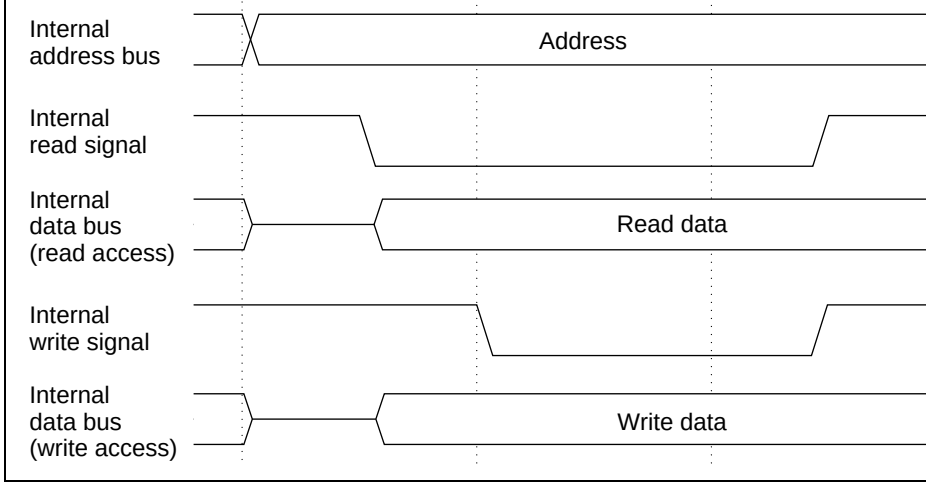
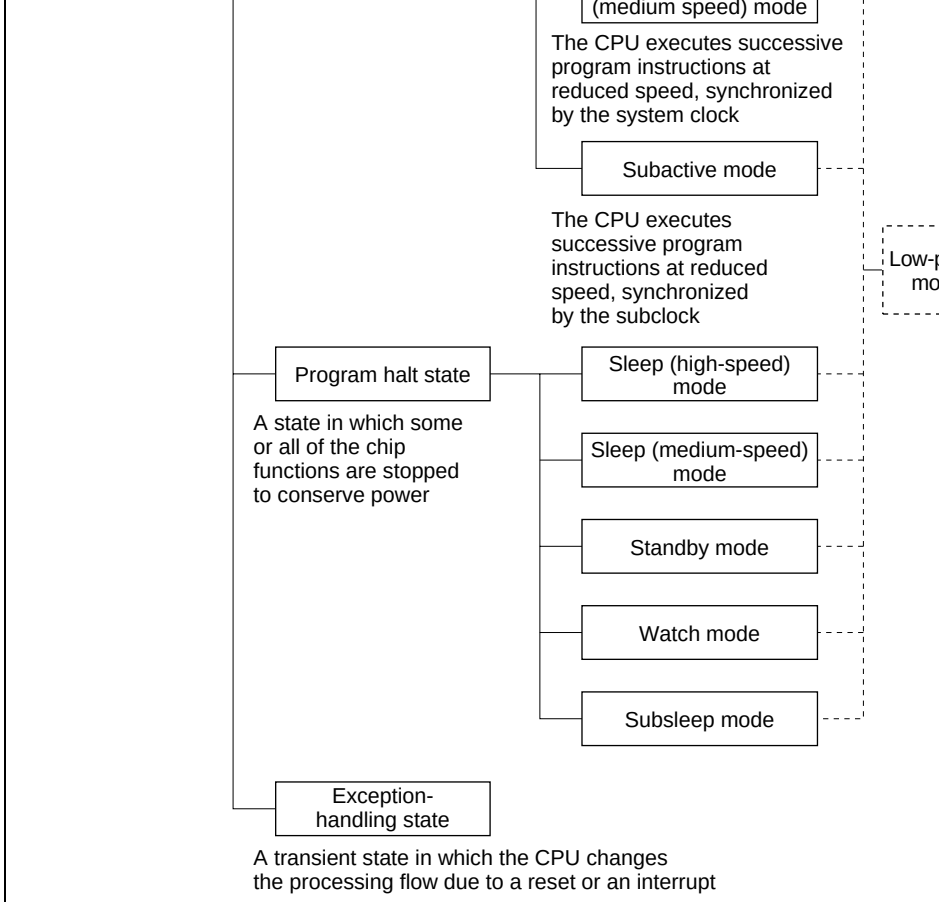


Figure 2.13 On-Chip Peripheral Module Access Cycle (3-State Acces

2.7 CPU States

2.7.1 Overview

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



Note: See section 5, Power-Down Modes, for details on the modes and their transitions

Figure 2.14 CPU Operation States

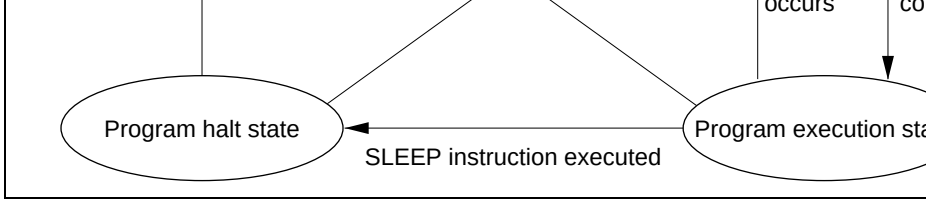


Figure 2.15 State Transitions

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) subactive mode. Operation is synchronized with the system clock in active mode (hig medium speed), and with the subclock in subactive mode. See section 5, Power-Down details on these modes.

2.7.3 Program Halt State

In the program halt state there are five modes: two sleep modes (high speed and mediu standby mode, watch mode, and subsleep mode. See section 5, Power-Down Modes f these modes.

2.7.4 Exception-Handling State

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

For details on interrupt handling, see section 3.3, Interrupts.

2.16 (5), and that of the H8/3827R, H8/3827S, H8/38327, and H8/38427 in figure 2.16

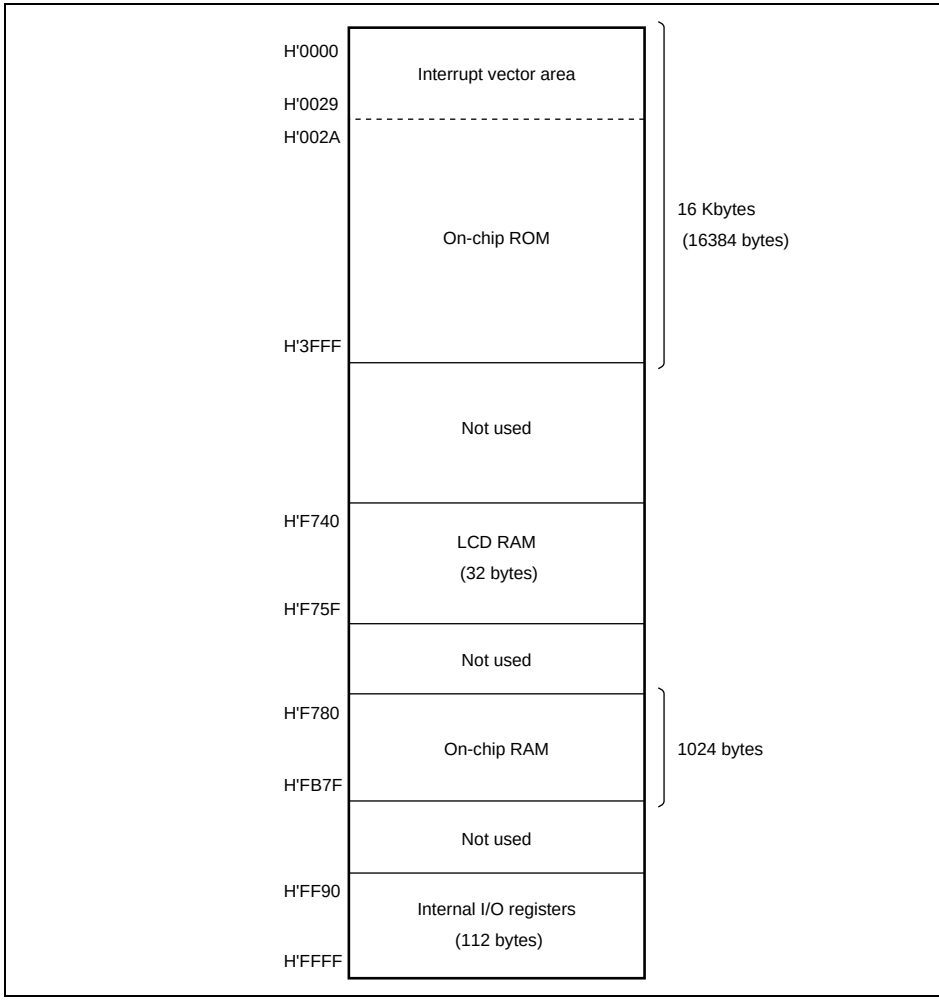


Figure 2.16 (1) H8/3822R, H8/38322, and H8/38422 Memory Map

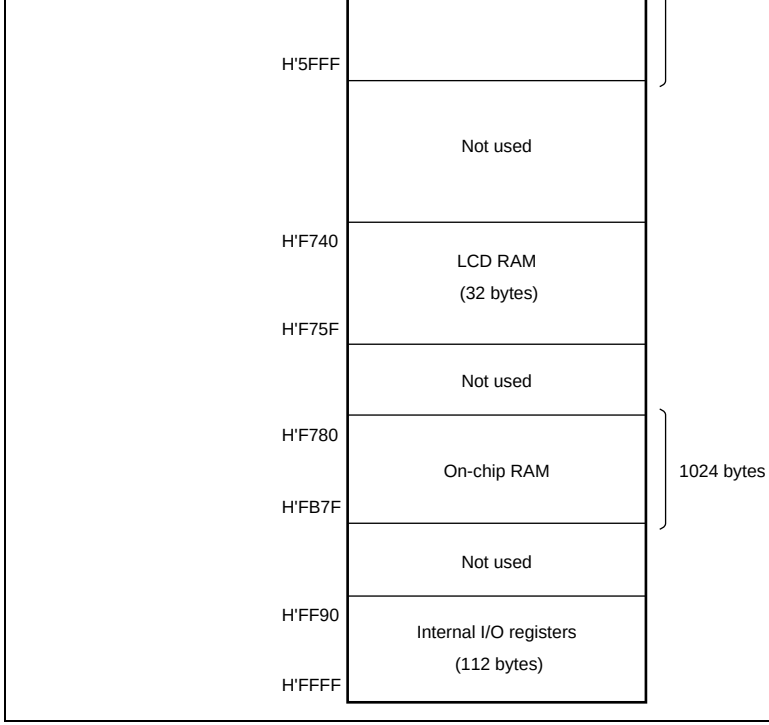
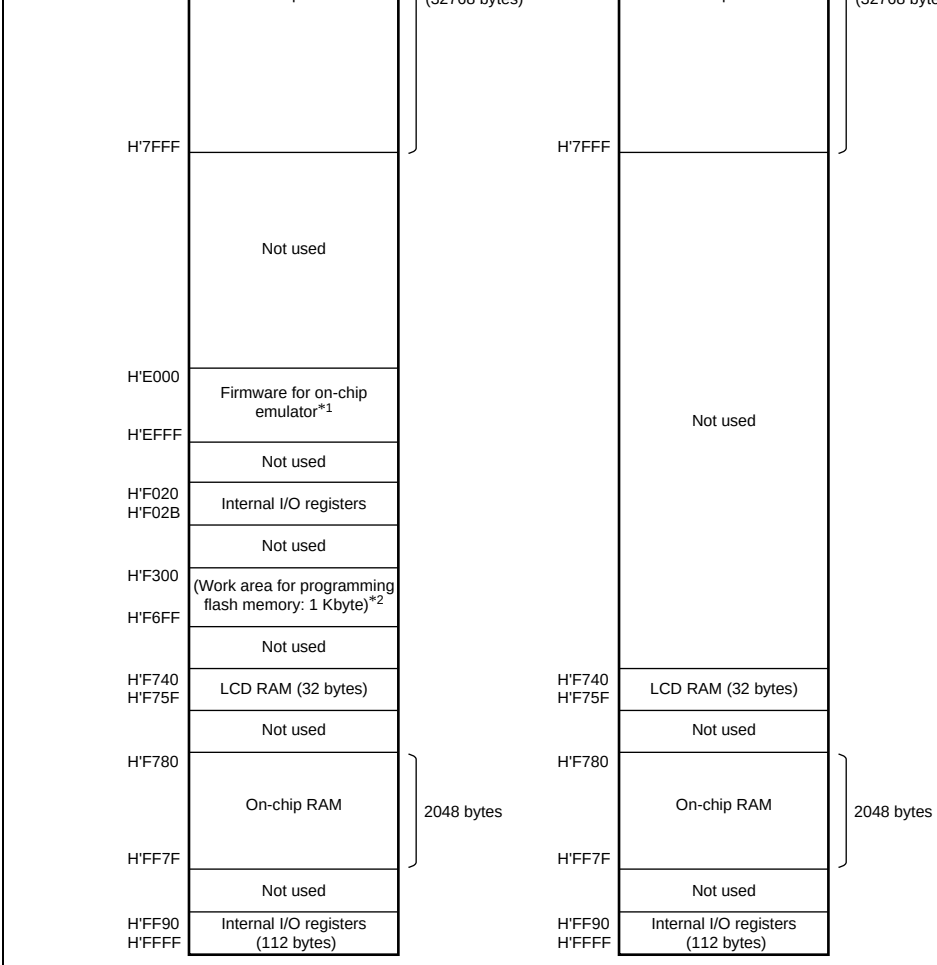


Figure 2.16 (2) H8/3823R, H8/38323, and H8/38423 Memory Map



Notes: 1. Not accessible by the user when the on-chip emulator is used.
2. A programming control program is used to program flash memory. Do not use a user program to perform programming when the on-chip emulator is used. This area is not used in the mask ROM version.

Figure 2.16 (3) H8/3824R, H8/3824S, H8/38324, and H8/38424 Memory M

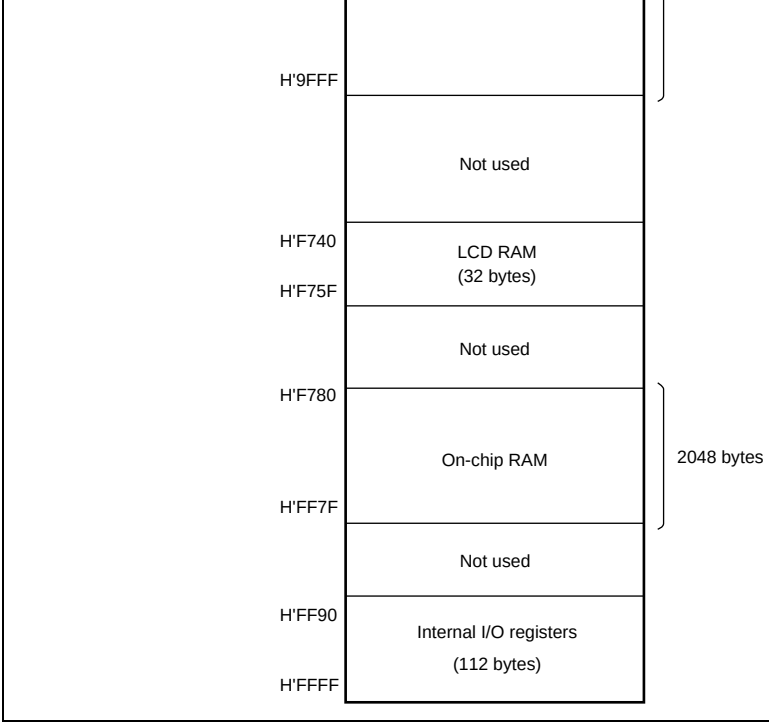


Figure 2.16 (4) H8/3825R, H8/3825S, H8/38325, and H8/38425 Memory

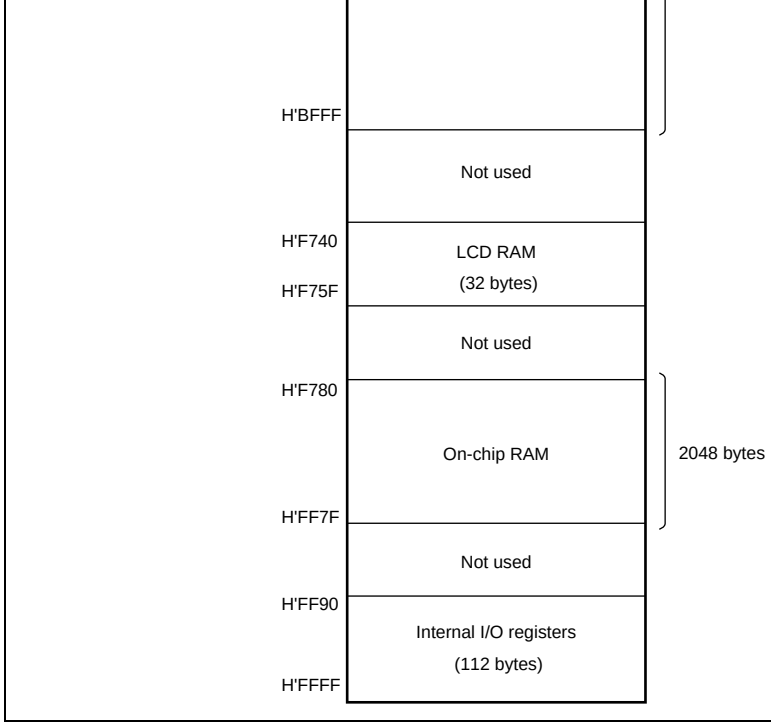
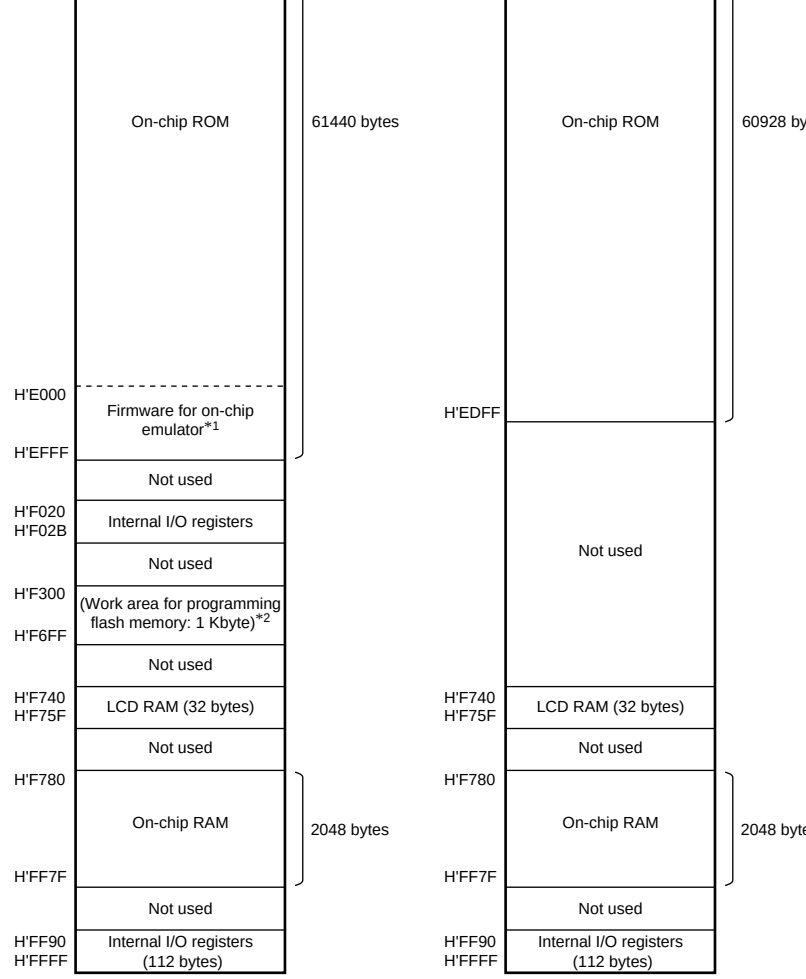


Figure 2.16 (5) H8/3826R, H8/3826S, H8/38326, and H8/38426 Memory M



Notes: 1. Not accessible by the user when the on-chip emulator is used.
2. A programming control program is used to program flash memory. Do not use a user program to perfo
programming when the on-chip emulator is used. This area is not used in the mask ROM version.

Figure 2.16 (6) H8/3827R, H8/3827S, H8/38327, and H8/38427 Memory

Data transfer from CPU to empty area:

The transferred data will be lost. This action may also cause the CPU to misope

Data transfer from empty area to CPU:

Unpredictable data is transferred.

2. Access to Internal I/O Registers:

   Internal data transfer to or from on-chip modules other than the ROM and RAM are use of an 8-bit data width. If word access is attempted to these areas, the following occur.

   Word access from CPU to I/O register area:

   Upper byte: Will be written to I/O register.

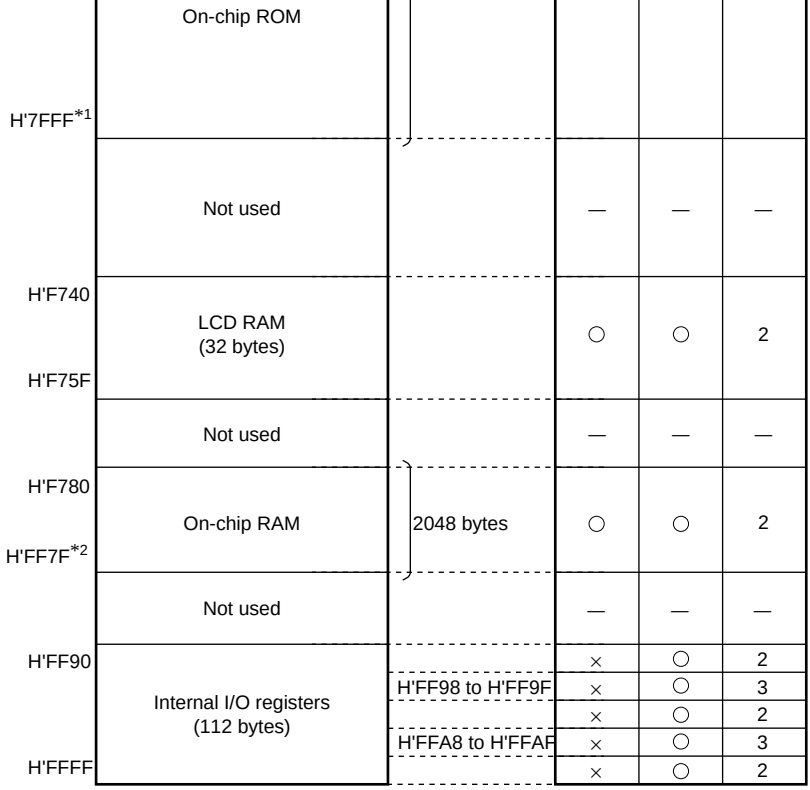
   Lower byte: Transferred data will be lost.

   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 re other than the on-chip ROM and RAM areas. Figure 2.17 shows the data size and num states in which on-chip peripheral modules can be accessed.



Notes: The example of the H8/3824R and H8/3824S is shown here.

1. This address is H'3FFF in the H8/3822R (16-Kbyte on-chip ROM), H'5FFF in th H8/3823R (24-Kbyte on-chip ROM), H'9FFF in the H8/3825R and H8/3825S (40 Kbyte on-chip ROM), H'BFFF in the H8/3826R and H8/3826S (48-Kbyte on-chip ROM), and H'EDFF in the H8/3827R and H8/3827S (60-Kbyte on-chip ROM).
2. This address is H'FB7F in the H8/3822R, and H8/3823R (1024 bytes of on-chip RAM).

Figure 2.17 Data Size and Number of States for Access to and from On-Chip Peripheral Modules

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

1. Bit Manipulation in Two Registers Assigned to the Same Address

Example 1: timer load register and timer counter

Figure 2.18 shows an example in which two timer registers share the same address. Wh
manipulation instruction accesses the timer load register and timer counter of a reloadab
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 ins
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
timer load register. As a result, bits other than the intended bit in the timer load registe
modified to the timer counter value.

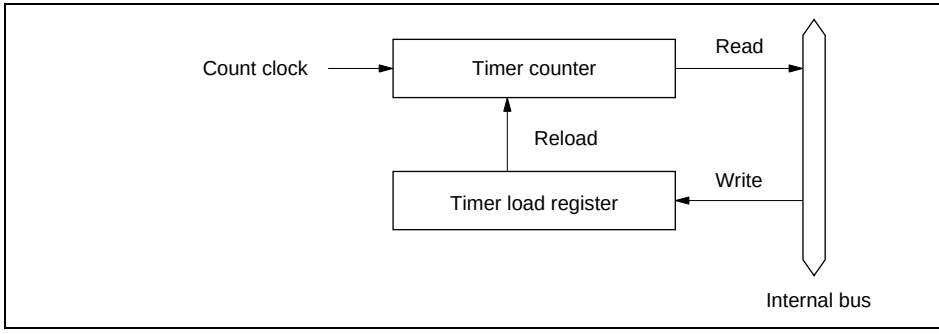


Figure 2.18 Timer Configuration Example

Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low lev
PCR3	0	0	1	1	1	1	1
PDR3	1	0	0	0	0	0	0

[B: BSET instruction executed]

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

The BSET instruction is executed designating por

[C: After executing BSET]

	$P3_7$	**$P3_6$**	**$P3_5$**	**$P3_4$**	**$P3_3$**	**$P3_2$**	**$P3_1$**
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low lev
PCR3	0	0	1	1	1	1	1
PDR3	0	1	0	0	0	0	0

[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-leve $P3_5$ to $P3_0$ are output pins, so the CPU reads the value in PDR3. In this example PDR 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. Finall 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 si However, bits 7 and 6 of PDR3 end up with different values.

	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low leve
PCR3	0	0	1	1	1	1	1
PDR3	1	0	0	0	0	0	0
RAM0	1	0	0	0	0	0	0

[B: BSET instruction executed]

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

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

[C: After executing BSET]

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

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

	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low leve
PCR3	0	0	1	1	1	1	1
PDR3	1	0	0	0	0	0	0
RAM0	1	0	0	0	0	0	0

[A: Prior to executing BCLR]

	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low lev
PCR3	0	0	1	1	1	1	1
PDR3	1	0	0	0	0	0	0

[B: BCLR instruction executed]

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

The BCLR instruction is executed designating PC

[C: After executing BCLR]

	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$
Input/output	Output	Output	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low lev
PCR3	1	1	1	1	1	1	1
PDR3	1	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 v
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, th
(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. Hov
and 6 in PCR3 change to 1, so that $P3_7$ and $P3_6$ change from input pins to output pins.

	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low leve
PCR3	0	0	1	1	1	1	1
PDR3	1	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1

[B: BCLR instruction executed]

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

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

[C: After executing BCLR]

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

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

	$P3_7$	$P3_6$	$P3_5$	$P3_4$	$P3_3$	$P3_2$	$P3_1$
Input/output	Input	Input	Output	Output	Output	Output	Output
Pin state	Low level	High level	Low level	Low level	Low level	Low level	Low leve
PCR3	0	0	1	1	1	1	1
PDR3	1	0	0	0	0	0	0
RAM0	0	0	1	1	1	1	1

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 A*	PDRA	H'FFDD

Note: * Port data registers have the same addresses as input pins.

Table 2.13 Registers with Write-Only Bits

Register Name	Abbr.	Address
Port control register 1	PCR1	H'FFE4
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 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

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

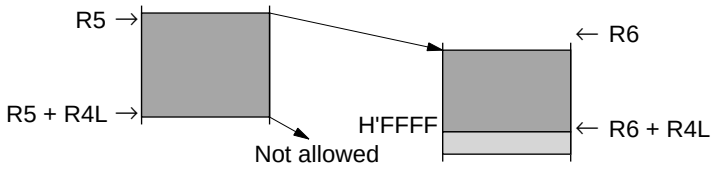


Table 3.1 Exception Handling Types and Priorities

Priority	Exception Source	Time of Start of Exception Handling
High	Reset	Exception handling starts as soon as the reset state
↑	Interrupt	When an interrupt is requested, exception handling s execution of the present instruction or the exception 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 registe 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 chip enters the res

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 stabilize
- Resetting during operation: Hold the $\overline{\text{RES}}$ pin low for at least 10 system clock cyc

Reset exception handling takes place as follows.

- The CPU internal state and the registers of on-chip peripheral modules are initializ 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'0 which the program starts executing from the address indicated in PC.

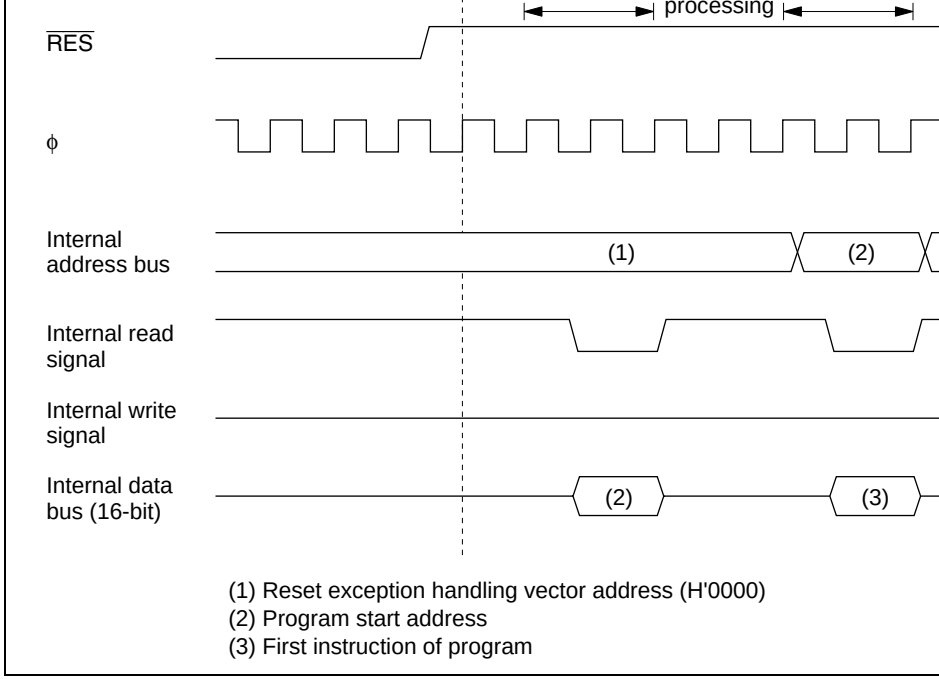


Figure 3.1 Reset Sequence

3.2.3 Interrupt Immediately after Reset

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

The interrupts have the following features:

- Internal and external interrupts can be masked by the I bit in CCR. When the I bit
  interrupt request flags can be set but the interrupts are not accepted.
- IRQ_4 to IRQ_0 and WKP_7 to WKP_0 can be set to either rising edge sensing or fallin
  sensing.

$\overline{IRQ_4}$	IRQ_4	8	H'0010 to H'0011
$\overline{WKP_0}$ $\overline{WKP_1}$ $\overline{WKP_2}$ $\overline{WKP_3}$ $\overline{WKP_4}$ $\overline{WKP_5}$ $\overline{WKP_6}$ $\overline{WKP_7}$	WKP_0 WKP_1 WKP_2 WKP_3 WKP_4 WKP_5 WKP_6 WKP_7	9	H'0012 to H'0013
Timer A	Timer A overflow	11	H'0016 to H'0017
Asynchronous counter	Asynchronous counter overflow	12	H'0018 to H'0019
Timer C	Timer C overflow or underflow	13	H'001A to H'001B
Timer FL	Timer FL compare match Timer FL overflow	14	H'001C to H'001D
Timer FH	Timer FH compare match Timer FH overflow	15	H'001E to H'001F
Timer G	Timer G input capture Timer G overflow	16	H'0020 to H'0021
SCI3-1	SCI3-1 transmit end SCI3-1 transmit data empty SCI3-1 receive data full SCI3-1 overrrun error SCI3-1 framing error SCI3-1 parity error	17	H'0022 to H'0023
SCI3-2	SCI3-2 transmit end SCI3-2 transmit data empty SCI3-2 receive data full SCI3-2 overrun error SCI3-2 framing error SCI3-2 parity error	18	H'0024 to H'0025
A/D	A/D conversion end	19	H'0026 to H'0027
(SLEEP instruction executed)	Direct transfer	20	H'0028 to H'0029

Note: Vector addresses H'0002 to H'0007 and H'0014 to H'0015 are reserved and can used.

Interrupt enable register 2	IENR2	R/W	H'00
Interrupt request register 1	IRR1	R/W*	H'20
Interrupt request register 2	IRR2	R/W*	H'00
Wakeup interrupt request register	IWPR	R/W*	H'00
Wakeup edge select register	WEGR	R/W	H'00

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

1. IRQ Edge Select Register (IEGR)

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

IEGR is an 8-bit read/write register used to designate whether pins $\overline{IRQ_4}$ to $\overline{IRQ_0}$ are s
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: IRQ_4 edge select (IEG4)

Bit 4 selects the input sensing of the $\overline{IRQ_4}$ pin and $\overline{ADTRG}$ pin.

Bit 4 IEG4	Description
0	Falling edge of $\overline{IRQ_4}$ and $\overline{ADTRG}$ pin input is detected
1	Rising edge of $\overline{IRQ_4}$ and $\overline{ADTRG}$ pin input is detected

Bit 2: IRQ_2 edge select (IEG2)

Bit 2 selects the input sensing of pin $\overline{IRQ_2}$.

Bit 2 IEG2	Description	
0	Falling edge of $\overline{IRQ_2}$ pin input is detected	(i
1	Rising edge of $\overline{IRQ_2}$ pin input is detected	

Bit 1: IRQ_1 edge select (IEG1)

Bit 3 selects the input sensing of the $\overline{IRQ_1}$ pin and TMIC pin.

Bit 1 IEG1	Description	
0	Falling edge of $\overline{IRQ_1}$ and TMIC pin input is detected	(i
1	Rising edge of $\overline{IRQ_1}$ and TMIC pin input is detected	

Bit 0: 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	(i
1	Rising edge of $\overline{IRQ_0}$ pin input is detected	

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 interrupt requests
1	Enables timer A interrupt requests

Bit 6: Reserved bit

Bit 6 is a readable/writable reserved bit. It is initialized to 0 by a reset.

Bit 5: Wakeup interrupt enable (IENWP)

Bit 5 enables or disables WKP_7 to WKP_0 interrupt requests.

Bit 5 IENWP	Description
0	Disables $\overline{WKP}_7$ to $\overline{WKP}_0$ interrupt requests
1	Enables $\overline{WKP}_7$ to $\overline{WKP}_0$ interrupt requests

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 requests from pin $\overline{IRQ}$n
1	Enables interrupt requests from pin $\overline{IRQ}$n

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	(i
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	(i
1	Enables A/D converter interrupt requests	

Bit 5: Reserved bit

Bit 5 is a readable/writable reserved bit. It is initialized to 0 by a reset.

Bit 4: Timer G interrupt enable (IENTG)

Bit 4 enables or disables timer G input capture or overflow interrupt requests.

Bit 4 IENTG	Description	
0	Disables timer G interrupt requests	(i
1	Enables timer G interrupt requests	

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 interrupt requests
1	Enables timer FL interrupt requests

Bit 1: Timer C interrupt enable (IENTC)

Bit 1 enables or disables timer C overflow and underflow interrupt requests.

Bit 1 IENTC	Description
0	Disables timer C interrupt requests
1	Enables timer C interrupt requests

Bit 0: Asynchronous event counter interrupt enable (IENEC)

Bit 0 enables or disables asynchronous event counter interrupt requests.

Bit 0 IENEC	Description
0	Disables asynchronous event counter interrupt requests
1	Enables asynchronous event counter interrupt requests

For details of SCI3-1 and SCI3-2 interrupt control, see section 10.2.6, Serial Control R
(SCR3).

IRR1 is an 8-bit read/write register, in which a corresponding flag is set to 1 when a tin
IRQ_4 to IRQ_0 interrupt is requested. The flags are not cleared automatically when an in
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 condition: When IRRTA = 1, it is cleared by writing 0	(i
1	Setting condition: When the timer A counter value overflows from H'FF to H'00	

Bit 6: Reserved bit

Bit 6 is a readable/writable reserved bit. It is initialized to 0 by a reset.

Bit 5: Reserved bit

Bit 5 is reserved; it is always read as 1 and cannot be modified.

	input

5. Interrupt Request Register 2 (IRR2)

Bit	7	6	5	4	3	2	1
	IRRDT	IRRAD	—	IRRTG	IRRTFH	IRRTFL	IRRTC
Initial value	0	0	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

IRR2 is an 8-bit read/write register, in which a corresponding flag is set to 1 when a d
transfer, A/D converter, Timer G, Timer FH, Timer FC, or Timer C interrupt is reques
flags are not cleared automatically when an interrupt is accepted. It is necessary to wr
each flag.

Bit 7: Direct transfer interrupt request flag (IRRDT)

Bit 7 IRRDT	Description
0	Clearing condition: (When IRRDT = 1, it is cleared by writing 0
1	Setting condition: When a direct transfer is made by executing a SLEEP instruction while D SYSCR2

Bit 5: Reserved bit

Bit 5 is a readable/writable reserved bit. It is initialized to 0 by a reset.

Bit 4: Timer G interrupt request flag (IRRTG)

Bit 4 IRRTG	Description
0	Clearing condition: (i When IRRTG = 1, it is cleared by writing 0
1	Setting condition: When the TMIG pin is designated for TMIG input and the designated signa input, or when TCG overflows while OVIE is set to 1 in TMG

Bit 3: Timer FH interrupt request flag (IRRTFH)

Bit 3 IRRTFH	Description
0	Clearing condition: (i When IRRTFH = 1, it is cleared by writing 0
1	Setting condition: When TCFH and OCRFH match in 8-bit timer mode, or when TCF (TCFL, and OCRF (OCRFL, OCRFH) match in 16-bit timer mode

Bit 1: Timer C interrupt request flag (IRRTC)

Bit 1 IRRTC	Description
0	Clearing condition: When IRRTC= 1, it is cleared by writing 0
1	Setting condition: When the timer C counter value overflows (from H'FF to H'00) or underfl (from H'00 to H'FF)

Bit 0: Asynchronous event counter interrupt request flag (IRREC)

Bit 0 IRREC	Description
0	Clearing condition: When IRREC = 1, it is cleared by writing 0
1	Setting condition: When ECH overflows in 16-bit counter mode, or ECH or ECL overflows i counter mode

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

IWPR is an 8-bit read/write register containing wakeup interrupt request flags. When c
$\overline{WKP}_7$ to $\overline{WKP}_0$ is designated for wakeup input and a rising or falling edge is input at th
corresponding flag in IWPR is set to 1. A flag is not cleared automatically when the
corresponding interrupt is accepted. Flags must be cleared by writing 0.

Bits 7 to 0: Wakeup interrupt request flags (IWPF7 to IWPF0)

Bit n IWPFn	Description	
0	Clearing condition: When IWPFn= 1, it is cleared by writing 0	(i
1	Setting condition: When pin $\overline{WKP}_n$ is designated for wakeup input and a rising or falling edg that pin	

WEGR is initialized to H'00 by a reset.

Bit n: $\overline{\text{WKP}}$n edge select (WKEGSn)

Bit n selects $\overline{\text{WKP}}$n pin input sensing.

Bit n WKEGSn	Description
0	$\overline{\text{WKP}}$n pin falling edge detected
1	$\overline{\text{WKP}}$n pin rising edge detected

3.3.3 External Interrupts

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

1. Interrupts WKP_7 to WKP_0

Interrupts WKP_7 to WKP_0 are requested by either rising or falling edge input to pins $\overline{\text{W}}$ $\overline{\text{WKP}}_0$. When these pins are designated as pins $\overline{\text{WKP}}_7$ to $\overline{\text{WKP}}_0$ in port mode register rising or falling edge is input, the corresponding bit in IWPR is set to 1, requesting an Recognition of wakeup interrupt requests can be disabled by clearing the IENWP bit t IENR1. These interrupts can all be masked by setting the I bit to 1 in CCR.

When WKP_7 to WKP_0 interrupt exception handling is initiated, the I bit is set to 1 in C number 9 is assigned to interrupts WKP_7 to WKP_0. All eight interrupt sources have th vector number, so the interrupt-handling routine must discriminate the interrupt source

to 0 in IENR1. These interrupts can all be masked by setting the I bit to 1 in CCR.

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

3.3.4 Internal Interrupts

There are 23 internal interrupts that can be requested by the on-chip peripheral modules peripheral module requests an interrupt, the corresponding bit in IRR1 or IRR2 is set to Recognition of individual interrupt requests can be disabled by clearing the correspond IENR1 or IENR2. All these interrupts can be masked by setting the I bit to 1 in CCR. W internal interrupt handling is initiated, the I bit is set to 1 in CCR. Vector numbers from are assigned to these interrupts. Table 3.2 shows the order of priority of interrupts from peripheral modules.

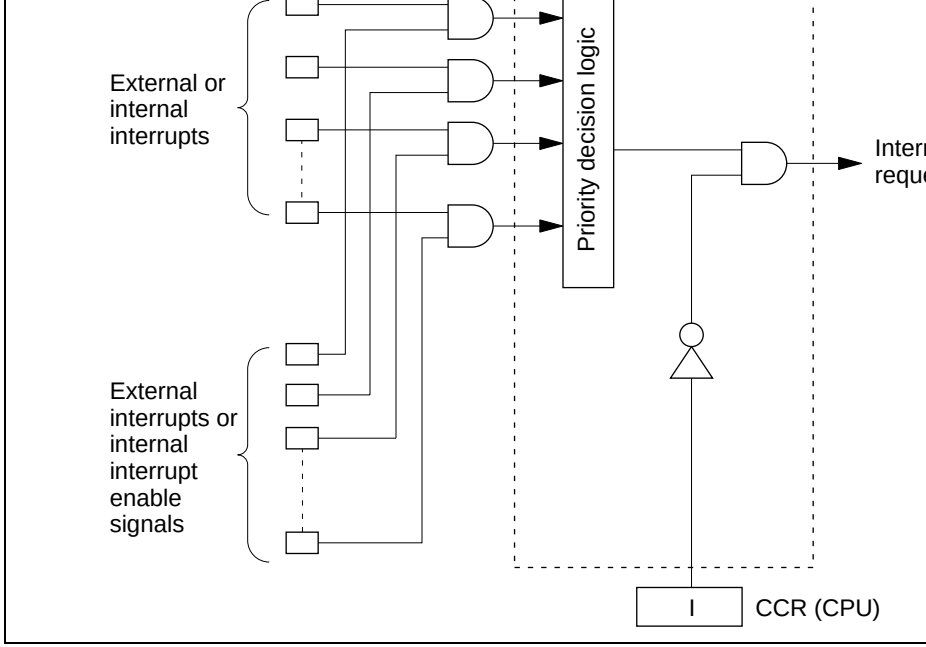


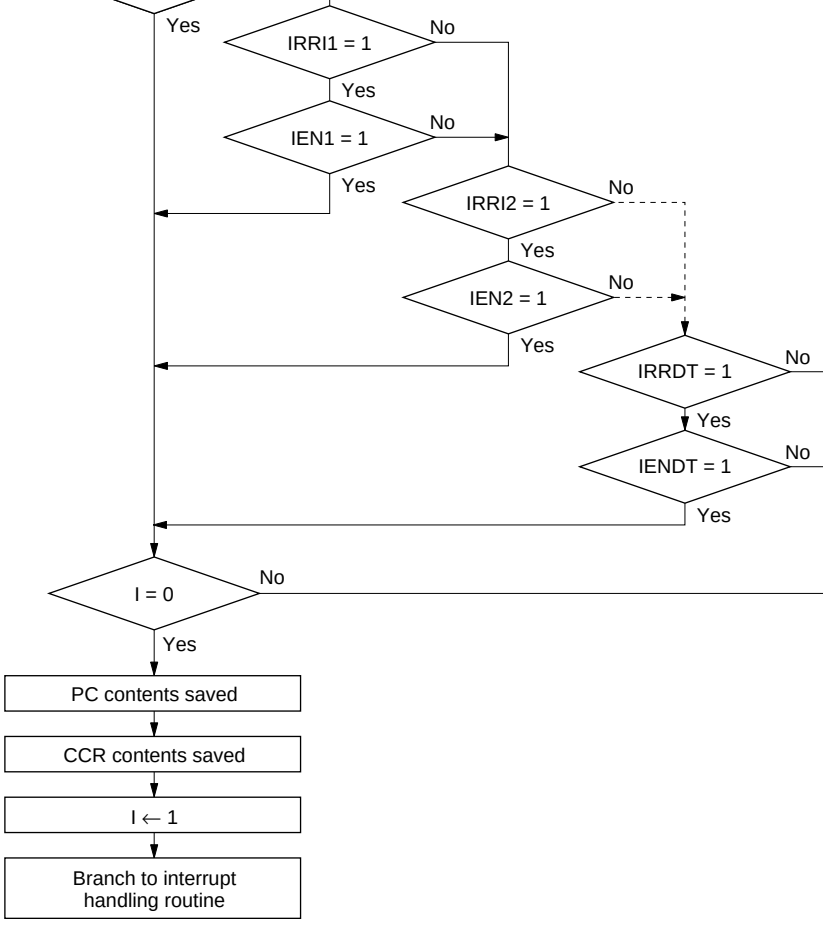
Figure 3.2 Block Diagram of Interrupt Controller

Interrupt operation is described as follows.

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

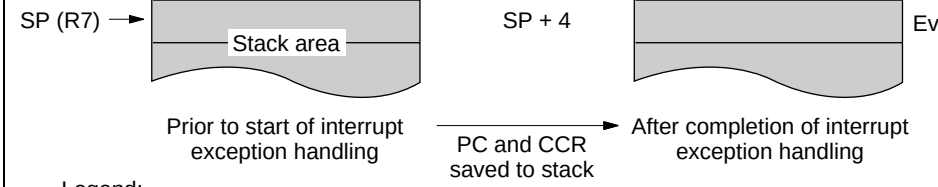
Notes: 1. When disabling interrupts by clearing bits in an interrupt enable register, or clearing bits in an interrupt request register, always do so while interrupts ar (I = 1).

2. If the above clear operations are performed while I = 0, and as a result a con between the clear instruction and an interrupt request, exception processing interrupt will be executed after the clear instruction has been executed.



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

Figure 3.3 Flow Up to Interrupt Acceptance



Legend:
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.

Figure 3.4 Stack State after Completion of Interrupt Exception Handlin

Figure 3.5 shows a typical interrupt sequence.

Figure 3.5 Interrupt Sequence

Saving of PC and CCR to stack	4
Vector fetch	2
Instruction fetch	4
Internal processing	4

Note: * Not including EEPMOV instruction.

@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

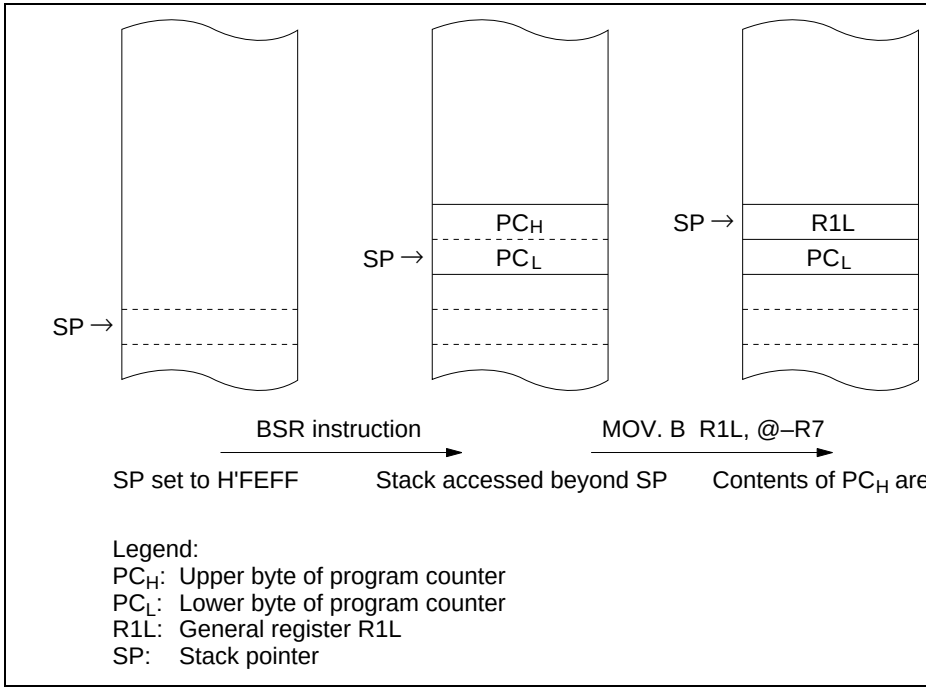


Figure 3.6 Operation when Odd Address is Set in SP

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

interrupt request flag to 0 after switching pin functions. Table 3.5 shows the conditions 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 PMR1 bit IRQ4 is changed from 0 to 1 while pin $\overline{IRQ}_4$ is low and IE = 0.
		When PMR1 bit IRQ4 is changed from 1 to 0 while pin $\overline{IRQ}_4$ is low and IE = 1.
	IRRI3	When PMR1 bit IRQ3 is changed from 0 to 1 while pin $\overline{IRQ}_3$ is low and IE = 0.
		When PMR1 bit IRQ3 is changed from 1 to 0 while pin $\overline{IRQ}_3$ is low and IE = 1.
	IRRI2	When PMR1 bit IRQ2 is changed from 0 to 1 while pin $\overline{IRQ}_2$ is low and IE = 0.
		When PMR1 bit IRQ2 is changed from 1 to 0 while pin $\overline{IRQ}_2$ is low and IE = 1.
	IRRI1	When PMR1 bit IRQ1 is changed from 0 to 1 while pin $\overline{IRQ}_1$ is low and IE = 0.
		When PMR1 bit IRQ1 is changed from 1 to 0 while pin $\overline{IRQ}_1$ is low and IE = 1.
	IRRI0	When PMR3 bit IRQ0 is changed from 0 to 1 while pin $\overline{IRQ}_0$ is low and IE = 0.
		When PMR3 bit IRQ0 is changed from 1 to 0 while pin $\overline{IRQ}_0$ is low and IE = 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.

An alternative method is to avoid the setting of interrupt request flags when pin functi
switched by keeping the pins at the high level so that the conditions in table 3.5 do no

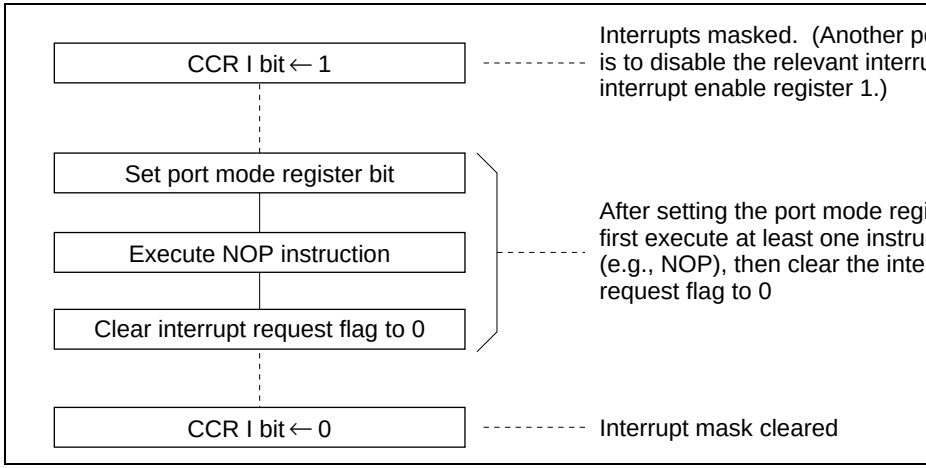


Figure 3.7 Port Mode Register Setting and Interrupt Request Flag Clearing I

given below.

```
BCLR  #1,  @IRR1:8
MOV.B R1L, @IRR1:8 (set the value of R1L to B'11111101)
```

- Example of a malfunction

  When flags are cleared with multiple instructions, other flags might be cleared duri
  execution of the instructions, even though they are currently set, and this will cause
  malfunction.

  Here is an example in which IRRI0 is cleared and disabled in the process of clearin
  (bit 1 of IRR1).

```
MOV.B @IRR1:8,R1L ......... IRRI0 = 0 at this time
AND.B #B'11111101,R1L ..... Here, IRRI0 = 1
MOV.B R1L,@IRR1:8 ......... IRRI0 is cleared to 0
```

  In the above example, it is assumed that an IRQ0 interrupt is generated while the A
  instruction is executing.

  The IRQ0 interrupt is disabled because, although the original objective is clearing I
  IRRI0 is also cleared.

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.

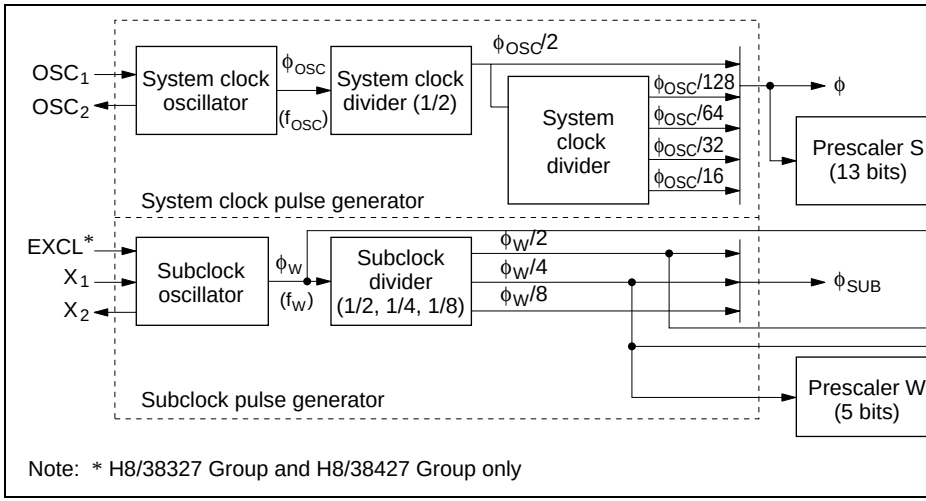


Figure 4.1 Block Diagram of Clock Pulse Generators

4.1.2 System Clock and Subclock

The basic clock signals that drive the CPU and on-chip peripheral modules are ϕ and ϕ of the clock signals have names: ϕ is the system clock, ϕ_{SUB} is the subclock, ϕ_{OSC} is the clock, and ϕ_W is the watch clock.

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

Characteristics. Please consult with the resonator manufacturer when selecting a resona

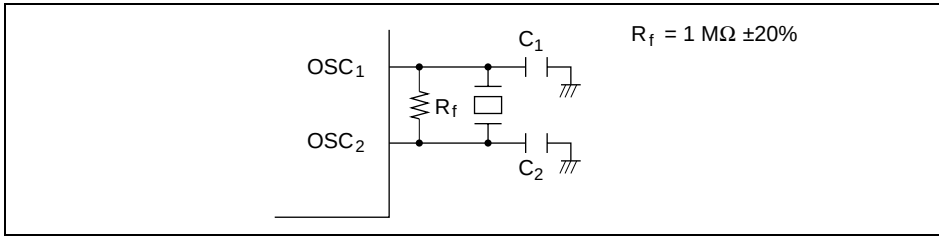


Figure 4.2 Typical Connection to Crystal Oscillator

2. Connecting a Ceramic Oscillator

Figure 4.3 shows a typical method of connecting a ceramic oscillator. For information c
recommended resonators, see the product AC characteristics listed in section 15, Electr
Characteristics. Please consult with the resonator manufacturer when selecting a resona

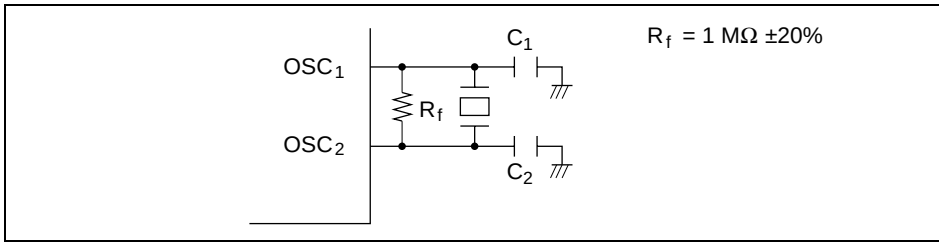


Figure 4.3 Typical Connection to Ceramic Oscillator

3. Notes on Board Design

When generating clock pulses by connecting a crystal or ceramic oscillator, pay careful
to the following points.

Avoid running signal lines close to the oscillator circuit, since the oscillator may be adv
affected by induction currents. (See figure 4.4.)

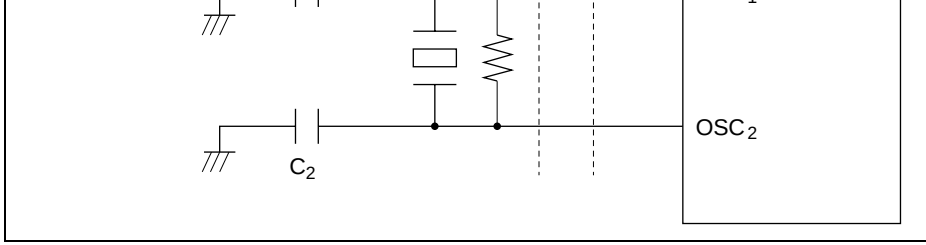


Figure 4.4 Board Design of Oscillator Circuit

4. External Clock Input Method

Connect an external clock signal to pin OSC_1, and leave pin OSC_2 open. Figure 4.5 sl
typical connection.

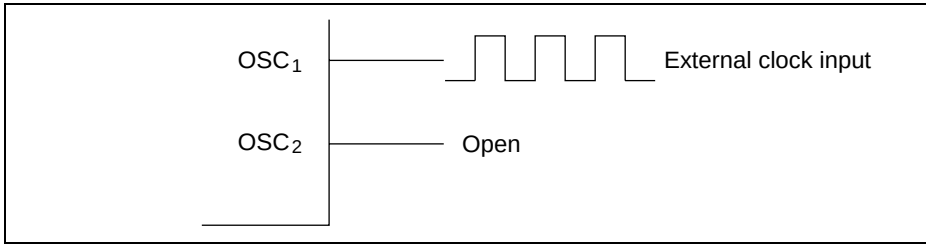
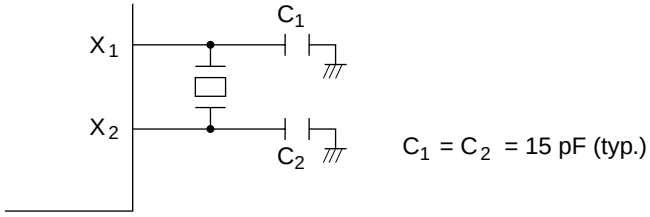


Figure 4.5 External Clock Input (Example)

Frequency	Oscillator Clock (ϕ_{osc})
Duty cycle	45% to 55%

Note: The circuit parameters above are recommended by the crystal or ceramic oscil
manufacturer.
The circuit parameters are affected by the crystal or ceramic oscillator and flo
capacitance when designing the board. When using the oscillator, consult wit
or ceramic oscillator manufacturer to determine the circuit parameters.



Note: Circuit constants should be determined in consultation with the resonator manufacturer.

Oscillation frequency	Manufacturer	Products Name
38.4 kHz	Seiko Instrument Inc.	VTC-200
32.768 kHz	Nihon Denpa Kogyo	MX73P

Figure 4.6 Typical Connection to 32.768 kHz/38.4 kHz Crystal Oscillator (Su

Figure 4.7 shows the equivalent circuit of the 32.768 kHz/38.4 kHz crystal oscillator.

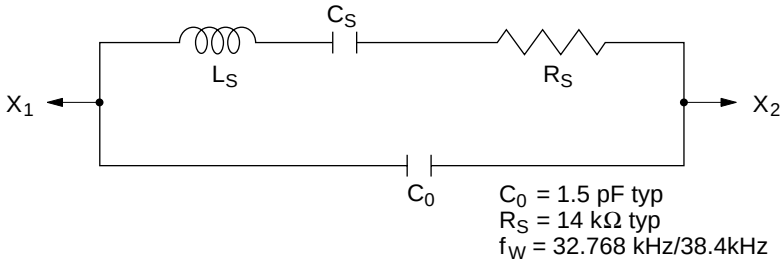


Figure 4.7 Equivalent Circuit of 32.768 kHz/38.4 kHz Crystal Oscillato

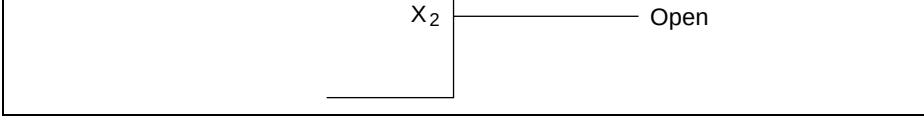


Figure 4.8 Pin Connection when not Using Subclock

3. External Clock Input

- H8/3827R Group and H8/3827S Group
  Connect the external clock to the X_1 pin and leave the X_2 pin open, as shown in fig

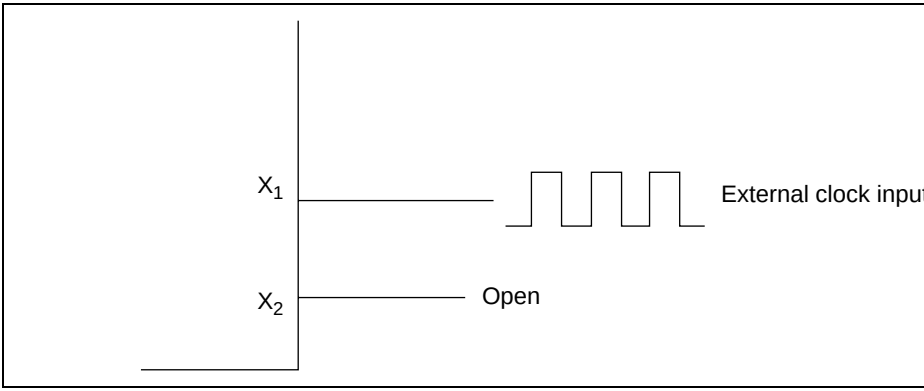


Figure 4.9 (a) Pin Connection when Inputting External Clock

Frequency	Subclock (ϕw)
Duty	45% to 55%

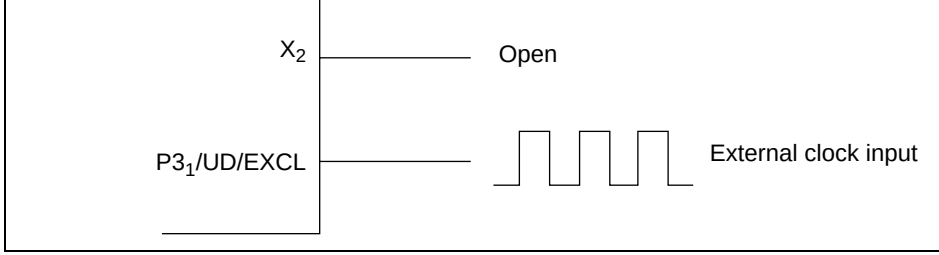


Figure 4.9 (b) Pin Connection when Inputting External Clock (H8/38327 Group and H8/38427 Group)

Frequency	Subclock (φw)
Duty	45% to 55%

1. Prescaler S (PSS)

Prescaler S is a 13-bit counter using the system clock (ϕ) as its input clock. It is incre per clock period.

Prescaler S is initialized to H'0000 by a reset, and starts counting on exit from the rese

In standby mode, watch mode, subactive mode, and subsleep mode, the system clock 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 C, timer F, timer G, SCI3-1, S A/D converter, the LCD controller, the watchdog timer, and the 14-bit PWM. The div can be set separately for each on-chip peripheral function.

In active (medium-speed) mode the clock input to prescaler S is ϕosc/16, ϕosc/32, ϕos ϕosc/128.

2. Prescaler W (PSW)

Prescaler W is a 5-bit counter using a 32.768 kHz/38.4 kHz signal divided by 4 ($\phi_W/4$) clock.

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

Even in standby mode, watch mode, subactive mode, or subsleep mode, prescaler W c functioning so long as clock signals are supplied to pins X1 and X2.

Prescaler W can be reset by setting 1s in bits TMA3 and TMA2 of timer mode registe

Output from prescaler W can be used to drive timer A, in which case timer A function base for timekeeping.

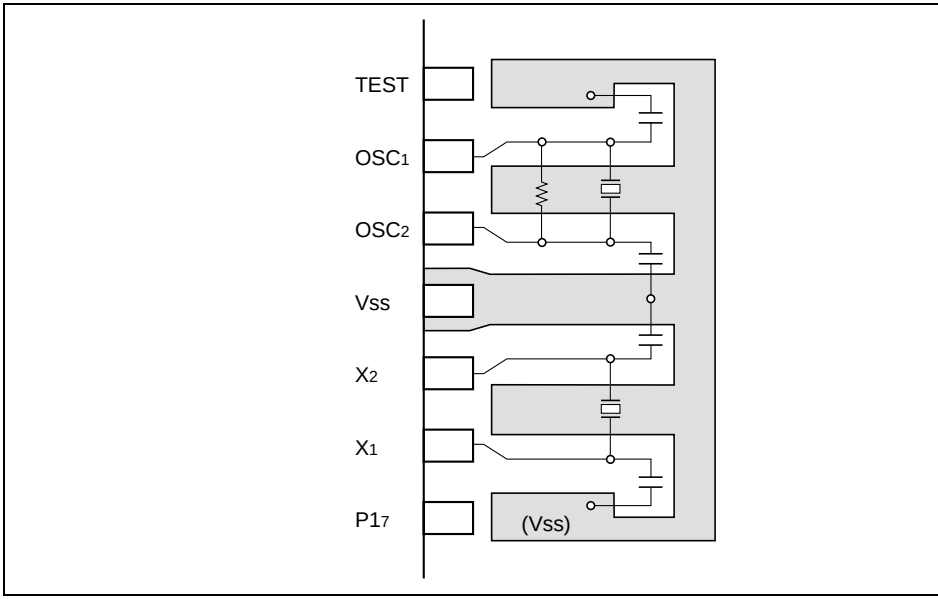


Figure 4.10 Example of Crystal and Ceramic Oscillator Element Arrange

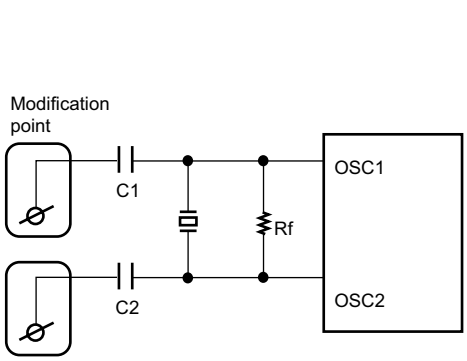
Figure 4.11 (1) shows an example measuring circuit with the negative resistance sugge
oscillator manufacturer. Note that if the negative resistance of the circuit is less than tha
by the oscillator manufacturer, it may be difficult to start the main oscillator.

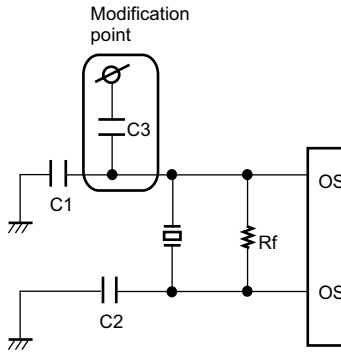
If it is determined that oscillation is not occurring because the negative resistance is lo
level suggested by the oscillator manufacturer, the circuit may be modified as shown in
(2) through (4). Which of the modification suggestions to use and the capacitor capacita
be decided based upon an evaluation of factors such as the negative resistance and the f
deviation.

addition of –R

(1) Negative Resistance Measuring Circuit

(2) Oscillator Circuit Modification Sug





(3) Oscillator Circuit Modification Suggestion 2

(4) Oscillator Circuit Modification Su

Figure 4.11 Negative Resistance Measurement and Circuit Modification Sug

4.5.1 Definition of Oscillation Stabilization Wait Time

Figure 4.12 shows the oscillation waveform (OSC_2), system clock (ϕ), and microcomp operating mode when a transition is made from standby mode, watch mode, or subacti active (high-speed/medium-speed) mode, with an oscillator element connected to the s oscillator.

As shown in figure 4.12, as the system clock oscillator is halted in standby mode, wat and subactive mode, when a transition is made to active (high-speed/medium-speed) m sum of the following two times (oscillation stabilization time and wait time) is require

waveform frequency and system clock have stabilized.

The wait time setting is selected with standby timer select bits 2 to 0 (STS2 to STS0) (b
system control register 1 (SYSCR1)).

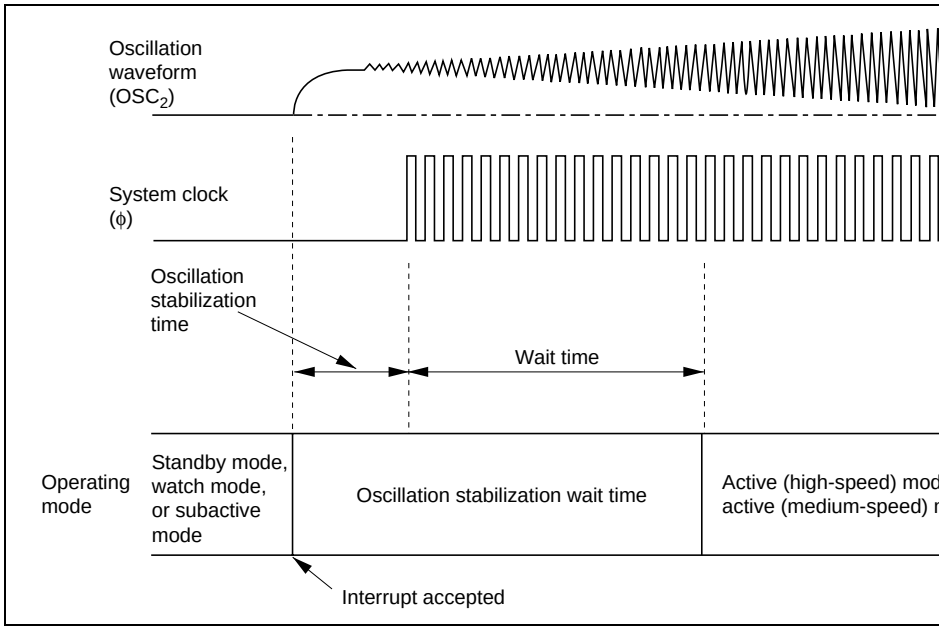


Figure 4.12 Oscillation Stabilization Wait Time

When standby mode, watch mode, or subactive mode is cleared by an interrupt or reset
transition is made to active (high-speed/medium-speed) mode, the oscillation waveform
change at the point at which the interrupt is accepted. Therefore, when an oscillator ele
connected in standby mode, watch mode, or subactive mode, since the system clock osc
halted, the time from the point at which this oscillation waveform starts to change until
amplitude of the oscillation waveform increases and the oscillation frequency stabilizes
the oscillation stabilization time—is required.

functions is the sum of the above described oscillation stabilization time and wait time
time is called the oscillation stabilization wait time, and is expressed by equation (1) b

Oscillation stabilization wait time = oscillation stabilization time + wait time
= t_{rc} + (8 to 131,072 states) .....

Therefore, when a transition is made from standby mode, watch mode, or subactive m
active (high-speed/medium-speed) mode, with an oscillator element connected to the
oscillator, careful evaluation must be carried out on the installation circuit before deci
oscillation stabilization wait time. In particular, since the oscillation stabilization time
by installation circuit constants, stray capacitance, and so forth, suitable constants sho
determined in consultation with the oscillator element manufacturer.

4.5.2 Notes on Use of Crystal Oscillator Element (Excluding Ceramic Oscilla Element)

When a microcomputer operates, the internal power supply potential fluctuates slightl
synchronization with the system clock. Depending on the individual crystal oscillator
characteristics, the oscillation waveform amplitude may not be sufficiently large imm
the oscillation stabilization wait time, making the oscillation waveform susceptible to
fluctuations in the power supply potential. In this state, the oscillation waveform may
disrupted, leading to an unstable system clock and erroneous operation of the microco

If erroneous operation occurs, change the setting of standby timer select bits 2 to 0 (S
STS0) (bits 6 to 4 in system control register 1 (SYSCR1)) to give a longer wait time.

For example, if erroneous operation occurs with a wait time setting of 16 states, check
operation with a wait time setting of 8,192 states or more.

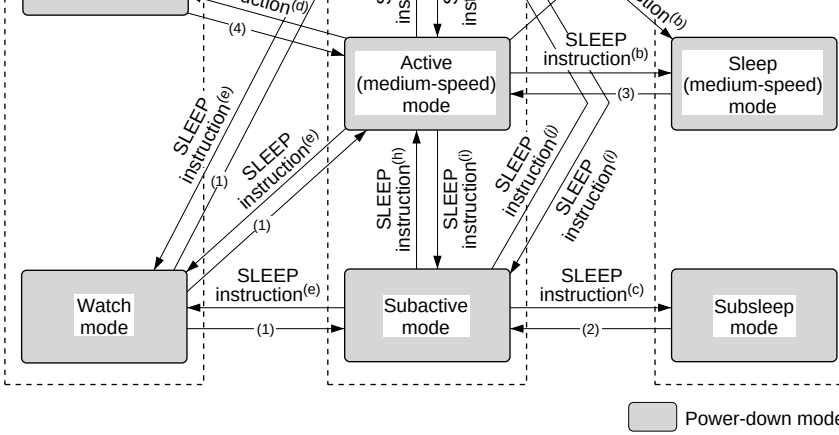
If the same kind of erroneous operation occurs after a reset as after a state transition, h
pin low for a longer period.

Table 5.1 Operating Modes

Operating Mode	Description
Active (high-speed) mode	The CPU and all on-chip peripheral functions are
	the system clock in high-speed operation
Active (medium-speed) mode	The CPU and all on-chip peripheral functions are
	the system clock in low-speed operation
Subactive mode	The CPU is operable on the subclock in low-spee
Sleep (high-speed) mode	The CPU halts. On-chip peripheral functions are c
	the system clock
Sleep (medium-speed) mode	The CPU halts. On-chip peripheral functions oper
	frequency of 1/64, 1/32, 1/16, or 1/8 of the system
	frequency
Subsleep mode	The CPU halts. The time-base function of timer A,
	timer G, timer F,WDT, SCI3-1, SCI3-2, AEC, and
	controller/driver are operable on the subclock
Watch mode	The CPU halts. The time-base function of timer A,
	timer G, AEC, and LCD controller/driver are opera
	subclock
Standby mode	The CPU and all on-chip peripheral functions halt
Module standby mode	Individual on-chip peripheral functions specified b
	enter standby mode and halt

Of these nine operating modes, all but the active (high-speed) mode are power-down m
this section the two active modes (high-speed and medium speed) will be referred to c
as active mode.

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



Mode Transition Conditions (1)

	LSON	MSON	SSBY	TMA3	DTON
(a)	0	0	0	*	0
(b)	0	1	0	*	0
(c)	1	*	0	1	0
(d)	0	*	1	0	0
(e)	*	*	1	1	0
(f)	0	0	0	*	1
(g)	0	1	0	*	1
(h)	0	1	1	1	1
(i)	1	*	1	1	1
(J)	0	0	1	1	1

* : Don't care

Mode Transition Conditions (2)

	Interrupt Sources
(1)	Timer A, Timer F, Timer G interrupt, IRQ_0 i WKP_7 to WKP_0 interrupts
(2)	Timer A, Timer C, Timer F, Timer G, SCI3-1 SCI3-2 interrupt, IRQ_4 to IRQ_0 interrupts, WKP_7 to WKP_0 interrupts, AEC
(3)	All interrupts
(4)	IRQ_1 or IRQ_0 interrupt, WKP_7 to WKP_0 inter

Notes: 1. A transition between different modes cannot be made to occur simply because an interrupt request is generated. Make sure that interrupt handling is performed after the interrupt is accepted.
2. Details on the mode transition conditions are given in the explanations of each mode, in sections 5.2 to 5.9.

Figure 5.1 Mode Transition Diagram

	I/O ports							
External interrupts	IRQ_0	Functions	Functions	Functions	Functions	Functions	Functions	Functions
	IRQ_1					Retained*6		
	IRQ_2							
	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 functions	Timer A	Functions	Functions	Functions	Functions	Functions*5	Functions*5	Functions
	Asynchronous counter					Functions*8	Functions	Functions
	Timer C					Retained	Functions/ Retained*2	Functions Retained
	WDT						Functions/ Retained*7	Retained
	Timer G, Timer F					Functions/ Retained*9	Functions/ Retained*2	Functions Retained
	SCI3-1					Reset	Functions/ Retained*3	Functions Retained
	SCI3-2							
	PWM					Retained	Retained	Retained
	A/D converter					Retained	Retained	Retained
	LCD					Functions/ Retained*4	Functions/ Retained*4	Functions Retained

Notes:
1. Register contents are retained, but output is high-impedance state.
2. Functions if an external clock or the $\phi_W/4$ internal clock is selected; otherwise halted and retained
3. Functions if $\phi_W/2$ is selected as the internal clock; otherwise halted and retained.
4. Functions if ϕ_W, $\phi_W/2$ or $\phi_W/4$ is selected as the operating clock; otherwise halted and retained.
5. Functions if the timekeeping time-base function is selected.
6. External interrupt requests are ignored. Interrupt request register contents are not altered.
7. Functions if $\phi_W/32$ is selected as the internal clock; otherwise halted and retained.
8. Incrementing is possible, but interrupt generation is not.
9. Functions if the $\phi_W/4$ internal clock is selected; otherwise halted and retained.

1. System Control Register 1 (SYSCR1)

Bit	7	6	5	4	3	2	1
	SSBY	STS2	STS1	STS0	LSON	—	MA1
Initial value	0	0	0	0	0	1	1
Read/Write	R/W	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.

Upon reset, SYSCR1 is initialized to H'07.

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, (i a transition is made to sleep mode • When a SLEEP instruction is executed in subactive mode, a transition subsleep mode
1	• When a SLEEP instruction is executed in active mode, a transition is r standby mode or watch mode • When a SLEEP instruction is executed in subactive mode, a transition watch mode

0	0	0	Wait time = 8,192 states	(
0	0	1	Wait time = 16,384 states	
0	1	0	Wait time = 32,768 states	
0	1	1	Wait time = 65,536 states	
1	0	0	Wait time = 131,072 states	
1	0	1	Wait time = 2 states	(External
1	1	0	Wait time = 8 states	
1	1	1	Wait time = 16 states	

Note: In the case that external clock is input, set up the “Standby timer select” selecti
External clock mode before Mode Transition. Also, do not set up to external clo
the case that it does not use external clock.

Bit 3: Low speed on flag (LSON)

This bit chooses the system clock (ϕ) or subclock (ϕ_{SUB}) as the CPU operating clock w
mode is cleared. The resulting operation mode depends on the combination of other c
and interrupt input.

Bit 3 LSON	Description	
0	The CPU operates on the system clock (ϕ)	(
1	The CPU operates on the subclock (ϕ_{SUB})	

Bits 2: Reserved bits

Bit 2 is reserved: it is always read as 1 and cannot be modified.

0	1	ϕ_{OSC}/32	
1	0	ϕ_{OSC}/64	
1	1	ϕ_{OSC}/128	(i

2. System Control Register 2 (SYSCR2)

Bit	7	6	5	4	3	2	1	
	—	—	—	NESEL	DTON	MSON	SA1	
Initial value	1	1	1	1	0	0	0	
Read/Write	—	—	—	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 (ϕ_W) generated by the sub
pulse generator is sampled, in relation to the oscillator clock (ϕ_{OSC}) generated by the sy
pulse generator. When ϕ_{OSC} = 2 to 16 MHz, clear NESEL to 0.

Bit 4 NESEL	Description	
0	Sampling rate is ϕ_{OSC}/16	
1	Sampling rate is ϕ_{OSC}/4	(i

0	• When a SLEEP instruction is executed in active mode, a transition is made to standby mode, watch mode, or sleep mode • When a SLEEP instruction is executed in subactive mode, a transitio watch mode or subsleep mode
1	• When a SLEEP instruction is executed in active (high-speed) mode, transition is made to active (medium-speed) mode if SSBY = 0, MSO LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = • When a SLEEP instruction is executed in active (medium-speed) mo transition is made to active (high-speed) mode if SSBY = 0, MSON = LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = • When a SLEEP instruction is executed in subactive mode, a direct tra made to active (high-speed) mode if SSBY = 1, TMA3 = 1, LSON = 0 = 0, or to active (medium-speed) mode if SSBY = 1, TMA3 = 1, LSO MSON = 1

Bit 2: Medium speed on flag (MSON)

After standby, watch, or sleep mode is cleared, this bit selects active (high-speed) or a (medium-speed) mode.

Bit 2 MSON	Description
0	Operation in active (high-speed) mode
1	Operation in active (medium-speed) mode

1	*	$\phi_w/2$

*

5.2 Sleep Mode

5.2.1 Transition to Sleep Mode

1. Transition to Sleep (High-Speed) Mode

The system goes from active mode to sleep (high-speed) mode when a SLEEP instructi
executed while the SSBY and LSON bits in SYSCR1 are cleared to 0, the MSON and I
in SYSCR2 are cleared to 0. In sleep mode CPU operation is halted but the on-chip per
functions. CPU register contents are retained.

2. Transition to Sleep (Medium-Speed) Mode

The system goes from active mode to sleep (medium-speed) mode when a SLEEP instr
executed while the SSBY and LSON bits in SYSCR1 are cleared to 0, the MSON bit in
is set to 1, and the DTON bit in SYSCR2 is cleared to 0. In sleep (medium-speed) mod
sleep (high-speed) mode, CPU operation is halted but the on-chip peripheral functions a
operational. The clock frequency in sleep (medium-speed) mode is determined by the M
MA0 bits in SYSCR1. CPU register contents are retained.
Furthermore, it sometimes acts with half state early timing at the time of transition to sl
(medium-speed) mode.

(medium-speed) mode to active (medium-speed) mode. Sleep mode is not cleared the condition code register (CCR) is set to 1 or the particular interrupt is disabled i interrupt enable register.

Interrupt signal and system clock are mutually asynchronous. Synchronization err maximum is 2/ϕ (s).

- 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 c

5.2.3 Clock Frequency in Sleep (Medium-Speed) Mode

Operation in sleep (medium-speed) mode is clocked at the frequency designated by th MA0 bits in SYSCR1.

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 exec the SSBY bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, and bit TMA is cleared to 0. In standby mode the clock pulse generator stops, so the CPU and peripheral modules stop functioning, but as long as the rated voltage is supplied, the c CPU registers, on-chip RAM, and some on-chip peripheral module registers are retain RAM contents will be further retained down to a minimum RAM data retention voltag ports go to the high-impedance state.

entire chip, standby mode is cleared, and interrupt exception handling starts. Opera resumes in active (high-speed) mode if MSON = 0 in SYSCR2, or active (medium- mode if MSON = 1. Standby mode is not cleared if the I bit of CCR is set to 1 or th interrupt is disabled in the interrupt enable register.

- Clearing by $\overline{\text{RES}}$ input

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

Table 5.4 Clock Frequency and Settling Time (Times are in ms)

STS2	STS1	STS0	Waiting Time	2 MHz
0	0	0	8,192 states	4.1
0	0	1	16,384 states	8.2
0	1	0	32,768 states	16.4
0	1	1	65,536 states	32.8
1	0	0	131,072 states	65.5
1	0	1	2 states (Use prohibited)	0.001
1	1	0	8 states	0.004
1	1	1	16 states	0.008

- When an external clock is used
  STS2 = 1, STS1 = 0, and STS0 = 1 should be set. Other values possible use, but C
  sometimes will start operation before waiting time completion.

5.3.4 Standby Mode Transition and Pin States

When a SLEEP instruction is executed in active (high-speed) mode or active (medium
mode while bit SSBY is set to 1 and bit LSON is cleared to 0 in SYSCR1, and bit TM
cleared to 0 in TMA, a transition is made to standby mode. At the same time, pins go
impedance state (except pins for which the pull-up MOS is designated as on). Figure
the timing in this case.

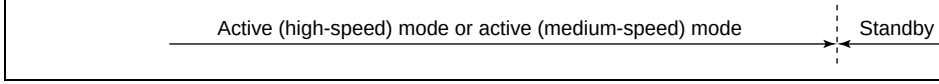


Figure 5.2 Standby Mode Transition and Pin States

5.3.5 Notes on External Input Signal Changes before/after Standby Mode

1. When external input signal changes before/after standby mode or watch mode

   When an external input signal such as $\overline{\text{IRQ}}$ or $\overline{\text{WKP}}$ is input, both the high- and low widths of the signal must be at least two cycles of system clock ϕ or subclock ϕ_{SUB} to together in this section as the internal clock). As the internal clock stops in stand and watch mode, the width of external input signals requires careful attention when is made via these operating modes. Ensure that external input signals conform to the stated in 3, Recommended timing of external input signals, below

2. When external input signals cannot be captured because internal clock stops

   The case of falling edge capture is illustrated in figure 5.3

   As shown in the case marked "Capture not possible," when an external input signal immediately after a transition to active (high-speed or medium-speed) mode or suba mode, after oscillation is started by an interrupt via a different signal, the external in cannot be captured if the high-level width at that point is less than 2 t_{cyc} or 2 t_{subcyc}.

3. Recommended timing of external input signals

   To ensure dependable capture of an external input signal, high- and low-level signal at least 2 t_{cyc} or 2 t_{subcyc} are necessary before a transition is made to standby mode or mode, as shown in "Capture possible: case 1."

   External input signal capture is also possible with the timing shown in "Capture pos 2" and "Capture possible: case 3," in which a 2 t_{cyc} or 2 t_{subcyc} level width is secured.

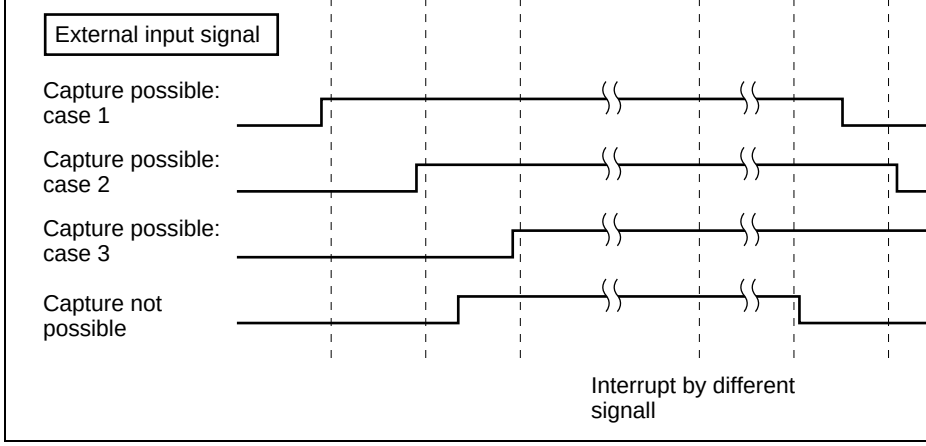


Figure 5.3 External Input Signal Capture when Signal Changes before/a Standby Mode or Watch Mode

4. Input pins to which these notes apply:
   $\overline{IRQ_4}$ to $\overline{IRQ_0}$, $\overline{WKP_7}$ to $\overline{WKP_0}$, $\overline{ADTRG}$, TMIC, TMIF, TMIG

timer G, AEC and the LCD controller/driver (for which operation or halting can be set)
As long as a minimum required voltage is applied, the contents of CPU registers, the on
RAM and some registers of the on-chip peripheral modules, are retained. I/O ports kee
states as before the transition.

5.4.2 Clearing Watch Mode

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

- Clearing by interrupt

  When watch mode is cleared by interrupt, the mode to which a transition is made de
  the settings of LSON in SYSCR1 and MSON in SYSCR2. If both LSON and MSO
  cleared to 0, transition is to active (high-speed) mode; if LSON = 0 and MSON = 1,
  is to active (medium-speed) mode; if LSON = 1, transition is to subactive mode. W
  transition is to active mode, after the time set in SYSCR1 bits STS2 to STS0 has ela
  stable clock signal is supplied to the entire chip, watch mode is cleared, and interrup
  handling starts. Watch mode is not cleared if the I bit of CCR is set to 1 or the parti
  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 2. Clearing by $\overline{RES}$ pin in
  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 section 5.3.3, Oscillator Settling
Standby Mode is Cleared.

The system goes from subactive mode to subsleep mode when a SLEEP instruction is while the SSBY bit in SYSCR1 is cleared to 0, LSON bit in SYSCR1 is set to 1, and T TMA is set to 1. In subsleep mode, operation of on-chip peripheral modules other tha converter WDT and PWM is halted. As long as a minimum required voltage is applie contents of CPU registers, the on-chip RAM and some registers of the on-chip periphe 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 F, timer G, asynchro counter, SCI3-2, SCI3-1, IRQ_4 to IRQ_0, WKP_7 to WKP_0) or by a low input at the $\overline{RES}$

- Clearing by interrupt

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

  Interrupt signal and system clock are mutually asynchronous. Synchronization err maximum is $2/\phi_{SUB}$ (s).

- Clearing by $\overline{RES}$ input

  Clearing by $\overline{RES}$ pin is the same as for standby mode; see 2. Clearing by $\overline{RES}$ pin 5.3.2, Clearing Standby Mode.

does not take place if the I bit of CCR is set to 1 or the particular interrupt is disabled in interrupt enable register.

5.6.2 Clearing Subactive Mode

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

- Clearing by SLEEP instruction

  If a SLEEP instruction is executed while the SSBY bit in SYSCR1 is set to 1 and T TMA is set to 1, subactive mode is cleared and watch mode is entered. If a SLEEP is executed while SSBY = 0 and LSON = 1 in SYSCR1 and TMA3 = 1 in TMA, su mode is entered. Direct transfer to active mode is also possible; see section 5.8, Dir Transfer, below.

- Clearing by $\overline{\text{RES}}$ pin

  Clearing by $\overline{\text{RES}}$ pin is the same as for standby mode; see 2. Clearing by $\overline{\text{RES}}$ pin in 5.3.2.

5.6.3 Operating Frequency in Subactive Mode

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

timer G, IRQ_0, or WKP_7 to WKP_0 interrupts in watch mode, or any interrupt in sleep
transition to active (medium-speed) mode does not take place if the I bit of CCR is set
particular interrupt is disabled in the interrupt enable register.
Furthermore, it sometimes acts with half state early timing at the time of transition to
(medium-speed) mode.

5.7.2 Clearing Active (Medium-Speed) Mode

Active (medium-speed) mode is cleared by a SLEEP instruction.

- Clearing by SLEEP instruction

  A transition to standby mode takes place if the SLEEP instruction is executed whi
  bit in SYSCR1 is set to 1, the LSON bit in SYSCR1 is cleared to 0, and the TMA3
  is cleared to 0. The system goes to watch mode if the SSBY bit in SYSCR1 is set
  TMA3 in TMA is set to 1 when a SLEEP instruction is executed.

  When both SSBY and LSON are cleared to 0 in SYSCR1 and a SLEEP instructio
  sleep mode is entered. Direct transfer to active (high-speed) mode or to subactive
  possible. See section 5.8, Direct Transfer, below for details.

- Clearing by $\overline{\text{RES}}$ pin

  When the $\overline{\text{RES}}$ pin is driven low, a transition is made to the reset state and active
  speed) mode is cleared.

5.7.3 Operating Frequency in Active (Medium-Speed) Mode

Operation in active (medium-speed) mode is clocked at the frequency designated by t
MA0 bits in SYSCR1.

exception handling starts.

If the direct transfer interrupt is disabled in interrupt enable register 2, a transition is ma
to sleep mode or watch mode. Note that if a direct transition is attempted while the I bi
set to 1, sleep mode or watch mode will be entered, and it will be impossible to clear th
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
  LSON bits in SYSCR1 are cleared to 0, the MSON bit in SYSCR2 is set to 1, and th
  bit in SYSCR2 is set to 1, a transition is made to active (medium-speed) mode via s

- 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 SS
  LSON bits in SYSCR1 are cleared to 0, the MSON bit in SYSCR2 is cleared to 0, a
  DTON bit in SYSCR2 is set to 1, a transition is made to active (high-speed) mode v
  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
  LSON bits in SYSCR1 are set to 1, the DTON bit in SYSCR2 is set to 1, and the TM
  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 SY
  set to 1, the LSON bit in SYSCR1 is cleared to 0, the MSON bit in SYSCR2 is clea
  the DTON bit in SYSCR2 is set to 1, and the TMA3 bit in TMA is set to 1, a transit
  directly to active (high-speed) mode via watch mode after the waiting time set in SY
  STS2 to STS0 has elapsed.

DTON bit in SYSCR2 is set to 1, and the TMA3 bit in TMA is set to 1, a transitio
directly to active (medium-speed) mode via watch mode after the waiting time set
bits STS2 to STS0 has elapsed.

5.8.2 Direct Transition Times

1. Time for Direct Transition from Active (High-Speed) Mode to Active (Medium-Speed) Mode

A direct transition from active (high-speed) mode to active (medium-speed) mode is p
executing a SLEEP instruction in active (high-speed) mode while bits SSBY and LSO
cleared to 0 in SYSCR1, and bits MSON and DTON are both set to 1 in SYSCR2. Th
execution of the SLEEP instruction to the end of interrupt exception handling (the dire
time) is given by equation (1) below.

Direct transition time = { (Number of SLEEP instruction execution states) + (number
processing states) } × (tcyc before transition) + (number of in
exception handling execution states) × (tcyc after transition)
.....

Example: Direct transition time = (2 + 1) × 2tosc + 14 × 16tosc = 230tosc (when ϕ/8
as the CPU operating clock)

Notation:
tosc: OSC clock cycle time
tcyc: System clock (ϕ) cycle time

Direct transition time = { (Number of SLEEP instruction execution states) + (number o
processing states) } × (tcyc before transition) + (number of int
exception handling execution states) × (tcyc after transition)
.....

Example: Direct transition time = (2 + 1) × 16tosc + 14 × 2tosc = 76tosc (when ϕ/8 is
the CPU operating clock)

Notation:
tosc: OSC clock cycle time
tcyc: System clock (ϕ) cycle time

3. Time for Direct Transition from Subactive Mode to Active (High-Speed) Mod

A direct transition from subactive mode to active (high-speed) mode is performed by ex
SLEEP instruction in subactive mode while bit SSBY is set to 1 and bit LSON is cleare
SYSCR1, bit MSON is cleared to 0 and bit DTON is set to 1 in SYSCR2, and bit TMA
in TMA. The time from execution of the SLEEP instruction to the end of interrupt exc
handling (the direct transition time) is given by equation (3) below.

Direct transition time = { (Number of SLEEP instruction execution states) + (number o
processing states) } × (tsubcyc before transition) + { (wait tim
STS2 to STS0) + (number of interrupt exception handling exec
states) } × (tcyc after transition) .....

Example: Direct transition time = (2 + 1) × 8tw + (8192 + 14) × 2tosc = 24tw + 16412
ϕw/8 is selected as the CPU operating clock, and wait time = 8192 states)

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

processing states) } × (tsubcyc before transition) + { (wait tim
STS2 to STS0) + (number of interrupt exception handling exe
states) } × (tcyc after transition) .....

Example: Direct transition time = (2 + 1) × 8tw + (8192 + 14) × 16tosc = 24tw + 131
(when ϕw/8 or ϕ8 is selected as the CPU operating clock, and wait time = 8

Notation:

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

5.8.3 Notes on External Input Signal Changes before/after Direct Transition

1. Direct transition from active (high-speed) mode to subactive mode
   Since the mode transition is performed via watch mode, see section 5.3.5, Notes on Input Signal Changes before/after Standby Mode.

2. Direct transition from active (medium-speed) mode to subactive mode
   Since the mode transition is performed via watch mode, see section 5.3.5, Notes on Input Signal Changes before/after Standby Mode.

3. Direct transition from subactive mode to active (high-speed) mode
   Since the mode transition is performed via watch mode, see section 5.3.5, Notes on Input Signal Changes before/after Standby Mode.

4. Direct transition from subactive mode to active (medium-speed) mode
   Since the mode transition is performed via watch mode, see section 5.3.5, Notes on Input Signal Changes before/after Standby Mode.

identical to standby mode.

Module standby mode is set for a particular module by setting the corresponding bit to
stop register 1 (CKSTPR1) or clock stop register 2 (CKSTPR2). (See table 5.5.)

5.9.2 Clearing Module Standby Mode

Module standby mode is cleared for a particular module by setting the corresponding b
clock stop register 1 (CKSTPR1) or clock stop register 2 (CKSTPR2). (See table 5.5.)

Following a reset, clock stop register 1 (CKSTPR1) and clock stop register 2 (CKSTPR
initialized to H'FF.

		0	Timer F is set to module standby mode
	TGCKSTP	1	Timer G module standby mode is cleared
		0	Timer G is set to module standby mode
	ADCKSTP	1	A/D converter module standby mode is c
		0	A/D converter is set to module standby m
	S32CKSTP	1	SCI3-2 module standby mode is cleared
		0	SCI3-2 is set to module standby mode
	S31CKSTP	1	SCI3-1 module standby mode is cleared
		0	SCI3-1 is set to module standby mode
CKSTPR2	LDCKSTP	1	LCD module standby mode is cleared
		0	LCD is set to module standby mode
	PWCKSTP	1	PWM module standby mode is cleared
		0	PWM is set to module standby mode
	WDCKSTP	1	Watchdog timer module standby mode is
		0	Watchdog timer is set to module standby
	AECKSTP	1	Asynchronous event counter module star is cleared
		0	Asynchronous event counter is set to mo mode

Note: For details of module operation, see the sections on the individual modules.

standby mode. The surest way to do this is to specify the module standby mode setting
interrupts are prohibited (interrupts prohibited using the interrupt enable register or inte
masked using bit CCR-I).

H8/3826R, H8/3826S, H8/38326, and H8/38426 have 48 Kbytes, and the H8/3827R, H
H8/38327, and H8/38427 have 60 Kbytes. The ROM is connected to the CPU by a 16
bus, allowing high-speed two-state access for both byte data and word data. The H8/3
ZTAT™ version with 60-Kbyte PROM.

The H8/3827S Group does not have a ZTAT™ version. The H8/3827R ZTAT™ vers
used.

The F-ZTAT™ versions of the H8/38327 and H8/38427 are equipped with 60 Kbytes
memory. The F-ZTAT™ versions of the H8/38324 and H8/38424 are equipped with 3
flash memory.

6.1.1 Block Diagram

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

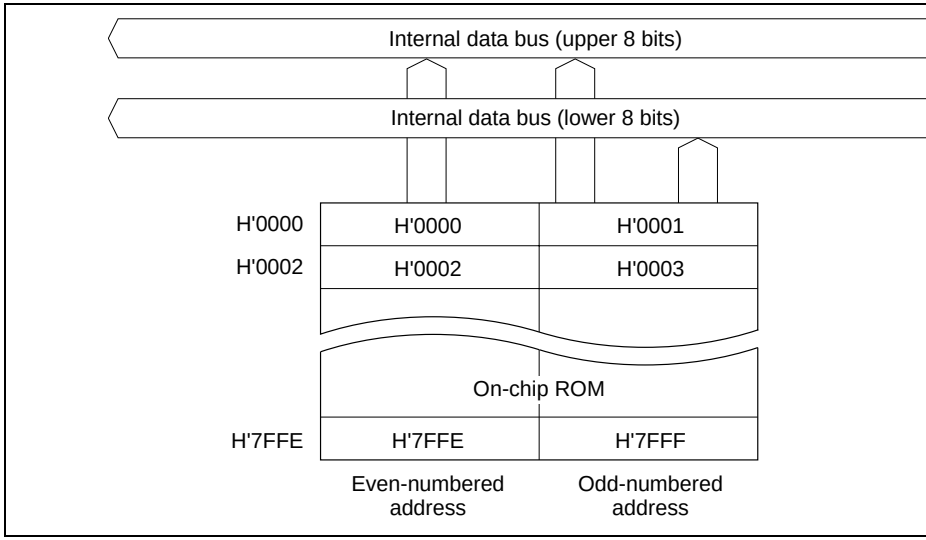


Figure 6.1 ROM Block Diagram (H8/3824R, H8/3824S, H8/38324, and H8/

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

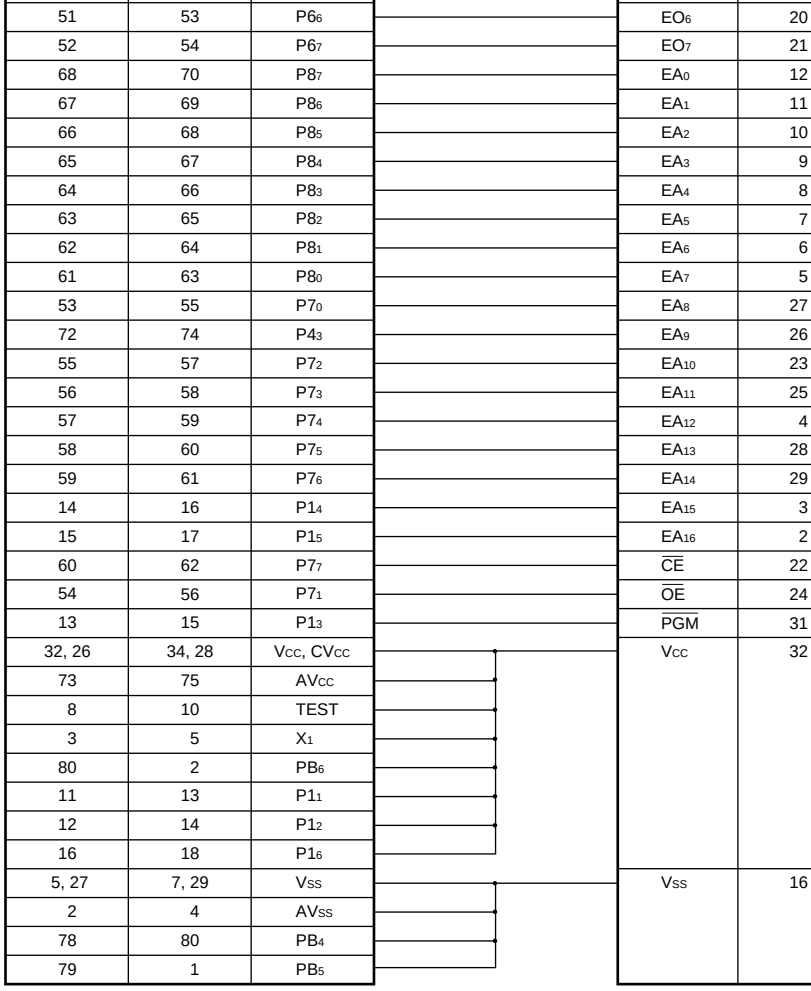
6.2.2 Socket Adapter Pin Arrangement and Memory Map

A standard PROM programmer can be used to program the PROM. A socket adapter is for conversion to 32 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 memo

Table 6.2 Socket Adapter

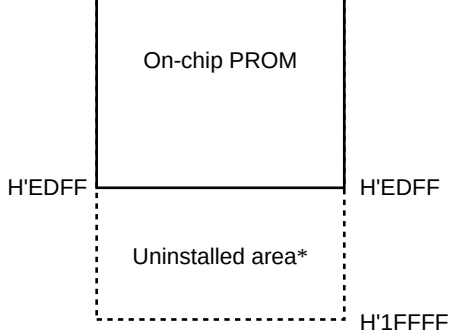
Package	Socket Adapters (Manufacturer)
80-pin (FP-80B)	ME3867ESFS1H (MINATO) H7386BQ080D3201 (DATA-I/O)
80-pin (FP-80A)	ME3867ESHS1H (MINATO) H7386AQ080D3201 (DATA-I/O)
80-pin (TFP-80C)	ME3867ESNS1H (MINATO) H7386CT080D3201 (DATA-I/O)



51	53	$P6_6$	EO_6	20
52	54	$P6_7$	EO_7	21
68	70	$P8_7$	EA_0	12
67	69	$P8_6$	EA_1	11
66	68	$P8_5$	EA_2	10
65	67	$P8_4$	EA_3	9
64	66	$P8_3$	EA_4	8
63	65	$P8_2$	EA_5	7
62	64	$P8_1$	EA_6	6
61	63	$P8_0$	EA_7	5
53	55	$P7_0$	EA_8	27
72	74	$P4_3$	EA_9	26
55	57	$P7_2$	EA_{10}	23
56	58	$P7_3$	EA_{11}	25
57	59	$P7_4$	EA_{12}	4
58	60	$P7_5$	EA_{13}	28
59	61	$P7_6$	EA_{14}	29
14	16	$P1_4$	EA_{15}	3
15	17	$P1_5$	EA_{16}	2
60	62	$P7_7$	$\overline{CE}$	22
54	56	$P7_1$	$\overline{OE}$	24
13	15	$P1_3$	$\overline{PGM}$	31
32, 26	34, 28	V_{CC}, CV_{CC}	V_{CC}	32
73	75	AV_{CC}		
8	10	TEST		
3	5	X_1		
80	2	PB_6		
11	13	$P1_1$		
12	14	$P1_2$		
16	18	$P1_6$		
5, 27	7, 29	V_{SS}	V_{SS}	16
2	4	AV_{SS}		
78	80	PB_4		
79	1	PB_5		

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

Figure 6.2 Socket Adapter Pin Correspondence (with HN27C101)



Note: * The output data is not guaranteed if this address area is read in PROM mode. T
when programming with a PROM programmer, be sure to specify addresses fro
to H'EDFF. If programming is inadvertently performed from H'EE00 onward, it m
be possible to continue PROM programming and verification.
When programming, H'FF should be set as the data in this address area (H'EE0
H'1FFFF).

Figure 6.3 H8/3827R Memory Map in PROM Mode

Write	L	H	L	V_{PP}	V_{CC}	Data input	Add
Verify	L	L	H	V_{PP}	V_{CC}	Data output	Add
Programming disabled	L	L	L	V_{PP}	V_{CC}	High impedance	Add
	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 are identical to those for the standard HN27 EPROM. However, page programming is not supported, and so page programming mo be set. A PROM programmer that only supports page programming mode cannot be u selecting a PROM programmer, ensure that it supports high-speed, high-reliability byt programming. Also, be sure to specify addresses from H'0000 to H'EDFF.

6.3.1 Writing and Verifying

An efficient, high-speed, high-reliability method is available for writing and verifying data. This method achieves high speed without voltage stress on the device and withou the reliability of written data. The basic flow of this high-speed, high-reliability progra method is shown in figure 6.4.

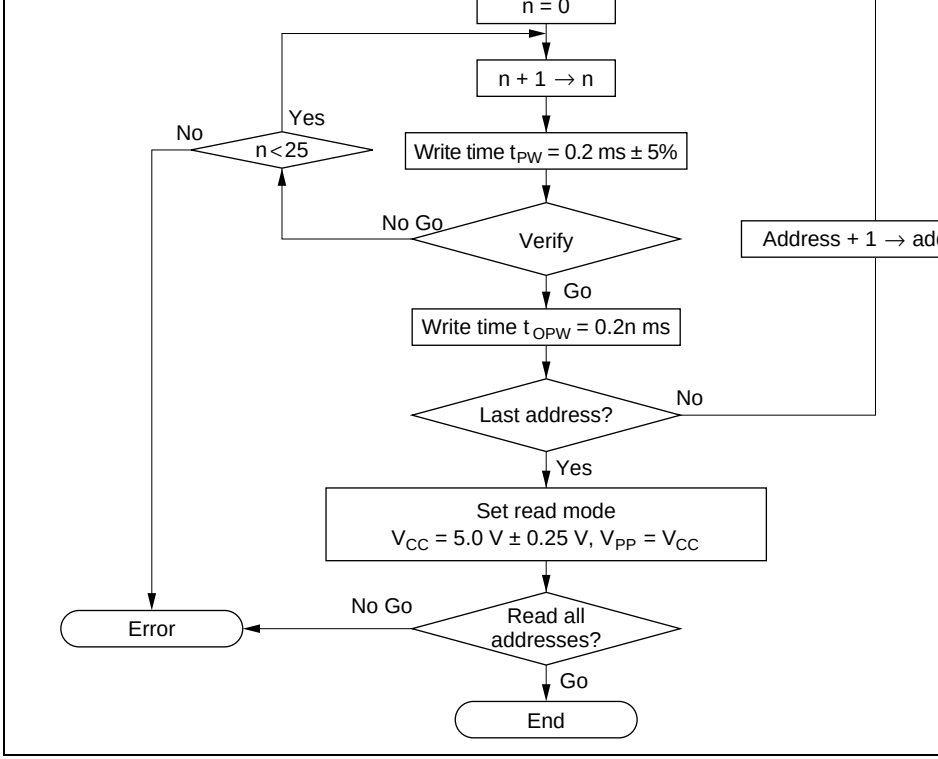


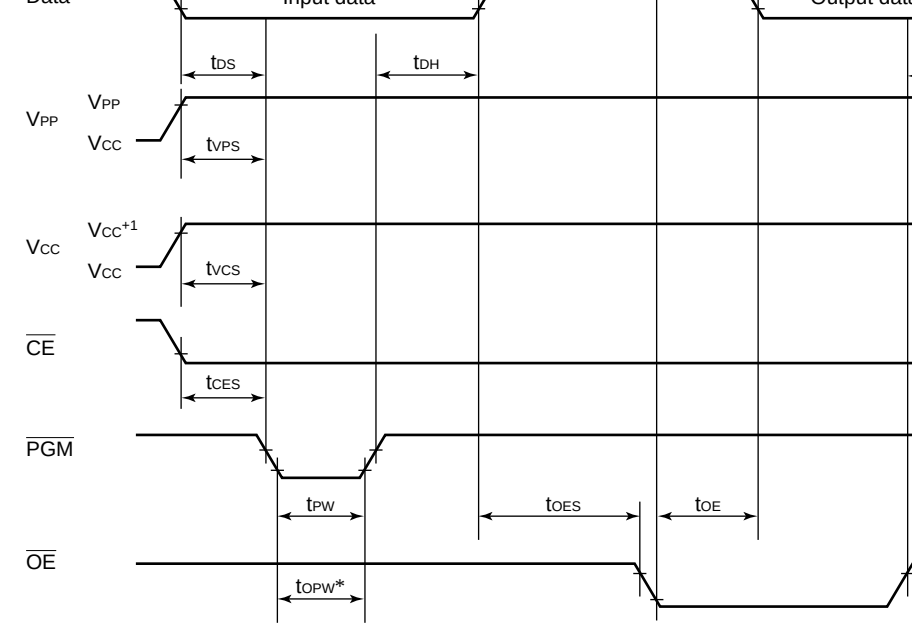
Figure 6.4 High-Speed, High-Reliability Programming Flow Chart

level voltage	$\overline{OE}$, $\overline{CE}$, $\overline{PGM}$						
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_O
Output low-level voltage	EO_7 to EO_0	V_{OL}	—	—	0.45	V	I_O
Input leakage current	EO_7 to EO_0, EA_{16} to EA_0 $\overline{OE}$, $\overline{CE}$, $\overline{PGM}$	$\|IL_I\|$	—	—	2	µA	V 0
V_{CC} current		I_{CC}	—	—	40	mA	
V_{PP} current		I_{PP}	—	—	40	mA	

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
$\overline{CE}$ setup time	t_{CES}	2	—	—	µs
V_{CC} setup time	t_{VCS}	2	—	—	µs
Data output delay time	t_{OE}	0	—	200	ns

Notes: 1. Input pulse level: 0.45 V to 2.2 V
Input rise time/fall time ≤ 20 ns
Timingreference 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 c read.
3. t_{OPW} is defined by the value given in figure 6.4, High-Speed, High-Reliability Programming Flow Chart.



Note: * t_{opw} is defined by the value shown in figure 6.4, High-Speed, High-Reliability Programmi

Figure 6.5 PROM Write/Verify Timing

- Make sure the index marks on the PROM programmer socket, socket adapter, and c
  properly aligned. If they are not, the chip may be destroyed by excessive current flo
  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
  faults and write errors.
- Take care when setting the programming mode, as page programming is not suppor
- When programming with a PROM programmer, be sure to specify addresses from H
  H'EDFF. If programming is inadvertently performed from H'EE00 onward, it may
  possible to continue PROM programming and verification. When programming, H
  be set as the data in address area H'EE00 to H'1FFFF.

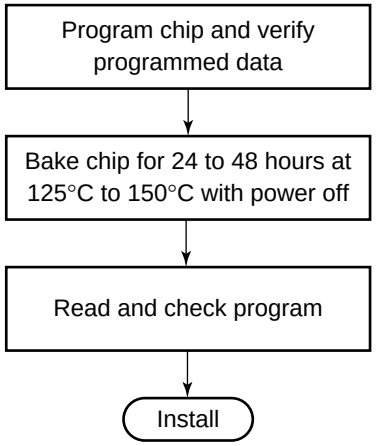
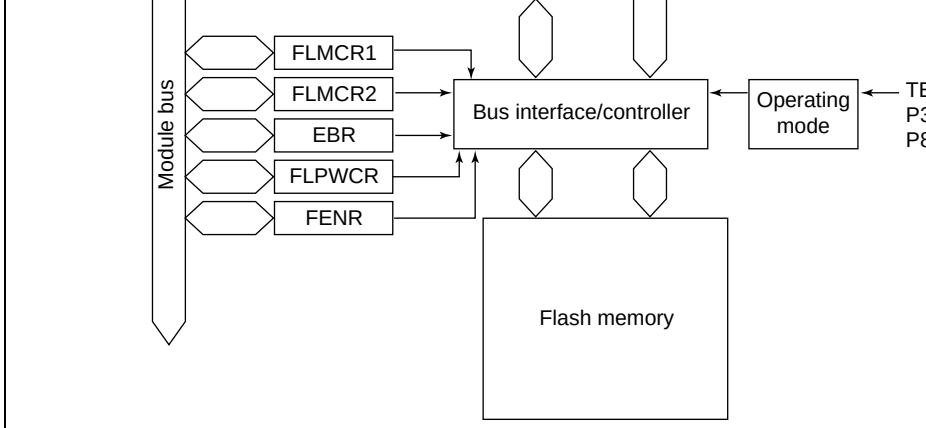


Figure 6.6 Recommended Screening Procedure

If a series of programming errors occurs while the same PROM programmer is in use, programming and check the PROM programmer and socket adapter for defects. Pleas Renesas Technology of any abnormal conditions noted during or after programming o screening of program data after high-temperature baking.

— The 60-Kbyte flash memory is programmed 128 bytes at a time. Erase is perfor
single-block units. The flash memory is configured as follows: 1 Kbyte × 4 blo
28 Kbytes × 1 block, 16 Kbytes × 1 block, 8 Kbytes × 1 block, 4 Kbytes × 1 blo
The 32-Kbyte flash memory is configured as follows: 1 Kbyte × 4 blocks, 28 Kb
block. To erase the entire flash memory, each block must be erased in turn.

- Reprogramming capability
  — The flash memory can be reprogrammed up to 1,000 times.
- On-board programming
  — On-board programming/erasing can be done in boot mode, in which the boot pr
  into the chip is started to erase or program of the entire flash memory. In norma
  program mode, individual blocks can be erased or programmed.
- Programmer mode
  — Flash memory can be programmed/erased in programmer mode using a PROM
  programmer, as well as in on-board programming mode.
- Automatic bit rate adjustment
  — For data transfer in boot mode, this LSI's bit rate can be automatically adjusted t
  the transfer bit rate of the host.
- Programming/erasing protection
  — Sets software protection against flash memory programming/erasing.
- Power-down mode
  — The power supply circuit is partly halted in the subactive mode and can be read
  power-down mode.



Legend:
FLMCR1: Flash memory control register 1
FLMCR2: Flash memory control register 2
EBR: Erase block register
FLPWCR: Flash memory power control register
FENR: Flash memory enable register

Figure 6.7 Block Diagram of Flash Memory

6.5.3 Block Configuration

Figure 6.8 shows the block configuration of flash memory. The thick lines indicate er
the narrow lines indicate programming units, and the values are addresses. The flash
divided into 1 Kbyte × 4 blocks, 28 Kbytes × 1 block, 16 Kbytes × 1 block, 8 Kbytes ×
and 4 Kbytes × 1 block. Erasing is performed in these units. Programming is perform
128-byte units starting from an address with lower eight bits H'00 or H'80.

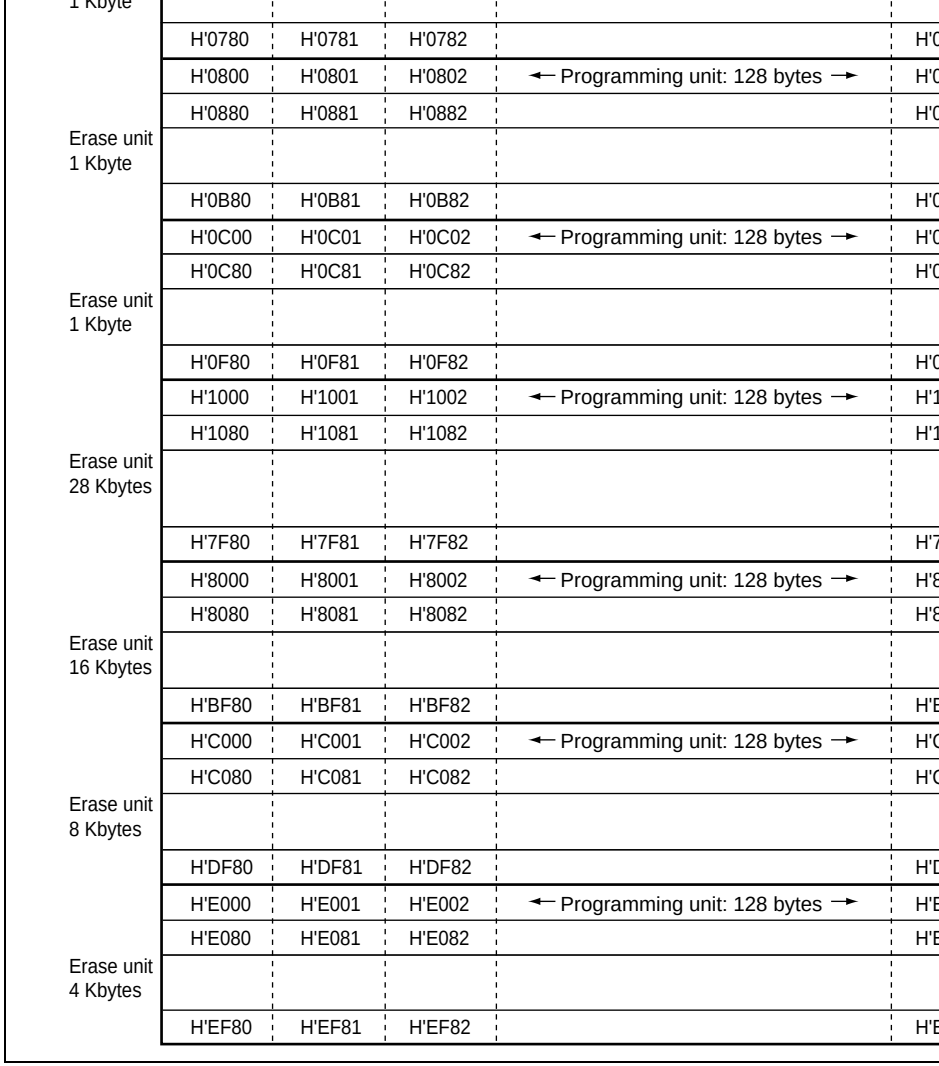


Figure 6.8 Flash Memory Block Configuration

Flash memory control register 2	FLMCR2	R	H'00
Flash memory power control register	FLPWCR	R/W	H'00
Erase block register	EBR	R/W	H'00
Flash memory enable register	FENR	R/W	H'00

Note: FLMCR1, FLMCR2, FLPWCR, EBR, and FENR are 8 bit registers. Only byte a
enabled which are two-state access. These registers are dedicated to the prod
flash memory is included. The product in which PROM or ROM is included doe
these registers. When the corresponding address is read in these products, the
undefined. A write is disabled.

Read/Write	—	—	—	—	—	—

FLMCR1 is a register that makes the flash memory change to program mode, program-mode, erase mode, or erase-verify mode. For details on register setting, refer to section Memory Programming/Erasing. By setting this register, the flash memory enters progra erase mode, program-verify mode, or erase-verify mode. Read the data in the state that of this register are cleared when using flash memory as normal built-in ROM.

Bit 7—Reserved

This bit is always read as 0 and cannot be modified.

Bit 6—Software Write Enable (SWE)

This bit is to set enabling/disabling of programming/enabling of flash memory (set whe 0 and the EBR register are to be set).

Bit 6 SWE	Description
0	Programming/erasing is disabled. Other FLMCR1 register bits and all EB cannot be set. (in
1	Flash memory programming/erasing is enabled.

Bit 5—Erase Setup (ESU)

This bit is to prepare for changing to erase mode. Set this bit to 1 before setting the E b FLMCR1 (do not set SWE, PSU, EV, PV, E, and P bits at the same time).

Bit 5 ESU	Description
0	The erase setup state is cancelled (in
1	The flash memory changes to the erase setup state. Set this bit to 1 befo the E bit to 1 in FLMCR1.

	setting the P bit to 1 in FLMCR1.

Bit 3—Erase-Verify (EV)

This bit is to set changing to or cancelling erase-verify mode (do not set SWE, ESU, P and P bits at the same time).

Bit 3 EV	Description	
0	Erase-verify mode is cancelled	(
1	The flash memory changes to erase-verify mode	

Bit 2—Program-Verify (PV)

This bit is to set changing to or cancelling program-verify mode (do not set SWE, ESU E, and P bits at the same time).

Bit 2 PV	Description	
0	Program-verify mode is cancelled	(
1	The flash memory changes to program-verify mode	

Bit 1—Erase (E)

This bit is to set changing to or cancelling erase mode (do not set SWE, ESU, PSU, E bits at the same time).

This bit is to set changing to or cancelling program mode (do not set SWE, ESU, PSU, and E bits at the same time).

Bit 0 P	Description
0	Program mode is cancelled (in
1	When this bit is set to 1, while the SWE = 1 and PSU = 1, the flash memo changes to program mode.

6.6.2 Flash Memory Control Register 2 (FLMCR2)

Bit	7	6	5	4	3	2	1
	FLER	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0
Read/Write	R	—	—	—	—	—	—

FLMCR2 is a register that displays the state of flash memory programming/erasing. FL read-only register, and should not be written to.

Bit 7—Flash Memory Error (FLER)

This bit is set when the flash memory detects an error and goes to the error-protection s programming or erasing to the flash memory. See section 6.9.3, Error Protection, for d

Bit 7 FLER	Description
0	The flash memory operates normally. (in
1	Indicates that an error has occurred during an operation on flash memory (programming or erasing).

Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

EBR specifies the flash memory erase area block. EBR is initialized to H'00 when the FLMCR1 is 0. Do not set more than one bit at a time, as this will cause all the bits in automatically cleared to 0. When each bit is set to 1 in EBR, the corresponding block erased. Other blocks change to the erase-protection state. See table 6.7 for the metho blocks of the flash memory. When the whole bits are to be erased, erase them in turn block.

Table 6.7 Division of Blocks to Be Erased

EBR	Bit Name	Block (Size)	Address
0	EB0	EB0 (1 Kbyte)	H'0000 to H'03FF
1	EB1	EB1 (1 Kbyte)	H'0400 to H'07FF
2	EB2	EB2 (1 Kbyte)	H'0800 to H'0BFF
3	EB3	EB3 (1 Kbyte)	H'0C00 to H'0FFF
4	EB4	EB4 (28 Kbytes)	H'1000 to H'7FFF
5	EB5	EB5 (16 Kbytes)	H'8000 to H'BFFF
6	EB6	EB6 (8 Kbytes)	H'C000 to H'DFFF
7	EB7	EB7 (4 Kbytes)	H'E000 to H'EFFF

FLPWCR enables or disables a transition to the flash memory power-down mode when switches to subactive mode. The power supply circuit can be read in the subactive mod it is partly halted in the power-down mode.

Bit 7—Power-down Disable (PDWND)

This bit selects the power-down mode of the flash memory when a transition to the sub mode is made.

Bit 7 PDWND	Description
0	When this bit is 0 and a transition is made to the subactive mode, the flas enters the power-down mode. (in
1	When this bit is 1, the flash memory remains in the normal mode even af transition is made to the subactive mode.

Bits 6 to 0—Reserved

These bits are always read as 0 and cannot be modified.

FENR controls CPU access to the flash memory control registers, FLMCR1, FLMCR
FLPWCR.

Bit 7—Flash Memory Control Register Enable (FLSHE)

This bit controls access to the flash memory control registers.

Bit 7 FLSHE	Description	
0	Flash memory control registers cannot be accessed	(
1	Flash memory control registers can be accessed	

Bits 6 to 0—Reserved

These bits are always read as 0 and cannot be modified.

When changing to boot mode, the boot program built into this LSI is initiated. The boo
transfers the programming control program from the externally-connected host to on-ch
via SCI32. After erasing the entire flash memory, the programming control program is
This can be used for programming initial values in the on-board state or for a forcible r
programming/erasing can no longer be done in user program mode. In user program m
individual blocks can be erased and programmed by branching to the user program/eras
program prepared by the user.

Table 6.8 Setting Programming Modes

TEST	P32	P86	PB0	PB1	PB2	LSI State after Reset En
0	1	X	X	X	X	User Mode
0	0	1	X	X	X	Boot Mode
1	X	X	0	0	0	Programmer Mode

X: Don't care

6.7.1 Boot Mode

Table 6.9 shows the boot mode operations between reset end and branching to the prog
control program. The device uses SCI32 in the boot mode.

1. When boot mode is used, the flash memory programming control program must be
   the host beforehand. Prepare a programming control program in accordance with th
   description in section 6.8, Flash Memory Programming/Erasing.
2. SCI3 should be set to asynchronous mode, and the transfer format as follows: 8-bit
   bit, and no parity. The inversion function of TXD and RXD pins by the SPCR regis
   "Not to be inverted," so do not put the circuit for inverting a value between the host
   LSI.
3. When the boot program is initiated, the chip measures the low-level period of asyn
   SCI communication data (H'00) transmitted continuously from the host. The chip th

transfer bit rate and system clock frequency of this LSI, there will be a discrepancy
the bit rates of the host and the chip. To operate the SCI properly, set the host's tra
rate and system clock frequency of this LSI within the ranges listed in table 6.10.

5. In boot mode, a part of the on-chip RAM area is used by the boot program. The ar
H'FEEF is the area to which the programming control program is transferred from
The boot program area cannot be used until the execution state in boot mode switc
programming control program.
6. Before branching to the programming control program, the chip terminates transfe
by SCI3 (by clearing the RE and TE bits in SCR to 0), however the adjusted bit ra
remains set in BRR. Therefore, the programming control program can still use it f
of write data or verify data with the host. The TXD pin is high (PCR42 = 1, P42 =
contents of the CPU general registers are undefined immediately after branching to
programming control program. These registers must be initialized at the beginning
programming control program, as the stack pointer (SP), in particular, is used impl
subroutine calls, etc.
7. Boot mode can be cleared by a reset. End the reset after driving the reset pin low,
least 20 states, and then setting the TEST pin and P32 pin. Boot mode is also clea
WDT overflow occurs.
8. Do not change the TEST pin and P32 pin input levels in boot mode.

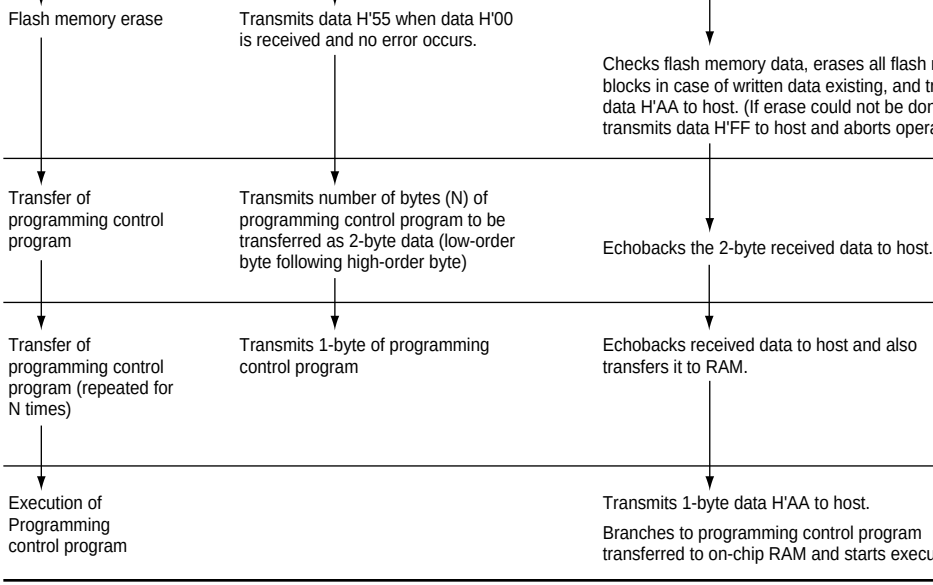


Table 6.10 Oscillating Frequencies (f_{OSC}) for which Automatic Adjustment of LS Is Possible

Product Group	Host Bit Rate	Oscillating Frequencies (f_{osc}) Range o
H8/38327F-ZTAT	19,200 bps	16 MHz
H8/38324F-ZTAT	9,600 bps	8 to 16 MHz
H8/38427F-ZTAT	4,800 bps	6 to 16 MHz
H8/38424F-ZTAT	2,400 bps	2 to 16 MHz
	1,200 bps	2 to 16 MHz

programming/erasing, transfer the user program/erase control program to on-chip RA
mode. Figure 6.9 shows a sample procedure for programming/erasing in user program
Prepare a user program/erase control program in accordance with the description in se
Flash Memory Programming/Erasing.

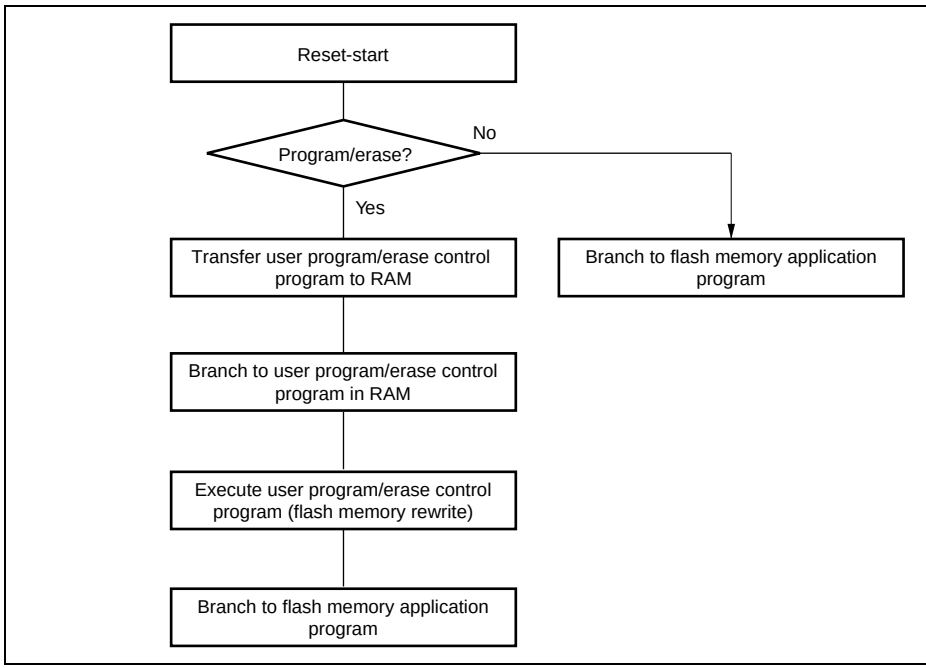


Figure 6.9 Programming/Erasing Flowchart Example in User Program M

6.8 Flash Memory Programming/Erasing

A software method using the CPU is employed to program and erase flash memory in board programming modes. Depending on the FLMCR1 setting, the flash memory op of the following four modes: Program mode, program-verify mode, erase mode, and e mode. The programming control program in boot mode and the user program/erase co

in figure 6.10 should be followed. Performing programming operations according to th
flowchart will enable data or programs to be written to the flash memory without subje
chip to voltage stress or sacrificing program data reliability.

1. Programming must be done to an empty address. Do not reprogram an address to w
programming has already been performed.
2. Programming should be carried out 128 bytes at a time. A 128-byte data transfer m
performed even if writing fewer than 128 bytes. In this case, H'FF data must be wri
extra addresses.
3. Prepare the following data storage areas in RAM: A 128-byte programming data ar
byte reprogramming data area, and a 128-byte additional-programming data area. F
reprogramming data computation according to table 6.11, and additional programm
computation according to table 6.12.
4. Consecutively transfer 128 bytes of data in byte units from the reprogramming data
additional-programming data area to the flash memory. The program address and 1
data are latched in the flash memory. The lower 8 bits of the start address in the fla
destination area must be H'00 or H'80.
5. The time during which the P bit is set to 1 is the programming time. Figure 6.12 sh
allowable programming times.
6. The watchdog timer (WDT) is set to prevent overprogramming due to program runa
An overflow cycle of approximately 6.6 ms is allowed.
7. For a dummy write to a verify address, write 1-byte data H'FF to an address whose
is b'0. Verify data can be read in word size from the address to which a dummy wri
performed.
8. The maximum number of repetitions of the program/program-verify sequence of th
is 1,000.

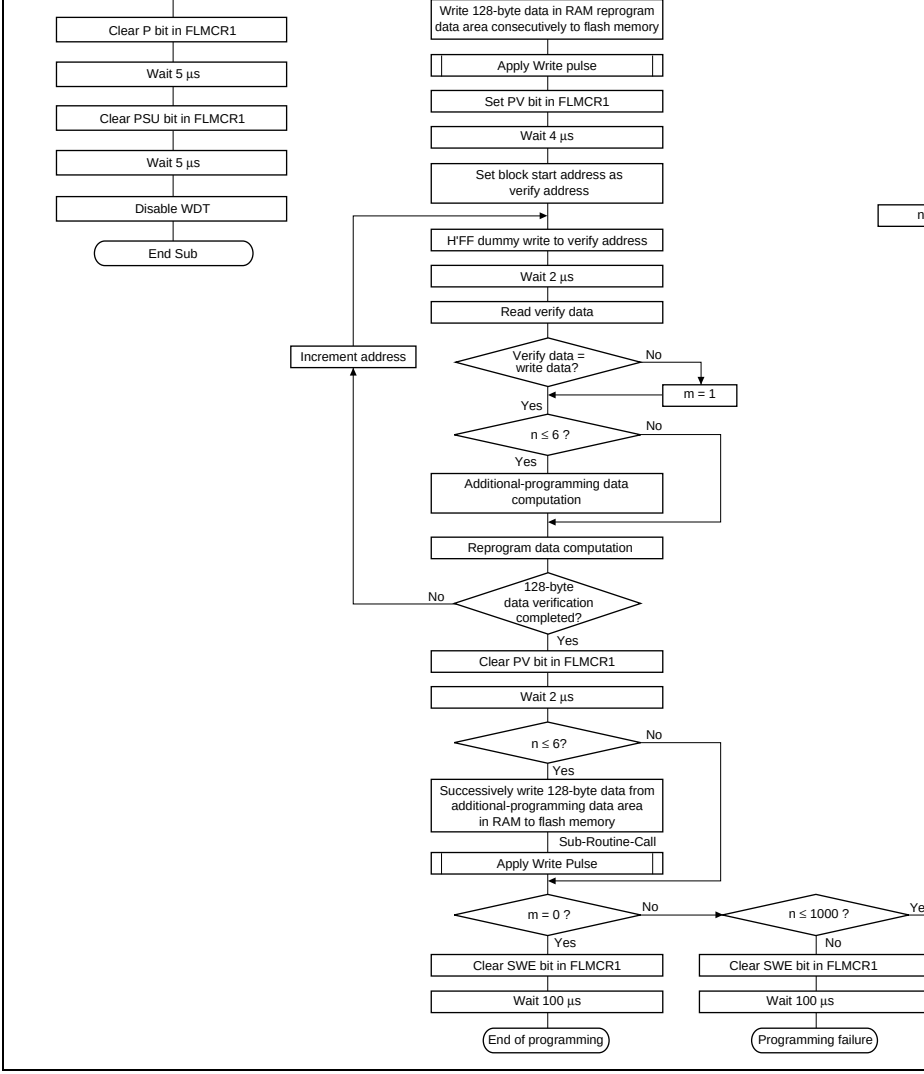


Figure 6.10 Program/Program-Verify Flowchart

Table 6.12 Additional-Program Data Computation Table

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

Table 6.13 Programming Time

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

Note: Time shown in µs.

3. The time during which the E bit is set to 1 is the flash memory erase time.
4. The watchdog timer (WDT) is set to prevent overerasing due to program runaway, overflow cycle of approximately 19.8 ms is allowed.
5. For a dummy write to a verify address, write 1-byte data H'FF to an address whose is b'0. Verify data can be read in word size from the address to which a dummy w performed.
6. If the read data is not erased successfully, set erase mode again, and repeat the eras verify sequence as before. The maximum number of repetitions of the erase/erase sequence is 100.

6.8.3 Interrupt Handling when Programming/Erasing Flash Memory

All interrupts, are disabled while flash memory is being programmed or erased, or wh program is executing, for the following three reasons:

1. Interrupt during programming/erasing may cause a violation of the programming c algorithm, with the result that normal operation cannot be assured.
2. If interrupt exception handling starts before the vector address is written or during programming/erasing, a correct vector cannot be fetched and the CPU malfunction
3. If an interrupt occurs during boot program execution, normal boot mode sequence carried out.

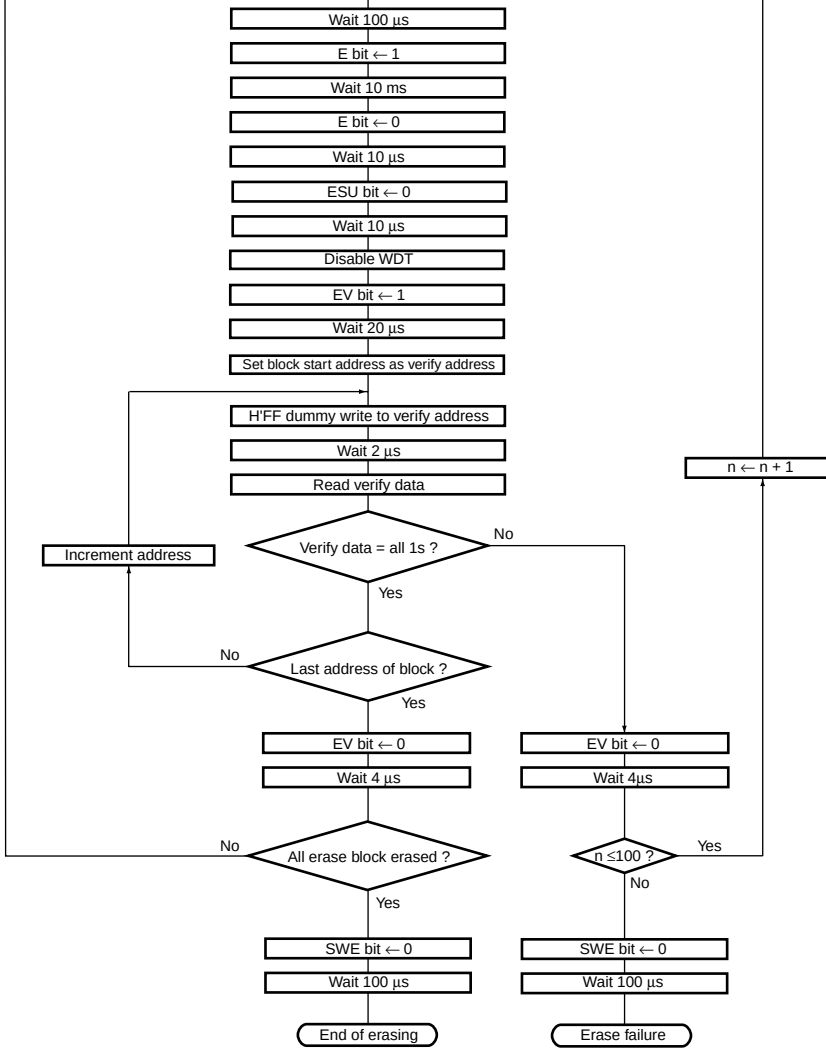


Figure 6.11 Erase/Erase-Verify Flowchart

disabled or aborted because of a transition to reset, subactive mode, subsleep mode, w
or standby mode. Flash memory control register 1 (FLMCR1), flash memory control
(FLMCR2), and erase block register (EBR) are initialized. In a reset via the $\overline{\text{RES}}$ pin,
state is not entered unless the $\overline{\text{RES}}$ pin is held low until oscillation stabilizes after pow
the case of a reset during operation, hold the $\overline{\text{RES}}$ pin low for the $\overline{\text{RES}}$ pulse width spe
AC Characteristics section.

6.9.2 Software Protection

Software protection can be implemented against programming/erasing of all flash mer
by clearing the SWE bit in FLMCR1. When software protection is in effect, setting th
in FLMCR1 does not cause a transition to program mode or erase mode. By setting th
block register (EBR), erase protection can be set for individual blocks. When EBR is
erase protection is set for all blocks.

6.9.3 Error Protection

In error protection, an error is detected when CPU runaway occurs during flash memo
programming/erasing, or operation is not performed in accordance with the program/e
algorithm, and the program/erase operation is aborted. Aborting the program/erase op
prevents damage to the flash memory due to overprogramming or overerasing.

When the following errors are detected during programming/erasing of flash memory,
bit in FLMCR2 is set to 1, and the error protection state is entered.

- When the flash memory of the relevant address area is read during programming/e
  (including vector read and instruction fetch)
- Immediately after exception handling excluding a reset during programming/erasi
- When a SLEEP instruction is executed during programming/erasing

The FLMCR1, FLMCR2, and EBR settings are retained, however program mode or e
aborted at the point at which the error occurred. Program mode or erase mode cannot

(F-ZTAT64V3). A 10-MHz input clock is required. For the conditions for transition to programmer mode, see table 6.8.

6.10.1 Socket Adapter

The socket adapter converts the pin allocation of the F-ZTAT device to that of the discr memory HN28F101. The address of the on-chip flash memory is H'0000 to H'EFFF. Fi shows a socket-adapter-pin correspondence diagram.

6.10.2 Programmer Mode Commands

The following commands are supported in programmer mode.

- Memory Read Mode
- Auto-Program Mode
- Auto-Erase Mode
- Status Read Mode

Status polling is used for auto-programming, auto-erasing, and status read modes. In sta mode, detailed internal information is output after the execution of auto-programming o erasing. Table 6.14 shows the sequence of each command. In auto-programming mode, are required since 128 bytes are written at the same time. In memory read mode, the nu cycles depends on the number of address write cycles (n).

n: the number of address write cycles

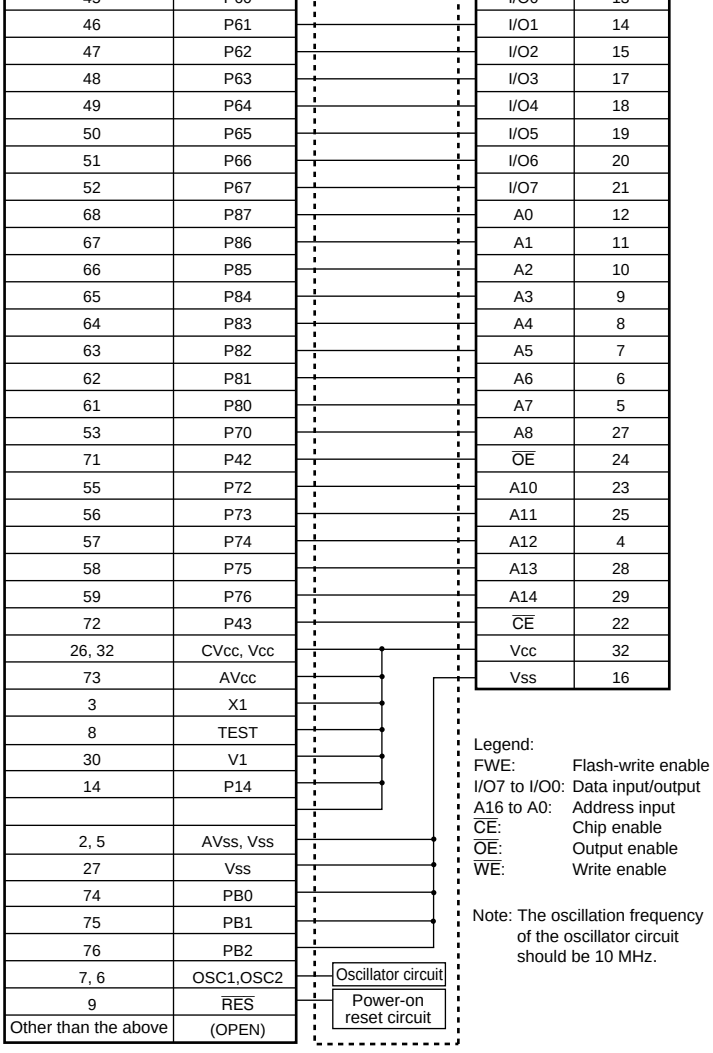


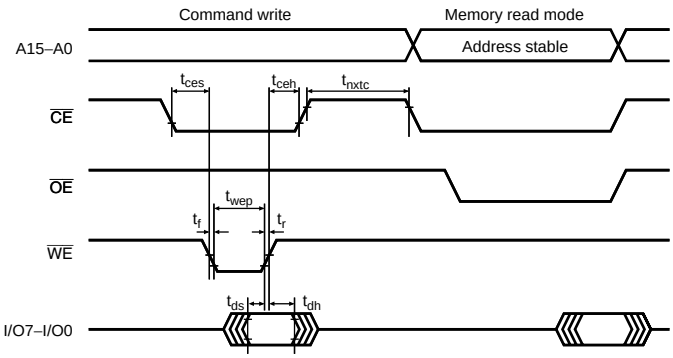
Figure 6.12 Socket Adapter Pin Correspondence Diagram

3. After powering on, memory read mode is entered.
4. Tables 6.15 to 6.17 show the AC characteristics.

Table 6.15 AC Characteristics in Transition to Memory Read Mode

Conditions: V_{CC} = 3.3 V ±0.3 V, V_{SS} = 0 V, T_a = 25°C ±5°C

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	Figure
$\overline{CE}$ hold time	t_{ceh}	0	—	ns	
$\overline{CE}$ setup time	t_{ces}	0	—	ns	
Data hold time	t_{dh}	50	—	ns	
Data setup time	t_{ds}	50	—	ns	
Write pulse width	t_{wep}	70	—	ns	
$\overline{WE}$ rise time	t_r	—	30	ns	
$\overline{WE}$ fall time	t_f	—	30	ns	



Note: Data is latched on the rising edge of $\overline{WE}$.

Figure 6.13 Timing Waveforms for Memory Read after Memory Wri

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

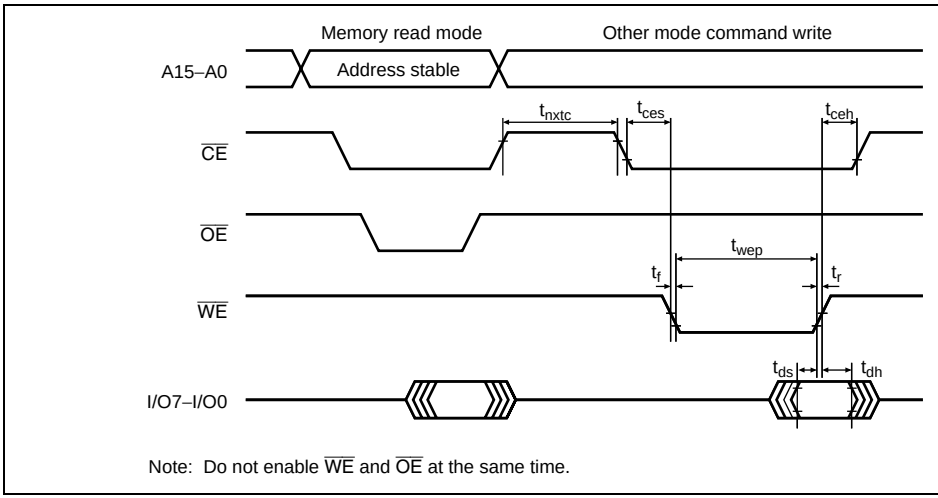


Figure 6.14 Timing Waveforms in Transition from Memory Read Mode to Anot

Output disable delay time	t_{df}	—	100	ns
Data output hold time	t_{oh}	5	—	ns

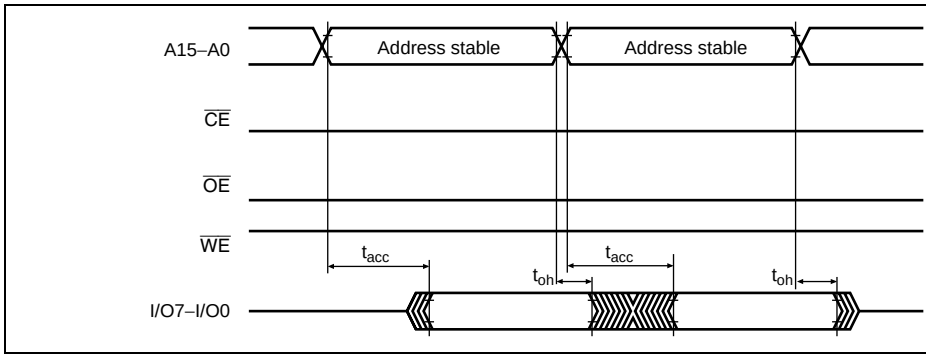


Figure 6.15 $\overline{CE}$ and $\overline{OE}$ Enable State Read Timing Waveforms

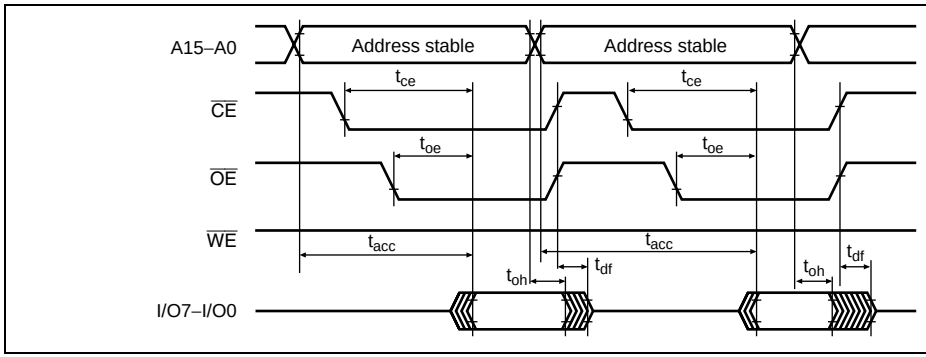


Figure 6.16 $\overline{CE}$ and $\overline{OE}$ Clock System Read Timing Waveforms

programming fewer than 128 bytes. In this case, H'FF data must be written to the ex
addresses.

4. The lower 7 bits of the transfer address must be low. If a value other than an effecti
is input, processing will switch to a memory write operation but a write error will b
5. Memory address transfer is performed in the second cycle (figure 6.17). Do not per
transfer after the third cycle.
6. Do not perform a command write during a programming operation.
7. Perform one auto-program operation for a 128-byte block for each address. Two or
additional programming operations cannot be performed on a previously programm
block.
8. Confirm normal end of auto-programming by checking I/O6. Alternatively, status r
can also be used for this purpose (I/O7 status polling uses the auto-program operati
decision pin).
9. Status polling I/O6 and I/O7 pin information is retained until the next command wri
as the next command write has not been performed, reading is possible by enabling
$\overline{OE}$.
10. Table 6.18 shows the AC characteristics.

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
Status polling start time	t_{wsts}	1	—	ms
Status polling access time	t_{spa}	—	150	ns
Address setup time	t_{as}	0	—	ns
Address hold time	t_{ah}	60	—	ns
Memory write time	t_{write}	1	3000	ms
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns

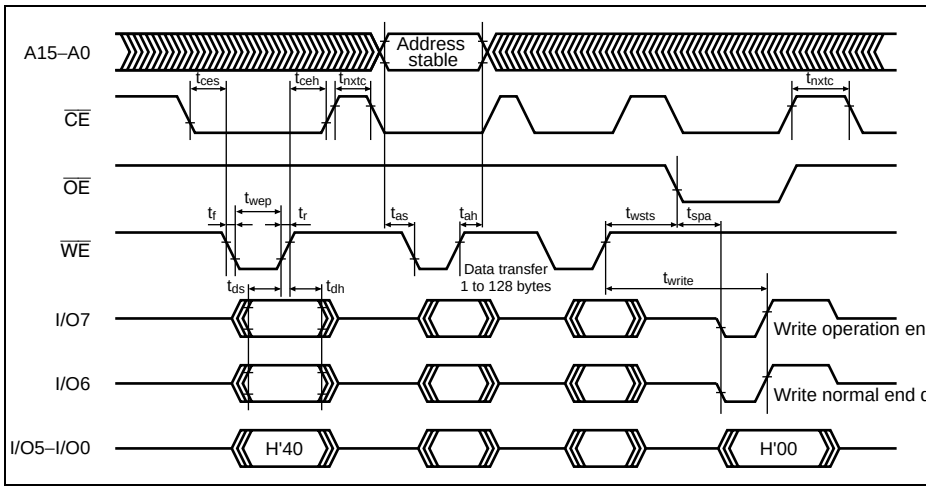


Figure 6.17 Auto-Program Mode Timing Waveforms

$\overline{OE}$.

5. Table 6.19 shows the AC characteristics.

Table 6.19 AC Characteristics in Auto-Erase Mode

Conditions: V_{CC} = 3.3 V ±0.3 V, V_{SS} = 0 V, T_a = 25°C ±5°C

Item	Symbol	Min	Max	Unit	Notes
Command write cycle	t_{nxtc}	20	—	µs	Figure 6
$\overline{CE}$ hold time	t_{ceh}	0	—	ns	
$\overline{CE}$ setup time	t_{ces}	0	—	ns	
Data hold time	t_{dh}	50	—	ns	
Data setup time	t_{ds}	50	—	ns	
Write pulse width	t_{wep}	70	—	ns	
Status polling start time	t_{ests}	1	—	ms	
Status polling access time	t_{spa}	—	150	ns	
Memory erase time	t_{erase}	100	40000	ms	
$\overline{WE}$ rise time	t_r	—	30	ns	
$\overline{WE}$ fall time	t_f	—	30	ns	

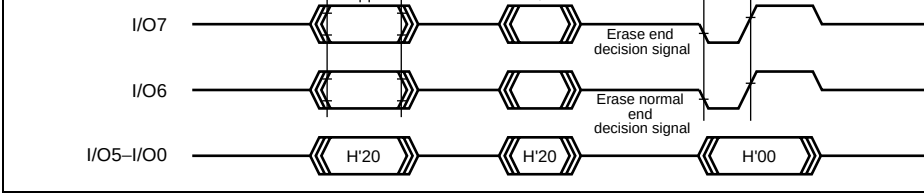
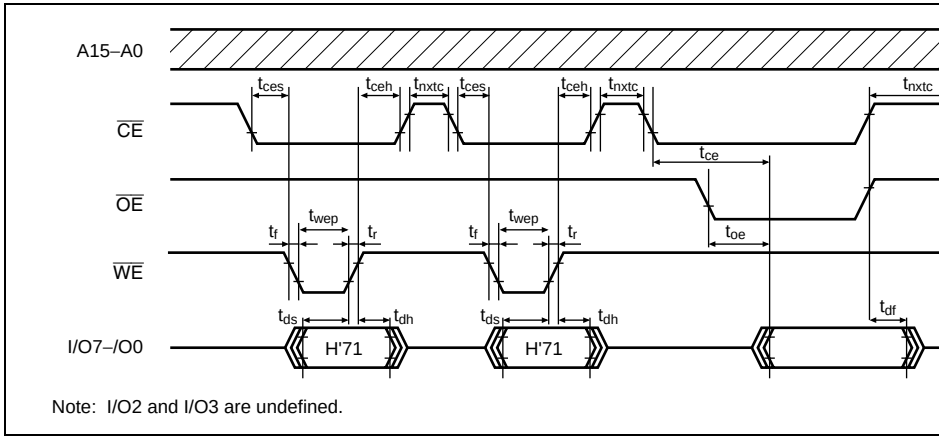


Figure 6.18 Auto-Erase Mode Timing Waveforms

6.10.6 Status Read Mode

1. Status read mode is provided to identify the kind of abnormal end. Use this mode w abnormal end occurs in auto-program mode or auto-erase mode.
2. The return code is retained until a command write other than a status read mode co write is executed.
3. Table 6.20 shows the AC characteristics and 6.21 shows the return codes.

Data hold time	t_{dh}	50	—	ns
Data setup time	t_{ds}	50	—	ns
Write pulse width	t_{wep}	70	—	ns
$\overline{OE}$ output delay time	t_{oe}	—	150	ns
Disable delay time	t_{df}	—	100	ns
$\overline{CE}$ output delay time	t_{ce}	—	150	ns
$\overline{WE}$ rise time	t_r	—	30	ns
$\overline{WE}$ fall time	t_f	—	30	ns



Note: I/O2 and I/O3 are undefined.

Figure 6.19 Status Read Mode Timing Waveforms

		0: Otherwise
I/O4	0	1: Erasing error 0: Otherwise
I/O3	0	—
I/O2	0	—
I/O1	0	1: Over counting of writing or erasing 0: Otherwise
I/O0	0	1: Effective address error 0: Otherwise

6.10.7 Status Polling

1. The I/O7 status polling flag indicates the operating status in auto-program/auto-era
2. The I/O6 status polling flag indicates a normal or abnormal end in auto-program/a mode.

Table 6.22 Status Polling Output Truth Table

I/O7	I/O6	I/O0 to 5	Status
0	0	0	During internal operati
1	0	0	Abnormal end
1	1	0	Normal end
0	1	0	—

Oscillation stabilization time(ceramic oscillator)	T_{osc1}	5	—	ms
Programmer mode setup time	T_{bmv}	10	—	ms
Vcc hold time	T_{dwn}	0	—	ms

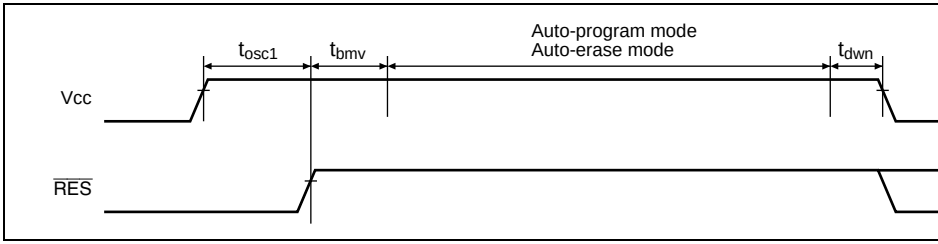


Figure 6.20 Oscillation Stabilization Time, Boot Program Transfer Tim and Power-Down Sequence

6.10.9 Notes on Memory Programming

1. When performing programming using programmer mode on a chip that has been programmed/erased in an on-board programming mode, auto-erasing is recommend carrying out auto-programming.
2. The flash memory is initially in the erased state when the device is shipped by Rene Technology. For other chips for which the erasure history is unknown, it is recomm auto-erasing be executed to check and supplement the initialization (erase) level.

consumption.

- Standby mode

  All flash memory circuits are halted.

Table 6.24 shows the correspondence between the operating modes of this LSI and the memory. In subactive mode, the flash memory can be set to operate in power-down m PDWND bit in FLPWCR. When the flash memory returns to its normal operating stat power-down mode or standby mode, a period to stabilize the power supply circuits tha stopped is needed. When the flash memory returns to its normal operating state, bits S STS0 in SYSCR1 must be set to provide a wait time of at least 20 µs, even when the e clock is being used.

Table 6.24 Flash Memory Operating States

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

H8/3827S, H8/38527, and H8/38427 have 2 Kbytes. The RAM is connected to the CP
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.

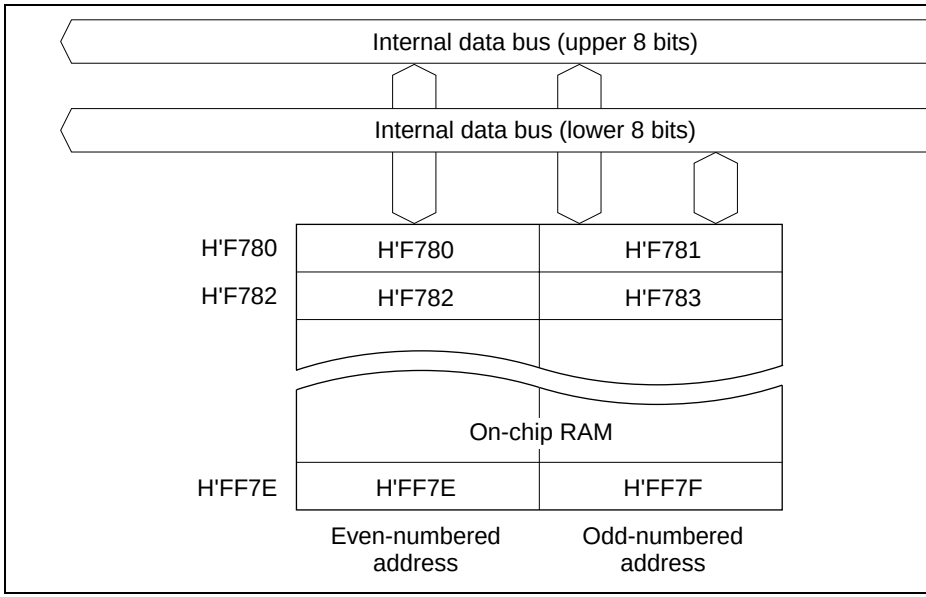


Figure 7.1 RAM Block Diagram (H8/3824R, H8/3824S, H8/38324, and H8/

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

Ports 5, 6, 7, 8, and A are also used as liquid crystal display segment and common pin
in 8-bit units.

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	F S R
Port 1	• 8-bit I/O port • MOS input pull-up option	$P1_7$ to $P1_5$/ $\overline{IRQ_3}$ to $\overline{IRQ_1}$/ TMIF, TMIC	External interrupts 3 to 1 Timer event interrupts TMIF, TMIC	F T T
		$P1_4/\overline{IRQ_4}/\overline{ADTRG}$	External interrupt 4 and A/D converter external trigger	F
		$P1_3$/TMIG	Timer G input capture input	F
		$P1_2$, $P1_1$/ TMOFH, TMOFL	Timer F output compare output	F
		$P1_0$/TMOW	Timer A clock output	F

	H8/38427 Group)	$P3_1$/UD/EXCL[*2] $P3_0$/PWM	count-up/down select input, 14-bit PWM output, and external subclock input[*2]	PM
Port 4	• 1-bit input port	$P4_3$/$\overline{IRQ_0}$	External interrupt 0	PM
	• 3-bit I/O port	$P4_2$/TXD_{32} $P4_1$/RXD_{32} $P4_0$/SCK_{32}	SCI3-2 data output (TXD_{32}), data input (RXD_{32}), clock input/output (SCK_{32})	SC SM
Port 5	• 8-bit I/O port • MOS input pull-up option	$P5_7$ to $P5_0$/ $\overline{WKP_7}$ to $\overline{WKP_0}$/ SEG_8 to SEG_1	Wakeup input ($\overline{WKP_7}$ to $\overline{WKP_0}$), segment output (SEG_8 to SEG_1)	PM LP
Port 6	• 8-bit I/O port • MOS input pull-up option	$P6_7$ to $P6_0$/ SEG_{16} to SEG_9	Segment output (SEG_{16} to SEG_9)	LP
Port 7	• 8-bit I/O port	$P7_7$ to $P7_0$/ SEG_{24} to SEG_{17}	Segment output (SEG_{24} to SEG_{17})	LP
Port 8	• 8-bit I/O port	$P8_7$/SEG_{32}/CL_1[*3] $P8_6$/SEG_{31}/CL_2[*3] $P8_5$/SEG_{30}/DO[*3] $P8_4$/SEG_{29}/M[*3] $P8_3$ to $P8_0$/ SEG_{28} to SEG_{25}	Segment output (SEG_{32} to SEG_{25}) Segment external expansion latch clock (CL_1)[*3], shift clock (CL_2)[*3], display data (DO)[*3], alternation signal (M)[*3]	LP
Port A	4-bit I/O port	PA_3 to PA_0/ COM_4 to COM_1	Common output (COM_4 to COM_1)	LP
Port B	8-bit input port	PB_7 to PB_0/ AN_7 to AN_0	A/D converter analog input	AM

Notes: 1. The $\overline{RESO}$ function is not implemented in the H8/38327 Group and H8/3842
2. The EXCL function is only implemented in the H8/38327 Group and H8/3842
3. The external expansion function for LCD segments is not implemented in the Group and H8/38427 Group.

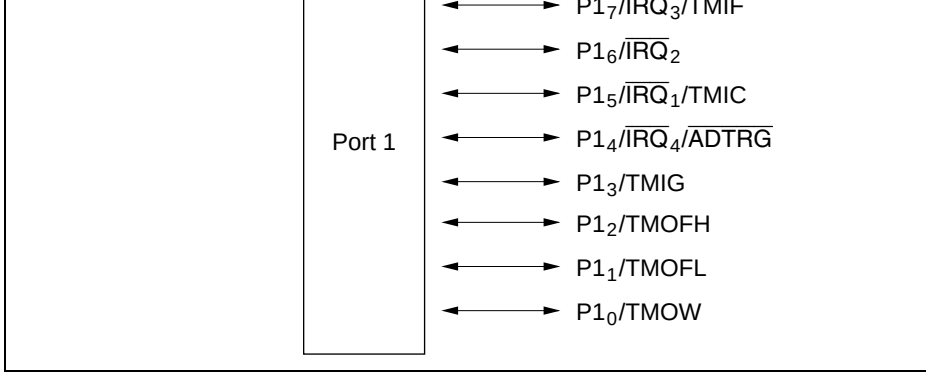


Figure 8.1 Port 1 Pin Configuration

8.2.2 Register Configuration and Description

Table 8.2 shows the port 1 register configuration.

Table 8.2 Port 1 Registers

Name	Abbr.	R/W	Initial Value
Port data register 1	PDR1	R/W	H'00
Port control register 1	PCR1	W	H'00
Port pull-up control register 1	PUCR1	R/W	H'00
Port mode register 1	PMR1	R/W	H'00

bits are set to 1, the values stored in PDR1 are read, regardless of the actual pin states. read while PCR1 bits are cleared to 0, the pin states are read.

Upon reset, PDR1 is initialized to H'00.

2. Port Control Register 1 (PCR1)

Bit	7	6	5	4	3	2	1
	$PCR1_7$	$PCR1_6$	$PCR1_5$	$PCR1_4$	$PCR1_3$	$PCR1_2$	$PCR1_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 $P1_0$ func input pin or output pin. Setting a PCR1 bit to 1 makes the corresponding pin an output clearing the bit to 0 makes the pin an input pin. The settings in PCR1 and in PDR1 are 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, which is always read as all 1s.

a PCR1 bit is cleared to 0, setting the corresponding PUCR1 bit to 1 turns on the MOS
the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR1 is initialized to H'00.

4. Port Mode Register 1 (PMR1)

Bit	7	6	5	4	3	2	1
	IRQ3	IRQ2	IRQ1	IRQ4	TMIG	TMOFH	TMOFL
Initial value	0	0	0	0	0	0	0
Read/Write	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

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
1	Functions as $\overline{IRQ_3}$/TMIF input pin

Note: Rising or falling edge sensing can be designated for IRQ_3, TMIF. For details on settings, see 3. Timer Control Register F (TCRF) in section 9.4.2.

Note: Rising or falling edge sensing can be designated for $\overline{IRQ_2}$.

Bit 5: $P1_5/\overline{IRQ_1}$/TMIC pin function switch (IRQ1)

This bit selects whether pin $P1_5/\overline{IRQ_1}$/TMIC is used as $P1_5$ or as $\overline{IRQ_1}$/TMIC.

Bit 5 IRQ1	Description	
0	Functions as $P1_5$ I/O pin	(i
1	Functions as $\overline{IRQ_1}$/TMIC input pin	

Note: Rising or falling edge sensing can be designated for $\overline{IRQ_1}$/TMIC.
For details of TMIC pin setting, see 1. Timer mode register C (TMC) in section 9.

Bit 4: $P1_4/\overline{IRQ_4}/\overline{ADTRG}$ pin function switch (IRQ4)

This bit selects whether pin $P1_4/\overline{IRQ_4}/\overline{ADTRG}$ is used as $P1_4$ or as $\overline{IRQ_4}/\overline{ADTRG}$.

Bit 4 IRQ4	Description	
0	Functions as $P1_4$ I/O pin	(i
1	Functions as $\overline{IRQ_4}/\overline{ADTRG}$ input pin	

Note: For details of $\overline{ADTRG}$ pin setting, see section 12.3.2, Start of A/D Conversion by Trigger Input.

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
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
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
1	Functions as TMOW output pin

and bit PCR1$_7$ in PCR1.

IRQ_3	0		1	
$PCR1_7$	0	1	*	
CKSL2 to CKSL0	*		Not 0**	
Pin function	$P1_7$ input pin	$P1_7$ output pin	$\overline{IRQ}_3$ input pin	$\overline{I}$

Note: When this pin is used as the TMIF input pin, clear bit IEN3 to 0 to disable the IRQ_3 interrupt.

$P1_6/\overline{IRQ}_2$

The pin function depends on bits IRQ2 in PMR1 and bit $PCR1_6$ in PCI

IRQ2	0		1
$PCR1_6$	0	1	*
Pin function	$P1_6$ input pin	$P1_6$ output pin	$\overline{IRQ}_2$ input

$P1_5/\overline{IRQ}_1$
TMIC

The pin function depends on bit IRQ1 in PMR1, bits TMC2 to TMC0 in bit $PCR1_5$ in PCR1.

IRQ1	0		1	
$PCR1_5$	0	1	*	
TMC2 to TMC0	*		Not 111	
Pin function	$P1_5$ input pin	$P1_5$ output pin	$\overline{IRQ}_1$ input pin	$\overline{I}$

Note: When this pin is used as the TMIC input pin, clear bit IEN1 to 0 to disable the IRQ_1 interrupt.

Note: When this pin is used as the $\overline{\text{ADTRG}}$ input pin, clear bit IEN4 IENR1 to disable the IRQ_4 interrupt.

$P1_3$/TMIG — The pin function depends on bit TMIG in PMR1 and bit $PCR1_3$ in PC

TMIG	0		1
$PCR1_3$	0	1	*
Pin function	$P1_3$ input pin	$P1_3$ output pin	TMIG inpu

$P1_2$/TMOFH — The pin function depends on bit TMOFH in PMR1 and bit $PCR1_2$ in P

TMOFH	0		1
$PCR1_2$	0	1	*
Pin function	$P1_2$ input pin	$P1_2$ output pin	TMOFH out

$P1_1$/TMOFL — The pin function depends on bit TMOFL in PMR1 and bit $PCR1_1$ in P

TMOFL	0		1
$PCR1_1$	0	1	*
Pin function	$P1_1$ input pin	$P1_1$ output pin	TMOFL out

$P1_0$/TMOW — The pin function depends on bit TMOW in PMR1 and bit $PCR1_0$ in P

TMOW	0		1
$PCR1_0$	0	1	*
Pin function	$P1_0$ input pin	$P1_0$ output pin	TMOW out

$P1_5/\overline{IRQ_1}$/TMIC		state	state		state
$P1_4/\overline{IRQ_4}$/$\overline{ADTRG}$					
$P1_3$/TMIG					
$P1_2$/TMOFH					
$P1_1$/TMOFL					
$P1_0$/TMOW					

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

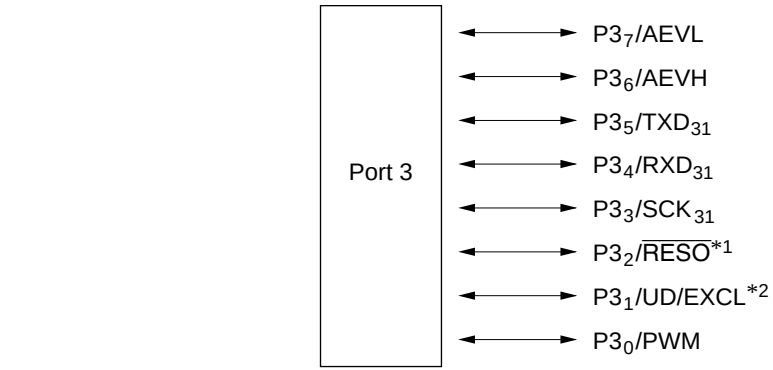
8.2.5 MOS Input Pull-Up

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

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

*

making connections to external circuitry. Note that in the mask ROM and ZTAT versi
continues to operate normally.



Notes: 1. The $\overline{\text{RESO}}$ function is not implemented in the H8/38327 Group and H8/384
2. The EXCL function only applies to the H8/38327 Group and H8/38427 Grou

Figure 8.2 Port 3 Pin Configuration

8.3.2 Register Configuration and Description

Table 8.5 shows the port 3 register configuration.

Table 8.5 Port 3 Registers

Name	Abbr.	R/W	Initial Value
Port data register 3	PDR3	R/W	H'00
Port control register 3	PCR3	W	H'00
Port pull-up control register 3	PUCR3	R/W	H'00
Port mode register 2	PMR2	R/W	H'58
Port mode register 3	PMR3	R/W	H'04

bits are set to 1, the values stored in PDR3 are read, regardless of the actual pin states. read while PCR3 bits are cleared to 0, the pin states are read.

Upon reset, PDR3 is initialized to H'00.

2. Port Control Register 3 (PCR3)

Bit	7	6	5	4	3	2	1
	$PCR3_7$	$PCR3_6$	$PCR3_5$	$PCR3_4$	$PCR3_3$	$PCR3_2$	$PCR3_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 $P3_0$ func input pin or output pin. Setting a PCR3 bit to 1 makes the corresponding pin an output clearing the bit to 0 makes the pin an input pin. The settings in PCR3 and in PDR3 are 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, which is always read as all 1s.

a PCR3 bit is cleared to 0, setting the corresponding PUCR3 bit to 1 turns on the MOS
the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR3 is initialized to H'00.

4. Port Mode Register 3 (PMR3)

Bit	7	6	5	4	3	2	1
	AEVL	AEVH	WDCKS	NCS	IRQ0	RESO*	UD
Initial value	0	0	0	0	0	1	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: * The RESO bit is not implemented in the H8/38327 Group and H8/38427 Gr

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

Upon reset, PMR3 is initialized to H'04.

Bit 7: $P3_7$/AEVL pin function switch (AEVL)

This bit selects whether pin $P3_7$/AEVL is used as $P3_7$ or as AEVL.

Bit 7 AEVL	Description
0	Functions as $P3_7$ I/O pin
1	Functions as AEVL input pin

Bit 5: Watchdog timer source clock select (WDCKS)

This bit selects the watchdog timer source clock.

Bit 5 WDCKS	Description	
0	ϕ/8192 selected	(i
1	ϕw/32 selected	

Bit 4: TMIG noise canceler select (NCS)

This bit controls the noise canceler for the input capture input signal (TMIG).

Bit 4 NCS	Description	
0	Noise cancellation function not used	(i
1	Noise cancellation function used	

Bit 3: $P4_3/\overline{IRQ}_0$ pin function switch (IRQ_0)

This bit selects whether pin $P4_3/\overline{IRQ}_0$ is used as $P4_3$ or as $\overline{IRQ}_0$.

Bit 3 IRQ_0	Description	
0	Functions as $P4_3$ input pin	(i
1	Functions as $\overline{IRQ}_0$ input pin	

1	Functions as $\overline{\text{RESO}}$ output pin

Bit 1: $P3_1$/UD pin function switch (SI1)

This bit selects whether pin $P3_1$/UD is used as $P3_1$ or as UD.

Bit 1 UD	Description
0	Functions as $P3_1$ I/O pin
1	Functions as UD input pin

Bit 0: $P3_0$/PWM pin function switch (PWM)

This bit selects whether pin $P3_0$/PWM is used as $P3_0$ or as PWM.

Bit 0 PWM	Description
0	Functions as $P3_0$ I/O pin
1	Functions as PWM output pin

Upon reset, PMR2 is initialized to H'58. The information on this register applies to the
Group and H8/38427 Group.

Bit 7: $P3_1$/UD/EXCL pin function switch (EXCL)

This bit selects whether pin $P3_1$/UD/EXCL is used as $P3_1$/UD or as EXCL. When the p
as EXCL an external clock should be input to it. See section 4, Clock Pulse Generators,
connection example.

Bit 7 EXCL	Description	
0	Functions as $P3_1$/UD I/O pin	(i
1	Functions as EXCL input pin	

Bit 6: Reserved bit

Bit 6 is a reserved bit. It is always read as 1 and cannot be modified.

Bit 5: Reserved bit

Bit 5 is a readable/writable reserved bit.

Bits 4 and 3: Reserved bits

Bits 4 and 3 are reserved bits. They are always read as 1 and cannot be modified.

Bits 2 to 0: Reserved bits

Bits 2 to 0 are readable/writable reserved bits.

AEVL	0		
$PCR3_7$	0	1	
Pin function	$P3_7$ input pin	$P3_7$ output pin	AEV

$P3_6$/AEVH — The pin function depends on bit AEVH in PMR3 and bit $PCR3_6$ in PC

AEVH	0		
$PCR3_6$	0	1	
Pin function	$P3_6$ input pin	$P3_6$ output pin	AEV

$P3_5$/TXD_{31} — The pin function depends on bit TE_{31} in SCR3-1, bit SPC31 in SPCR $PCR3_5$ in PCR3.

SPC31	0		
TE_{31}	0		
$PCR3_5$	0	1	
Pin function	$P3_5$ input pin	$P3_5$ output pin	TXD_3

$P3_4$/RXD_{31} — The pin function depends on bit RE_{31} in SCR3-1 and bit $PCR3_4$ in P

RE_{31}	0		
$PCR3_4$	0	1	
Pin function	$P3_4$ input pin	$P3_4$ output pin	RXD

$P3_3$/SCK_{31} — The pin function depends on bits CKE311, CKE310, and SMR31 in bit $PCR3_3$ in PCR3.

CKE311	0				
CKE310	0			1	
$COM3_1$	0		1	*	
$PCR3_3$	0	1	*		
Pin function	$P3_3$ input pin	$P3_3$ output pin	SCK_{31} output pin		

H8/38427)

The pin function depends on bit $PCR3_2$ in PCR3.

$PCR3_2$	0	1
Pin function	$P3_2$ input pin	$P3_2$ output pin

$P3_1$/UD (H8/3827R, H8/3827S)

- H8/3827R Group and H8/3827S Group

  The pin function depends on bit UD in PMR3 and bit $PCR3_1$ in PC

UD	0		1
$PCR3_1$	0	1	*
Pin function	$P3_1$ input pin	$P3_1$ output pin	UD input pin

$P3_1$/UD/EXCL (H8/38327, H8/38427)

- H8/38327 Group and H8/38427 Group

  The pin function depends on bit EXCL in PMR2, bit UD in PMR3, $PCR3_1$ in PCR3.

EXCL	0			
UD	0		1	
$PCR3_1$	0	1	*	
Pin function	$P3_1$ input pin	$P3_1$ output pin	UD input pin	EX

$P3_0$/PWM

The pin function depends on bit PWM in PMR3 and bit $PCR3_0$ in PCR

PWM	0		
$PCR3_0$	0	1	
Pin function	$P3_0$ input pin	$P3_0$ output pin	PWM

*

$P3_5$/TXD_{31} $P3_4$/RXD_{31} $P3_3$/SCK_{31}					
$P3_2$/$\overline{RESO}$*2	RESO output				
$P3_2$*4	Pull-up MOS on				
$P3_2$*3 $P3_1$/UD*2 $P3_1$/UD/EXCL*3*4 $P3_0$/PWM	High-impedance				

Notes: 1. A high-level signal is output when the MOS pull-up is in the on state.
2. Applies to H8/3827R Group and H8/3827S Group.
3. Applies to the mask ROM version of the H8/38327 Group and H8/38427 Gr
4. Applies to the F-ZTAT version of the H8/38327 Group and H8/38427 Group

8.3.5 MOS Input Pull-Up

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

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

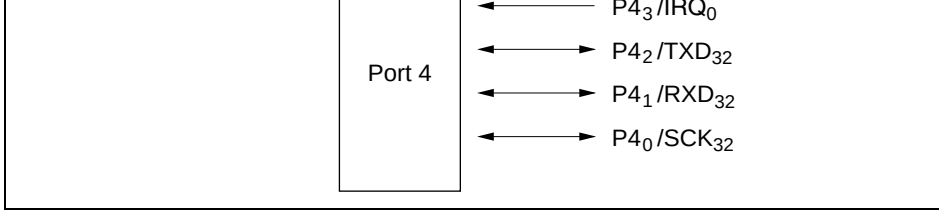


Figure 8.3 Port 4 Pin Configuration

8.4.2 Register Configuration and Description

Table 8.8 shows the port 4 register configuration.

Table 8.8 Port 4 Registers

Name	Abbr.	R/W	Initial Value
Port data register 4	PDR4	R/W	H'F8
Port control register 4	PCR4	W	H'F8

bits are set to 1, the values stored in PDR4 are read, regardless of the actual pin states. read while PCR4 bits are cleared to 0, the pin states are read.

Upon reset, PDR4 is initialized to H'F8.

2. Port Control Register 4 (PCR4)

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

PCR4 is an 8-bit register for controlling whether each of port 4 pins $P4_2$ to $P4_0$ functio input pin or output pin. Setting a PCR4 bit to 1 makes the corresponding pin an outpu clearing the bit to 0 makes the pin an input pin. PCR4 and PDR4 settings are valid wh corresponding pins are designated for general-purpose input/output by SCR3-2.

Upon reset, PCR4 is initialized to H'F8.

PCR4 is a write-only register, which always reads all 1s.

IRQ0	0	1
Pin function	$P4_3$ input pin	$\overline{IRQ}_0$ input

$P4_2/TXD_{32}$ The pin function depends on bit TE_{32} in SCR3-2, bit SPC32 in SPCR, and bit $PCR4_2$ in PCR4.

SPC32	0		
TE_{32}	0		
$PCR4_2$	0	1	
Pin function	$P4_2$ input pin	$P4_2$ output pin	TXD_{32}

$P4_1/RXD_{32}$ The pin function depends on bit RE_{32} in SCR3-2 and bit $PCR4_1$ in PC

RE_{32}	0		
$PCR4_1$	0	1	
Pin function	$P4_1$ input pin	$P4_1$ output pin	RXD_3

$P4_0/SCK_{32}$ The pin function depends on bit CKE321 and CKE320 in SCR3-2, bit C/A in SMR32, and bit $PCR4_0$ in PCR4.

CKE321	0			
CKE320	0		1	
COM32	0		1	*
$PCR4_0$	0	1	*	
Pin function	$P4_0$ input pin	$P4_0$ output pin	SCK_{32} output pin	S

*

$P4_1/RXD_{32}$ $P4_0/SCK_{32}$		Impedance	Previous state	Previous state	Impedance	Previous state

8.5 Port 5

8.5.1 Overview

Port 5 is an 8-bit I/O port, configured as shown in figure 8.4.

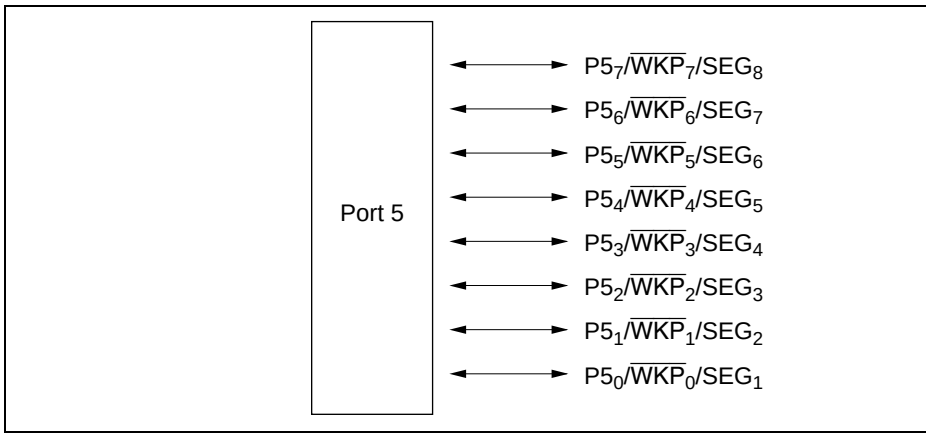


Figure 8.4 Port 5 Pin Configuration

Port pull-up control register 5	PUCR5	R/W	H'00
Port mode register 5	PMR5	R/W	H'00

1. Port Data Register 5 (PDR5)

Bit	7	6	5	4	3	2	1
	$P5_7$	$P5_6$	$P5_5$	$P5_4$	$P5_3$	$P5_2$	$P5_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 wh bits are set to 1, the values stored in PDR5 are read, regardless of the actual pin states. read while PCR5 bits are cleared to 0, the pin states are read.

Upon reset, PDR5 is initialized to H'00.

2. Port Control Register 5 (PCR5)

Bit	7	6	5	4	3	2	1
	$PCR5_7$	$PCR5_6$	$PCR5_5$	$PCR5_4$	$PCR5_3$	$PCR5_2$	$PCR5_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 $P5_0$ func input pin or output pin. Setting a PCR5 bit to 1 makes the corresponding pin an output clearing the bit to 0 makes the pin an input pin. PCR5 and PDR5 settings are valid whe corresponding pins are designated for general-purpose input/output by PMR5 and bits S SGS0 in LPCR.

Upon reset, PCR5 is initialized to H'00.

PUCR5 controls whether the MOS pull-up of each of port 5 pins $P5_7$ to $P5_0$ is on or of
PCR5 bit is cleared to 0, setting the corresponding PUCR5 bit to 1 turns on the MOS p
the corresponding pin, while clearing the bit to 0 turns off the MOS pull-up.

Upon reset, PUCR5 is initialized to H'00.

4. Port Mode Register 5 (PMR5)

Bit	7	6	5	4	3	2	1
	WKP_7	WKP_6	WKP_5	WKP_4	WKP_3	WKP_2	WKP_1
Initial value	0	0	0	0	0	0	0
Read/Write	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

Upon reset, PMR5 is initialized to H'00.

Bit n: $P5_n/\overline{WKP}_n/SEG_{n+1}$ pin function switch (WKPn)

When pin P5n/$\overline{WKP}$n/SEGn+1 is not used as SEG_{n+1}, these bits select whether the pin
P5n or $\overline{WKP}_n$.

Bit n WKPn	Description
0	Functions as P5n I/O pin
1	Functions as $\overline{WKP}_n$ input pin

Note: For use as SEG_{n+1}, see section 13.2.1, LCD Port Control Register (LPCR).

$P5_0/\overline{WKP}_0/SEG_1$

(n

SGS3 to SGS0	0***			
WKP_n	0		1	
$PCR5_n$	0	1	*	
Pin function	$P5_n$ input pin	$P5_n$ output pin	$\overline{WKP}n$ input pin	

*

8.5.4 Pin States

Table 8.13 shows the port 5 pin states in each operating mode.

Table 8.13 Port 5 Pin States

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

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

MOS input pull-up	Off	On	Off

8.6 Port 6

8.6.1 Overview

Port 6 is an 8-bit I/O port. The port 6 pin configuration is shown in figure 8.5.

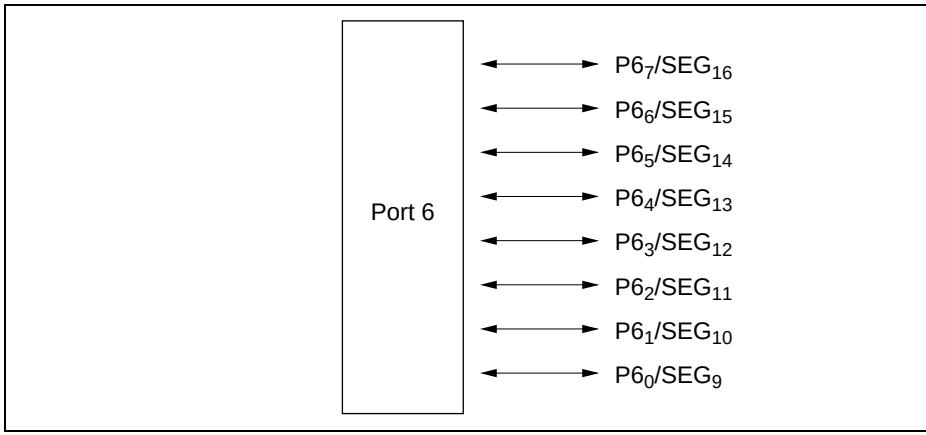


Figure 8.5 Port 6 Pin Configuration

Port pull-up control register 6	PUCR6	R/W	H'00

1. Port Data Register 6 (PDR6)

Bit	7	6	5	4	3	2	1
	$P6_7$	$P6_6$	$P6_5$	$P6_4$	$P6_3$	$P6_2$	$P6_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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, regard
actual pin states. If port 6 is read while PCR6 bits are cleared to 0, the pin states are rea

Upon reset, PDR6 is initialized to H'00.

2. Port Control Register 6 (PCR6)

Bit	7	6	5	4	3	2	1
	$PCR6_7$	$PCR6_6$	$PCR6_5$	$PCR6_4$	$PCR6_3$	$PCR6_2$	$PCR6_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 $P6_0$ func
input pin or output pin.

Setting a PCR6 bit to 1 makes the corresponding pin ($P6_7$ to $P6_0$) an output pin, while c
bit to 0 makes the pin an input pin. PCR6 and PDR6 settings are valid when the corres
pins are designated for general-purpose input/output by bits SGS3 to SGS0 in LPCR.

Upon reset, PCR6 is initialized to H'00.

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

Upon reset, PUCR6 is initialized to H'00.

8.6.3 Pin Functions

Table 8.15 shows the port 6 pin functions.

Table 8.15 Port 6 Pin Functions

Pin	Pin Functions and Selection Method		
$P6_7/SEG_{16}$ to $P6_0/SEG_9$	The pin function depends on bit PCR6n in PCR6 and bits SGS3 to S LPCR.		(n
	SEG3 to SEGS0	00**, 010*	01
	$PCR6_n$	0	1
	Pin function	$P6_n$ input pin	$P6_n$ output pin

$P6_0/SEG_9$	impedance	previous state	previous state	impedance	previous state

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

$PCR6_n$	0	0	1
$PUCR6_n$	0	1	*
MOS input pull-up	Off	On	Off

*

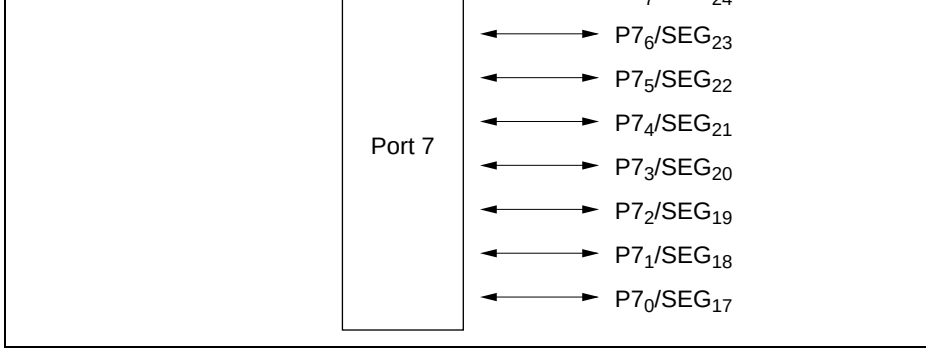


Figure 8.6 Port 7 Pin Configuration

8.7.2 Register Configuration and Description

Table 8.17 shows the port 7 register configuration.

Table 8.17 Port 7 Registers

Name	Abbr.	R/W	Initial Value
Port data register 7	PDR7	R/W	H'00
Port control register 7	PCR7	W	H'00

PDR7 is an 8-bit register that stores data for port 7 pins $P7_7$ to $P7_0$. If port 7 is read wh
bits are set to 1, the values stored in PDR7 are read, regardless of the actual pin states.
read while PCR7 bits are cleared to 0, the pin states are read.

Upon reset, PDR7 is initialized to H'00.

2. Port Control Register 7 (PCR7)

Bit	7	6	5	4	3	2	1
	$PCR7_7$	$PCR7_6$	$PCR7_5$	$PCR7_4$	$PCR7_3$	$PCR7_2$	$PCR7_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 $P7_0$ func
input pin or output pin. Setting a PCR7 bit to 1 makes the corresponding pin an output
clearing the bit to 0 makes the pin an input pin. PCR7 and PDR7 settings are valid whe
corresponding pins are designated for general-purpose input/output by bits SGS3 to SG
LPCR.

Upon reset, PCR7 is initialized to H'00.

PCR7 is a write-only register, which always reads as all 1s.

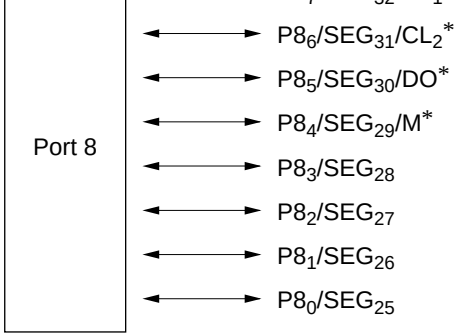
SEGS3 to SEGS0	00**		01
$PCR7_n$	0	1	
Pin function	$P7_n$ input pin	$P7_n$ output pin	SEG_n

8.7.4 Pin States

Table 8.19 shows the port 7 pin states in each operating mode.

Table 8.19 Port 7 Pin States

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



Note: * The CL_1, CL_2, DO, and M functions are not implemented on the H8/38327 G
H8/38427 Group.

Figure 8.7 Port 8 Pin Configuration

8.8.2 Register Configuration and Description

Table 8.20 shows the port 8 register configuration.

Table 8.20 Port 8 Registers

Name	Abbr.	R/W	Initial Value
Port data register 8	PDR8	R/W	H'00
Port control register 8	PCR8	W	H'00

bits are set to 1, the values stored in PDR8 are read, regardless of the actual pin states. read while PCR8 bits are cleared to 0, the pin states are read.

Upon reset, PDR8 is initialized to H'00.

2. Port Control Register 8 (PCR8)

Bit	7	6	5	4	3	2	1
	$PCR8_7$	$PCR8_6$	$PCR8_5$	$PCR8_4$	$PCR8_3$	$PCR8_2$	$PCR8_1$
Initial value	0	0	0	0	0	0	0
Read/Write	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 $P8_0$ fun input or output pin. Setting a PCR8 bit to 1 makes the corresponding pin an output pi clearing the bit to 0 makes the pin an input pin. PCR8 and PDR8 settings are valid wh corresponding pins are designated for general-purpose input/output by bits SGS3 to S LPCR.

Upon reset, PCR8 is initialized to H'00.

PCR8 is a write-only register, which is always read as all 1s.

CL_1

in LPCR.

SEGS3 to SEGS0	000*		001*, 01**, 1***	
SGX	0		0	
$PCR8_7$	0	1	*	
Pin function	$P8_7$ input pin	$P8_7$ output pin	SEG_{32} output pin	CL

$P8_6/SEG_{31}/CL_2$ The pin function depends on bit $PCR8_6$ in PCR8 and bits SGX and SGS3 in LPCR.

SEGS3 to SEGS0	000*		001*, 01**, 1***	
SGX	0		0	
$PCR8_6$	0	1	*	
Pin function	$P8_6$ input pin	$P8_6$ output pin	SEG_{31} output pin	CL

$P8_5/SEG_{30}/DO$ The pin function depends on bit $PCR8_5$ in PCR8 and bits SGX and SGS3 in LPCR.

SEGS3 to SEGS0	000*		001*, 01**, 1***	
SGX	0		0	
$PCR8_5$	0	1	*	
Pin function	$P8_5$ input pin	$P8_5$ output pin	SEG_{30} output pin	D0

$P8_4/SEG_{29}/M$ The pin function depends on bit $PCR8_4$ in PCR8 and bits SGX and SGS3 in LPCR.

SEGS3 to SEGS0	000*		001*, 01**, 1***	
SGX	0		0	
$PCR8_4$	0	1	*	
Pin function	$P8_4$ input pin	$P8_4$ output pin	SEG_{29} output pin	M

$P8_3/SEG_{28}$ to $P8_0/SEG_{25}$ The pin function depends on bit $PCR8_n$ in PCR8 and bits SGS3 to SGS0 (n =

SEGS3 to SEGS0	000*		001*, 0
$PCR8_n$	0	1	
Pin function	$P8_n$ input pin	$P8_n$ output pin	SEG_{n+25}

*

$P8_5/SEG_{30}/DO$ $P8_4/SEG_{29}/M$ $P8_3/SEG_{28}$ to $P8_0/SEG_{25}$	state	state		state

8.9 Port A

8.9.1 Overview

Port A is a 4-bit I/O port, configured as shown in figure 8.8.

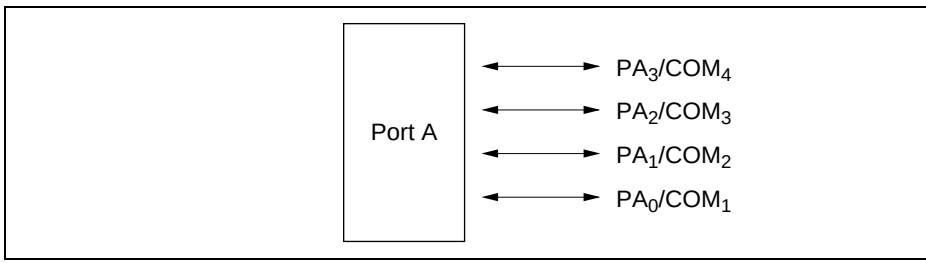


Figure 8.8 Port A Pin Configuration

Port control register A	PCRA	W	H'F0

1. Port Data Register A (PDRA)

Bit	7	6	5	4	3	2	1
	—	—	—	—	PA_3	PA_2	PA_1
Initial value	1	1	1	1	0	0	0
Read/Write	—	—	—	—	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 w
PCRA bits are set to 1, the values stored in PDRA are read, regardless of the actual pin
port A is read while PCRA bits are cleared to 0, the pin states are read.

Upon reset, PDRA is initialized to H'F0.

2. Port Control Register A (PCRA)

Bit	7	6	5	4	3	2	1
	—	—	—	—	$PCRA_3$	$PCRA_2$	$PCRA_1$
Initial value	1	1	1	1	0	0	0
Read/Write	—	—	—	—	R/W	R/W	R/W

PCRA controls whether each of port A pins PA_3 to PA_0 functions as an input pin or out
Setting a PCRA bit to 1 makes the corresponding pin an output pin, while clearing the b
makes the pin an input pin. PCRA and PDRA settings are valid when the correspondin
designated for general-purpose input/output by LPCR.

Upon reset, PCRA is initialized to H'F0.

PCRA is a write-only register, which always reads all 1s.

SEGS3 to SEGS0	0000	0000	N
$PCRA_3$	0	1	
Pin function	PA_3 input pin	PA_3 output pin	COM

PA_2/COM_3 The pin function depends on bit $PCRA_2$ in PCRA and bits SGS3 to S

SEGS3 to SEGS0	0000	0000	N
$PCRA_2$	0	1	
Pin function	PA_2 input pin	PA_2 output pin	COM

PA_1/COM_2 The pin function depends on bit $PCRA_1$ in PCRA and bits SGS3 to S

SEGS3 to SEGS0	0000	0000	N
$PCRA_1$	0	1	
Pin function	PA_1 input pin	PA_1 output pin	COM

PA_0/COM_1 The pin function depends on bit $PCRA_0$ in PCRA and bits SGS3 to S

SEGS3 to SEGS0	0000		N
$PCRA_0$	0	1	
Pin function	PA_0 input pin	PA_0 output pin	COM

PA_2/COM_3					
PA_1/COM_2	Impedance	previous	previous	Impedance	previous
PA_0/COM_1		state	state		state

8.10 Port B

8.10.1 Overview

Port B is an 8-bit input-only port, configured as shown in figure 8.9.

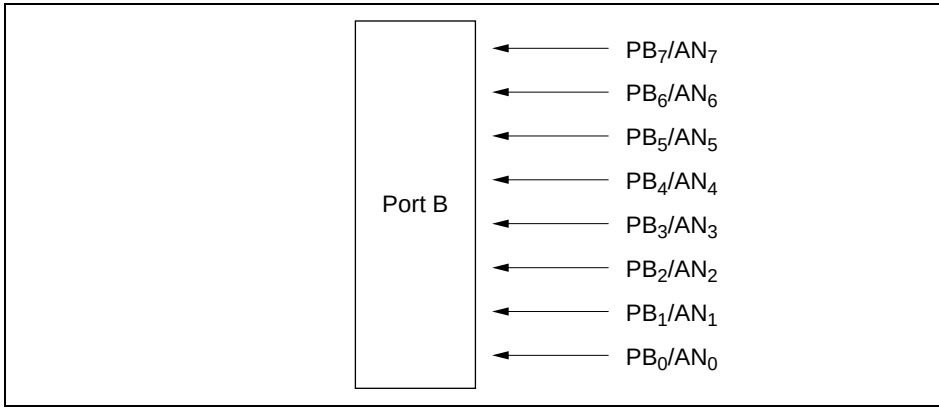


Figure 8.9 Port B Pin Configuration

Port Data Register B (PDRB)

Bit	7	6	5	4	3	2	1
	PB_7	PB_6	PB_5	PB_4	PB_3	PB_2	PB_1
Read/Write	R	R	R	R	R	R	R

Reading PDRB always gives the pin states. However, if a port B pin is selected as an
channel for the A/D converter by AMR bits CH3 to CH0, that pin reads 0 regardless o
voltage.

8.11 Input/Output Data Inversion Function

8.11.1 Overview

With input pins $\overline{WKP_0}$ to $\overline{WKP_7}$, RXD_{31}, and RXD_{32}, and output pins TXD_{31} and TXD
can be handled in inverted form.

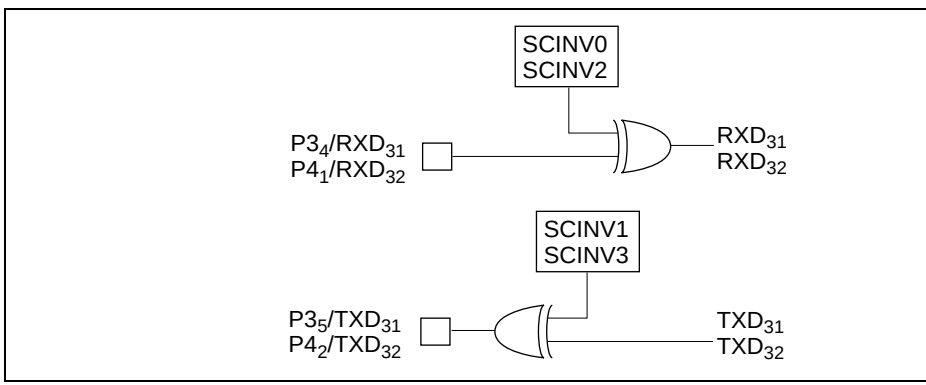


Figure 8.10 Input/Output Data Inversion Function

Serial Port Control Register (SPCR)

Bit	7	6	5	4	3	2	1
	—	—	SPC32	SPC31	SCINV3	SCINV2	SCINV1
Initial value	1	1	0	0	0	0	0
Read/Write	—	—	R/W	R/W	R/W	R/W	R/W

SPCR is an 8-bit readable/writable register that performs RXD_{31}, RXD_{32}, TXD_{31}, and T input/output data inversion switching. SPCR is initialized to H'C0 by a reset.

Bits 7 and 6: Reserved bits

Bits 7 and 6 are reserved; they are always read as 1 and cannot be modified.

Bit 5: $P4_2/TXD_{32}$ pin function switch (SPC32)

This bit selects whether pin $P4_2/TXD_{32}$ is used as $P4_2$ or as TXD_{32}.

Bit 5 SPC32	Description	
0	Functions as $P4_2$ I/O pin	(i
1	Functions as TXD_{32} output pin*	

Note: * Set the TE bit in SCR3 after setting this bit to 1.

Note: Set the TE bit in SCR3 after setting this bit to 1.

Bit 3: TXD_{32} pin output data inversion switch

Bit 3 specifies whether or not TXD_{32} pin output data is to be inverted.

Bit 3 SCINV3	Description
0	TXD_{32} output data is not inverted
1	TXD_{32} output data is inverted

Bit 2: RXD_{32} pin input data inversion switch

Bit 2 specifies whether or not RXD_{32} pin input data is to be inverted.

Bit 2 SCINV2	Description
0	RXD_{32} input data is not inverted
1	RXD_{32} input data is inverted

Bit 1: TXD_{31} pin output data inversion switch

Bit 1 specifies whether or not TXD_{31} pin output data is to be inverted.

Bit 1 SCINV1	Description
0	TXD_{31} output data is not inverted
1	TXD_{31} output data is inverted

8.11.3 Note on Modification of Serial Port Control Register

When a serial port control register is modified, the data being input or output up to that
inverted immediately after the modification, and an invalid data change is input or outp
modifying a serial port control register, do so in a state in which data changes are invali

8.12 Application Note

8.12.1 The Management of the Un-Use Terminal

If an I/O pin not used by the user system is floating, pull it up or down.

- If an unused pin is an input pin, handle it in one of the following ways:
  - Pull it up to V_{CC} with an on-chip pull-up MOS.
  - Pull it up to V_{CC} with an external resistor of approximately 100 kΩ.
  - Pull it down to V_{SS} with an external resistor of approximately 100 kΩ.
  - For a pin also used by the A/D converter, pull it up to AV_{CC}.
- If an unused pin is an output pin, handle it in one of the following ways:
  - Set the output of the unused pin to high and pull it up to V_{CC} with an external re approximately 100 kΩ.
  - Set the output of the unused pin to low and pull it down to V_{SS} with an external approximately 100 kΩ.

Table 9.1 Timer Functions

Name	Functions	Internal Clock	Event Input Pin	Waveform Output Pin	Rem
Timer A	• 8-bit interval timer	ϕ/8 to ϕ/8192 (8 choices)	—	—	
	• Interval function				
	• Time base	ϕ_W/128 (choice of 4 overflow periods)			
	• Clock output	ϕ/4 to ϕ/32 ϕ_W, ϕ_W/4 to ϕ_W/32 (9 choices)	—	TMOW	
Timer C	• 8-bit timer • Interval function • Event counting function • Up-count/down-count selectable	ϕ/4 to ϕ/8192, ϕ_W/4 (7 choices)	TMIC	—	Up-c cou cont soft hard
Timer F	16-bit timer Event counting function Also usable as two independent8-bit timers Output compare output function	ϕ/4 to ϕ/32, ϕ_W/4 (4 choices)	TMIF	TMOFL TMOFH	
Timer G	• 8-bit timer • Input capture function • Interval function	ϕ/2 to ϕ/64, ϕ_W/4 (4 choices)	TMIG	—	• Co cle • Bu inp no

• Counts events asynchronous to ϕ and ϕw

9.2 Timer A

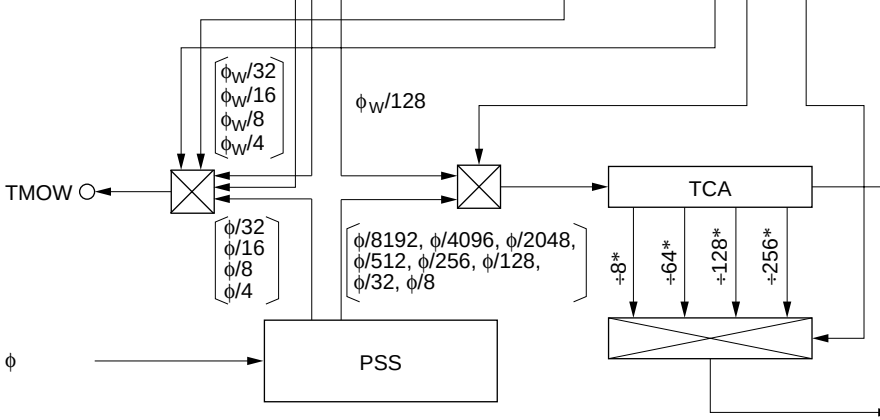
9.2.1 Overview

Timer A is an 8-bit timer with interval timing and real-time clock time-base functions. time-base function is available when a 32.768-kHz crystal oscillator is connected. A cl divided from 32.768 kHz, from 38.4 kHz (if a 38.4 kHz crystal oscillator is connected), the system clock, can be output at the TMOW pin.

1. Features

Features of timer A are given below.

- Choice of eight internal clock sources (ϕ/8192, ϕ/4096, ϕ/2048, ϕ/512, ϕ/256, ϕ/128 ϕ/8).
- Choice of four overflow periods (1 s, 0.5 s, 0.25 s, 31.25 ms) when timer A is used time base (using a 32.768 kHz crystal oscillator).
- An interrupt is requested when the counter overflows.
- Any of nine clock signals can be output at the TMOW pin: 32.768 kHz divided by 3 4 (1 kHz, 2 kHz, 4 kHz, 8 kHz, 32,768 kHz) or 38.4 kHz divided by 32, 16, 8, or 4 (2.4 kHz, 4.8 kHz, 9.6 kHz, 38.4 kHz), and the system clock divided by 32, 16, 8, or
- Use of module standby mode enables this module to be placed in standby mode ind when not used.



Legend:

TMA:	Timer mode register A
TCA:	Timer counter A
IRRTA:	Timer A overflow interrupt request flag
PSW:	Prescaler W
PSS:	Prescaler S
CWOSR:	Subclock output select register

Note: * Can be selected only when the prescaler W output (ϕ_W/128) is used as the TCA input

Figure 9.1 Block Diagram of Timer A

3. Pin Configuration

Table 9.2 shows the timer A pin configuration.

Table 9.2 Pin Configuration

Name	Abbr.	I/O	Function
Clock output	TMOW	Output	Output of waveform generated by timer A ou

Clock stop register 1	CKSTPR1	R/W	H'FF
Subclock output select register	CWOSR	R/W	H'FE

9.2.2 Register Descriptions

1. Timer Mode Register A (TMA)

Bit	7	6	5	4	3	2	1
	TMA7	TMA6	TMA5	—	TMA3	TMA2	TMA1
Initial value	0	0	0	1	0	0	0
Read/Write	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 c

Upon reset, TMA is initialized to H'10.

CWOS	TMA7	TMA6	TMA5	Clock Output
0	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	$\phi_w/4$
1	*	*	*	ϕ_w

Bit 4: Reserved bit

Bit 4 is reserved; it is always read as 1, and cannot be modified.

			1	PSS, ϕ/4096	
		1	0	PSS, ϕ/2048	
			1	PSS, ϕ/512	
	1	0	0	PSS, ϕ/256	
			1	PSS, ϕ/128	
		1	0	PSS, ϕ/32	
			1	PSS, ϕ/8	
1	0	0	0	PSW, 1 s	Cloc (whe 32.7
			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		

source for input to this counter is selected by bits TMA3 to TMA0 in timer mode regi
(TMA). TCA values can be read by the CPU in active mode, but cannot be read in su
mode. When TCA overflows, the IRRTA bit in interrupt request register 1 (IRR1) is s

TCA is cleared by setting bits TMA3 and TMA2 of TMA to 11.

Upon reset, TCA is initialized to H'00.

3. Clock Stop Register 1 (CKSTPR1)

Bit	7	6	5	4	3	2	1
	—	S31CKSTP	S32CKSTP	ADCKSTP	TGCKSTP	TFCKSTP	TCCKSTI
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CKSTPR1 is an 8-bit read/write register that performs module standby mode control f
modules. Only the bit relating to timer A is described here. For details of the other bi
sections on the relevant modules.

Bit 0: Timer A module standby mode control (TACKSTP)

Bit 0 controls setting and clearing of module standby mode for timer A.

TACKSTP	Description
0	Timer A is set to module standby mode
1	Timer A module standby mode is cleared

CWOSR is initialized to H'FE by a reset.

Bits 7 to 1: Reserved bits

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

Bit 0: TMOW pin clock select (CWOS)

Bit 0 selects the clock to be output from the TMOW pin.

Bit 0 CWOS	Description	
0	Clock output from timer A is output (see TMA)	(i
1	ϕ_w is output	

timing resume immediately. The clock input to timer A is selected by bits TMA2 to T
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
overflow, setting bit IRRTA to 1 in interrupt request register 1 (IRR1). If IENTA = 1
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
an interval timer that generates an overflow output at intervals of 256 input clock puls

Note: * For details on interrupts, see section 3.3, Interrupts.

2. 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 t
clock signals output by prescaler W. The overflow period of timer A is set by bits TM
TMA0 in TMA. A choice of four periods is available. In time base operation (TMA3
bit TMA2 to 1 clears both TCA and prescaler W to their initial values of H'00.

3. Clock Output

Setting bit TMOW in port mode register 1 (PMR1) to 1 causes a clock signal to be ou
TMOW. Nine different clock output signals can be selected by means of bits TMA7 t
TMA and bit CWOS in CWOSR. The system clock divided by 32, 16, 8, or 4 can be
active mode and sleep mode. A 32.768 kHz or 38.4 kHz signal divided by 32, 16, 8, o
output in active mode, sleep mode, watch mode, subactive mode, and subsleep mode.
kHz or 38.4 kHz clock is output in all modes except the reset state.

TSA	Interval	Reset	Functions	Functions	Halted	Halted	Halted	Halted
	Clock time base	Reset	Functions	Functions	Functions	Functions	Functions	Halted
TMA	CWOSR	Reset	Functions	Retained	Retained	Functions	Retained	Retained

Note: When the real-time clock time base function is selected as the internal clock of T
active mode or sleep mode, the internal clock is not synchronous with the systen
it is synchronized by a synchronizing circuit. This may result in a maximum error
the count cycle.

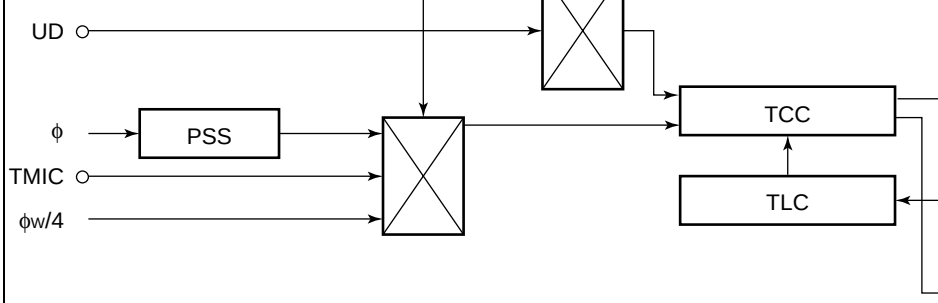
9.2.5 Application Note

When bit 0 (TACKSTP) of the clock stop register 1 (CKSTPR1) is cleared to 0, bit 3 (T
the timer mode register A (TMA) cannot be rewritten.

Set bit 0 (TACKSTP) of the clock stop register 1 (CKSTPR1) to 1 before rewriting bit
of the timer mode register A (TMA).

Features of timer C are given below.

- Choice of seven internal clock sources (ϕ/8192, ϕ/2048, ϕ/512, ϕ/64, ϕ/16, ϕ/4, ϕw external clock (can be used to count external events).
- An interrupt is requested when the counter overflows.
- Up/down-counter switching is possible by hardware or software.
- Subactive mode and subsleep mode operation is possible when ϕw/4 is selected as clock, or when an external clock is selected.
- Use of module standby mode enables this module to be placed in standby mode in when not used.



Legend:
TMC: Timer mode register C
TCC: Timer counter C
TLC: Timer load register C
IRRTC: Timer C overflow interrupt request flag
PSS: Prescaler S

Figure 9.2 Block Diagram of Timer C

3. Pin Configuration

Table 9.5 shows the timer C pin configuration.

Table 9.5 Pin Configuration

Name	Abbr.	I/O	Function
Timer C event input	TMIC	Input	Input pin for event in
Timer C up/down-count selection	UD	Input	Timer C up/down sel

Timer load register C	TLC	W	H'00
Clock stop register 1	CKSTPR1	R/W	H'FF

9.3.2 Register Descriptions

1. Timer Mode Register C (TMC)

Bit	7	6	5	4	3	2	1
	TMC7	TMC6	TMC5	—	—	TMC2	TMC1
Initial value	0	0	0	1	1	0	0
Read/Write	R/W	R/W	R/W	—	—	R/W	R/W

TMC is an 8-bit read/write register for selecting the auto-reload function and input clo performing up/down-counter control.

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
1	Auto-reload function selected

1	*	Hardware control by UD pin input UD pin input high: Down-counter UD pin input low: Up-counter

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 event counting, either the rising edge can be selected.

Bit 2 TMC2	Bit 1 TMC1	Bit 0 TMC0	Description
0	0	0	Internal clock: ϕ/8192
0	0	1	Internal clock: ϕ/2048
0	1	0	Internal clock: ϕ/512
0	1	1	Internal clock: ϕ/64
1	0	0	Internal clock: ϕ/16
1	0	1	Internal clock: ϕ/4
1	1	0	Internal clock: ϕw/4
1	1	1	External event (TMIC): rising or falling edge*

Note: * The edge of the external event signal is selected by bit IEG1 in the IRQ edge register (IEGR). See 1. IRQ edge select register (IEGR) in 3.3.2 for details. be set to 1 in port mode register 1 (PMR1) before setting 111 in bits TMC2 to

input. The clock source for input to this counter is selected by bits TMC2 to TMC0 in 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 fro H'FF or to the value set in TLC, the IRRTC bit in IRR2 is set to 1.

TCC is allocated to the same address as TLC.

Upon reset, TCC is initialized to H'00.

3. Timer Load Register C (TLC)

Bit	7	6	5	4	3	2	1
	TLC7	TLC6	TLC5	TLC4	TLC3	TLC2	TLC1
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

TLC is an 8-bit write-only register for setting the reload value of timer counter C (TC

When a reload value is set in TLC, the same value is loaded into timer counter C as w starts counting up from that value. When TCC overflows or underflows during operat reload mode, the TLC value is loaded into TCC. Accordingly, overflow/underflow pe 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.

modules. Only the bit relating to timer C is described here. For details of the other bits sections on the relevant modules.

Bit 1: Timer C module standby mode control (TCCKSTP)

Bit 1 controls setting and clearing of module standby mode for timer C.

TCCKSTP	Description	
0	Timer C is set to module standby mode	
1	Timer C module standby mode is cleared	(i

9.3.3 Timer Operation

1. Interval Timer Operation

When bit TMC7 in timer mode register C (TMC) is cleared to 0, timer C functions as a interval timer.

Upon reset, TCC is initialized to H'00 and TMC to H'18, so TCC continues up-counting interval up-counter without halting immediately after a reset. The timer C operating clo selected from seven internal clock signals output by prescalers S and W, or an external at pin TMIC. The selection is made by bits TMC2 to TMC0 in TMC.

TCC up/down-count control can be performed either by software or hardware. The sele made by bits TMC6 and TMC5 in TMC.

After the count value in TCC reaches H'FF (H'00), the next clock input causes timer C (underflow), setting bit IRRTC to 1 in IRR2. If IENTC = 1 in interrupt enable register a CPU interrupt is requested.

At overflow (underflow), TCC returns to H'00 (H'FF) and starts counting up (down) ag

TCC starts its count.

After the count value in TCC reaches H'FF (H'00), the next clock signal input causes t
overflow/underflow. The TLC value is then loaded into TCC, and the count continues
value. The overflow/underflow period can be set within a range from 1 to 256 input c
depending on the TLC value.

The clock sources, up/down control, and interrupts in auto-reload mode are the same a
mode.

In auto-reload mode (TMC7 = 1), when a new value is set in TLC, the TLC value is al
TCC.

3. Event Counter Operation

Timer C can operate as an event counter, counting rising or falling edges of an externa
signal input at pin TMIC. External event counting is selected by setting bits TMC2 to
timer mode register C to all 1s (111).

When timer C is used to count external event input, bit IRQ1 in PMR1 should be set t
IEN1 in IENR1 cleared to 0 to disable interrupt IRQ_1 requests.

4. TCC Up/Down Control by Hardware

With timer C, TCC up/down control can be performed by UD pin input. When bit TM
1 in TMC, TCC functions as an up-counter when UD pin input is high, and as a down-
when low.

When using UD pin input, set bit UD to 1 in PMR3.

TCC	Interval	Reset	Functions	Functions	Halted	Functions/ Halted*	Functions/ Halted*	Halted
	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 TCC internal clock in active mode or sleep mo
the system clock and internal clock are mutually asynchronous, synchroniza
maintained by a synchronization circuit. This results in a maximum count cy
$1/\phi$ (s). When the counter is operated in subactive mode or subsleep mode,
select $\phi w/4$ as the internal clock or select an external clock. The counter wil
operate on any other internal clock. If $\phi w/4$ is selected as the internal clock f
counter when $\phi w/8$ has been selected as subclock ϕ_{SUB}, the lower 2 bits of th
operate on the same cycle, and the operation of the least significant bit is un
the operation of the counter.

- An external event (TMIC) is used in subsleep mode.

Symptom

- The counter increments or decrements twice for a single external event input.

Approximate rate of occurrence

The approximate rate of occurrence in cases where the external event input is r synchronized with internal operation is defined by the following equation.

Approximate rate of occurrence P = 30 ns / tsubcyc

For example, if tsubcyc = 61.06 µs (subclock ϕw/2), P = 0.0005 (0.05%). If 2,0 event inputs occur, there is a likelihood that one of them will cause the counter increment or decrement twice (+2 or –2).

The symptom described is caused by the internal circuit configuration of the devic therefore difficult to avoid. Therefore, it is not advisable to use the clock counter f applications requiring a high degree of accuracy.

and timer FL).

1. Features

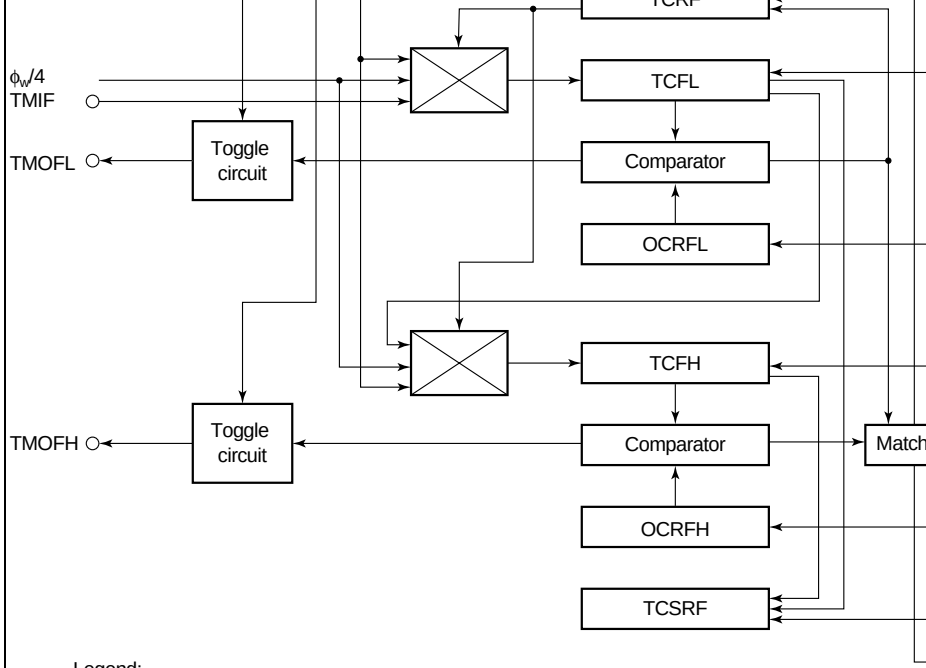
Features of timer F are given below.

- Choice of four internal clock sources (φ/32, φ/16, φ/4, φw/4) or an external clock (ca as an external event counter)
- TMOFH pin (TMOFL pin) toggle output provided using a single compare match sig output initial value can be set)
- Counter resetting by a compare match signal
- Two interrupt sources: one compare match, one overflow
- Can operate as two independent 8-bit timers (timer FH and timer FL) (in 8-bit mode

	Timer FH 8-Bit Timer*	**Timer FL 8-Bit Timer/Event Counte**
Internal clock	Choice of 4 (φ/32, φ/16, φ/4, φw/4)	
Event input	—	TMIF pin
Toggle output	One compare match signal, output to TMOFH pin(initial value settable)	One compare match signal TMOFL pin (initial value set
Counter reset	Counter can be reset by compare match signal	
Interrupt sources	One compare match One overflow	

Note: * When timer F operates as a 16-bit timer, it operates on the timer FL overflow

- Operation in watch mode, subactive mode, and subsleep mode
  When φw/4 is selected as the internal clock, timer F can operate in watch mode, sub mode, and subsleep mode.
- Use of module standby mode enables this module to be placed in standby mode ind when not used.



Legend:
TCRF: Timer control register F
TCSRF: Timer control/status register F
TCFH: 8-bit timer counter FH
TCFL: 8-bit timer counter FL
OCRFH: Output compare register FH
OCRFL: Output compare register FL
IRRTFH: Timer FH interrupt request flag
IRRTFL: Timer FL interrupt request flag
PSS: Prescaler S

Figure 9.3 Block Diagram of Timer F

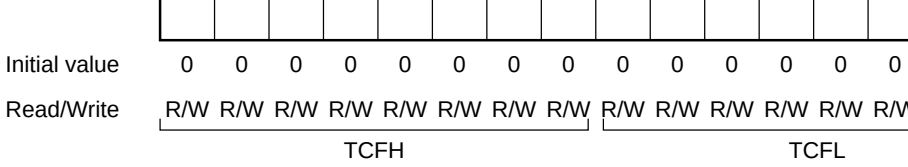
Timer FL output	TMOFL	Output	Timer FL toggle output p

4. Register Configuration

Table 9.9 shows the register configuration of timer F.

Table 9.9 Timer F Registers

Name	Abbr.	R/W	Initial Value
Timer control register F	TCRF	W	H'00
Timer control/status register F	TCSRF	R/W	H'00
8-bit timer counter FH	TCFH	R/W	H'00
8-bit timer counter FL	TCFL	R/W	H'00
Output compare register FH	OCRFH	R/W	H'FF
Output compare register FL	OCRFL	R/W	H'FF
Clock stop register 1	CKSTPR1	R/W	H'FF



TCF is a 16-bit read/write up-counter configured by cascaded connection of 8-bit time
TCFH and TCFL. In addition to the use of TCF as a 16-bit counter with TCFH as the
and TCFL as the lower 8 bits, TCFH and TCFL can also be used as independent 8-bit

TCFH and TCFL can be read and written by the CPU, but when they are used in 16-b
transfer to and from the CPU is performed via a temporary register (TEMP). For deta
see section 9.4.3, CPU Interface.

TCFH and TCFL are each initialized to H'00 upon reset.

a. 16-bit mode (TCF)

   When CKSH2 is cleared to 0 in TCRF, TCF operates as a 16-bit counter. The TC
   is selected by bits CKSL2 to CKSL0 in TCRF.

   TCF can be cleared in the event of a compare match by means of CCLRH in TCSI

   When TCF overflows from H'FFFF to H'0000, OVFH is set to 1 in TCSRF. If OV
   TCSRF is 1 at this time, IRRTFH is set to 1 in IRR2, and if IENTFH in IENR2 is
   interrupt request is sent to the CPU.

b. 8-bit mode (TCFL/TCFH)

   When CKSH2 is set to 1 in TCRF, TCFH and TCFL operate as two independent 8
   counters. The TCFH (TCFL) input clock is selected by bits CKSH2 to CKSH0 (C
   CKSL0) in TCRF.

   TCFH (TCFL) can be cleared in the event of a compare match by means of CCLR
   in TCSRF.

   When TCFH (TCFL) overflows from H'FF to H'00, OVFH (OVFL) is set to 1 in T
   OVIEH (OVIEL) in TCSRF is 1 at this time, IRRTFH (IRRTFL) is set to 1 in IRR
   IENTFH (IENTFL) in IENR2 is 1, an interrupt request is sent to the CPU.

Initial value	1	1	1	1	1	1	1	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	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	OCRFH								OCRFL						

OCRF is a 16-bit read/write register composed of the two registers OCRFH and OCRF addition to the use of OCRF as a 16-bit register with OCRFH as the upper 8 bits and O the lower 8 bits, OCRFH and OCRFL can also be used as independent 8-bit registers.

OCRFH and OCRFL can be read and written by the CPU, but when they are used in 16 data transfer to and from the CPU is performed via a temporary register (TEMP). For d TEMP, see section 9.4.3, CPU Interface.

OCRFH and OCRFL are each initialized to H'FF upon reset.

a. 16-bit mode (OCRF)

When CKSH2 is cleared to 0 in TCRF, OCRF operates as a 16-bit register. OCRF are constantly compared with TCF, and when both values match, CMFH is set to 1 At the same time, IRRTFH is set to 1 in IRR2. If IENTFH in IENR2 is 1 at this tim interrupt request is sent to the CPU.

Toggle output can be provided from the TMOFH pin by means of compare matches output level can be set (high or low) by means of TOLH in TCRF.

b. 8-bit mode (OCRFH/OCRFL)

When CKSH2 is set to 1 in TCRF, OCRFH and OCRFL operate as two independen registers. OCRFH contents are compared with TCFH, and OCRFL contents are wit When the OCRFH (OCRFL) and TCFH (TCFL) values match, CMFH (CMFL) is s TCSRF. At the same time, IRRTFH (IRRTFL) is set to 1 in IRR2. If IENTFH (IE IENR2 is 1 at this time, an interrupt request is sent to the CPU.

Toggle output can be provided from the TMOFH pin (TMOFL pin) by means of co matches, and the output level can be set (high or low) by means of TOLH (TOLL) i

TCRF is an 8-bit write-only register that switches between 16-bit mode and 8-bit mod
input clock from among four internal clock sources or external event input, and sets th
level of the TMOFH and TMOFL pins.

TCRF is initialized to H'00 upon reset.

Bit 7: Toggle output level H (TOLH)

Bit 7 sets the TMOFH pin output level. The output level is effective immediately afte
written.

Bit 7 TOLH	Description
0	Low level
1	High level

Bits 6 to 4: Clock select H (CKSH2 to CKSH0)

Bits 6 to 4 select the clock input to TCFH from among four internal clock sources or T
overflow.

Bit 6 CKSH2	Bit 5 CKSH1	Bit 4 CKSH0	Description
0	0	0	16-bit mode, counting on TCFL overflow signal
0	0	1	
0	1	0	
0	1	1	Use prohibited
1	0	0	Internal clock: Counting on $\phi/32$
1	0	1	Internal clock: Counting on $\phi/16$
1	1	0	Internal clock: Counting on $\phi/4$
1	1	1	Internal clock: Counting on $\phi w/4$

Bits 2 to 0: Clock select L (CKSL2 to CKSL0)

Bits 2 to 0 select the clock input to TCFL from among four internal clock sources or ex
event input.

Bit 2 CKSL2	Bit 1 CKSL1	Bit 0 CKSL0	Description
0	0	0	Counting on external event (TMIF) rising/falling edge
0	0	1	(i
0	1	0	
0	1	1	Use prohibited
1	0	0	Internal clock: Counting on ϕ/32
1	0	1	Internal clock: Counting on ϕ/16
1	1	0	Internal clock: Counting on ϕ/4
1	1	1	Internal clock: Counting on ϕw/4

Note: * External event edge selection is set by IEG3 in the IRQ edge select register
For details, see 1. IRQ edge select register (IEGR) in section 3.3.2.

Note that the timer F counter may increment if the setting of IRQ3 in port mo
1 (PMR1) is changed from 0 to 1 while the TMIF pin is low in order to change
pin function.

TCSRF is an 8-bit read/write register that performs counter clear selection, overflow f
and compare match flag setting, and controls enabling of overflow interrupt requests.

TCSRF is initialized to H'00 upon reset.

Bit 7: Timer overflow flag H (OVFH)

Bit 7 is a status flag indicating that TCFH has overflowed from H'FF to H'00. This fla
hardware and cleared by software. It cannot be set by software.

Bit 7 OVFH	Description
0	Clearing condition: After reading OVFH = 1, cleared by writing 0 to OVFH
1	Setting condition: Set when TCFH overflows from H'FF to H'00

Bit 6: Compare match flag H (CMFH)

Bit 6 is a status flag indicating that TCFH has matched OCRFH. This flag is set by ha
cleared by software. It cannot be set by software.

Bit 6 CMFH	Description
0	Clearing condition: After reading CMFH = 1, cleared by writing 0 to CMFH
1	Setting condition: Set when the TCFH value matches the OCRFH value

Bit 4: Counter clear H (CCLRH)

In 16-bit mode, bit 4 selects whether TCF is cleared when TCF and OCRF match.

In 8-bit mode, bit 4 selects whether TCFH is cleared when TCFH and OCRFH match.

Bit 4 CCLRH	Description	
0	16-bit mode: TCF clearing by compare match is disabled	
	8-bit mode: TCFH clearing by compare match is disabled	(i
1	16-bit mode: TCF clearing by compare match is enabled	
	8-bit mode: TCFH clearing by compare match is enabled	

Bit 3: Timer overflow flag L (OVFL)

Bit 3 is a status flag indicating that TCFL has overflowed from H'FF to H'00. This flag hardware and cleared by software. It cannot be set by software.

Bit 3 OVFL	Description	
0	Clearing condition:	(i
	After reading OVFL = 1, cleared by writing 0 to OVFL	
1	Setting condition:	
	Set when TCFL overflows from H’FF to H’00	

1	Setting condition: Set when the TCFL value matches the OCRFL value

Bit 1: Timer overflow interrupt enable L (OVIEL)

Bit 1 selects enabling or disabling of interrupt generation when TCFL overflows.

Bit 1 OVIEL	Description
0	TCFL overflow interrupt request is disabled
1	TCFL overflow interrupt request is enabled

Bit 0: Counter clear L (CCLRL)

Bit 0 selects whether TCFL is cleared when TCFL and OCRFL match.

Bit 0 CCLRL	Description
0	TCFL clearing by compare match is disabled
1	TCFL clearing by compare match is enabled

modules. Only the bit relating to timer F is described here. For details of the other bits sections on the relevant modules.

Bit 2: Timer F module standby mode control (TFCKSTP)

Bit 2 controls setting and clearing of module standby mode for timer F.

TFCKSTP	Description
0	Timer F is set to module standby mode
1	Timer F module standby mode is cleared (i

9.4.3 CPU Interface

TCF and OCRF are 16-bit read/write registers, but the CPU is connected to the on-chip modules by an 8-bit data bus. When the CPU accesses these registers, it therefore uses temporary register (TEMP).

In 16-bit mode, TCF read/write access and OCRF write access must be performed 16 b (using two consecutive byte-size MOV instructions), and the upper byte must be access the lower byte. Data will not be transferred correctly if only the upper byte or only the is accessed.

In 8-bit mode, there are no restrictions on the order of access.

1. Write Access

Write access to the upper byte results in transfer of the upper-byte write data to TEMP. write access to the lower byte results in transfer of the data in TEMP to the upper regist and direct transfer of the lower-byte write data to the lower register byte.

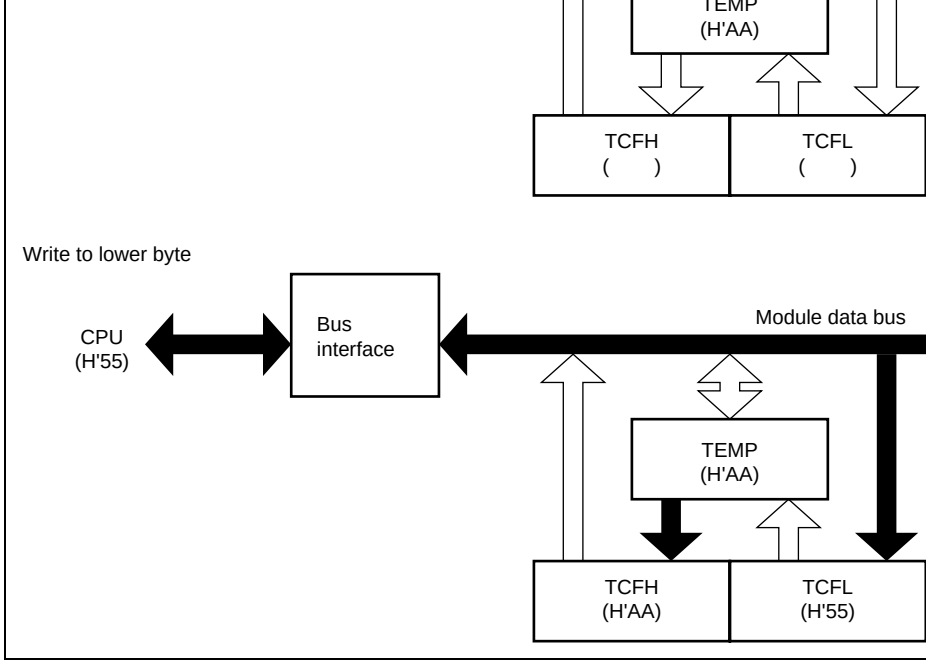


Figure 9.4 Write Access to TCR (CPU → TCF)

Figure 9.5 shows an example in which TCF is read when it contains H'AAFF.

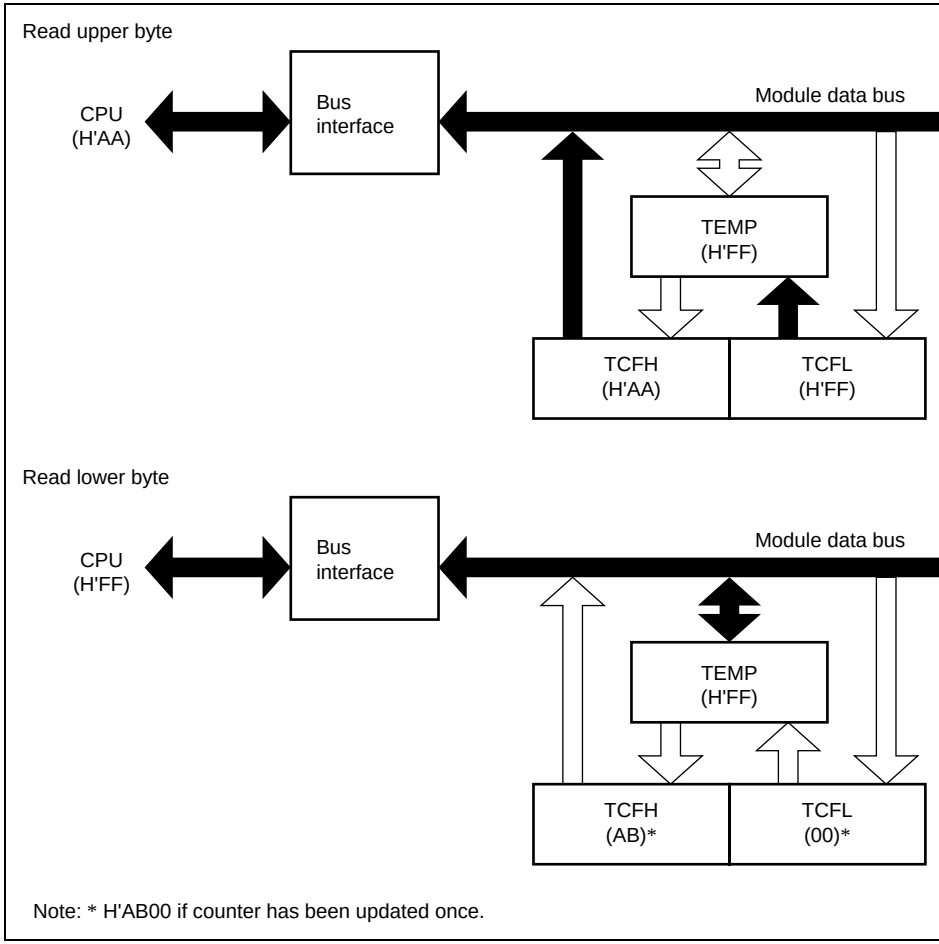


Figure 9.5 Read Access to TCF (TCF → CPU)

Timer F has two operating modes, 16-bit timer mode and 8-bit timer mode. The opera
of these modes is described below.

a. Operation in 16-bit timer mode

When CKSH2 is cleared to 0 in timer control register F (TCRF), timer F operates
timer.

Following a reset, timer counter F (TCF) is initialized to H'0000, output compare r
(OCRF) to H'FFFF, and timer control register F (TCRF) and timer control/status r
(TCSRF) to H'00. The counter starts incrementing on external event (TMIF) input
external event edge selection is set by IEG3 in the IRQ edge select register (IEGR

The timer F operating clock can be selected from four internal clocks or an externa
means of bits CKSL2 to CKSL0 in TCRF.

OCRF contents are constantly compared with TCF, and when both values match,
to 1 in TCSRF. If IENTFH in IENR2 is 1 at this time, an interrupt request is sent
and at the same time, TMOFH pin output is toggled. If CCLRH in TCSRF is 1, T
cleared. TMOFH pin output can also be set by TOLH in TCRF.

When TCF overflows from H'FFFF to H'0000, OVFH is set to 1 in TCSRF. If OV
TCSRF and IENTFH in IENR2 are both 1, an interrupt request is sent to the CPU.

b. Operation in 8-bit timer mode

When CKSH2 is set to 1 in TCRF, TCF operates as two independent 8-bit timers,
TCFL. The TCFH/TCFL input clock is selected by CKSH2 to CKSH0/CKSL2 to
TCRF.

When the OCRFH/OCRFL and TCFH/TCFL values match, CMFH/CMFL is set t
TCSRF. If IENTFH/IENTFL in IENR2 is 1, an interrupt request is sent to the CP
same time, TMOFH pin/TMOFL pin output is toggled. If CCLRH/CCLRL in TC
TCFH/TCFL is cleared. TMOFH pin/TMOFL pin output can also be set by TOL
TCRF.

When TCFH/TCFL overflows from H'FF to H'00, OVFH/OVFL is set to 1 in TCS
OVIEH/OVIEL in TCSRF and IENTFH/IENTFL in IENR2 are both 1, an interrup
sent to the CPU.

External event input is selected by clearing CKSL2 to 0 in TCRF. TCF can increme either the rising or falling edge of external event input. External event edge selectio IEG3 in the interrupt controller's IEGR register. An external event pulse width of a system clocks (ϕ) is necessary. Shorter pulses will not be counted correctly.

3. TMOFH/TMOFL Output Timing

In TMOFH/TMOFL output, the value set in TOLH/TOLL in TCRF is output. The outp toggled by the occurrence of a compare match. Figure 9.6 shows the output timing.

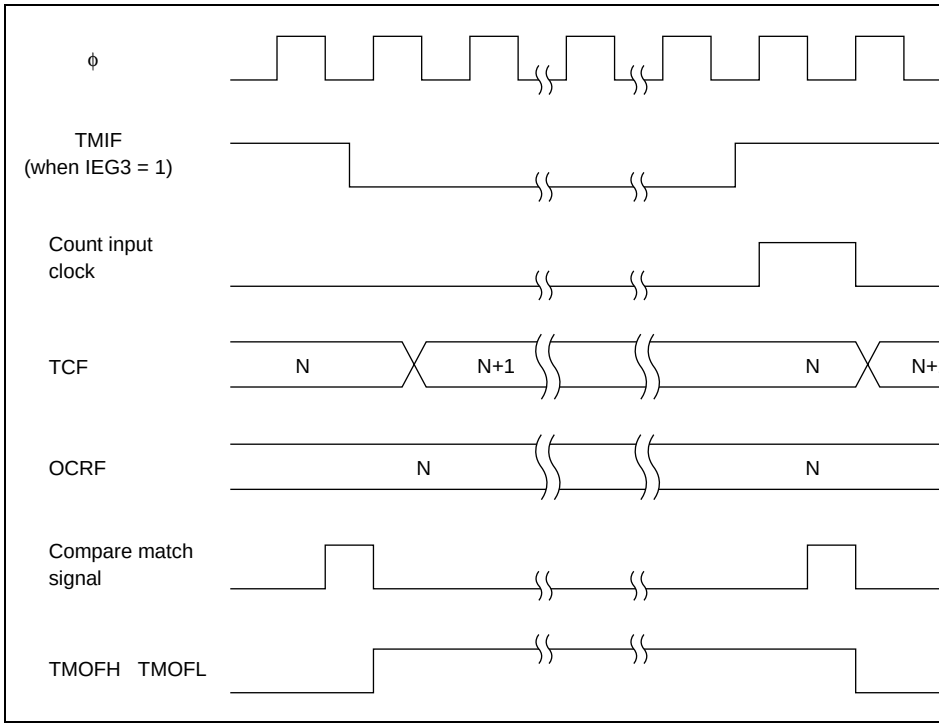


Figure 9.6 TMOFH/TMOFL Output Timing

The compare match flag (CMFH or CMFL) is set to 1 when the TCF and OCRF value
The compare match signal is generated in the last state during which the values match
is updated from the matching value to a new value). When TCF matches OCRF, the c
match signal is not generated until the next counter clock.

7. Timer F Operation Modes

Timer F operation modes are shown in table 9.10.

Table 9.10 Timer F Operation Modes

Operation Mode	Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Stand
TCF	Reset	Functions	Functions	Functions/ Halted*	Functions/ Halted*	Functions/ Halted*	Halted
OCRF	Reset	Functions	Held	Held	Functions	Held	Held
TCRF	Reset	Functions	Held	Held	Functions	Held	Held
TCSRF	Reset	Functions	Held	Held	Functions	Held	Held

Note: * When $\phi_w/4$ is selected as the TCF internal clock in active mode or sleep mo
the system clock and internal clock are mutually asynchronous, synchroniz
maintained by a synchronization circuit. This results in a maximum count c
$1/\phi$ (s). When the counter is operated in subactive mode, watch mode, or s
mode, $\phi_w/4$ must be selected as the internal clock. The counter will not ope
other internal clock is selected.

signal occur simultaneously, TOLH data is output to the TMOFH pin as a result of the write. TMOFL pin output is unstable in 16-bit mode, and should not be used; the TMO should be used as a port pin.

If an OCRFL write and compare match signal generation occur simultaneously, the com match signal is invalid. However, if the written data and the counter value match, a com match signal will be generated at that point. As the compare match signal is output in synchronization with the TCFL clock, a compare match will not result in compare matc generation if the clock is stopped.

Compare match flag CMFH is set when all 16 bits match and a compare match signal is Compare match flag CMFL is set if the setting conditions for the lower 8 bits are satisf

When TCF overflows, OVFH is set. OVFL is set if the setting conditions are satisfied lower 8 bits overflow. If a TCFL write and overflow signal output occur simultaneousl overflow signal is not output.

2. 8-bit Timer Mode

a. TCFH, OCRFH

In toggle output, TMOFH pin output is toggled when a compare match occurs. If a write by a MOV instruction and generation of the compare match signal occur simu TOLH data is output to the TMOFH pin as a result of the TCRF write.

If an OCRFH write and compare match signal generation occur simultaneously, the match signal is invalid. However, if the written data and the counter value match, a match signal will be generated at that point. The compare match signal is output in synchronization with the TCFH clock.

If a TCFH write and overflow signal output occur simultaneously, the overflow sign output.

b. TCFL, OCRFL

In toggle output, TMOFL pin output is toggled when a compare match occurs. If a write by a MOV instruction and generation of the compare match signal occur simu TOLL data is output to the TMOFL pin as a result of the TCRF write.

5. Clear Timer FH, Timer FL Interrupt Request Flags (IRRTFH, IRRTFL), Timer Overflow Flags H, L (OVFH, OVFL) and Compare Match Flags H, L (CMFH, CMFL)

When ϕw/4 is selected as the internal clock, “Interrupt factor generation signal” will b with ϕw and the signal will be outputted with ϕw width. And, “Overflow signal” and “ match signal” are controlled with 2 cycles of ϕw signals. Those signals are outputted v width of ϕw (figure 9.7)

In active (high-speed, medium-speed) mode, even if you cleared interrupt request flag term of validity of “Interrupt factor generation signal”, same interrupt request flag is s 9.7 (1)) And, you cannot be cleared timer overflow flag and compare match flag durin of validity of “Overflow signal” and “Compare match signal”.

For interrupt request flag is set right after interrupt request is cleared, interrupt proces timer FH, timer FL interrupt might be repeated. (figure 9.7 (2)) Therefore, to definitel interrupt request flag in active (high-speed, medium-speed) mode, clear should be pro the time that calculated with below (1) formula. And, to definitely clear timer overflo compare match flag, clear should be processed after read timer control status register l after the time that calculated with below (1) formula. For ST of (1) formula, please su longest number of execution states in used instruction. (10 states of RTE instruction w MULXU, DIVXU instruction is not used, 14 states when MULXU, DIVXU instructio subactive mode, there are not limitation for interrupt request flag, timer overflow flag, compare match flag clear.

The term of validity of “Interrupt factor generation signal”

= 1 cycle of ϕw + waiting time for completion of executing instruction
+ interrupt time synchronized with ϕ = 1/ϕw + ST × (1/ϕ) + (2/ϕ) (second).....(1)

ST: Executing number of execution states

4. Operate interrupt permission (set IENFH, IENFL to 1).

Method 2

1. Set interrupt handling routine time to more than time that calculated with (1) for
2. Clear interrupt request flags (IRRTFH, IRRTFL) at the end of interrupt handling
3. After read timer control status register F (TCSRF), clear timer overflow flags (O OVFL) and compare match flags (CMFH, CMFL).

All above attentions are also applied in 16-bit mode and 8-bit mode.

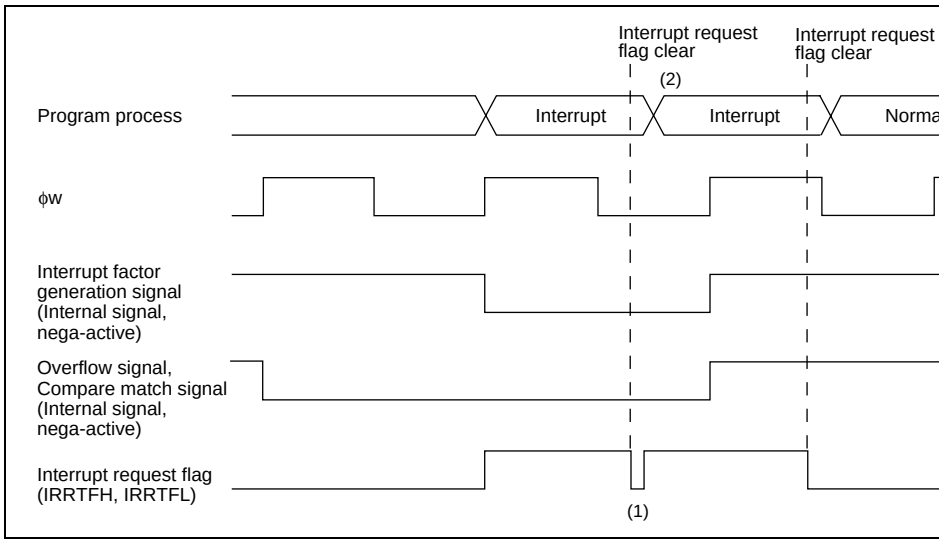


Figure 9.7 Clear Interrupt Request Flag when Interrupt Factor Generation Sign

In subactive mode, even ϕw/4 is selected as the internal clock, normal read/write TCF

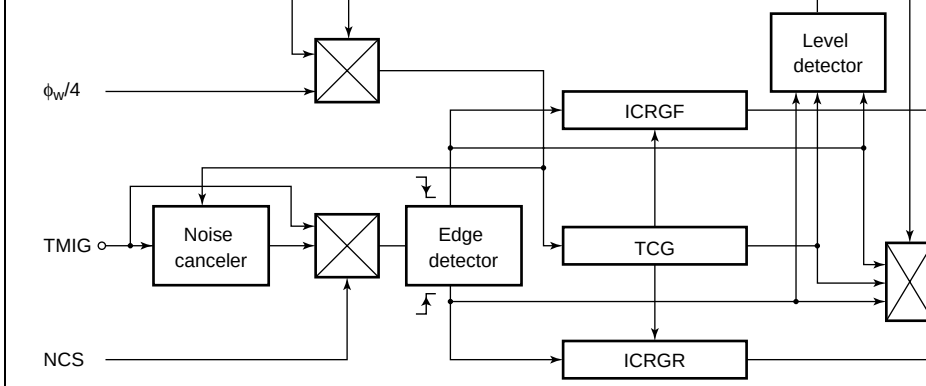
accurate measurement of the input capture input signal duty cycle. If input capture inp
timer G functions as an 8-bit interval timer.

1. Features

Features of timer G are given below.

- Choice of four internal clock sources (ϕ/64, ϕ/32, ϕ/2, ϕw/4)
- Dedicated input capture functions for rising and falling edges
- Level detection at counter overflow

  It is possible to detect whether overflow occurred when the input capture input sign
  or when it was low.
- Selection of whether or not the counter value is to be cleared at the input capture inp
  rising edge, falling edge, or both edges
- Two interrupt sources: one input capture, one overflow. The input capture input sig
  or falling edge can be selected as the interrupt source.
- A built-in noise canceler eliminates high-frequency component noise in the input ca
  signal.
- Watch mode, subactive mode and subsleep mode operation is possible when ϕw/4 i
  as the internal clock.
- Use of module standby mode enables this module to be placed in standby mode ind
  when not used.



Legend:
TMG: Timer mode register G
TCG: Timer counter G
ICRGF: Input capture register GF
ICRGR: Input capture register GR
IRRTG: Timer G interrupt request flag
NCS: Noise canceler select
PSS: Prescaler S

Figure 9.8 Block Diagram of Timer G

4. Register Configuration

Table 9.12 shows the register configuration of timer G.

Table 9.12 Timer G Registers

Name	Abbr.	R/W	Initial Value
Timer control register G	TMG	R/W	H'00
Timer counter G	TCG	—	H'00
Input capture register GF	ICRGF	R	H'00
Input capture register GR	ICRGR	R	H'00
Clock stop register 1	CKSTPR1	R/W	H'FF

Read/Write

TCG is an 8-bit up-counter which is incremented by clock input. The input clock is se
bits CKS1 and CKS0 in TMG.

TMIG in PMR1 is set to 1 to operate TCG as an input capture timer, or cleared to 0 to
TCG as an interval timer*. In input capture timer operation, the TCG value can be cle
rising edge, falling edge, or both edges of the input capture input signal, according to t
made in TMG.

When TCG overflows from H'FF to H'00, if OVIE in TMG is 1, IRRTG is set to 1 in I
IENTG in IENR2 is 1, an interrupt request is sent to the CPU.

For details of the interrupt, see section 3.3, Interrupts.

TCG cannot be read or written by the CPU. It is initialized to H'00 upon reset.

Note: * An input capture signal may be generated when TMIG is modified.

detected, the current TCG value is transferred to ICRGF. If IIEGS in TMG is 1 at this
IRRTG is set to 1 in IRR2, and if IENTG in IENR2 is 1, an interrupt request is sent to t

For details of the interrupt, see section 3.3, Interrupts.

To ensure dependable input capture operation, the pulse width of the input capture inpu
must be at least 2ϕ or 2ϕ_{SUB} (when the noise canceler is not used).

ICRGF is initialized to H'00 upon reset.

3. Input Capture Register GR (ICRGR)

Bit	7	6	5	4	3	2	1
	ICRGR7	ICRGR6	ICRGR5	ICRGR4	ICRGR3	ICRGR2	ICRGR1
Initial value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

ICRGR is an 8-bit read-only register. When a rising edge of the input capture input sig
detected, the current TCG value is transferred to ICRGR. If IIEGS in TMG is 1 at this
IRRTG is set to 1 in IRR2, and if IENTG in IENR2 is 1, an interrupt request is sent to t

For details of the interrupt, see section 3.3, Interrupts.

To ensure dependable input capture operation, the pulse width of the input capture inpu
must be at least 2ϕ or 2ϕ_{SUB} (when the noise canceler is not used).

ICRGR is initialized to H'00 upon reset.

TMG is an 8-bit read/write register that performs TCG clock selection from four inter
sources, counter clear selection, and edge selection for the input capture input signal i
request, controls enabling of overflow interrupt requests, and also contains the overflo

TMG is initialized to H'00 upon reset.

Bit 7: Timer overflow flag H (OVFH)

Bit 7 is a status flag indicating that TCG has overflowed from H'FF to H'00 when the
input signal is high. This flag is set by hardware and cleared by software. It cannot b
software.

Bit 7 OVFH	Description
0	Clearing condition: After reading OVFH = 1, cleared by writing 0 to OVFH
1	Setting condition: Set when TCG overflows from H'FF to H'00

Bit 6: Timer overflow flag L (OVFL)

Bit 6 is a status flag indicating that TCG has overflowed from H'FF to H'00 when the
input signal is low, or in interval operation. This flag is set by hardware and cleared b
It cannot be set by software.

Bit 5: Timer overflow interrupt enable (OVIE)

Bit 5 selects enabling or disabling of interrupt generation when TCG overflows.

Bit 5 OVIE	Description	
0	TCG overflow interrupt request is disabled	(i
1	TCG overflow interrupt request is enabled	

Bit 4: Input capture interrupt edge select (IIEGS)

Bit 4 selects the input capture input signal edge that generates an interrupt request.

Bit 4 IIEGS	Description	
0	Interrupt generated on rising edge of input capture input signal	(i
1	Interrupt generated on falling edge of input capture input signal	

Bits 3 and 2: Counter clear 1 and 0 (CCLR1, CCLR0)

Bits 3 and 2 specify whether or not TCG is cleared by the rising edge, falling edge, or b
of the input capture input signal.

Bit 3 CCLR1	Bit 2 CCLR0	Description	
0	0	TCG clearing is disabled	(i
0	1	TCG cleared by falling edge of input capture input signal	
1	0	TCG cleared by rising edge of input capture input signal	
1	1	TCG cleared by both edges of input capture input signal	

1	0	Internal clock: Counting on ϕ/2
1	1	Internal clock: Counting on ϕw/4

5. Clock Stop Register 1 (CKSTPR1)

Bit	7	6	5	4	3	2	1
	—	S31CKSTP	S32CKSTP	ADCKSTP	TGCKSTP	TFCKSTP	TCCKSTI
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CKSTPR1 is an 8-bit read/write register that performs module standby mode control f
modules. Only the bit relating to timer G is described here. For details of the other bi
sections on the relevant modules.

Bit 3: Timer G module standby mode control (TGCKSTP)

Bit 3 controls setting and clearing of module standby mode for timer G.

TGCKSTP	Description
0	Timer G is set to module standby mode
1	Timer G module standby mode is cleared

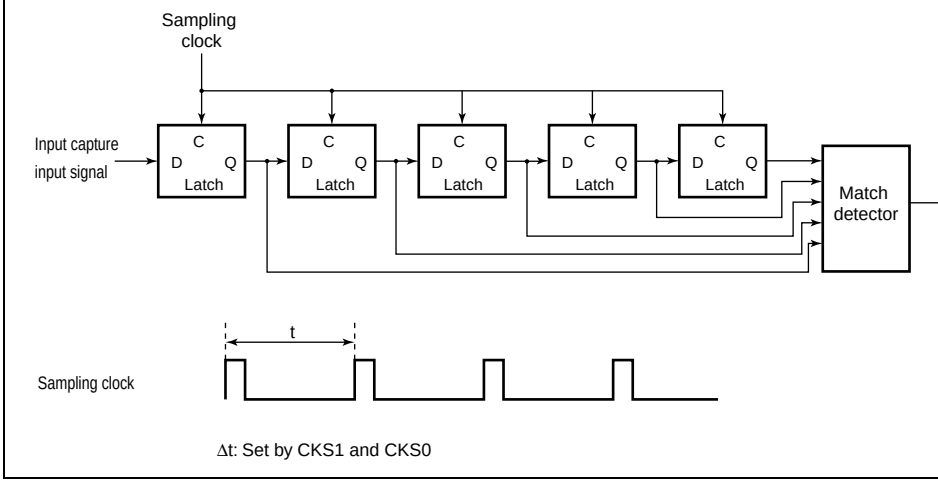


Figure 9.9 Noise Canceler Block Diagram

The noise canceler consists of five latch circuits connected in series and a match detect
When the noise cancellation function is not used (NCS = 0), the system clock is selecte
sampling clock. When the noise cancellation function is used (NCS = 1), the sampling
internal clock selected by CKS1 and CKS0 in TMG, the input capture input is sampled
rising edge of this clock, and the data is judged to be correct when all the latch outputs
all the outputs do not match, the previous value is retained. After a reset, the noise can
is initialized when the falling edge of the input capture input signal has been sampled fi
Therefore, after making a setting for use of the noise cancellation function, a pulse with
five times the width of the sampling clock is a dependable input capture signal. Even i
cancellation is not used, an input capture input signal pulse width of at least 2ϕ or $2\phi_{SUI}$
necessary to ensure that input capture operations are performed properly

Note: * An input capture signal may be generated when the NCS bit is modified.

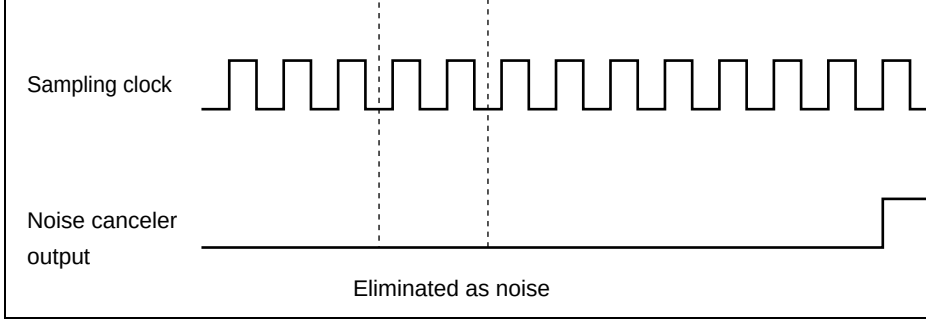


Figure 9.10 Noise Canceler Timing (Example)

The operation of these two functions is described below.

a. Input capture timer operation

When the TMIG bit is set to 1 in port mode register 1 (PMR1), timer G functions as capture timer*.

In a reset, timer mode register G (TMG), timer counter G (TCG), input capture regi (ICRGF), and input capture register GR (ICRGR) are all initialized to H'00.

Following a reset, TCG starts incrementing on the $\phi/64$ internal clock.

The input clock can be selected from four internal clock sources by bits CKS1 and C TMG.

When a rising edge/falling edge is detected in the input capture signal input from th pin, the TCG value at that time is transferred to ICRGR/ICRGF. When the edge sel IIEGS in TMG is input, IRRTG is set to 1 in IRR2, and if the IENTG bit in IENR2 time, an interrupt request is sent to the CPU. For details of the interrupt, see section Interrupts.

TCG can be cleared by a rising edge, falling edge, or both edges of the input capture according to the setting of bits CCLR1 and CCLR0 in TMG. If TCG overflows wh input capture signal is high, the OVFH bit is set in TMG; if TCG overflows when th capture signal is low, the OVFL bit is set in TMG. If the OVIE bit in TMG is 1 wh bits are set, IRRTG is set to 1 in IRR2, and if the IENTG bit in IENR2 is 1, timer G interrupt request to the CPU. For details of the interrupt, see section 3.3, Interrupts.

Timer G has a built-in noise canceler that enables high-frequency component noise eliminated from pulses input from the TMIG pin. For details, see section 9.5.3, Noi Canceler.

Note: * An input capture signal may be generated when TMIG is modified.

2. Increment Timing

TCG is incremented by internal clock input. Bits CKS1 and CKS0 in TMG select one
internal clock sources (ϕ/64, ϕ/32, ϕ/2, or ϕw/4) created by dividing the system clock
clock (ϕw).

3. Input Capture Input Timing

a. Without noise cancellation function

For input capture input, dedicated input capture functions are provided for rising a
edges.

Figure 9.11 shows the timing for rising/falling edge input capture input.

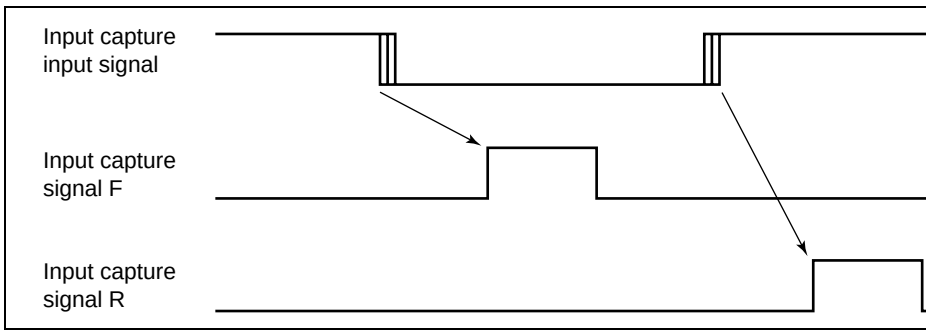


Figure 9.11 Input Capture Input Timing (without Noise Cancellation Fun

b. With noise cancellation function

When noise cancellation is performed on the input capture input, the passage of th
capture signal through the noise canceler results in a delay of five sampling clock
the input capture input signal edge.

Figure 9.12 shows the timing in this case.

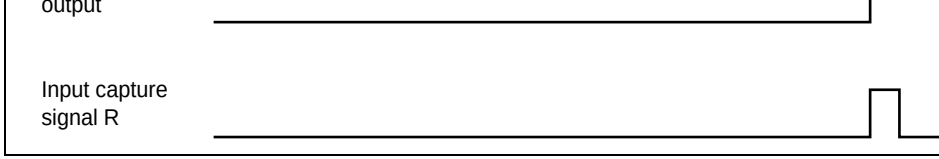


Figure 9.12 Input Capture Input Timing (with Noise Cancellation Functi

4. Timing of Input Capture by Input Capture Input

Figure 9.13 shows the timing of input capture by input capture input

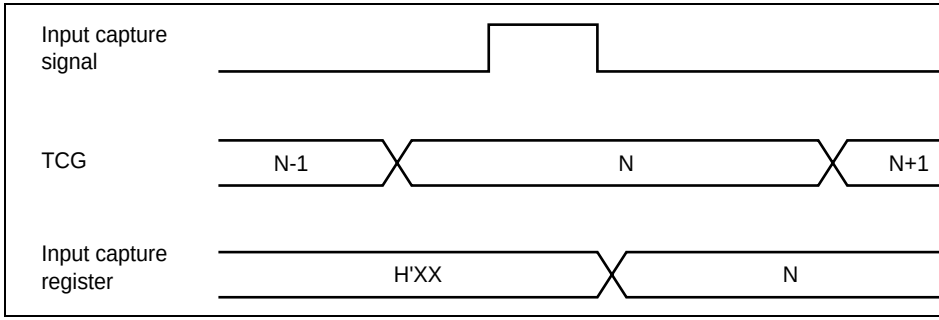


Figure 9.13 Timing of Input Capture by Input Capture Input

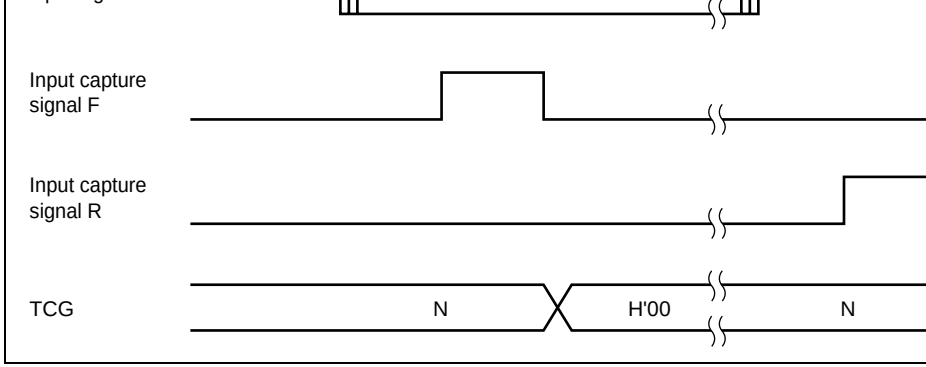


Figure 9.14 TCG Clear Timing

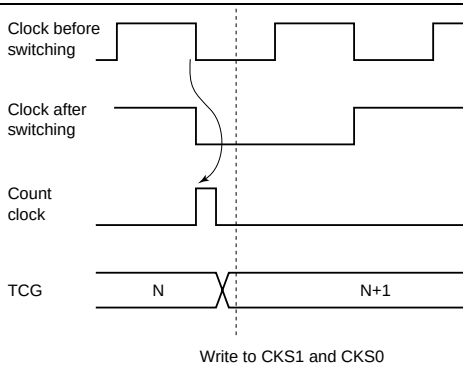
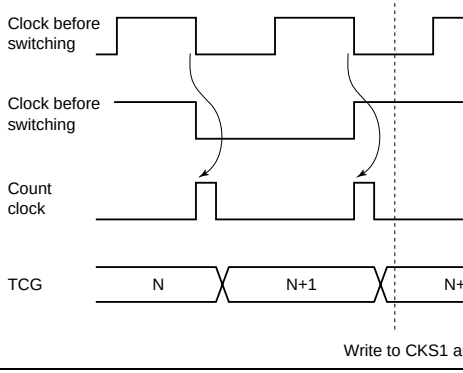
					halted	halted	halted	
	Interval	Reset	Functions*	Functions*	Functions/ halted*	Functions/ halted*	Functions/ halted*	Halted
ICRGF		Reset	Functions*	Functions*	Functions/ halted*	Functions/ halted*	Functions/ halted*	Held
ICRGR		Reset	Functions*	Functions*	Functions halted*	Functions/ halted*	Functions/ halted*	Held
TMG		Reset	Functions	Held	Held	Functions	Held	Held

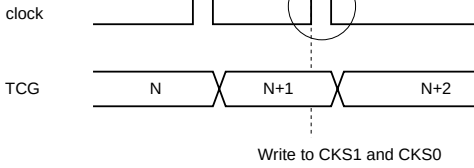
Note: * When ϕw/4 is selected as the TCG internal clock in active mode or sleep mo
the system clock and internal clock are mutually asynchronous, synchroniza
maintained by a synchronization circuit. This results in a maximum count cy
1/ϕ (s). When ϕw/4 is selected as the TCG internal clock in watch mode, TC
noise canceler operate on the ϕw/4 internal clock without regard to the ϕ_{SUB} s
(ϕw/8, ϕw/4, ϕw/2). Note that when another internal clock is selected, TCG a
noise canceler do not operate, and input of the input capture input signal doe
in input capture.

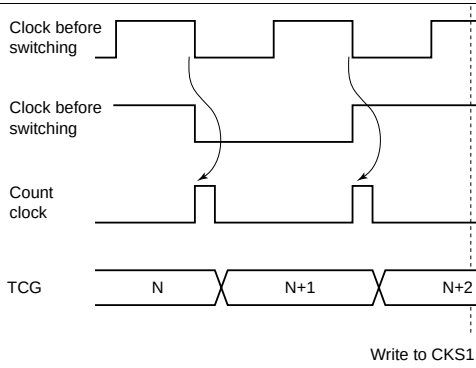
To be operated Timer G in subactive mode or subsleep mode, select ϕw/4 fo
clock of TCG and also select ϕw/2 for sub clock ϕ_{SUB}. When another interna
selected and when another sub clock (ϕw/8, ϕw/4) is selected, TCG and nois
do not operate.

which TCG is internally clocked, an increment pulse is generated on detection of the f
of an internal clock signal, which is divided from the system clock (φ) or subclock (φw
reason, in a case like No. 3 in table 9.14 where the switch is from a high clock signal t
clock signal, the switchover is seen as a falling edge, causing TCG to increment.

Table 9.14 Internal Clock Switching and TCG Operation

No.	Clock Levels 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	



4	Goes from high level to high level	

Note: * The switchover is seen as a falling edge, and TCG is incremented.

signal input edges, and the conditions for their occurrence, are summarized in tabl

Table 9.15 Input Capture Input Signal Input Edges Due to Input Capture Inpu Switching, and Conditions for Their Occurrence

Input Capture Input Signal Input Edge	Conditions
Generation of rising edge	When TMIG is modified from 0 to 1 while the TMIG p
	When NCS is modified from 0 to 1 while the TMIG p then TMIG is modified from 0 to 1 before the signal i five times by the noise canceler
Generation of falling edge	When TMIG is modified from 1 to 0 while the TMIG p
	When NCS is modified from 0 to 1 while the TMIG p then TMIG is modified from 0 to 1 before the signal i five times by the noise canceler
	When NCS is modified from 0 to 1 while the TMIG p then TMIG is modified from 1 to 0 after the signal is times by the noise canceler

Note: When the $P1_3$ pin is not set as an input capture input pin, the timer G input capt signal is low.

Switching, and Conditions for Their Occurrence

Input Capture Input Signal Input Edge	Conditions
Generation of rising edge	When the TMIG pin level is switched from low to high TMIG is set to 1, then NCS is modified from 0 to 1 be signal is sampled five times by the noise canceler
Generation of falling edge	When the TMIG pin level is switched from high to low TMIG is set to 1, then NCS is modified from 1 to 0 be signal is sampled five times by the noise canceler

When the pin function is switched and an edge is generated in the input capture inpu this edge matches the edge selected by the input capture interrupt select (IIEGS) bit interrupt request flag will be set to 1. The interrupt request flag should therefore be 0 before use. Figure 9.15 shows the procedure for port mode register manipulation interrupt request flag clearing. When switching the pin function, set the interrupt-di state before manipulating the port mode register, then, after the port mode register c has been performed, wait for the time required to confirm the input capture input sig input capture signal (at least two system clocks when the noise canceler is not used; five sampling clocks when the noise canceler is used), before clearing the interrupt to 0. There are two ways of preventing interrupt request flag setting when the pin fu switched: by controlling the pin level so that the conditions shown in tables 9.15 an not satisfied, or by setting the opposite of the generated edge in the IIEGS bit in TM

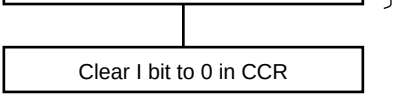


Figure 9.15 Port Mode Register Manipulation and Interrupt Enable Flag C Procedure

9.5.6 Timer G Application Example

Using timer G, it is possible to measure the high and low widths of the input capture i as absolute values. For this purpose, CCLR1 and CCLR0 should both be set to 1 in T

Figure 9.16 shows an example of the operation in this case.

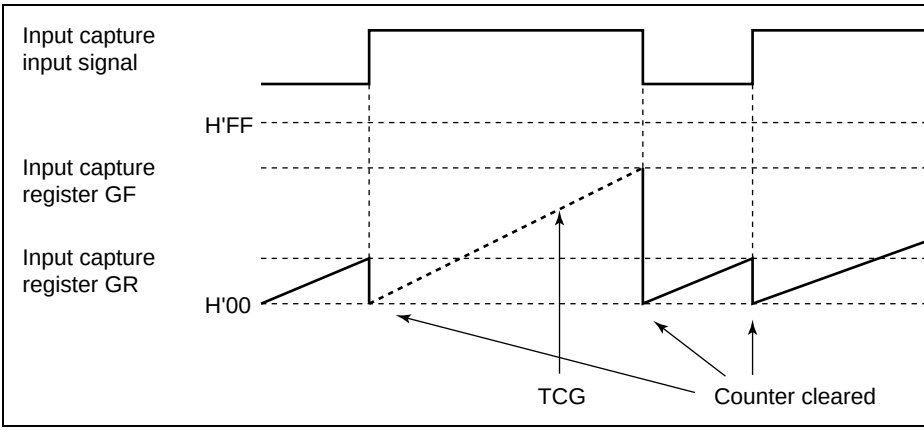


Figure 9.16 Timer G Application Example

1. Features

Features of the watchdog timer are given below.

- Incremented by internal clock source (ϕ/8192 or ϕw/32).
- A reset signal is generated when the counter overflows. The overflow period can be from 1 to 256 times 8192/ϕ or 32/ϕw (from approximately 4 ms to 1000 ms when ϕ MHz).
- Use of module standby mode enables this module to be placed in standby mode ind when not used.

2. Block Diagram

Figure 9.17 shows a block diagram of the watchdog timer.

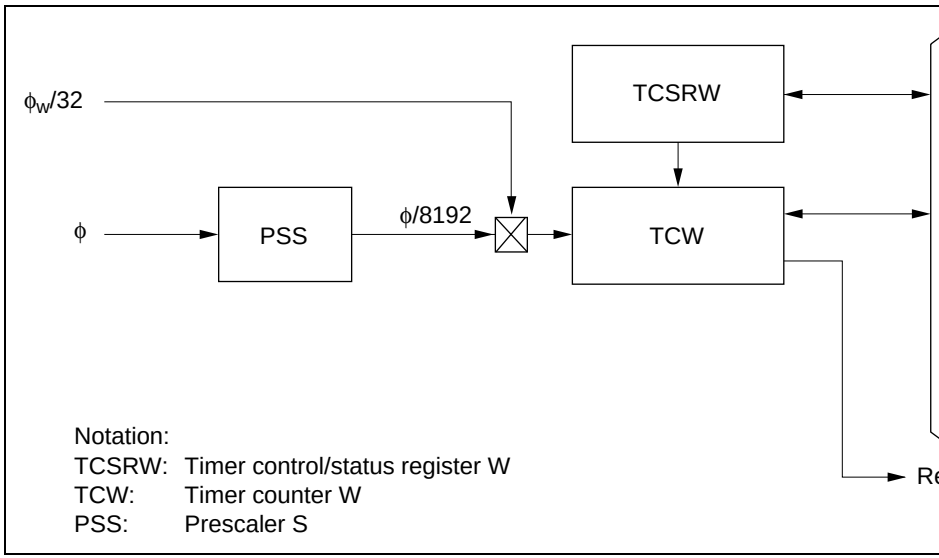


Figure 9.17 Block Diagram of Watchdog Timer

Clock stop register 2	CKSTP2	R/W	H'FF
Port mode register 3	PMR3	R/W	H'00

9.6.2 Register Descriptions

1. Timer Control/Status Register W (TCSRW)

Bit	7	6	5	4	3	2	1
	B6WI	TCWE	B4WI	TCSRWE	B2WI	WDON	B0WI
Initial value	1	0	1	0	1	0	1
Read/Write	R	R/(W)*	R	R/(W)*	R	R/(W)*	R

Note: * Write is permitted only under certain conditions, which are given in the desc
the individual bits.

TCSRW is an 8-bit read/write register that controls write access to TCW and TCSRW controls watchdog timer operations, and indicates operating status.

Bit 7: Bit 6 write inhibit (B6WI)

Bit 7 controls the writing of data to bit 6 in TCSRW.

Bit 7 B6WI	Description
0	Bit 6 is write-enabled
1	Bit 6 is write-protected

This bit is always read as 1. Data written to this bit is not stored.

Bit 5: Bit 4 write inhibit (B4WI)

Bit 5 controls the writing of data to bit 4 in TCSRW.

Bit 5 B4WI	Description	
0	Bit 4 is write-enabled	
1	Bit 4 is write-protected	(i

This bit is always read as 1. Data written to this bit is not stored.

Bit 4: Timer control/status register W write enable (TCSRWE)

Bit 4 controls the writing of data to TCSRW bits 2 and 0.

Bit 4 TCSRWE	Description	
0	Data cannot be written to bits 2 and 0	(i
1	Data can be written to bits 2 and 0	

Bit 3: Bit 2 write inhibit (B2WI)

Bit 3 controls the writing of data to bit 2 in TCSRW.

Bit 3 B2WI	Description	
0	Bit 2 is write-enabled	
1	Bit 2 is write-protected	(i

This bit is always read as 1. Data written to this bit is not stored.

	Reset, or when TCSRWE = 1 and 0 is written in both B2WI and WDON
1	Watchdog timer operation is enabled Setting condition: When TCSRWE = 1 and 0 is written in B2WI and 1 is written in WDON

Counting starts when this bit is set to 1, and stops when this bit is cleared to 0.

Bit 1: Bit 0 write inhibit (B0WI)

Bit 1 controls the writing of data to bit 0 in TCSRW.

Bit 1 B0WI	Description
0	Bit 0 is write-enabled
1	Bit 0 is write-protected

This bit is always read as 1. Data written to this bit is not stored.

Bit 0: Watchdog timer reset (WRST)

Bit 0 indicates that TCW has overflowed, generating an internal reset signal. The inter signal generated by the overflow resets the entire chip. WRST is cleared to 0 by a rese $\overline{\text{RES}}$ pin, or when software writes 0.

Bit 0 WRST	Description
0	Clearing condition: Reset by $\overline{\text{RES}}$ pin When TCSRWE = 1, and 0 is written in both B0WI and WRST
1	Setting condition: When TCW overflows and an internal reset signal is generated

clock is ϕ/8192 or ϕw/32. The TCW value can always be written or read by the CPU.

When TCW overflows from H'FF to H'00, an internal reset signal is generated and WR
1 in TCSRW. Upon reset, TCW is initialized to H'00.

3. Clock Stop Register 2 (CKSTPR2)

Bit	7	6	5	4	3	2	1
	—	—	—	—	AECKSTP	WDCKSTP	PWCKSTP
Initial value	1	1	1	1	1	1	1
Read/Write	—	—	—	—	R/W	R/W	R/W

CKSTPR2 is an 8-bit read/write register that performs module standby mode control fo
modules. Only the bit relating to the watchdog timer is described here. For details of t
bits, see the sections on the relevant modules.

Bit 2: Watchdog timer module standby mode control (WDCKSTP)

Bit 2 controls setting and clearing of module standby mode for the watchdog timer.

WDCKSTP	Description
0	Watchdog timer is set to module standby mode
1	Watchdog timer module standby mode is cleared (i

Note: WDCKSTP is valid when the WDON bit is cleared to 0 in timer control/status reg
(TCSRW). If WDCKSTP is set to 0 while WDON is set to 1 (during watchdog tim
operation), 0 will be set in WDCKSTP but the watchdog timer will continue its wa
function and will not enter module standby mode. When the watchdog function e
WDON is cleared to 0 by software, the WDCKSTP setting will become valid and
watchdog timer will enter module standby mode.

pins. Only the bit relating to the watchdog timer is described here. For details of the
see section 8, I/O Ports.

Bit 5: Watchdog timer source clock select (WDCKS)

WDCKS	Description
0	ϕ/8192 selected
1	ϕw/32 selected

9.6.3 Timer Operation

The watchdog timer has an 8-bit counter (TCW) that is incremented by clock input (ϕ
ϕw/32). The input clock is selected by bit WDCKS in port mode register 3 (PMR3): ϕ
selected when WDCKS is cleared to 0, and ϕw/32 when set to 1. When TCSRWE = 1
if 0 is written in B2WI and 1 is simultaneously written in WDON, TCW starts countin
the TCW count reaches H'FF, the next clock input causes the watchdog timer to overf
internal reset signal is generated one reference clock (ϕ or ϕ_{SUB}) cycle later. The intern
signal is output for 512 clock cycles of the ϕ_{OSC} clock. It is possible to write to TCW,
TCW to count up from the written value. The overflow period can be set in the range
256 input clocks, depending on the value written in TCW.

Figure 9.18 shows an example of watchdog timer operations.

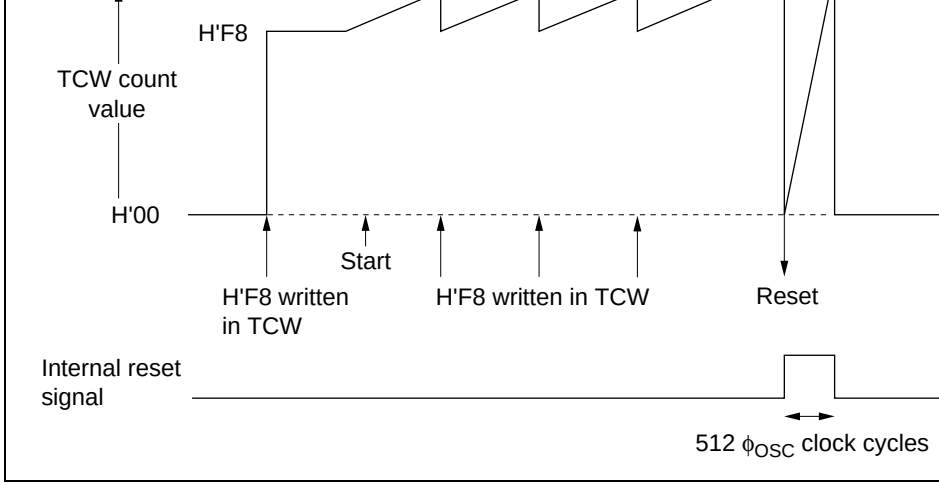


Figure 9.18 Typical Watchdog Timer Operations (Example)

9.6.4 Watchdog Timer Operation States

Table 9.18 summarizes the watchdog timer operation states.

Table 9.18 Watchdog Timer Operation States

Operation Mode	Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Standby
TCW	Reset	Functions	Functions	Halted	Functions/ Halted*	Halted	Halted
TCSRW	Reset	Functions	Functions	Retained	Functions/ Halted*	Retained	Retained

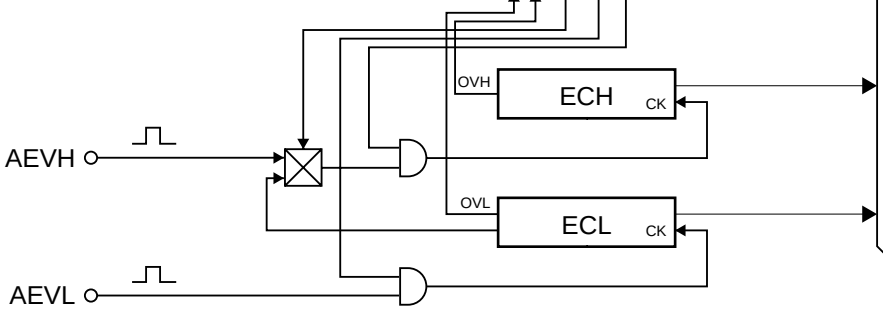
Note: * Functions when ϕw/32 is selected as the input clock.

Features of the asynchronous event counter are given below.

- Can count asynchronous events

  Can count external events input asynchronously without regard to the operation of ϕ and ϕ_{SUB}.

  The counter has a 16-bit configuration, enabling it to count up to 65536 (2^{16}) event
  - Can also be used as two independent 8-bit event counter channels.
  - Counter resetting and halting of the count-up function controllable by software
  - Automatic interrupt generation on detection of event counter overflow
  - Use of module standby mode enables this module to be placed in standby mod independently when not used.



Legend:
ECCSR: Event counter control/status register
ECH: Event counter H
ECL: Event counter L
AEVH: Asynchronous event input H
AEVL: Asynchronous event input L
IRREC: Event counter overflow interrupt request flag

Figure 9.19 Block Diagram of Asynchronous Event Counter

3. Pin Configuration

Table 9.19 shows the asynchronous event counter pin configuration.

Table 9.19 Pin Configuration

Name	Abbr.	I/O	Function
Asynchronous event input H	AEVH	Input	Event input pin for input to event
Asynchronous event input L	AEVL	Input	Event input pin for input to event

Event counter L	ECL	R	H'00
Clock stop register 2	CKSTP2	R/W	H'FF

9.7.2 Register Descriptions

1. Event Counter Control/Status Register (ECCSR)

Bit	7	6	5	4	3	2	1
	OVH	OVL	—	CH2	CUEH	CUEL	CRCH
Initial Value	0	0	0	0	0	0	0
Read/Write	R/W*	R/W*	R/W	R/W	R/W	R/W	R/W

Note: * Bits 7 and 6 can only be written with 0, for flag clearing.

ECCSR is an 8-bit read/write register that controls counter overflow detection, counte and halting of the count-up function.

ECCSR is initialized to H'00 upon reset.

Bit 7 OVH	Description
0	ECH has not overflowed (i Clearing condition: After reading OVH = 1, cleared by writing 0 to OVH
1	ECH has overflowed Setting condition: Set when ECH overflows from H'FF to H'00

Bit 6: Counter overflow flag L (OVL)

Bit 6 is a status flag indicating that ECL has overflowed from H'FF to H'00. This flag i ECL overflows. It is cleared by software but cannot be set by software. OVL is cleare reading it when set to 1, then writing 0.

Bit 6 OVL	Description
0	ECL has not overflowed (i Clearing condition: After reading OVL = 1, cleared by writing 0 to OVL
1	ECL has overflowed Setting condition: Set when ECL overflows from H'FF to H'00 while CH2 is set to 1

Bit 5: Reserved bit

Bit 5 is reserved; it can be read and written, and is initialized to 0 upon reset.

asynchronous event input.

Bit 4 CH2	Description
0	ECH and ECL are used together as a single-channel 16-bit event counte
1	ECH and ECL are used as two independent 8-bit event counter channels

Bit 3: Count-up enable H (CUEH)

Bit 3 enables event clock input to ECH. When 1 is written to this bit, event clock inpu and increments the counter. When 0 is written to this bit, event clock input is disabled ECH value is held. The AEVH pin or the ECL overflow signal can be selected as the source by bit CH2.

Bit 3 CUEH	Description
0	ECH event clock input is disabled ECH value is held
1	ECH event clock input is enabled

Bit 2: Count-up enable L (CUEL)

Bit 3 enables event clock input to ECL. When 1 is written to this bit, event clock inpu and increments the counter. When 0 is written to this bit, event clock input is disabled ECL value is held.

Bit 2 CUEL	Description
0	ECL event clock input is disabled ECL value is held
1	ECL event clock input is enabled

Bit 0: Counter reset control L (CRCL)

Bit 0 controls resetting of ECL. When this bit is cleared to 0, ECL is reset. When 1 is this bit, the counter reset is cleared and the ECL count-up function is enabled.

Bit 0 CRCL	Description
0	ECL is reset (i
1	ECL reset is cleared and count-up function is enabled

2. Event Counter H (ECH)

Bit	7	6	5	4	3	2	1
	ECH7	ECH6	ECH5	ECH4	ECH3	ECH2	ECH1
Initial Value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

ECH is an 8-bit read-only up-counter that operates either as an independent 8-bit event as the upper 8-bit up-counter of a 16-bit event counter configured in combination with Either the external asynchronous event AEVH pin or the overflow signal from lower 8- ECL can be selected as the input clock source by bit CH2. ECH can be cleared to H'00 software, and is also initialized to H'00 upon reset.

	ECL7	ECL6	ECL5	ECL4	ECL3	ECL2	ECL1
Initial Value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

4. Clock Stop Register 2 (CKSTPR2)

Bit	7	6	5	4	3	2	1
	—	—	—	—	AECKSTP	WDCKSTP	PWCKST
Initial value	1	1	1	1	1	1	1
Read/Write	—	—	—	—	R/W	R/W	R/W

CKSTPR2 is an 8-bit read/write register that performs module standby mode control f
modules. Only the bit relating to the asynchronous event counter is described here. F
the other bits, see the sections on the relevant modules.

Bit 3: Asynchronous event counter module standby mode control (AECKSTP)

Bit 3 controls setting and clearing of module standby mode for the asynchronous even

AECKSTP	Description
0	Asynchronous event counter is set to module standby mode
1	Asynchronous event counter module standby mode is cleared

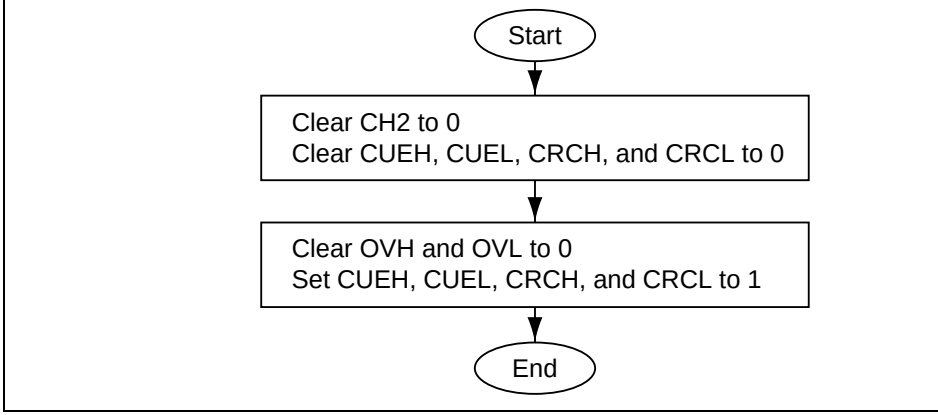


Figure 9.20 Example of Software Processing when Using ECH and ECL as 16-
Counter

As CH2 is cleared to 0 by a reset, ECH and ECL operate as a 16-bit event counter after
They can also be used as a 16-bit event counter by carrying out the software processing
the example in figure 9.20. The operating clock source is asynchronous event input fro
AEVL pin. When the next clock is input after the count value reaches H'FF in both EC
ECL, ECH and ECL overflow from H'FFFF to H'0000, the OVH flag is set to 1 in ECC
ECH and ECL count values each return to H'00, and counting up is restarted. When ov
occurs, the IRREC bit is set to 1 in IRR2. If the IENEC bit in IENR2 is 1 at this time, a
request is sent to the CPU.

2. 8-bit Event Counter Operation

When bit CH2 is set to 1 in ECCSR, ECH and ECL operate as independent 8-bit event
Figure 9.21 shows an example of the software processing when ECH and ECL are used
event counters.

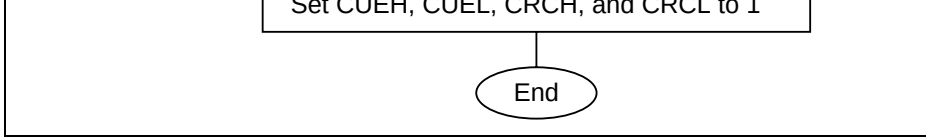


Figure 9.21 Example of Software Processing when Using ECH and ECL as 8-Counters

ECH and ECL can be used as 8-bit event counters by carrying out the software proces in the example in figure 9.20. The 8-bit event counter operating clock source is asyn event input from the AEVH pin for ECH, and asynchronous event input from the AE ECL. When the next clock is input after the ECH count value reaches H'FF, ECH ov OVH flag is set to 1 in ECCSR, the ECH count value returns to H'00, and counting up Similarly, when the next clock is input after the ECL count value reaches H'FF, ECL the OVL flag is set to 1 in ECCSR, the ECL count value returns to H'00, and counting restarted. When overflow occurs, the IRREC bit is set to 1 in IRR2. If the IENEC bit 1 at this time, an interrupt request is sent to the CPU.

9.7.4 Asynchronous Event Counter Operation Modes

Asynchronous event counter operation modes are shown in table 9.21.

Table 9.21 Asynchronous Event Counter Operation Modes

Operation Mode	Reset	Active	Sleep	Watch	Sub-active	Sub-sleep	Standb
ECCSR	Reset	Functions	Functions	Held*	Functions	Functions	Held*
ECH	Reset	Functions	Functions	Functions*	Functions	Functions	Functio
ECL	Reset	Functions	Functions	Functions*	Functions	Functions	Functio

Note: * When an asynchronous external event is input, the counter increments but overflow H/L flags are not affected.

4.5 to 5.5 V, 10 MHz when Vcc = 2.7 to 5.5 V, and 4 MHz when Vcc = 1.8 to 5.5 V
internal power step-down circuit is used, the maximum clock frequency to be input
when Vcc = 2.7 to 5.5 V, and 4 MHz when Vcc = 1.8 to 5.5 V. In the H8/3827S Gr
maximum clock frequency to be input is 10 MHz when Vcc = 2.7 to 3.6 V, and 4 M
Vcc = 1.8 to 3.6 V. In the H8/38327 Group and H8/38427 Group, the maximum clo
frequency to be input is 16 MHz. In addition, ensure that the high and low widths o
are at least 32 ns. The duty cycle is immaterial.

			used
			V_{CC} = 2.7 to 5
			V_{CC} = 1.8 to 5
			H8/3827S Grou
			V_{CC} = 2.7 to 3
			V_{CC} = 1.8 to 3
			H8/38327 Grou
			V_{CC} = 2.7 to 5
			H8/38427 Grou
			V_{CC} = 4.5 to 5
			V_{CC} = 2.7 to 5
8-bit mode	Active (medium-speed), sleep (medium-speed)	(ϕ/16)	$2 \cdot f_{OSC}$
		(ϕ/32)	f_{OSC}
		(ϕ/64)	$1/2 \cdot f_{OSC}$
	f_{OSC} = 1 MHz to 16 MHz	(ϕ/128)	$1/4 \cdot f_{OSC}$
8-bit mode	Watch, subactive, subsleep, standby	(ϕw/2)	1000 kHz
		(ϕw/4)	500 kHz
	ϕ_W = 32.768 kHz or 38.4 kHz	(ϕw/8)	250 kHz

3. When AEC uses with 16-bit mode, set CUEH in ECCSR to “1” first, set CRCH in “1” second, or set both CUEH and CRCH to “1” at same time before clock entry. is operating on 16-bit mode, do not change CUEH. Otherwise, ECH will be misc Also, to reset the counter, clear CRCH and CRCL to 0 simultaneously or clear CR CRCH to 0 sequentially, in that order.

Serial communication interface 3 (SCI3) can carry out serial data communication in
asynchronous or synchronous mode. It is also provided with a multiprocessor commu
function that enables serial data to be transferred among processors.

10.1.1 Features

Features of SCI3 are listed below.

- Choice of asynchronous or synchronous mode for serial data communication
  - Asynchronous mode

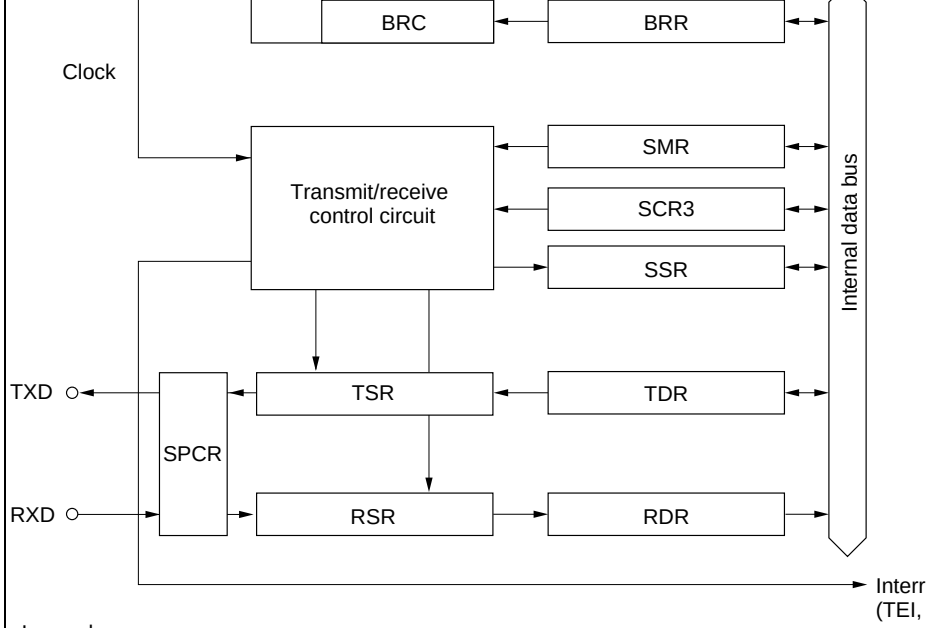
    Serial data communication is performed asynchronously, with synchronization
    character by character. In this mode, serial data can be exchanged with standar
    asynchronous communication LSIs such as a Universal Asynchronous
    Receiver/Transmitter (UART) or Asynchronous Communication Interface Ada
    (ACIA). A multiprocessor communication function is also provided, enabling
    communication among processors.

    There is a choice of 16 data transfer formats.

Data length	7, 8, 5 bits
Stop bit length	1 or 2 bits
Parity	Even, odd, or none
Multiprocessor bit	“1” or “0”
Receive error detection	Parity, overrun, and framing errors
Break detection	Break detected by reading the RXD_{3x} pin level dire framing error occurs

Separate transmission and reception units are provided, enabling transmission and r
be carried out simultaneously. The transmission and reception units are both doubl
allowing continuous transmission and reception.

- On-chip baud rate generator, allowing any desired bit rate to be selected
- Choice of an internal or external clock as the transmit/receive clock source
- Six interrupt sources: transmit end, transmit data empty, receive data full, overrun e
framing error, and parity error



Legend:

RSR:	Receive shift register
RDR:	Receive data register
TSR:	Transmit shift register
TDR:	Transmit data register
SMR:	Serial mode register
SCR3:	Serial control register 3
SSR:	Serial status register
BRR:	Bit rate register
BRC:	Bit rate counter
SPCR:	Serial port control register

Figure 10.1 SCI3 Block Diagram

SCI3 receive data input	RXD_{3x}	Input	SCI3 receive data in
SCI3 transmit data output	TXD_{3x}	Output	SCI3 transmit data o

10.1.4 Register Configuration

Table 10.2 shows the SCI3 register configuration.

Table 10.2 Registers

Name	Abbr.	R/W	Initial Value	Add
Serial mode register	SMR	R/W	H'00	H'F
Bit rate register	BRR	R/W	H'FF	H'F
Serial control register 3	SCR3	R/W	H'00	H'F
Transmit data register	TDR	R/W	H'FF	H'F
Serial data register	SSR	R/W	H'84	H'F
Receive data register	RDR	R	H'00	H'F
Transmit shift register	TSR	Protected	—	—
Receive shift register	RSR	Protected	—	—
Bit rate counter	BRC	Protected	—	—
Clock stop register 1	CKSTPR1	R/W	H'FF	H'F
Serial port control register	SPCR	R/W	H'C0	H'F

Read/Write	—	—	—	—	—	—	—

RSR is a register used to receive serial data. Serial data input to RSR from the RXD_{3x} the order in which it is received, starting from the LSB (bit 0), and converted to parall When one byte of data is received, it is transferred to RDR automatically.

RSR cannot be read or written directly by the CPU.

10.2.2 Receive Data Register (RDR)

Bit	7	6	5	4	3	2	1
	RDR7	RDR6	RDR5	RDR4	RDR3	RDR2	RDR1
Initial value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

RDR is an 8-bit register that stores received serial data.

When reception of one byte of data is finished, the received data is transferred from R and the receive operation is completed. RSR is then able to receive data. RSR and RI double-buffered, allowing consecutive receive operations.

RDR is a read-only register, and cannot be written by the CPU.

RDR is initialized to H'00 upon reset, and in standby, module standby or watch mode.

TSR is a register used to transmit serial data. Transmit data is first transferred from TD and serial data transmission is carried out by sending the data to the TXD_{3x} pin in order from the LSB (bit 0). When one byte of data is transmitted, the next byte of transmit da transferred to TDR, and transmission started, automatically. Data transfer from TDR to not performed if no data has been written to TDR (if bit TDRE is set to 1 in the serial st register (SSR)).

TSR cannot be read or written directly by the CPU.

10.2.4 Transmit Data Register (TDR)

Bit	7	6	5	4	3	2	1
	TDR7	TDR6	TDR5	TDR4	TDR3	TDR2	TDR1
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

TDR is an 8-bit register that stores transmit data. When TSR is found to be empty, the data written in TDR is transferred to TSR, and serial data transmission is started. Cont transmission is possible by writing the next transmit data to TDR during TSR serial dat transmission.

TDR can be read or written by the CPU at any time.

TDR is initialized to H'FF upon reset, and in standby, module standby, or watch mode.

SMR is an 8-bit register used to set the serial data transfer format and to select the clo
the baud rate generator.

SMR can be read or written by the CPU at any time.

SMR is initialized to H'00 upon reset, and in standby, module standby, or watch mode

Bit 7: Communication mode (COM)

Bit 7 selects whether SCI3 operates in asynchronous mode or synchronous mode.

Bit 7 COM	Description
0	Asynchronous mode
1	Synchronous mode

Bit 6: Character length (CHR)

Bit 6 selects either 7 or 8 bits as the data length to be used in asynchronous mode. In
mode the data length is always 8 bits, irrespective of the bit 6 setting.

Bit 6 CHR	Description
0	8-bit data/5-bit data*2
1	7-bit data*1/5-bit data*2

Notes: 1. When 7-bit data is selected, the MSB (bit 7) of TDR is not transmitted.
2. When 5-bit data is selected, set both PE and MP to 1. The three most sign
(bits 7, 6, and 5) of TDR are not transmitted.

1	Parity bit addition and checking enabled*1*2

Notes: 1. When PE is set to 1, even or odd parity, as designated by bit PM, is added t data before it is sent, and the received parity bit is checked against the parity designated by bit PM.
2. For the case where 5-bit data is selected, see table 10.11.

Bit 4: Parity mode (PM)

Bit 4 selects whether even or odd parity is to be used for parity addition and checking. setting is only valid in asynchronous mode when bit PE is set to 1, enabling parity bit a checking. The PM bit setting is invalid in synchronous mode, and in asynchronous mo bit addition and checking is disabled.

Bit 4 PM	Description
0	Even parity*1 (i
1	Odd parity*2

Notes: 1. When even parity is selected, a parity bit is added in transmission so that the number of 1 bits in the transmit data plus the parity bit is an even number; in a check is carried out to confirm that the number of 1 bits in the receive data parity bit is an even number.
2. When odd parity is selected, a parity bit is added in transmission so that the number of 1 bits in the transmit data plus the parity bit is an odd number; in r check is carried out to confirm that the number of 1 bits in the receive data p parity bit is an odd number.

1	2 stop bits*2

Notes: 1. In transmission, a single 1 bit (stop bit) is added at the end of a transmit cha
2. In transmission, two 1 bits (stop bits) are added at the end of a transmit cha

In reception, only the first of the received stop bits is checked, irrespective of the STO
If the second stop bit is 1 it is treated as a stop bit, but if 0, it is treated as the start bit o
transmit character.

Bit 2: Multiprocessor mode (MP)

Bit 2 enables or disables the multiprocessor communication function. When the multi
communication function is disabled, the parity settings in the PE and PM bits are inva
bit setting is only valid in asynchronous mode. When synchronous mode is selected t
should be set to 0. For details on the multiprocessor communication function, see sec
Multiprocessor Communication Function.

Bit 2 MP	Description
0	Multiprocessor communication function disabled*
1	Multiprocessor communication function enabled*

Note: * For the case where 5-bit data is selected, see table 10.11.

0	0	φ clock	(i
0	1	φ w/2 clock*1/φ w clock*2	
1	0	φ/16 clock	
1	1	φ/64 clock	

Notes: 1. φ w/2 clock in active (medium-speed/high-speed) mode and sleep mode
2. φ w clock in subactive mode and subsleep mode
In subactive or subsleep mode, SCI3 can be operated when CPU clock is φw

10.2.6 Serial Control Register 3 (SCR3)

Bit	7	6	5	4	3	2	1
	TIE	RIE	TE	RE	MPIE	TEIE	CKE1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

SCR3 is an 8-bit register for selecting transmit or receive operation, the asynchronous m output, interrupt request enabling or disabling, and the transmit/receive clock source.

SCR3 can be read or written by the CPU at any time.

SCR3 is initialized to H'00 upon reset, and in standby, module standby or watch mode.

Bit 7: Transmit interrupt enable (TIE)

Bit 7 selects enabling or disabling of the transmit data empty interrupt request (TXI) wh transmit data is transferred from the transmit data register (TDR) to the transmit shift re (TSR), and bit TDRE in the serial status register (SSR) is set to 1.

TXI can be released by clearing bit TDRE or bit TIE to 0.

error interrupt request (ERI) when receive data is transferred from the receive shift reg
to the receive data register (RDR), and bit RDRF in the serial status register (SSR) is s
There are three kinds of receive error: overrun, framing, and parity.

RXI can be released by clearing bit RDRF or the FER, PER, or OER error flag to 0, o
bit RIE to 0.

Bit 6 RIE	Description
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

Bit 5: Transmit enable (TE)

Bit 5 selects enabling or disabling of the start of transmit operation.

Bit 5 TE	Description
0	Transmit operation disabled*1 (TXD pin is I/O port)
1	Transmit operation enabled*2 (TXD pin is transmit data pin)

Notes: 1. Bit TDRE in SSR is fixed at 1.

2. When transmit data is written to TDR in this state, bit TDR in SSR is cleare
serial data transmission is started. Be sure to carry out serial mode registe
settings, and setting of bit SPC31 or SPC32 in SPCR, to decide the transm
before setting bit TE to 1.

Notes: 1. Note that the RDRF, FER, PER, and OER flags in SSR are not affected wh
cleared to 0, and retain their previous state.

2. In this state, serial data reception is started when a start bit is detected in as
mode or serial clock input is detected in synchronous mode. Be sure to carr
mode register (SMR) settings to decide the reception format before setting b

Bit 3: Multiprocessor interrupt enable (MPIE)

Bit 3 selects enabling or disabling of the multiprocessor interrupt request. The MPIE b
only valid when asynchronous mode is selected and reception is carried out with bit MI
set to 1. The MPIE bit setting is invalid when bit COM is set to 1 or bit MP is cleared t

Bit 3 MPIE	Description
0	Multiprocessor interrupt request disabled (normal receive operation) (i Clearing condition: When data is received in which the multiprocessor bit is set to 1
1	Multiprocessor interrupt request enabled*

Note: * Receive data transfer from RSR to RDR, receive error detection, and setting
RDRF, FER, and OER status flags in SSR is not performed. RXI, ERI, and s
the RDRF, FER, and OER flags in SSR, are disabled until data with the mult
bit set to 1 is received. When a receive character with the multiprocessor bit
received, bit MPBR in SSR is set to 1, bit MPIE is automatically cleared to 0,
and ERI requests (when bits TIE and RIE in serial control register 3 (SCR3)
1) and setting of the RDRF, FER, and OER flags are enabled.

Note: * TEI can be released by clearing bit TDRE to 0 and clearing bit TEND to 0 i
clearing bit TEIE to 0.

Bits 1 and 0: Clock enable 1 and 0 (CKE1, CKE0)

Bits 1 and 0 select the clock source and enabling or disabling of clock output from the
The combination of CKE1 and CKE0 determines whether the SCK_{3x} pin functions as
a clock output pin, or a clock input pin.

The CKE0 bit setting is only valid in case of internal clock operation (CKE1 = 0) in a
mode. In synchronous mode, or when external clock operation is used (CKE1 = 1), b
should be cleared to 0.

After setting bits CKE1 and CKE0, set the operating mode in the serial mode register

For details on clock source selection, see table 10.9 in 10.3.1, Overview.

Bit 1 CKE1	Bit 0 CKE0	Description		
		Communication Mode	**Clock Source**	**SCK_{3x} Pin Fu**
0	0	Asynchronous	Internal clock	I/O port*1
		Synchronous	Internal clock	Serial clock ou
0	1	Asynchronous	Internal clock	Clock output*2
		Synchronous	Reserved	
1	0	Asynchronous	External clock	Clock input*3
		Synchronous	External clock	Serial clock in
1	1	Asynchronous	Reserved	
		Synchronous	Reserved	

Notes: 1. Initial value
2. A clock with the same frequency as the bit rate is output.
3. Input a clock with a frequency 16 times the bit rate.

SSR is an 8-bit register containing status flags that indicate the operational status of SC multiprocessor bits.

SSR can be read or written by the CPU at any time, but only a write of 1 is possible to l RDRF, OER, PER, and FER. In order to clear these bits by writing 0, 1 must first be re

Bits TEND and MPBR are read-only bits, and cannot be modified.

SSR is initialized to H'84 upon reset, and in standby, module standby, or watch mode.

Bit 7: Transmit data register empty (TDRE)

Bit 7 indicates that transmit data has been transferred from TDR to TSR.

Bit 7 TDRE	Description
0	Transmit data written in TDR has not been transferred to TSR Clearing conditions: After reading TDRE = 1, cleared by writing 0 to TDRE When data is written to TDR by an instruction
1	Transmit data has not been written to TDR, or transmit data written in TDR transferred to TSR Setting conditions: When bit TE in SCR3 is cleared to 0 When data is transferred from TDR to TSR (i

	After reading RDRF = 1, cleared by writing 0 to RDRF
	When RDR data is read by an instruction
1	There is receive data in RDR
	Setting condition:
	When reception ends normally and receive data is transferred from RSR

Note: If an error is detected in the receive data, or if the RE bit in SCR3 has been cle RDR and bit RDRF are not affected and retain their previous state.
Note that if data reception is completed while bit RDRF is still set to 1, an overr (OER) will result and the receive data will be lost.

Bit 5: Overrun error (OER)

Bit 5 indicates that an overrun error has occurred during reception.

Bit 5 OER	Description
0	Reception in progress or completed*1
	Clearing condition:
	After reading OER = 1, cleared by writing 0 to OER
1	An overrun error has occurred during reception*2
	Setting condition:
	When reception is completed with RDRF set to 1

Notes: 1. When bit RE in SCR3 is cleared to 0, bit OER is not affected and retains its state.
2. RDR retains the receive data it held before the overrun error occurred, and received after the error is lost. Reception cannot be continued with bit OER and in synchronous mode, transmission cannot be continued either.

	After reading PER = 1, cleared by writing 0 to PER
1	A framing error has occurred during reception Setting condition: When the stop bit at the end of the receive data is checked for a value of 1 of reception, and the stop bit is 0[*2]

Notes: 1. When bit RE in SCR3 is cleared to 0, bit FER is not affected and retains its p state.
2. Note that, in 2-stop-bit mode, only the first stop bit is checked for a value of 1 second stop bit is not checked. When a framing error occurs the receive dat transferred to RDR but bit RDRF is not set. Reception cannot be continued FER set to 1. In synchronous mode, neither transmission nor reception is po when bit FER is set to 1.

Bit 3: Parity error (PER)

Bit 3 indicates that a parity error has occurred during reception with parity added in asy mode.

Bit 3 PER	Description
0	Reception in progress or completed[*1] (i Clearing condition: After reading PER = 1, cleared by writing 0 to PER
1	A parity error has occurred during reception[*2] Setting condition: When the number of 1 bits in the receive data plus parity bit does not mat parity designated by bit PM in the serial mode register (SMR)

Notes: 1. When bit RE in SCR3 is cleared to 0, bit PER is not affected and retains its p state.
2. Receive data in which it a parity error has occurred is still transferred to RDR RDRF is not set. Reception cannot be continued with bit PER set to 1. In sy mode, neither transmission nor reception is possible when bit FER is set to 1

	Clearing conditions: After reading TDRE = 1, cleared by writing 0 to TDRE When data is written to TDR by an instruction	
1	Transmission ended Setting conditions: When bit TE in SCR3 is cleared to 0 When bit TDRE is set to 1 when the last bit of a transmit character is ser	(

Bit 1: Multiprocessor bit receive (MPBR)

Bit 1 stores the multiprocessor bit in a receive character during multiprocessor format asynchronous mode.

Bit 1 is a read-only bit and cannot be modified.

Bit 1 MPBR	Description	
0	Data in which the multiprocessor bit is 0 has been received*	(
1	Data in which the multiprocessor bit is 1 has been received	

Note: * When bit RE is cleared to 0 in SCR3 with the multiprocessor format, bit MP affected and retains its previous state.

1	A 1 multiprocessor bit is transmitted

10.2.8 Bit Rate Register (BRR)

Bit	7	6	5	4	3	2	1
	BRR7	BRR6	BRR5	BRR4	BRR3	BRR2	BRR1
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

BRR is an 8-bit register that designates the transmit/receive bit rate in accordance with
rate generator operating clock selected by bits CKS1 and CKS0 of the serial mode regis

BRR can be read or written by the CPU at any time.

BRR is initialized to H'FF upon reset, and in standby, module standby, or watch mode.

Table 10.3 shows examples of BRR settings in asynchronous mode. The values shown
active (high-speed) mode.

200		0	2	0	0	155	0.16	3	2	0	—
250		—	—	—	0	124	0	0	153	–0.26	0
300		0	1	0	0	103	0.16	3	1	0	2
600		0	0	0	0	51	0.16	3	0	0	0
1200		—	—	—	0	25	0.16	2	1	0	0
2400		—	—	—	0	12	0.16	2	0	0	0
4800		—	—	—	—	—	—	0	7	0	0
9600		—	—	—	—	—	—	0	3	0	—
19200		—	—	—	—	—	—	0	1	0	—
31250		—	—	—	0	0	0	—	—	—	0
38400		—	—	—	—	—	—	0	0	0	—

200	2	48	-0.35	2	77	0.16
250	2	38	0.16	2	62	-0.79
300	—	—	—	2	51	0.16
600	—	—	—	2	25	0.16
1200	0	129	0.16	0	207	0.16
2400	0	64	0.16	0	103	0.16
4800	—	—	—	0	51	0.16
9600	—	—	—	0	25	0.16
19200	—	—	—	0	12	0.16
31250	0	4	0	0	7	0
38400	—	—	—	—	—	—

Notes: 1. The setting should be made so that the error is not more than 1%.

2. The value set in BRR is given by the following equation:

$$N = \frac{OSC}{(64 \times 2^{2n} \times B)} - 1$$

where B: Bit rate (bit/s)
N: Baud rate generator BRR setting ($0 \leq N \leq 255$)
OSC: Value of ϕ_{OSC} (Hz)
n: Baud rate generator input clock number (n = 0, 2, or 3)
(The relation between n and the clock is shown in table 10.4.)

3. The error in table 10.3 is the value obtained from the following equation, ro
two decimal places.

$$\text{Error (\%)} = \frac{\text{B (rate obtained from n, N, OSC)} - \text{R(bit rate in left-hand column in table 10.3.)}}{\text{R (bit rate in left-hand column in table 10.3.)}}$$

Notes: 1. φ w/2 clock in active (medium-speed/high-speed) mode and sleep mode
2. φ w clock in subactive mode and subsleep mode
In subactive or subsleep mode, SCI3 can be operated when CPU clock is φ

Table 10.5 shows the maximum bit rate for each frequency. The values shown are for
(high-speed) mode.

Table 10.5 Maximum Bit Rate for Each Frequency (Asynchronous Mode)

		Setting	
OSC (MHz)	**Maximum Bit Rate (bit/s)**	**n**	**N**
0.0384*	600	0	0
2	31250	0	0
2.4576	38400	0	0
4	62500	0	0
10	156250	0	0
16	250000	0	0

*: When SMR is set up to CKS1 = “0”, CKS0 = “1”.

Table 10.6 shows examples of BRR settings in synchronous mode. The values shown
active (high-speed) mode.

500	—	—	—	—	—	—	—	—	—	2
1k	0	249	0	—	—	—	—	—	—	2
2.5k	0	99	0	0	199	0	—	—	—	2
5k	0	49	0	0	99	0	0	249	0	2
10k	0	24	0	0	49	0	0	124	0	0
25k	0	9	0	0	19	0	0	49	0	0
50k	0	4	0	0	9	0	0	24	0	0
100k	—	—	—	0	4	0	—	—	—	0
250k	0	0	0	0	1	0	0	4	0	0
500k				0	0	0	—	—	—	0
1M							—	—	—	0

Blank: Cannot be set.

—: A setting can be made, but an error will result.

Notes: The value set in BRR is given by the following equation:

$$N = \frac{OSC}{(8 \times 2^{2n} \times B)} - 1$$

where B: Bit rate (bit/s)

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

OSC: Value of ϕ_{OSC} (Hz)

n: Baud rate generator input clock number (n = 0, 2, or 3)

(The relation between n and the clock is shown in table 10.7.)

Notes: 1. φw/2 clock in active (medium-speed/high-speed) mode and sleep mode
2. φw clock in subactive mode and subsleep mode
In subactive or subsleep mode, SCI3 can be operated when CPU clock is φ

10.2.9 Clock Stop Register 1 (CKSTPR1)

Bit	7	6	5	4	3	2	1
	—	S31CKSTP	S32CKSTP	ADCKSTP	TGCKSTP	TFCKSTP	TCCKST
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CKSTPR1 is an 8-bit read/write register that performs module standby mode control f
modules. Only the bits relating to SCI3 are described here. For details of the other bi
sections on the relevant modules.

Bit 6: SCI3-1 module standby mode control (S31CKSTP)

Bit 6 controls setting and clearing of module standby mode for SCI31.

S31CKSTP	Description
0	SCI3-1 is set to module standby mode
1	SCI3-1 module standby mode is cleared

Note: All SCI31 register is initialized in module standby mode.

Bit 5: SCI3-2 module standby mode control (S32CKSTP)

Bit 5 controls setting and clearing of module standby mode for SCI32.

Bit	7	6	5	4	3	2	1	
	—	—	SPC32	SPC31	SCINV3	SCINV2	SCINV1	
Initial value	1	1	0	0	0	0	0	
Read/Write	—	—	R/W	R/W	R/W	R/W	R/W	

SPCR is an 8-bit readable/writable register that performs RXD_{31}, RXD_{32}, TXD_{31}, and T input/output data inversion switching. SPCR is initialized to H'C0 by a reset.

Bits 7 to 6: Reserved bits

Bits 7 to 6 are reserved; they are always read as 1 and cannot be modified.

Bit 5: $P4_2/TXD_{32}$ pin function switch (SPC32)

This bit selects whether pin $P4_2/TXD_{32}$ is used as $P4_2$ or as TXD_{32}.

Bit 5 SPC32	Description	
0	Functions as $P4_2$ I/O pin	(i
1	Functions as TXD_{32} output pin*	

Note: * Set the TE bit in SCR3 after setting this bit to 1.

Bit 4: $P3_5/TXD_{31}$ pin function switch (SPC31)

This bit selects whether pin $P3_5/TXD_{31}$ is used as $P3_5$ or as TXD_{31}.

Bit 4 SPC31	Description	
0	Functions as $P3_5$ I/O pin	(i
1	Functions as TXD_{31} output pin*	

Note: * Set the TE bit in SCR3 after setting this bit to 1.

Bit 2: RXD_{32} pin input data inversion switch

Bit 2 specifies whether or not RXD_{32} pin input data is to be inverted.

Bit 2 SCINV2	Description
0	RXD_{32} input data is not inverted
1	RXD_{32} input data is inverted

Bit 1: TXD_{31} pin output data inversion switch

Bit 1 specifies whether or not TXD_{31} pin output data is to be inverted.

Bit 1 SCINV1	Description
0	TXD_{31} output data is not inverted
1	TXD_{31} output data is inverted

Bit 0: RXD_{31} pin input data inversion switch

Bit 0 specifies whether or not RXD_{31} pin input data is to be inverted.

Bit 0 SCINV0	Description
0	RXD_{31} input data is not inverted
1	RXD_{31} input data is inverted

asynchronous or synchronous mode and the data transfer format, as shown in table 10.

The clock source for SCI3 is determined by bit COM in SMR and bits CKE1 and CKE
as shown in table 10.9.

1. Synchronous Mode

- Choice of 5-, 7-, or 8-bit data length
- Choice of parity addition, multiprocessor bit addition, and addition of 1 or 2 stop bi
  combination of these parameters determines the data transfer format and the charact
- Framing error (FER), parity error (PER), overrun error (OER), and break detection
  reception
- Choice of internal or external clock as the clock source

  When internal clock is selected: SCI3 operates on the baud rate generator clock, and
  with the same frequency as the bit rate can be output.

  When external clock is selected: A clock with a frequency 16 times the bit rate mus
  (The on-chip baud rate generator is not used.)

2. Synchronous Mode

- Data transfer format: Fixed 8-bit data length
- Overrun error (OER) detection during reception
- Choice of internal or external clock as the clock source

  When internal clock is selected: SCI3 operates on the baud rate generator clock, and
  clock is output.

  When external clock is selected: The on-chip baud rate generator is not used, and S
  operates on the input serial clock.

				1				
	1		0	0		7-bit data		No
				1				
			1	0				Yes
				1				
	0	1	0	0		8-bit data	Yes	No
				1				
			1	0		5-bit data	No	
				1				
	1		0	0		7-bit data	Yes	
				1				
			1	0		5-bit data	No	Yes
				1				
1	*	0	*	*	Synchronous mode	8-bit data	No	No

1	0	0	Synchronous mode	Internal	Outputs serial clock
	1	0		External	Inputs serial clock
0	1	1	Reserved (Do not specify these combinations)		
1	0	1			
1	1	1			

	RIE	normally and receive data is transferred from RSR to RDR, bit RDRF is set to 1, and if bit RIE is set to 1 at this time, RXI is enabled and an interrupt is requested. (See figure 10.2(a).)	receive data transferred clears bit RDRF to 0. C reception can be perforn repeating the above ope reception of the next RS completed.
TXI	TDRE TIE	When TSR is found to be empty (on completion of the previous transmission) and the transmit data placed in TDR is transferred to TSR, bit TDRE is set to 1. If bit TIE is set to 1 at this time, TXI is enabled and an interrupt is requested. (See figure 10.2(b).)	The TXI interrupt routine next transmit data to TD bit TDRE to 0. Continuc transmission can be per repeating the above ope the data transferred to T been transmitted.
TEI	TEND TEIE	When the last bit of the character in TSR is transmitted, if bit TDRE is set to 1, bit TEND is set to 1. If bit TEIE is set to 1 at this time, TEI is enabled and an interrupt is requested. (See figure 10.2(c).)	TEI indicates that the ne data has not been writte when the last bit of the t character in TSR is sent

Figure 10.2 (a) RDRF Setting and RXI Interrupt

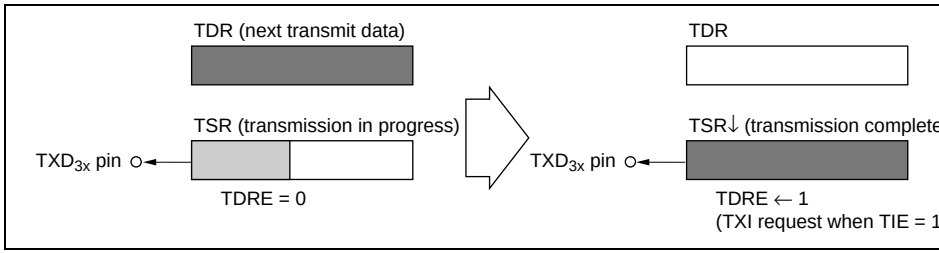


Figure 10.2 (b) TDRE Setting and TXI Interrupt

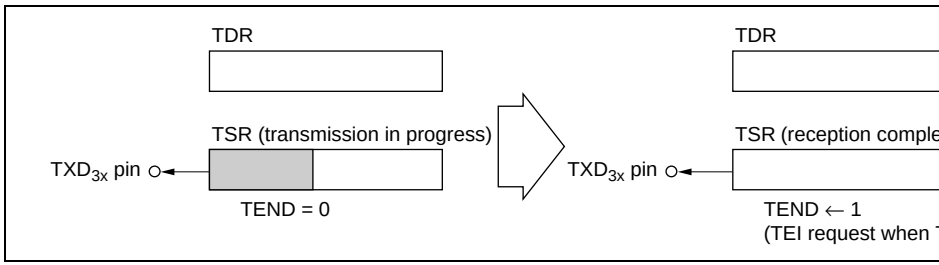


Figure 10.2 (c) TEND Setting and TEI Interrupt

and read during reception, making possible continuous transmission and reception.

1. Data Transfer Format

The general data transfer format in asynchronous communication is shown in figure 1

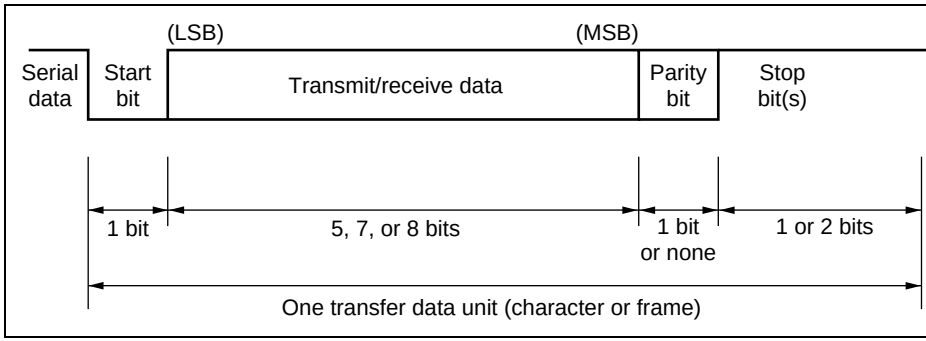


Figure 10.3 Data Format in Asynchronous Communication

In asynchronous communication, the communication line is normally in the mark stat level). SCI3 monitors the communication line and when it detects a space (low level) this as a start bit and begins serial data communication.

One transfer data character consists of a start bit (low level), followed by transmit/rec (LSB-first format, starting from the least significant bit), a parity bit (high or low leve finally one or two stop bits (high level).

In asynchronous mode, synchronization is performed by the falling edge of the start b reception. The data is sampled on the 8th pulse of a clock with a frequency 16 times t period, so that the transfer data is latched at the center of each bit.

Table 10.11 shows the 16 data transfer formats that can be set in asynchronous mode. is selected by the settings in the serial mode register (SMR).

0	0	1	0	S 8-bit data MPB STOP
0	0	1	1	S 8-bit data MPB STOP STOP
0	1	0	0	S 8-bit data P STOP
0	1	0	1	S 8-bit data P STOP STOP
0	1	1	0	S 5-bit data STOP
0	1	1	1	S 5-bit data STOP STOP
1	0	0	0	S 7-bit data STOP
1	0	0	1	S 7-bit data STOP STOP
1	0	1	0	S 7-bit data MPB STOP
1	0	1	1	S 7-bit data MPB STOP STOP
1	1	0	0	S 7-bit data P STOP
1	1	0	1	S 7-bit data P STOP STOP
1	1	1	0	S 5-bit data P STOP
1	1	1	1	S 5-bit data P STOP STOP

Legend:
S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multiprocessor bit

When SCI3 operates on an internal clock, the clock can be output at the SCK_{3x} pin. In the frequency of the output clock is the same as the bit rate, and the phase is such that rises at the center of each bit of transmit/receive data, as shown in figure 10.4.

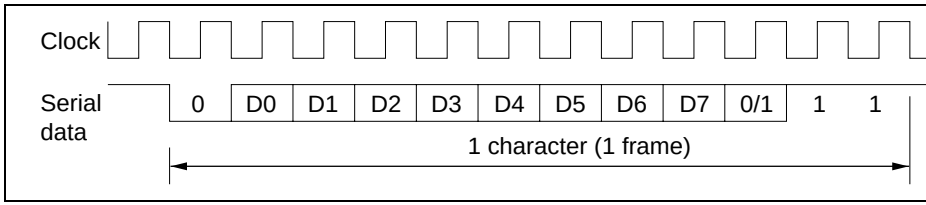
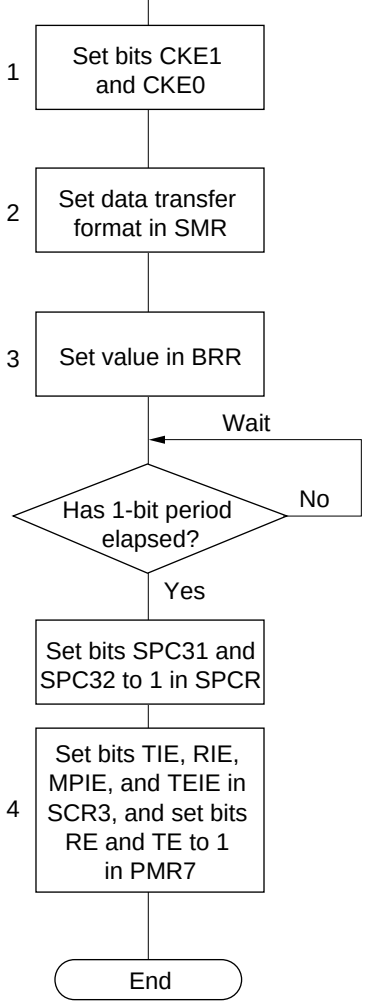


Figure 10.4 Phase Relationship between Output Clock and Transfer D (Asynchronous Mode) (8-bit data, parity, 2 stop bits)

3. Data Transfer Operations

SCI3 initialization: Before data is transferred on SCI3, bits TE and RE in SCR3 mus cleared to 0, and then SCI3 must be initialized as follows.

Note: If the operation mode or data transfer format is changed, bits TE and RE must cleared to 0.
When bit TE is cleared to 0, bit TDRE is set to 1.
Note that the RDRF, PER, FER, and OER flags and the contents of RDR are n when RE is cleared to 0.
When an external clock is used in asynchronous mode, the clock should not b during operation, including initialization. When an external clock is used in s mode, the clock should not be supplied during operation, including initializati



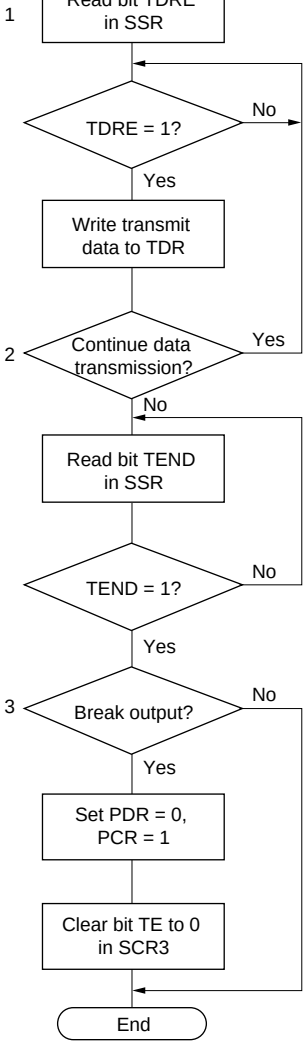
1. Set clock selection in SCR3. Be sure clear the other bits to 0. If clock outpu is selected in asynchronous mode, the clock is output immediately after settin bits CKE1 and CKE0. If clock output i selected for reception in synchronous mode, the clock is output immediately after bits CKE1, CKE0, and RE are set to 1.

2. Set the data transfer format in the seri mode register (SMR).

3. Write the value corresponding to the transfer rate in BRR. This operation is not necessary when an external clock is selected.

4. Wait for at least one bit period, then se bits TIE, RIE, MPIE, and TEIE in SCR and set bits RE and TE to 1 in PMR7. Setting bits TE and RE enables the TX and RXD3x pins to be used. In asynch mode the mark state is established wh transmitting, and the idle state waiting a start bit when receiving.

Figure 10.5 Example of SCI3 Initialization Flowchart



1. Read the serial status register (SSR) and check that bit TDRE is set to 1, then write transmit data to the transmit data register (TDR). When data is written to TDR, bit TDRE is cleared to 0 automatically.
   (After the TE bit is set to 1, one frame of 1s is output, then transmission is possible.)

2. When continuing data transmission, be sure to read TDRE = 1 to confirm that a write can be performed before writing data to TDR. When data is written to TDR, bit TDRE is cleared to 0 automatically.

3. If a break is to be output when data transmission ends, set the port PCR to 1 and clear the port PDR to 0, then clear bit TE in SCR3 to 0.

Figure 10.6 Example of Data Transmission Flowchart (Asynchronous M

next frame. If bit TDRE is set to 1, bit TEND in SSR bit is set to 1the mark state, in wh
transmitted, is established after the stop bit has been sent. If bit TEIE in SCR3 is set to
time, a TEI request is made.

Figure 10.12 shows an example of the operation when transmitting in asynchronous mo

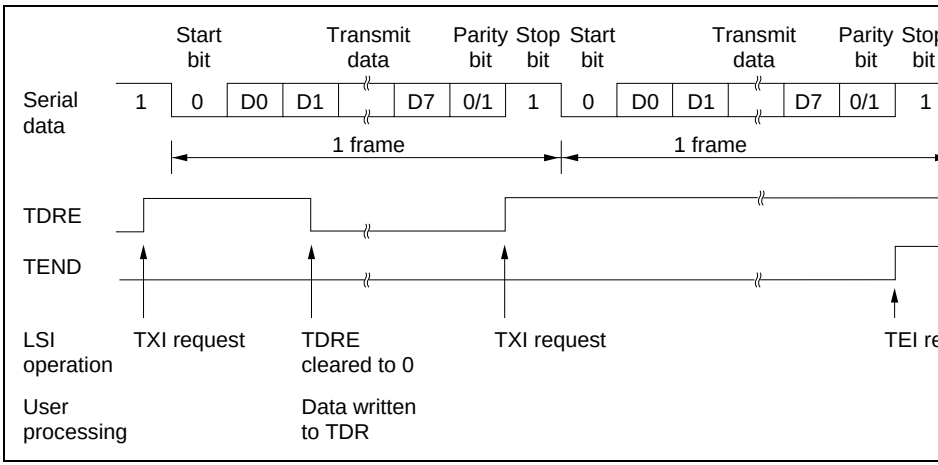
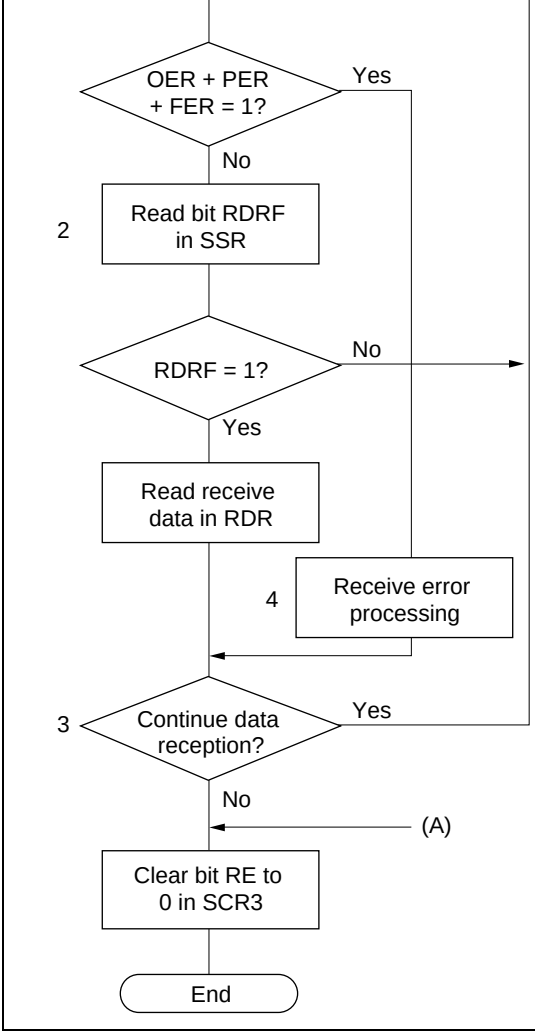


Figure 10.7 Example of Operation when Transmitting in Asynchronous M
(8-bit data, parity, 1 stop bit)

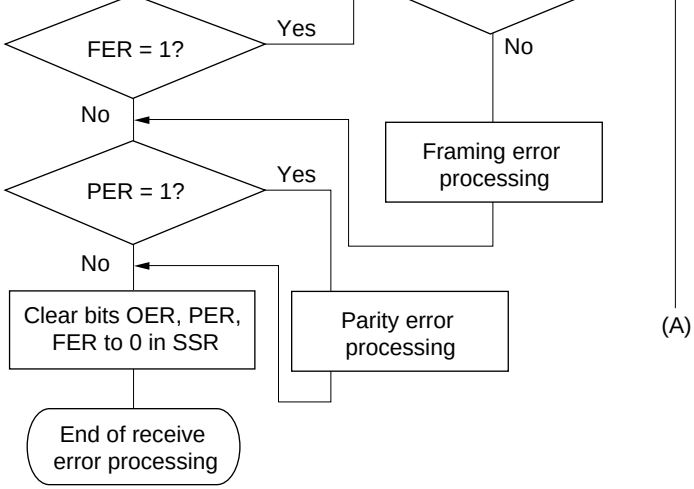


if there is an error. If a receive e
occurred, execute receive error
processing.

2. Read SSR and check that bit RD
set to 1. If it is, read the receive
in RDR. When the RDR data is
bit RDRF is cleared to 0 automa

3. When continuing data reception,
reading of bit RDRF and RDR be
receiving the stop bit of the curre
frame. When the data in RDR is
bit RDRF is cleared to 0 automa

Figure 10.8 Example of Data Reception Flowchart (Asynchronous Mo



bits is set to 1. In
of a framing error
can be detected b
the value of the R

Figure 10.8 Example of Data Reception Flowchart (Asynchronous Mode) (

set in bit PM in the serial mode register (SMR).

- Stop bit check

  SCI3 checks that the stop bit is 1. If two stop bits are used, only the first is checke

- Status check

  SCI3 checks that bit RDRF is set to 0, indicating that the receive data can be trans RSR to RDR.

If no receive error is found in the above checks, bit RDRF is set to 1, and the receive d in RDR. If bit RIE is set to 1 in SCR3, an RXI interrupt is requested. If the error chec a receive error, bit OER, PER, or FER is set to 1 depending on the kind of error. Bit R its state prior to receiving the data. If bit RIE is set to 1 in SCR3, an ERI interrupt is r

Table 10.12 shows the conditions for detecting a receive error, and receive data proces

Note: No further receive operations are possible while a receive error flag is set. Bit FER, PER, and RDRF must therefore be cleared to 0 before resuming receptio

Table 10.12 Receive Error Detection Conditions and Receive Data Processing

Receive Error	Abbr.	Detection Conditions	Receive Data Proc
Overrun error	OER	When the next date receive operation is completed while bit RDRF is still set to 1 in SSR	Receive data is not from RSR to RDR
Framing error	FER	When the stop bit is 0	Receive data is tra from RSR to RDR
Parity error	PER	When the parity (odd or even) set in SMR is different from that of the received data	Receive data is tra from RSR to RDR

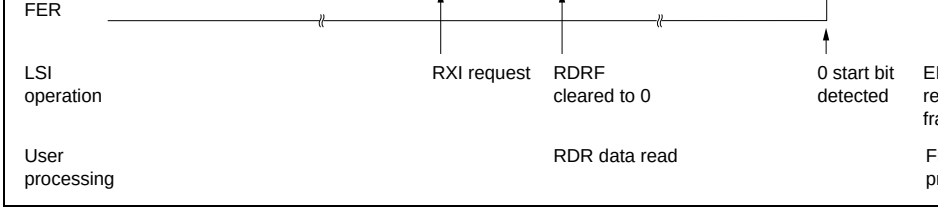


Figure 10.9 Example of Operation when Receiving in Asynchronous Mo (8-bit data, parity, 1 stop bit)

10.3.3 Operation in Synchronous Mode

In synchronous mode, SCI3 transmits and receives data in synchronization with clock p This mode is suitable for high-speed serial communication.

SCI3 has separate transmission and reception units, allowing full-duplex communicatio shared clock.

As the transmission and reception units are both double-buffered, data can be written d transmission and read during reception, making possible continuous transmission and r

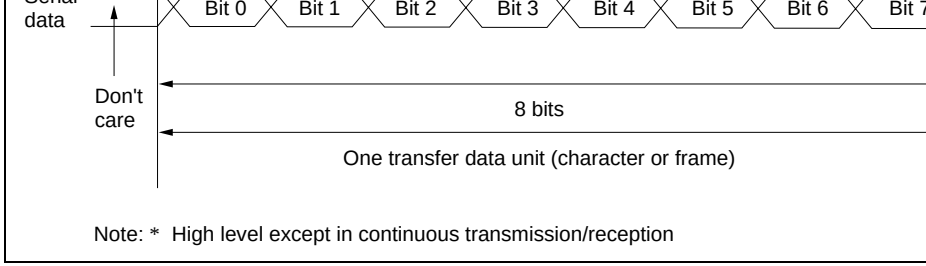


Figure 10.10 Data Format in Synchronous Communication

In synchronous communication, data on the communication line is output from one fa the serial clock until the next falling edge. Data confirmation is guaranteed at the risi the serial clock.

One transfer data character begins with the LSB and ends with the MSB. After outpu MSB, the communication line retains the MSB state.

When receiving in synchronous mode, SCI3 latches receive data at the rising edge of clock.

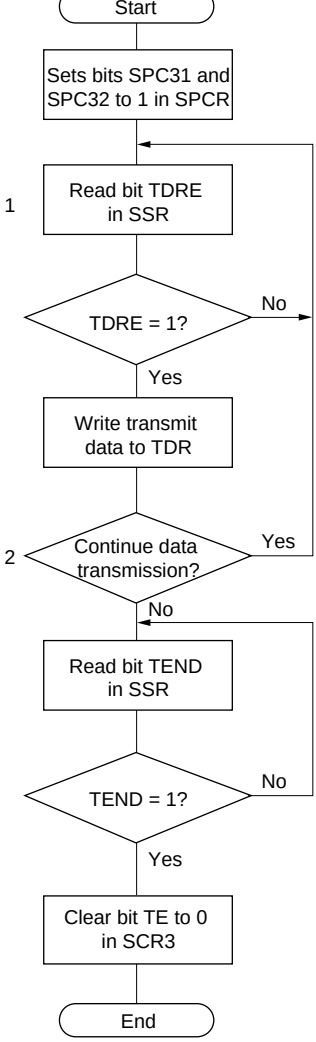
The data transfer format uses a fixed 8-bit data length.

Parity and multiprocessor bits cannot be added.

2. Clock

Either an internal clock generated by the baud rate generator or an external clock inpu SCK_{3x} pin can be selected as the SCI3 serial clock. The selection is made by means o SMR and bits CKE1 and CKE0 in SCR3. See table 10.9 for details on clock source se

When SCI3 operates on an internal clock, the serial clock is output at the SCK_{3x} pin. of the serial clock are output in transmission or reception of one character, and when S transmitting or receiving, the clock is fixed at the high level.



1. Read the serial status register (SSR) and check that bit TDRE is set to 1, then write transmit data to the transmit data register (TDR). When data is written to TDR, bit TDRE is cleared to 0 automatically, the clock is output, and data transmission is started. When clock output is selected, the clock is output and data transmission started when data is written to TDR.

2. When continuing data transmission, be sure to read TDRE = 1 to confirm that a write can be performed before writing data to TDR. When data is written to TDR, bit TDRE is cleared to 0 automatically.

Figure 10.11 Example of Data Transmission Flowchart (Synchronous Mo

Serial data is transmitted from the TXD3x pin in order from the LSB (bit 0) to the MS
When the MSB (bit 7) is sent, checks bit TDRE. If bit TDRE is cleared to 0, SCI3 tra
from TDR to TSR, and starts transmission of the next frame. If bit TDRE is set to 1, S
TEND to 1 in SSR, and after sending the MSB (bit 7), retains the MSB state. If bit TI
is set to 1 at this time, a TEI request is made.

After transmission ends, the SCK pin is fixed at the high level.

Note: Transmission is not possible if an error flag (OER, FER, or PER) that indicate
reception status is set to 1. Check that these error flags are all cleared to 0 be
transmit operation.

Figure 10.12 shows an example of the operation when transmitting in synchronous mo

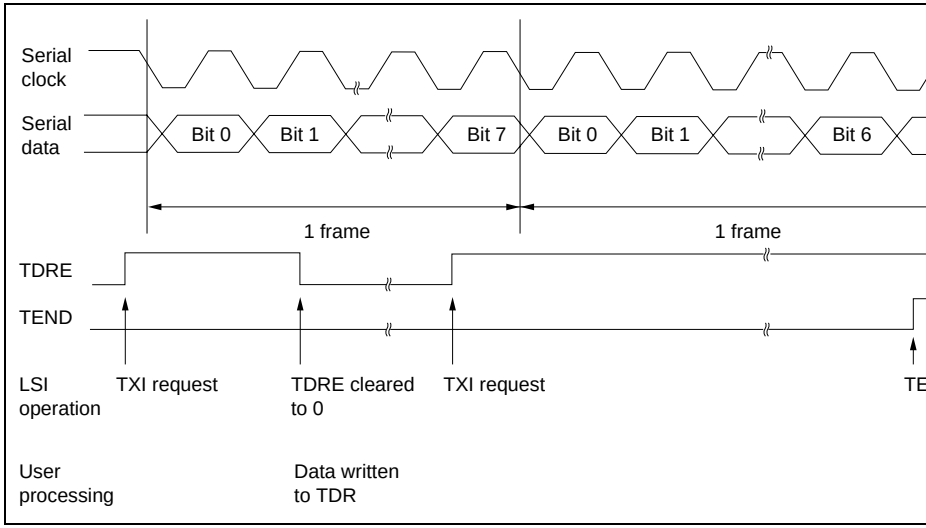
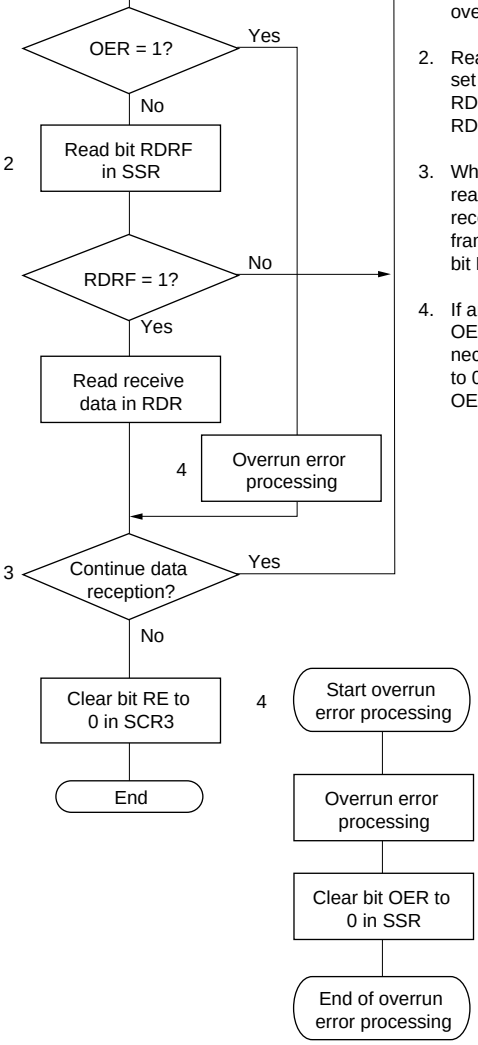


Figure 10.12 Example of Operation when Transmitting in Synchronous M



overrun error processing.

2. Read SSR and check that bit RDRF is set to 1. If it is, read the receive data in RDR. When the RDR data is read, bit RDRF is cleared to 0 automatically.

3. When continuing data reception, finish reading of bit RDRF and RDR before receiving the MSB (bit 7) of the current frame. When the data in RDR is read, bit RDRF is cleared to 0 automatically.

4. If an overrun error has occurred, read bit OER in SSR, and after carrying out the necessary error processing, clear bit OER to 0. Reception cannot be resumed if bit OER is set to 1.

Figure 10.13 Example of Data Reception Flowchart (Synchronous Mod

If this check shows that there is no overrun error, bit RDRF is set to 1, and the receive
stored in RDR. If bit RIE is set to 1 in SCR3, an RXI interrupt is requested. If the ch
identifies an overrun error, bit OER is set to 1.

Bit RDRF remains set to 1. If bit RIE is set to 1 in SCR3, an ERI interrupt is requeste

See table 10.12 for the conditions for detecting a receive error, and receive data proce

Note: No further receive operations are possible while a receive error flag is set. Bit
FER, PER, and RDRF must therefore be cleared to 0 before resuming receptio

Figure 10.14 shows an example of the operation when receiving in synchronous mode

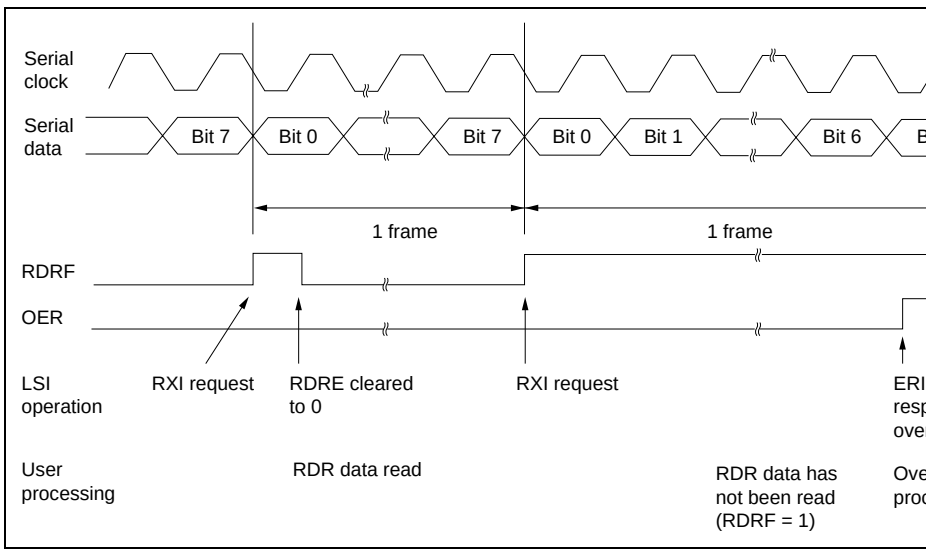
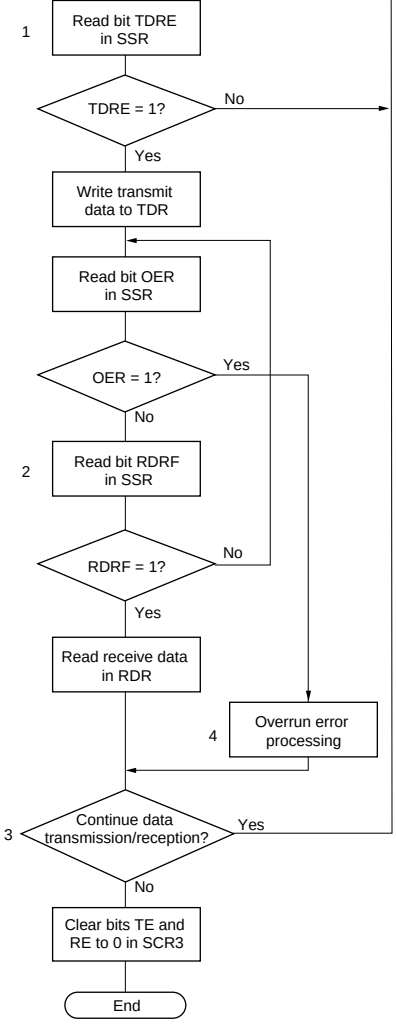


Figure 10.14 Example of Operation when Receiving in Synchronous M



1. Read the serial status register (SSR) and check that bit TDRE is set to 1, then write transmit data to the transmit data register (TDR). When data is written to TDR, bit TDRE is cleared to 0 automatically.
2. Read SSR and check that bit RDRF is set to 1. If it is, read the receive data in RDR. When the RDR data is read, bit RDRF is cleared to 0 automatically.
3. When continuing data transmission/reception, finish reading of bit RDRF and RDR before receiving the MSB (bit 7) of the current frame. Before receiving the MSB (bit 7) of the current frame, also read TDRE = 1 to confirm that a write can be performed, then write data to TDR. When data is written to TDR, bit TDRE is cleared to 0 automatically, and when the data in RDR is read, bit RDRF is cleared to 0 automatically.
4. If an overrun error has occurred, read bit OER in SSR, and after carrying out the necessary error processing, clear bit OER to 0. Transmission and reception cannot be resumed if bit OER is set to 1.
   See figure 10.13 for details on overrun error processing.

Figure 10.15 Example of Simultaneous Data Transmission/Reception Flow (Synchronous Mode)

10.5.4 Multiprocessor Communication Function

The multiprocessor communication function enables data to be exchanged among a nu processors on a shared communication line. Serial data communication is performed asynchronous mode using the multiprocessor format (in which a multiprocessor bit is transfer data).

In multiprocessor communication, each receiver is assigned its own ID code. The seri communication cycle consists of two cycles, an ID transmission cycle in which the rec specified, and a data transmission cycle in which the transfer data is sent to the specifi These two cycles are differentiated by means of the multiprocessor bit, 1 indicating an transmission cycle, and 0, a data transmission cycle.

The sender first sends transfer data with a 1 multiprocessor bit added to the ID code o it wants to communicate with, and then sends transfer data with a 0 multiprocessor bit transmit data. When a receiver receives transfer data with the multiprocessor bit set to compares the ID code with its own ID code, and if they are the same, receives the tran sent next. If the ID codes do not match, it skips the transfer data until data with the m bit set to 1 is sent again.

In this way, a number of processors can exchange data among themselves.

Figure 10.16 shows an example of communication between processors using the multi format.

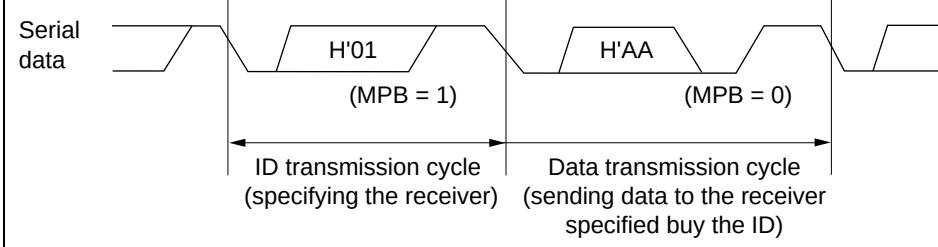
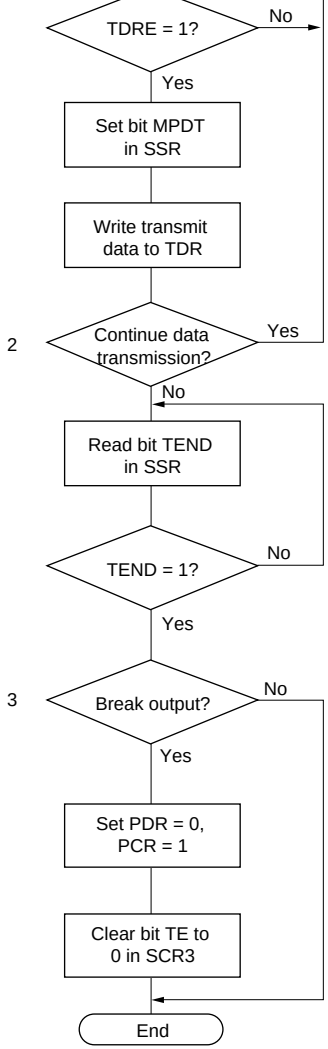


Figure 10.16 Example of Inter-Processor Communication Using Multiprocesso (Sending Data H'AA to Receiver A)

There is a choice of four data transfer formats. If a multiprocessor format is specified, bit specification is invalid. See table 10.11 for details.

For details on the clock used in multiprocessor communication, see section 10.3.2, Ope Asynchronous Mode.

Multiprocessor transmitting: Figure 10.17 shows an example of a flowchart for multi data transmission. This procedure should be followed for multiprocessor data transmis initializing SCI3.



register (TDR). When data is written to TDR, bit TDRE is cleared to 0 automatically.

2. When continuing data transmission, be sure to read TDRE = 1 to confirm that a write can be performed before writing data to TDR. When data is written to TDR, bit TDRE is cleared to 0 automatically.

3. If a break is to be output when data transmission ends, set the port PCR to 1 and clear the port PDR to 0, then clear bit TE in SCR3 to 0.

Figure 10.17 Example of Multiprocessor Data Transmission Flowcha

bit TDRE is set to 1 bit TEND in SSR bit is set to 1, the mark state, in which 1s are tra
established after the stop bit has been sent. If bit TEIE in SCR3 is set to 1 at this time,
request is made.

Figure 10.18 shows an example of the operation when transmitting using the multiproc
format.

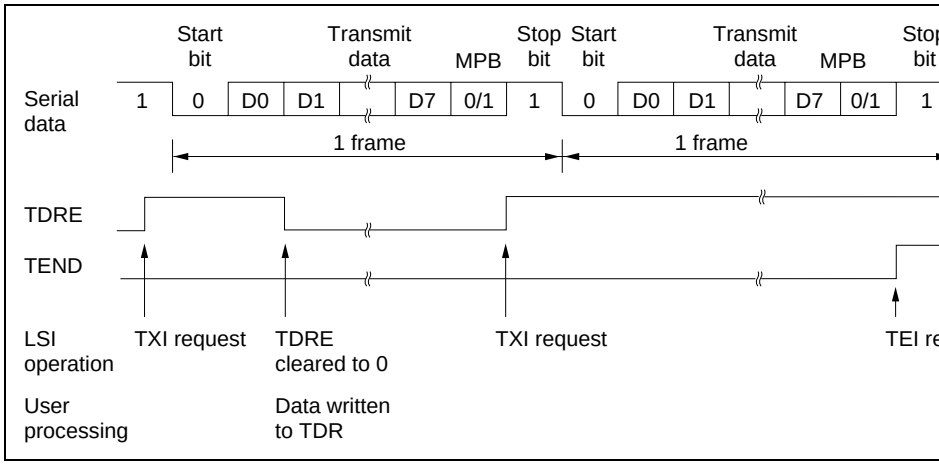
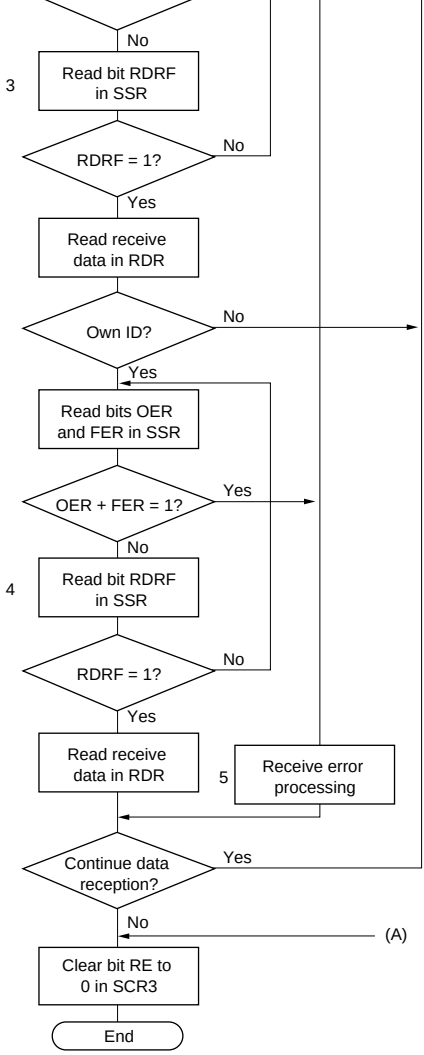


Figure 10.18 Example of Operation when Transmitting Using Multiprocessor (8-bit data, multiprocessor bit, 1 stop bit)

Multiprocessor receiving: Figure 10.19 shows an example of a flowchart for multipro
reception. This procedure should be followed for multiprocessor data reception after in
SCI3.



RDR and compare it with this receiver's own ID. If the ID is not this receiver's, set bit MPIE to 1 again. When the RDR data is read, bit RDRF is cleared to 0 automatically.

4. Read SSR and check that bit RDRF is set to 1, then read the data in RDR.

5. If a receive error has occurred, read bits OER and FER in SSR to identify the error, and after carrying out the necessary error processing, ensure that bits OER and FER are both cleared to 0. Reception cannot be resumed if either of these bits is set to 1. In the case of a framing error, a break can be detected by reading the value of the RXD_{3x} pin.

Figure 10.19 Example of Multiprocessor Data Reception Flowchart

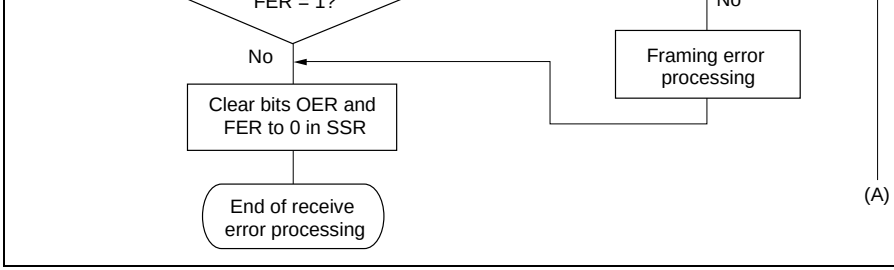
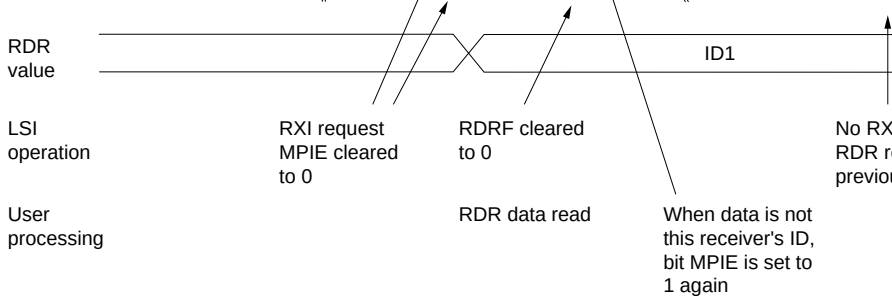
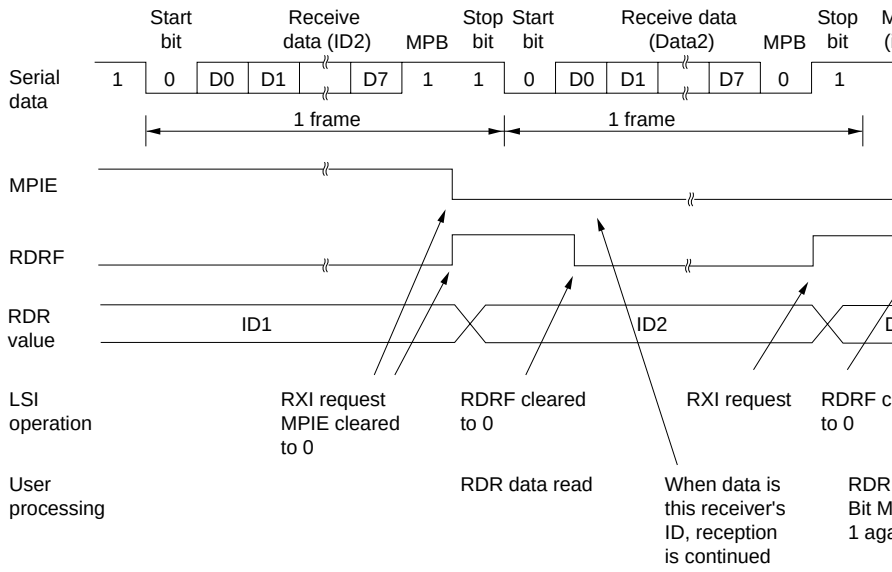


Figure 10.19 Example of Multiprocessor Data Reception Flowchart (cor

Figure 10.20 shows an example of the operation when receiving using the multiprocess



(a) When data does not match this receiver's ID



(b) When data matches this receiver's ID

Figure 10.20 Example of Operation when Receiving Using Multiprocessor
(8-bit data, multiprocessor bit, 1 stop bit)

Interrupt Abbr.	Interrupt Request	Vecto
RXI	Interrupt request initiated by receive data full flag (RDRF)	H'002
TXI	Interrupt request initiated by transmit data empty flag (TDRE)	
TEI	Interrupt request initiated by transmit end flag (TEND)	
ERI	Interrupt request initiated by receive error flag (OER, FER, PER)	

Each interrupt request can be enabled or disabled by means of bits TIE and RIE in SCR

When bit TDRE is set to 1 in SSR, a TXI interrupt is requested. When bit TEND is set SSR, a TEI interrupt is requested. These two interrupts are generated during transmissi

The initial value of bit TDRE in SSR is 1. Therefore, if the transmit data empty interru (TXI) is enabled by setting bit TIE to 1 in SCR3 before transmit data is transferred to T interrupt will be requested even if the transmit data is not ready.

Also, the initial value of bit TEND in SSR is 1. Therefore, if the transmit end interrupt (TEI) is enabled by setting bit TEIE to 1 in SCR3 before transmit data is transferred to TEI interrupt will be requested even if the transmit data has not been sent.

Effective use of these interrupt requests can be made by having processing that transfer data to TDR carried out in the interrupt service routine.

To prevent the generation of these interrupt requests (TXI and TEI), on the other hand, bits for these interrupt requests (bits TIE and TEIE) should be set to 1 after transmit dat transferred to TDR.

When bit RDRF is set to 1 in SSR, an RXI interrupt is requested, and if any of bits OEI FER is set to 1, an ERI interrupt is requested. These two interrupt requests are generate reception.

For further details, see section 3.3, Interrupts.

Data can be written to TDR irrespective of the state of bit TDRE, but if new data is wr
TDR while bit TDRE is cleared to 0, the data previously stored in TDR will be lost of
yet been transferred to TSR. Accordingly, to ensure that serial transmission is perform
dependably, you should first check that bit TDRE is set to 1, then write the transmit da
once only (not two or more times).

2. Operation when a Number of Receive Errors Occur Simultaneously

If a number of receive errors are detected simultaneously, the status flags in SSR will
states shown in table 10.14. If an overrun error is detected, data transfer from RSR to
not be performed, and the receive data will be lost.

Table 10.14 SSR Status Flag States and Receive Data Transfer

SSR Status Flags				Receive Data Transfer	
RDRF*	OER	FER	PER	RSR → RDR	Receive Error Status
1	1	0	0	X	Overrun error
0	0	1	0	O	Framing error
0	0	0	1	O	Parity error
1	1	1	0	X	Overrun error + framing error
1	1	0	1	X	Overrun error + parity error
0	0	1	1	O	Framing error + parity error
1	1	1	1	X	Overrun error + framing error

O : Receive data is transferred from RSR to RDR.

X : Receive data is not transferred from RSR to RDR.

Note: * Bit RDRF retains its state prior to data reception. However, note that if RDI
after an overrun error has occurred in a frame because reading of the recei
the previous frame was delayed, RDRF will be cleared to 0.

4. Mark State and Break Detection

When bit TE is cleared to 0, the TXD_{3x} pin functions as an I/O port whose input/output and level are determined by PDR and PCR. This fact can be used to set the TXD_{3x} pin state, or to detect a break during transmission.

To keep the communication line in the mark state (1 state) until bit TE is set to 1, set PC PDR = 1. Since bit TE is cleared to 0 at this time, the TXD_{3x} pin functions as an I/O po output.

To detect a break, clear bit TE to 0 after setting PCR = 1 and PDR = 0.

When bit TE is cleared to 0, the transmission unit is initialized regardless of the current transmission state, the TXD_{3x} pin functions as an I/O port, and 0 is output from the TXI

5. Receive Error Flags and Transmit Operation (Synchronous Mode Only)

When a receive error flag (OER, PER, or FER) is set to 1, transmission cannot be starte bit TDRE is cleared to 0. The receive error flags must be cleared to 0 before starting tr

Note also that receive error flags cannot be cleared to 0 even if bit RE is cleared to 0.

6. Receive Data Sampling Timing and Receive Margin in Asynchronous Mode

In asynchronous mode, SCI3 operates on a basic clock with a frequency 16 times the tr When receiving, SCI3 performs internal synchronization by sampling the falling edge o bit with the basic clock. Receive data is latched internally at the 8th rising edge of the b This is illustrated in figure 10.21.

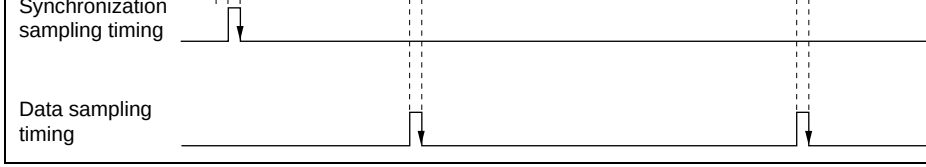


Figure 10.21 Receive Data Sampling Timing in Asynchronous Mode

Consequently, the receive margin in asynchronous mode can be expressed as shown in (1).

$$M = \{(0.5 - \frac{1}{2N}) - \frac{D - 0.5}{N} - (L - 0.5)\,F\} \times 100\ [\%] \quad \text{..... Equation (}$$

where M: Receive margin (%)
N: Ratio of bit rate to clock (N = 16)
D: Clock duty (D = 0.5 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute value of clock frequency deviation

Substituting 0 for F (absolute value of clock frequency deviation) and 0.5 for D (clock equation (1), a receive margin of 46.875% is given by equation (2).

When D = 0.5 and F = 0,

$$M = \{0.5 - 1/(2 \times 16)\} \times 100\ [\%]$$
$$= 46.875\% \quad \text{..... Equation (2)}$$

However, this is only a computed value, and a margin of 20% to 30% should be allow carrying out system design.

0, if the read operation coincides with completion of reception of a frame, the next fram
may be read. This is illustrated in figure 10.22.

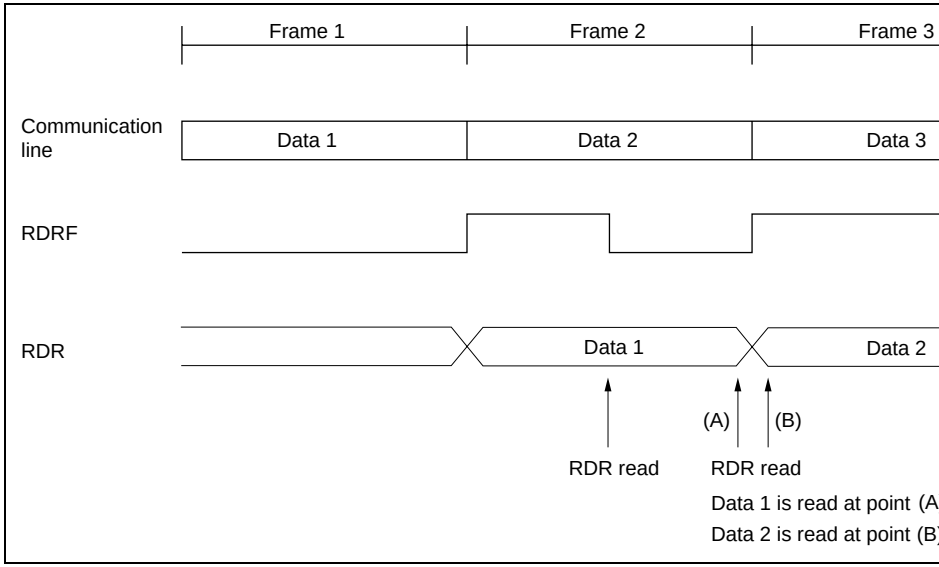


Figure 10.22 Relation between RDR Read Timing and Data

In this case, only a single RDR read operation (not two or more) should be performed a
checking that bit RDRF is set to 1. If two or more reads are performed, the data read th
should be transferred to RAM, etc., and the RAM contents used. Also, ensure that ther
sufficient margin in an RDR read operation before reception of the next frame is compl
be precise in terms of timing, the RDR read should be completed before bit 7 is transfe
synchronous mode, or before the STOP bit is transferred in asynchronous mode.

a system clock (ϕ) cycle immediately after it is switched.

This can be prevented by either of the following methods according to the situation.

a. When an SCK_{3x} 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 CKE1 and CKE0 in SCR3 to 1 and 0, respectively. In this case, bit COM in SMR left 1. The above prevents SCK_{3x} from being used as a general input/output pin. T intermediate level of voltage from being applied to SCK_{3x}, the line connected to S be pulled up to the V_{CC} level via a resistor, or supplied with output from an externa

b. When an SCK_{3x} function is switched from clock output to general input/output

   When stopping data transfer,

   (i) Issue one instruction to clear bits TE and RE to 0 and to set bits CKE1 and C SCR3 to 1 and 0, respectively.

   (ii) Clear bit COM in SMR to 0

   (iii) Clear bits CKE1 and CKE0 in SCR3 to 0

   Note that special care is also needed here to avoid an intermediate level of voltage applied to SCK_{3x}.

10. Setup at Subactive or Subsleep Mode

At subactive or subsleep mode, SCI3 becomes possible use only at CPU clock is ϕw/2

11.1.1 Features

Features of the 14-bit PWM are as follows.

- Choice of two conversion periods
  Any of the following four conversion periods can be chosen:
  — 131,072/ϕ, with a minimum modulation width of 8/ϕ (PWCR1 = 1, PWCR0 =
  — 65,536/ϕ, with a minimum modulation width of 4/ϕ (PWCR1 = 1, PWCR0 = 0
  — 32,768/ϕ, with a minimum modulation width of 2/ϕ (PWCR1 = 0, PWCR0 = 1
  — 16,384/ϕ, with a minimum modulation width of 1/ϕ (PWCR1 = 0, PWCR0 = 0
- Pulse division method for less ripple
- Use of module standby mode enables this module to be placed in standby mode in
  when not used.

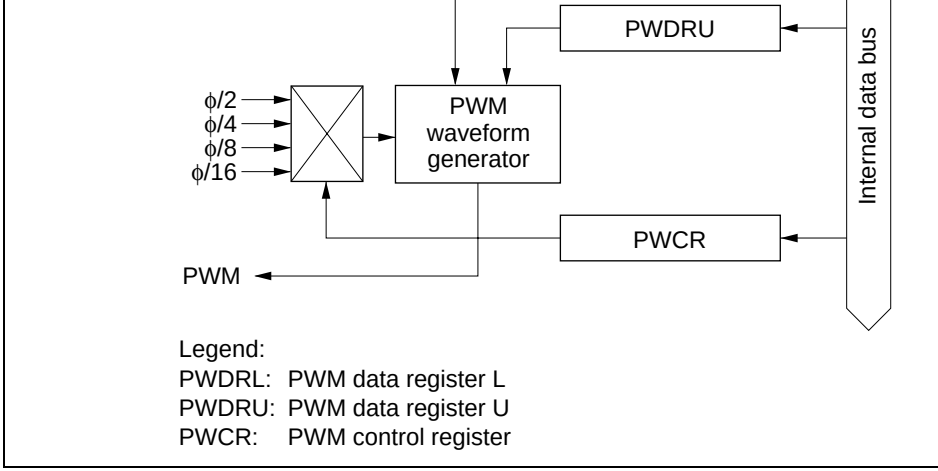


Figure 11.1 Block Diagram of the 14 Bit PWM

11.1.3 Pin Configuration

Table 11.1 shows the output pin assigned to the 14-bit PWM.

Table 11.1 Pin Configuration

Name	Abbr.	I/O	Function
PWM output pin	PWM	Output	Pulse-division PWM wavefor

PWM data register U	PWDRU	W	H'C0
PWM data register L	PWDRL	W	H'00
Clock stop register 2	CKSTPR2	R/W	H'FF

11.2 Register Descriptions

11.2.1 PWM Control Register (PWCR)

Bit	7	6	5	4	3	2	1
	—	—	—	—	—	—	PWCR1
Initial value	1	1	1	1	1	1	0
Read/Write	—	—	—	—	—	—	W

PWCR is an 8-bit write-only register for input clock selection.

Upon reset, PWCR is initialized to H'FC.

Bits 7 to 2: Reserved bits

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

		modulation width of 1/φ
0	1	The input clock is φ/4 (tφ* = 4/φ) The conversion period is 32,768/φ, with a minimum modulation width of 2/φ
1	0	The input clock is φ/8 (tφ* = 8/φ) The conversion period is 65,536/φ, with a minimum modulation width of 4/φ
1	1	The input clock is φ/16 (tφ* = 16/φ) The conversion period is 131,072/φ, with a minimum modulation width of 8/φ

Note: * Period of PWM input clock.

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

PWDRU

Bit	7	6	5	4	3	2	1
	—	—	PWDRU5	PWDRU4	PWDRU3	PWDRU2	PWDRU1
Initial value	1	1	0	0	0	0	0
Read/Write	—	—	W	W	W	W	W

PWDRL

Bit	7	6	5	4	3	2	1
	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

PWDRU and PWDRL form a 14-bit write-only register, with the upper 6 bits assigned and the lower 8 bits to PWDRL. The value written to PWDRU and PWDRL gives the level width of one PWM waveform cycle.

Upon reset, PWDRU and PWDRL are initialized to H'C000.

11.2.3 Clock Stop Register 2 (CKSTPR2)

Bit	7	6	5	4	3	2	1
	—	—	—	—	AECKSTP	WDCKSTP	PWCKST
Initial value	1	1	1	1	1	1	1
Read/Write	—	—	—	—	R/W	R/W	R/W

CKSTPR2 is an 8-bit read/write register that performs module standby mode control f
modules. Only the bit relating to the PWM is described here. For details of the other
sections on the relevant modules.

Bit 1: PWM module standby mode control (PWCKSTP)

Bit 1 controls setting and clearing of module standby mode for the PWM.

PWCKSTP	Description
0	PWM is set to module standby mode
1	PWM module standby mode is cleared

2. Set bits PWCR1 and PWCR0 in the PWM control register (PWCR) to select a conv period of 131,072/ϕ (PWCR1 = 1, PWCR0 = 1), 65,536/ϕ (PWCR1 = 1, PWCR0 = 32,768/ϕ (PWCR1 = 0, PWCR0 = 1), or 16,384/ϕ (PWCR1 = 0, PWCR0 = 0).
3. Set the output waveform data in PWM data registers U and L (PWDRU/L). Be sur the correct sequence, first PWDRL then PWDRU. When data is written to PWDRU in these registers will be latched in the PWM waveform generator, updating the PW waveform generation in synchronization with internal signals.

One conversion period consists of 64 pulses, as shown in figure 11.2. The total of the h pulse widths during this period (T_H) corresponds to the data in PWDRU and PWDRL. relation can be represented as follows.

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

where tϕ is the PWM input clock period: 2/ϕ (PWCR = H'0), 4/ϕ (PWCR = H'1), 8/ϕ (P H'2), or 16/ϕ (PWCR = H'3).

Example: Settings in order to obtain a conversion period of 32,768 µs:

When PWCR1 = 0 and PWCR0 = 0, the conversion period is 16,384/ϕ, so ϕ 0.5 MHz. In this case, tfn = 512 µs, with 1/ϕ (resolution) = 2.0 µs.

When PWCR1 = 0 and PWCR0 = 1, the conversion period is 32,768/ϕ, so ϕ MHz. In this case, tfn = 512 µs, with 2/ϕ (resolution) = 2.0 µs.

When PWCR1 = 1 and PWCR0 = 0, the conversion period is 65,536/ϕ , so ϕ MHz. In this case, tfn = 512 µs, with 4/ϕ (resolution) = 2.0 µs.

Accordingly, for a conversion period of 32,768 µs, the system clock frequen must be 0.5 MHz, 1 MHz, or 2 MHz.

$T_H = t_{H1} + t_{H2} + t_{H3} + \quad ..... \quad t_{H64}$

$t_{f1} = t_{f2} = t_{f3} \quad ..... \quad = t_{f64}$

Figure 11.2 PWM Output Waveform

11.3.2 PWM Operation Modes

PWM operation modes are shown in table 11.3.

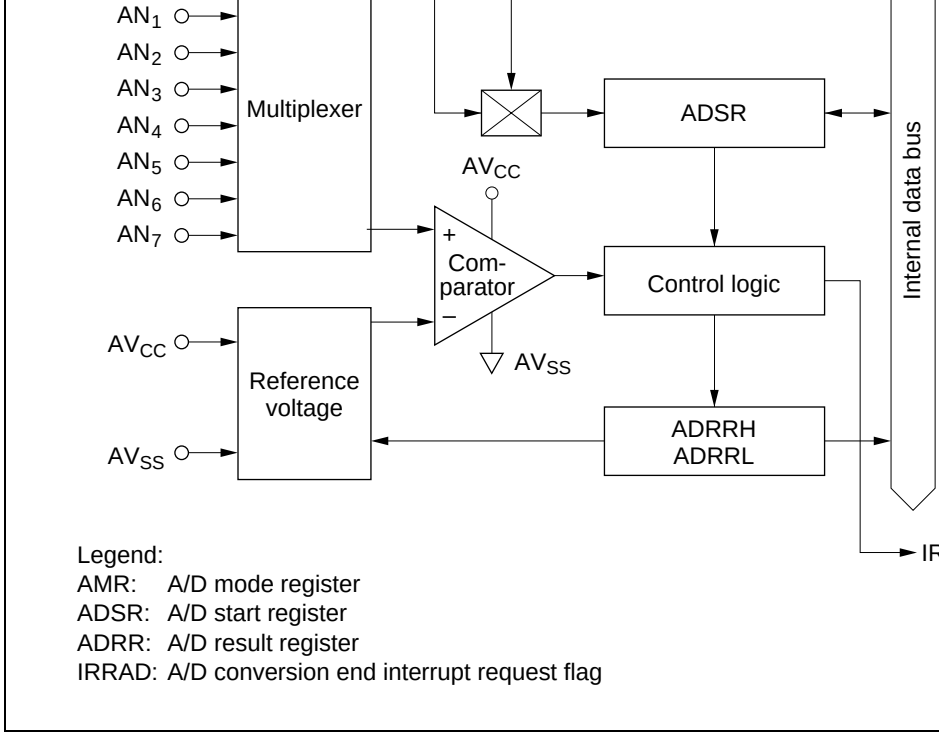
Table 11.3 PWM Operation Modes

Operation Mode	Reset	Active	Sleep	Watch	Subactive	Subsleep	Standby
PWCR	Reset	Functions	Functions	Held	Held	Held	Held
PWDRU	Reset	Functions	Functions	Held	Held	Held	Held
PWDRL	Reset	Functions	Functions	Held	Held	Held	Held

12.1.1 Features

The A/D converter has the following features.

- 10-bit resolution
- Eight 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
- Use of module standby mode enables this module to be placed in standby mode in
  when not used.



Legend:
AMR: A/D mode register
ADSR: A/D start register
ADRR: A/D result register
IRRAD: A/D conversion end interrupt request flag

Figure 12.1 Block Diagram of the A/D Converter

Analog ground	AV_{SS}	Input	Ground and reference voltage of analog
Analog input 0	AN_0	Input	Analog input channel 0
Analog input 1	AN_1	Input	Analog input channel 1
Analog input 2	AN_2	Input	Analog input channel 2
Analog input 3	AN_3	Input	Analog input channel 3
Analog input 4	AN_4	Input	Analog input channel 4
Analog input 5	AN_5	Input	Analog input channel 5
Analog input 6	AN_6	Input	Analog input channel 6
Analog input 7	AN_7	Input	Analog input channel 7
External trigger input	$\overline{ADTRG}$	Input	External trigger input for starting A/D co

12.1.4 Register Configuration

Table 12.2 shows the A/D converter register configuration.

Table 12.2 Register Configuration

Name	Abbr.	R/W	Initial Value
A/D mode register	AMR	R/W	H'30
A/D start register	ADSR	R/W	H'7F
A/D result register H	ADRRH	R	Not fixed
A/D result register L	ADRRL	R	Not fixed
Clock stop register 1	CKSTPRT1	R/W	H'FF

Read/Write	R	R	R	R	R	R	R	R	R	R	—	—	—	—
	ADRRH								ADRRL					

ADRRH and ADRRL together comprise a 16-bit read-only register for holding the resu
analog-to-digital conversion. The upper 8 bits of the data are held in ADRRH, and the
bits in ADRRL.

ADRRH and ADRRL can be read by the CPU at any time, but the ADRRH and ADRR
during A/D conversion are not fixed. After A/D conversion is complete, the conversion
stored as 10-bit data, and this data is held until the next conversion operation starts.

ADRRH and ADRRL are not cleared on reset.

12.2.2 A/D Mode Register (AMR)

Bit	7	6	5	4	3	2	1
	CKS	TRGE	—	—	CH3	CH2	CH1
Initial value	0	0	1	1	0	0	0
Read/Write	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 t
option, and the analog input pins.

Upon reset, AMR is initialized to H'30.

Note: * For information on conversion time settings for which operation is guarante
section 15, Electrical Characteristics.

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
1	Enables start of A/D conversion by rising or falling edge of external trigge $\overline{\text{ADTRG}}$*

Note: * The external trigger ($\overline{\text{ADTRG}}$) edge is selected by bit IEG4 of IEGR. See 1
select register (IEGR) in section 3.3.2 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.

1	0	0	0	AN4
1	0	0	1	AN5
1	0	1	0	AN6
1	0	1	1	AN7
1	1	*	*	Setting prohibited

Note: * Don't care

12.2.3 A/D Start Register (ADSR)

Bit	7	6	5	4	3	2	1
	ADSF	—	—	—	—	—	—
Initial value	0	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 conversion.

A/D conversion is started by writing 1 to the A/D start flag (ADSF) or by input of the d edge of the external trigger signal, which also sets ADSF to 1. When conversion is cor converted data is set in ADRRH and ADRRL, and at the same time ADSF is cleared to

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.

12.2.4 Clock Stop Register 1 (CKSTPR1)

Bit	7	6	5	4	3	2	1
	—	S31CKSTP	S32CKSTP	ADCKSTP	TGCKSTP	TFCKSTP	TCCKST
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CKSTPR1 is an 8-bit read/write register that performs module standby mode control f
modules. Only the bit relating to the A/D converter is described here. For details of t
see the sections on the relevant modules.

Bit 4: A/D converter module standby mode control (ADCKSTP)

Bit 4 controls setting and clearing of module standby mode for the A/D converter.

ADCKSTP	Description
0	A/D converter is set to module standby mode
1	A/D converter module standby mode is cleared

value of 1 during A/D conversion, and is cleared to 0 automatically when conversion is

The completion of conversion also sets bit IRRAD in interrupt request register 2 (IRR2 A/D conversion end interrupt is requested if bit IENAD in interrupt enable register 2 (I set to 1.

If the conversion time or input channel needs to be changed in the A/D mode register (A during A/D conversion, bit ADSF should first be cleared to 0, stopping the conversion 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 External trigger input is enabled at pin $\overline{\text{ADTRG}}$ when bit IRQ4 in PMR1 is set to 1 and in AMR is set to 1. Then when the input signal edge designated in bit IEG4 of interrup select register (IEGR) is detected at pin $\overline{\text{ADTRG}}$, bit ADSF in ADSR will be set to 1, s conversion.

Figure 12.2 shows the timing.

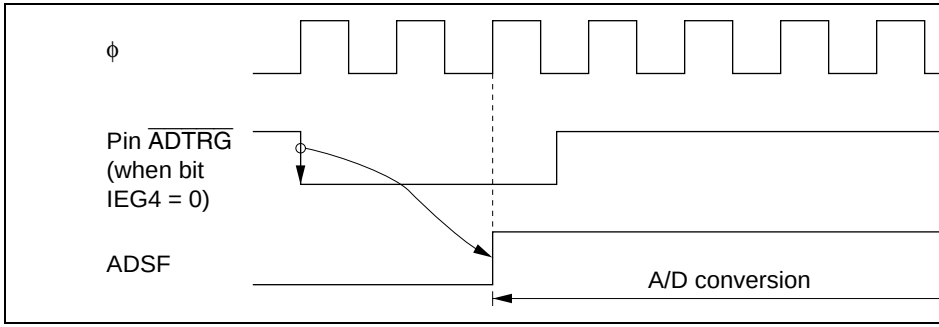


Figure 12.2 External Trigger Input Timing

ADSR	Reset	Functions	Functions	Held	Held	Held	Held
ADRRH	Held*	Functions	Functions	Held	Held	Held	Held
ADRRL	Held*	Functions	Functions	Held	Held	Held	Held

Note: * Undefined in a power-on reset.

12.4 Interrupts

When A/D conversion ends (ADSF changes from 1 to 0), bit IRRAD in interrupt requ
(IRR2) is set to 1.

A/D conversion end interrupts can be enabled or disabled by means of bit IENAD in i
enable register 2 (IENR2).

For further details see section 3.3, Interrupts.

2. When A/D conversion is complete, bit IRRAD is set to 1, and the A/D conversion r stored is stored in ADRRH and ADRRL. At the same time ADSF is cleared to 0, a converter goes to the idle state.
3. Bit IENAD = 1, so an A/D conversion end interrupt is requested.
4. The A/D interrupt handling routine starts.
5. The A/D conversion result is read and processed.
6. The A/D interrupt handling routine ends.

If ADSF is set to 1 again afterward, A/D conversion starts and steps 2 to 6 take place.

Figures 12.4 and 12.5 show flow charts of procedures for using the A/D converter.

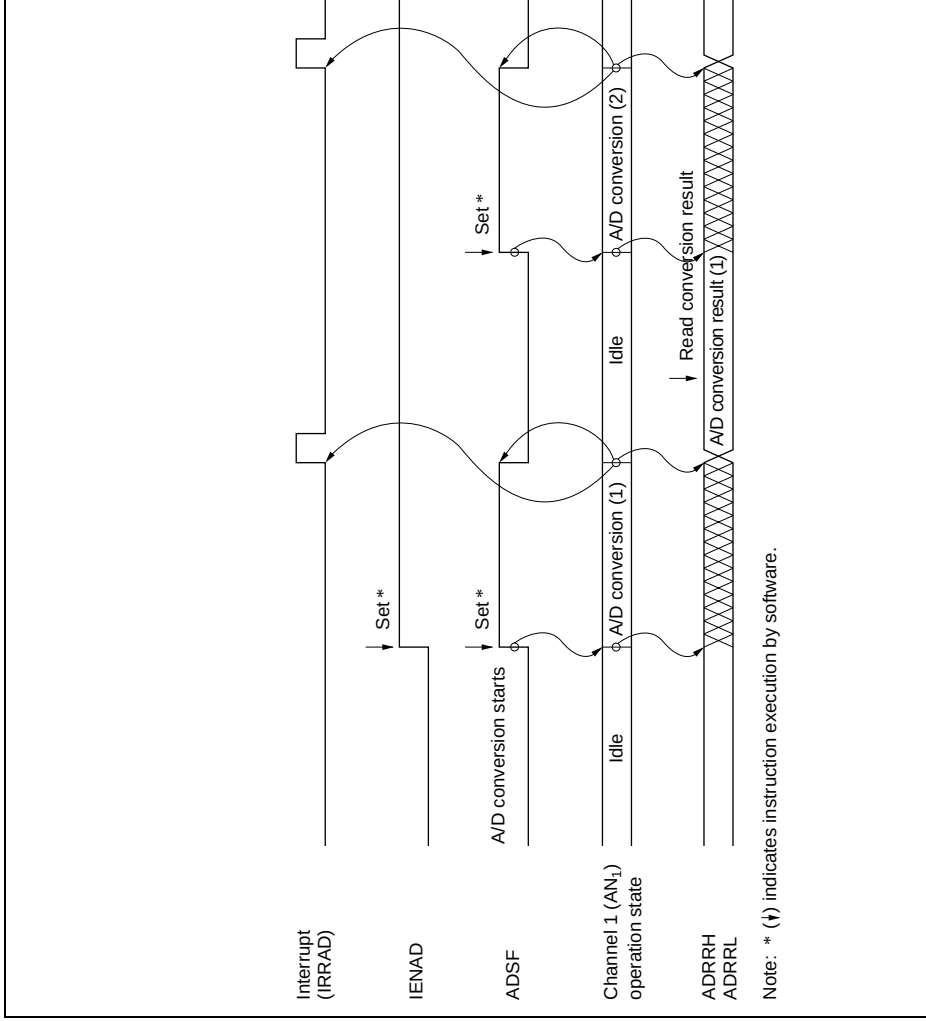


Figure 12.3 Typical A/D Converter Operation Timing

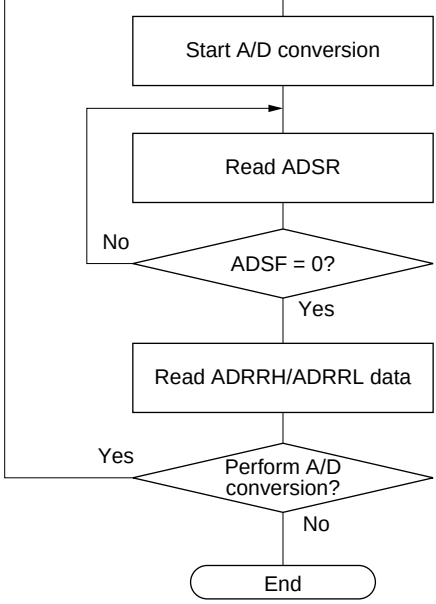


Figure 12.4 Flow Chart of Procedure for Using A/D Converter (Polling by So

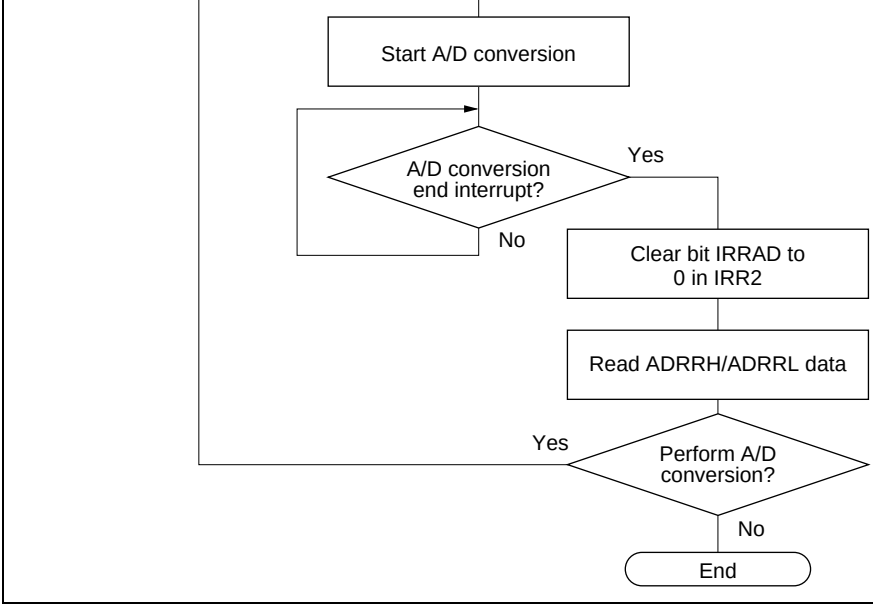


Figure 12.5 Flow Chart of Procedure for Using A/D Converter (Interrupts

affect conversion accuracy.

- When A/D conversion is started after clearing module standby mode, wait for 10 ϕ cycles before starting.
- In active mode or sleep mode, analog power supply current (AI_{STOP1}) flows into the resistance even when the A/D converter is not operating. Therefore, if the A/D con used, it is recommended that AV_{CC} be connected to the system power supply and th ADCKSTP(A/D converter module standby mode control) bit be cleared to 0 in cloc register 1 (CKSTPR1).

12.6.2 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion precision is guaranteed for an signal for which the signal source impedance is 10 kΩ or less. This specification is pro enable the A/D converter's sample-and-hold circuit input capacitance to be charged wit sampling time; if the sensor output impedance exceeds 10 kΩ, charging may be insuffi may not be possible to guarantee A/D conversion precision. However, a large capacitar provided externally, the input load will essentially comprise only the internal input resi 10 kΩ, and the signal source impedance is ignored. However, as a low-pass filter effect in this case, it may not be possible to follow an analog signal with a large differential c (e.g., 5 mV/μs or greater) (see figure 12.6). When converting a high-speed analog signa impedance buffer should be inserted.

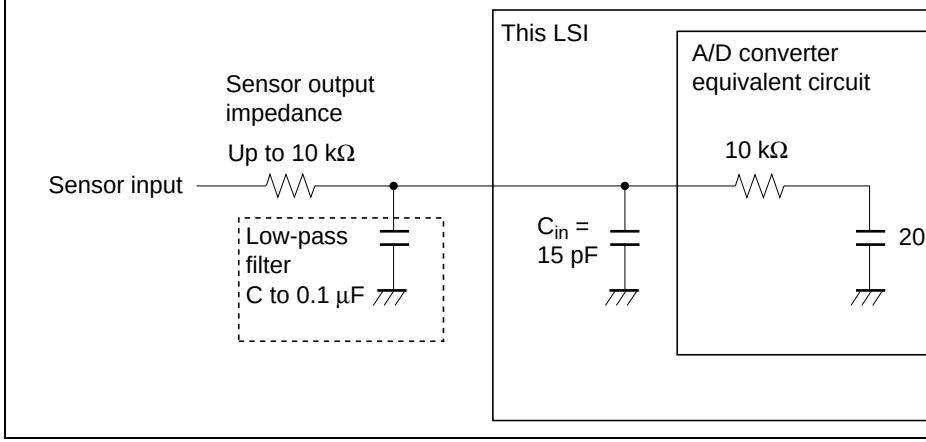


Figure 12.6 Analog Input Circuit Example

13.1.1 Features

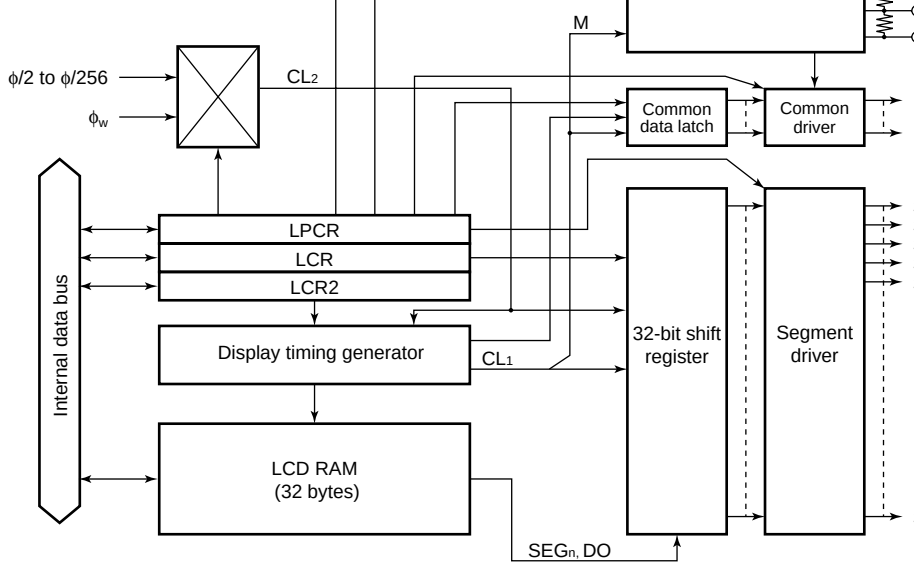
1. Features

Features of the LCD controller/driver are given below.

- Display capacity

Duty Cycle	Internal Driver	Segment External Expansion Driver
Static	32 seg	256 seg
1/2	32 seg	128 seg
1/3	32 seg	64 seg
1/4	32 seg	64 seg

- LCD RAM capacity
  8 bits × 32 bytes (256 bits)
- Word access to LCD RAM
- All eight segment output pins can be used individually as port pins.
- Common output pins not used because of the duty cycle can be used for common o
  buffering (parallel connection).
- Display possible in operating modes other than standby mode
- Choice of 11 frame frequencies
- Built-in power supply split-resistance, supplying LCD drive power
- Use of module standby mode enables this module to be placed in standby mode in
  when not used.
- A or B waveform selectable by software



Legend:
LPCR: LCD port control register
LCR: LCD control register
LCR2: LCD control register 2

Note: * The external expansion function for LCD segments is not implemented in the H8/38327 Group and H8/38427 Group.

Figure 13.1 Block Diagram of LCD Controller/Driver

			All pins are multiplexed as port (setting programmable)
Common output pins	COM_4 to COM_1	Output	LCD common drive pins Pins can be used in parallel w 1/2 duty
Segment external expansion signal pins*	CL_1	Output	Display data latch clock, multip SEG_{32}
	CL_2	Output	Display data shift clock, multip SEG_{31}
	M	Output	LCD alternation signal, multipl SEG_{29}
	DO	Output	Serial display data, multiplexe
LCD power supply pins	V_0, V_1, V_2, V_3	—	Used when a bypass capacito connected externally, and whe external power supply circuit is

Note: * The external expansion function for LCD segments is not implemented in th Group and H8/38427 Group.

13.1.4 Register Configuration

Table 13.2 shows the register configuration of the LCD controller/driver.

Table 13.2 LCD Controller/Driver Registers

Name	Abbr.	R/W	Initial Value	Add
LCD port control register	LPCR	R/W	H'00	H'FF
LCD control register	LCR	R/W	H'80	H'FF
LCD control register 2	LCR2	R/W	H'60	H'FF
LCD RAM	—	R/W	Undefined	H'F7
Clock stop register 2	CKSTPR2	R/W	H'FF	H'FF

Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

LPCR is an 8-bit read/write register which selects the duty cycle and LCD driver pin fu

LPCR is initialized to H'00 upon reset.

Bits 7 to 5: Duty cycle select 1 and 0 (DTS1, DTS0), common function select (CMX)

The combination of DTS1 and DTS0 selects static, 1/2, 1/3, or 1/4 duty. CMX specifie
or not the same waveform is to be output from multiple pins to increase the common dr
when not all common pins are used because of the duty setting.

Bit 7 DTS1	Bit 6 DTS0	Bit 5 CMX	Duty Cycle	Common Drivers	Notes
0	0	0	Static	COM_1 (initial value)	Do not use COM_4, COM COM_2.
		1		COM_4 to COM_1	COM_4, COM_3, and COM the same waveform as
0	1	0	1/2 duty	COM_2 to COM_1	Do not use COM_4 and C
		1		COM_4 to COM_1	COM_4 outputs the same as COM_3, and COM_2 ou same waveform as COM
1	0	0	1/3 duty	COM_3 to COM_1	Do not use COM_4.
		1		COM_4 to COM_1	Do not use COM_4.
1	1	0	1/4 duty	COM_4 to COM_1	—
		1			

Bit 4 SGX	Description
0	Pins SEG_{32} to SEG_{29}*
1	Pins CL_1, CL_2, DO, M

Note: * These pins function as ports when the setting of SGS3 to SGS0 is 0000 or

Bits 3 to 0: Segment driver select 3 to 0 (SGS3 to SGS0)

Bits 3 to 0 select the segment drivers to be used. The SGX = 0 setting is selected on th
and H8/38427.

					Function of Pins SEG_{32} to SEG_1			
Bit 4 SGX	**Bit 3 SGS3**	**Bit 2 SGS2**	**Bit 1 SGS1**	**Bit 0 SGS0**	**SEG_{32} to SEG_{25}**	**SEG_{24} to SEG_{17}**	**SEG_{16} to SEG_9**	**SEG_8 to SEG_1**
0	0	0	0	0	Port	Port	Port	Port
	0	0	0	1	Port	Port	Port	Port
	0	0	1	*	SEG	Port	Port	Port
	0	1	0	*	SEG	SEG	Port	Port
	0	1	1	*	SEG	SEG	SEG	Port
	1	*	*	*	SEG	SEG	SEG	SEG
1	0	0	0	0	Port*1	Port	Port	Port
	*	*	*	*	Setting prohibited			

Note: 1. SEG_{32} to SEG_{29} are external expansion pins.

LCR is an 8-bit read/write register which performs LCD drive power supply on/off con
display data control, and selects the frame frequency.

LCR is initialized to H'80 upon reset.

Bit 7: Reserved bit

Bit 7 is reserved; it is always read as 1 and cannot be modified.

Bit 6: LCD drive power supply on/off control (PSW)

Bit 6 can be used to turn the LCD drive power supply off when LCD display is not requ
power-down mode, or when an external power supply is used. When the ACT bit is cle
or in standby mode, the LCD drive power supply is turned off regardless of the setting

Bit 6 PSW	Description	
0	LCD drive power supply off	(i
1	LCD drive power supply on	

Bit 5: Display function activate (ACT)

Bit 5 specifies whether or not the LCD controller/driver is used. Clearing this bit to 0 h
operation of the LCD controller/driver. The LCD drive power supply is also turned off
of the setting of the PSW bit. However, register contents are retained.

Bit 5 ACT	Description	
0	LCD controller/driver operation halted	(i
1	LCD controller/driver operates	

Bits 3 to 0: Frame frequency select 3 to 0 (CKS3 to CKS0)

Bits 3 to 0 select the operating clock and the frame frequency. In subactive mode, wa and subsleep mode, the system clock (ϕ) is halted, and therefore display operations are performed if one of the clocks from ϕ/2 to ϕ/256 is selected. If LCD display is require modes, ϕw, ϕw/2, or ϕw/4 must be selected as the operating clock.

Bit 3 CKS3	Bit 2 CKS2	Bit 1 CKS1	Bit 0 CKS0	Operating Clock	Frame Frequen ϕ = 2 MHz	ϕ = 25
0	*	0	0	ϕw	128 Hz[*3] (initial value)	
0	*	0	1	ϕw/2	64 Hz[*3]	
0	*	1	*	ϕw/4	32 Hz[*3]	
1	0	0	0	ϕ/2	—	244 H
1	0	0	1	ϕ/4	977 Hz	122 H
1	0	1	0	ϕ/8	488 Hz	61 Hz
1	0	1	1	ϕ/16	244 Hz	30.5 H
1	1	0	0	ϕ/32	122 Hz	—
1	1	0	1	ϕ/64	61 Hz	—
1	1	1	0	ϕ/128	30.5 Hz	—
1	1	1	1	ϕ/256	—	—

Notes: 1. This is the frame frequency in active (medium-speed, ϕosc/16) mode when
2. When 1/3 duty is selected, the frame frequency is 4/3 times the value show
3. This is the frame frequency when ϕw = 32.768 kHz.

waveform, and selects the duty cycle of the charge/discharge pulses which control disc
of the power supply split-resistance from the power supply circuit.

LCR2 is initialized to H'60 upon reset.

Bit 7: A waveform/B waveform switching control (LCDAB)

Bit 7 specifies whether the A waveform or B waveform is used as the LCD drive wave

Bit 7 LCDAB	Description	
0	Drive using A waveform	(i
1	Drive using B waveform	

Bits 6 and 5: Reserved bits

Bits 6 and 5 are reserved; they are always read as 1 and cannot be modified.

Bit 4: Reserved bit

Bit 4 is reserved; it is always read as 0 and must not be written with 1.

0	1	0	0	4/8	
0	1	0	1	5/8	
0	1	1	0	6/8	
0	1	1	1	0	Fixed low
1	0	*	*	1/16	
1	1	*	*	1/32	

Bits 3 to 0 select the duty cycle while the power supply split-resistance is connected to
supply circuit.

When a 0 duty cycle is selected, the power supply split-resistance is permanently disc
from the power supply circuit, so power should be supplied to pins V_1, V_2, and V_3 by
circuit.

Figure 13.2 shows the waveform of the charge/discharge pulses. The duty cycle is Tc

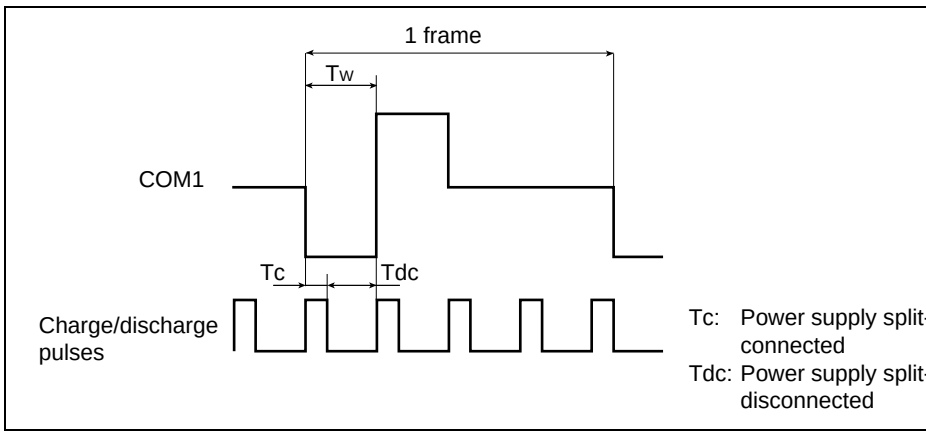


Figure 13.2 Example of A Waveform with 1/2 Duty and 1/2 Bias

modules. Only the bit relating to the LCD controller/driver is described here. For deta
other bits, see the sections on the relevant modules.

Bit 0: LCD controller/driver module standby mode control (LDCKSTP)

Bit 0 controls setting and clearing of module standby mode for the LCD controller/driv

Bit 0 LDCKSTP	Description	
0	LCD controller/driver is set to module standby mode	
1	LCD controller/driver module standby mode is cleared	(i

a. Using 1/2 duty

When 1/2 duty is used, interconnect pins V_2 and V_3 as shown in figure 13.3.

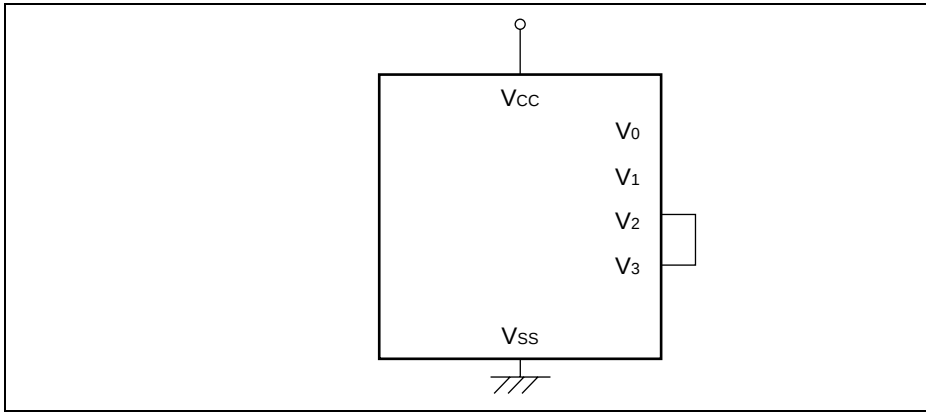


Figure 13.3 Handling of LCD Drive Power Supply when Using 1/2 Du

b. Large-panel display

As the impedance of the built-in power supply split-resistance is large, it may not b
for driving a large panel. If the display lacks sharpness when using a large panel,
section 13.3.6, Boosting the LCD Drive Power Supply. When static or 1/2 duty is
common output drive capability can be increased. Set CMX to 1 when selecting th
cycle. In this mode, with a static duty cycle pins COM_4 to COM_1 output the same
and with 1/2 duty the COM_1 waveform is output from pins COM_2 and COM_1, and
waveform is output from pins COM_4 and COM_3.

c. Luminance adjustment function (V_0 pin)

Connecting a resistance between the V_0 and V_1 pins enables the luminance to be a
details, see section 13.3.3, Luminance Adjustment Function (V_0 Pin).

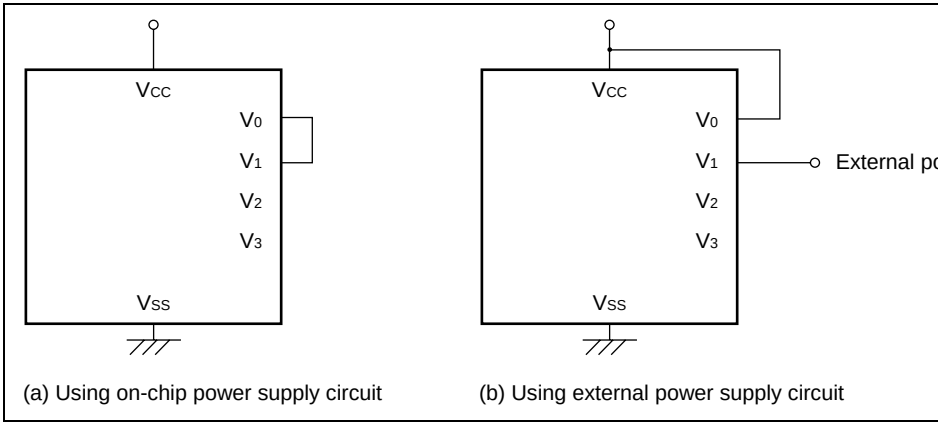


Figure 13.4 Examples of LCD Power Supply Pin Connections

e. Low-power-consumption LCD drive system

Use of a low-power-consumption LCD drive system enables the power consumptio for LCD drive to be optimized. For details, see section 13.3.4, Low-Power-Consum Drive System.

f. Segment external expansion

The number of segments can be increased by connecting an HD66100 externally. F see section 13.3.7, Connection to HD66100.

c. Frame frequency selection

The frame frequency can be selected by setting bits CKS_3 to CKS_0. The frame fre
should be selected in accordance with the LCD panel specification. For the clock
method in watch mode, subactive mode, and subsleep mode, see section 13.3.5, O
Power-Down Modes.

d. A or B waveform selection

Either the A or B waveform can be selected as the LCD waveform to be used by m
LCDAB.

using the same kind of instruction as for ordinary RAM, and display is started automati
turned on. Word- or byte-access instructions can be used for RAM setting.

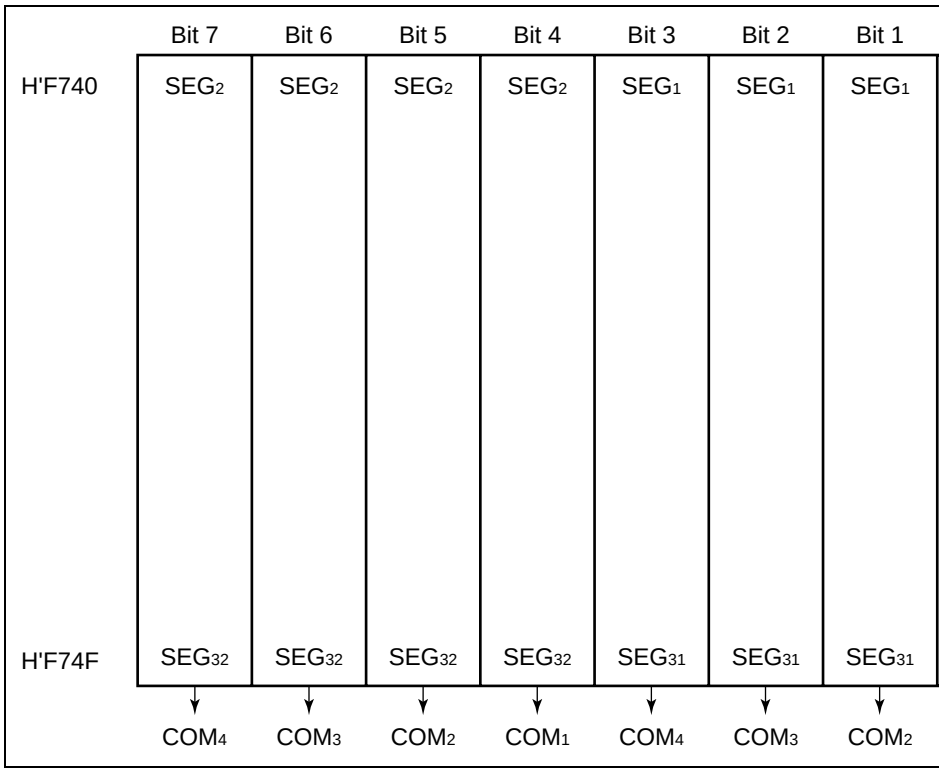


Figure 13.5 LCD RAM Map when Not Using Segment External Expansion (1/

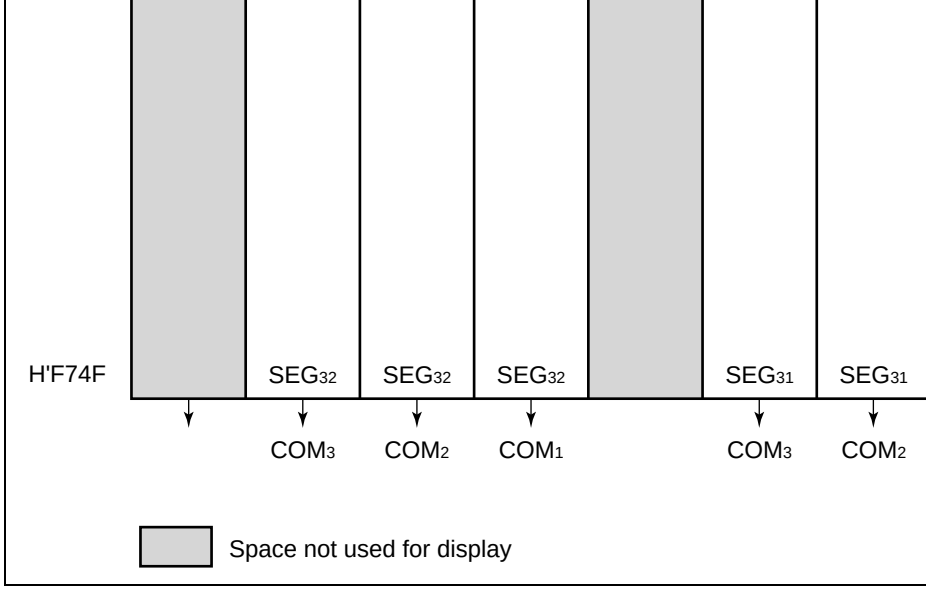


Figure 13.6 LCD RAM Map when Not Using Segment External Expansion (

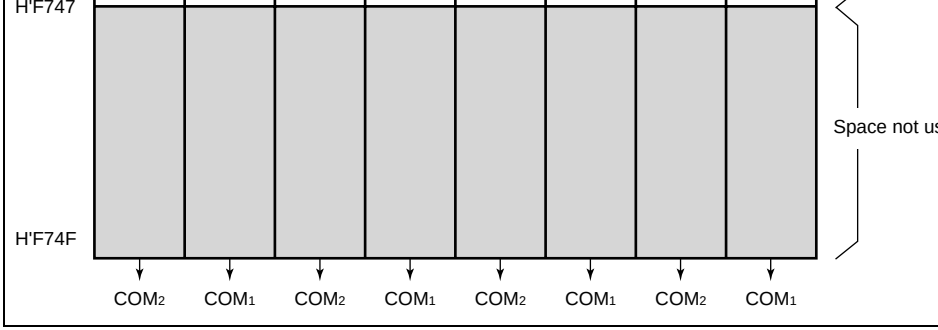


Figure 13.7 LCD RAM Map when Not Using Segment External Expansion (1/

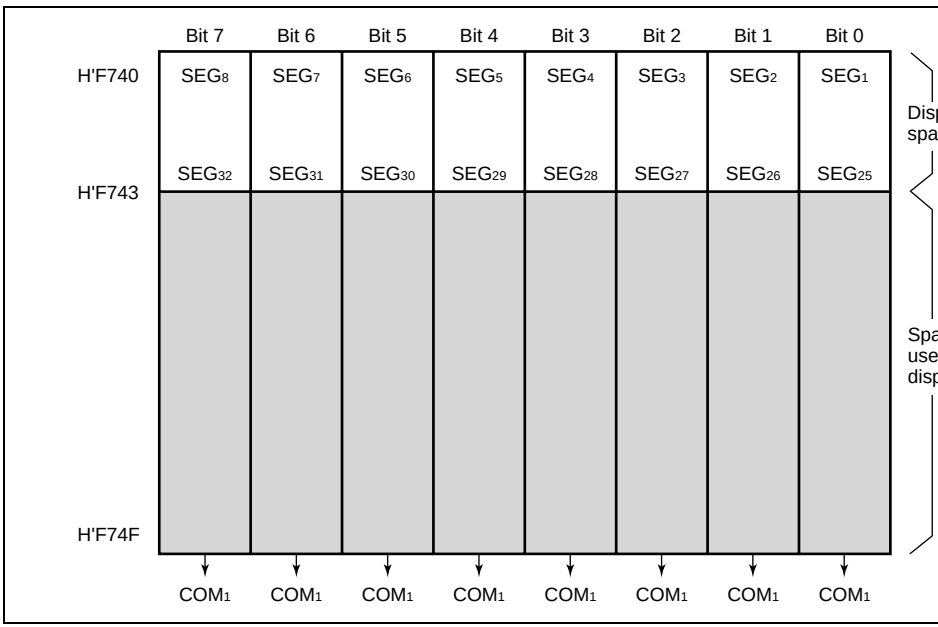


Figure 13.8 LCD RAM Map when Not Using Segment External Expansion (Sta

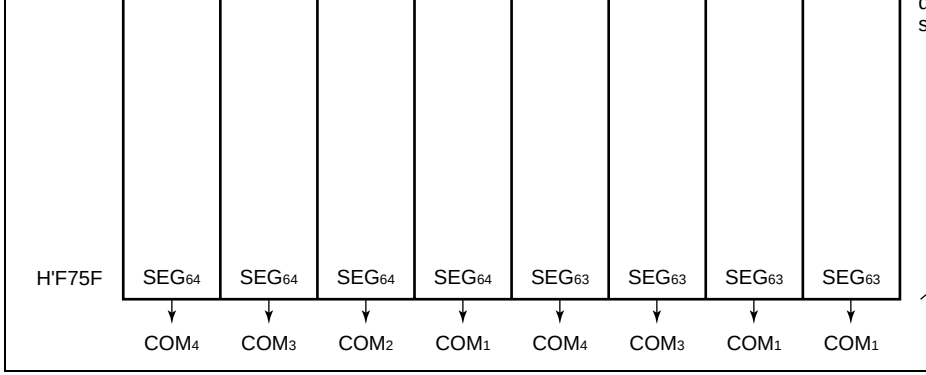


Figure 13.9 LCD RAM Map when Using Segment External Expansio
(SGX = "1", SGS3 to SGS0 = "0000" 1/4 Duty)

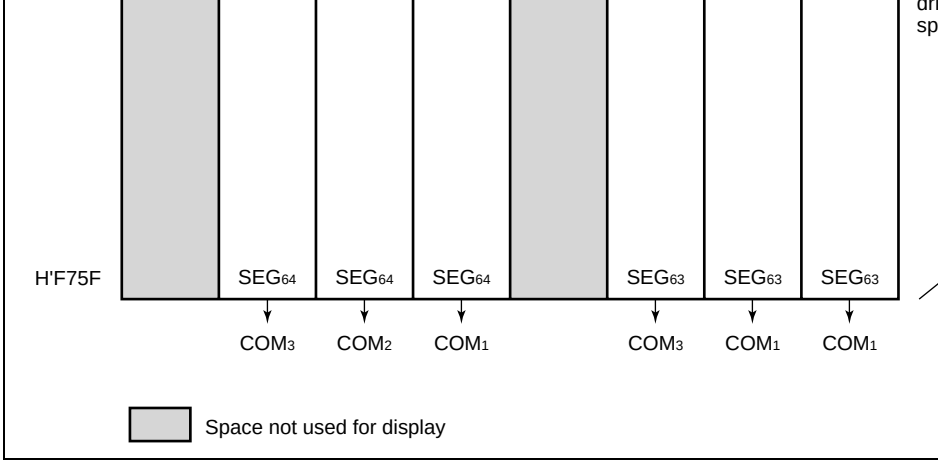


Figure 13.10 LCD RAM Map when Using Segment External Expansion (SGX = "1", SGS3 to SGS0 = "0000" 1/3 Duty)

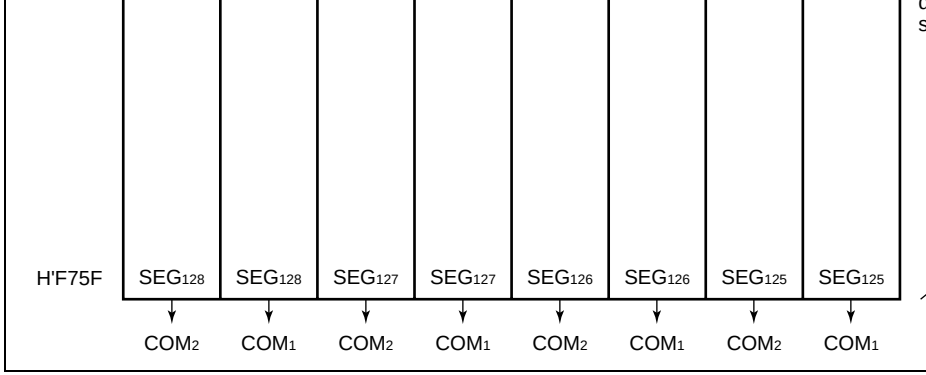


Figure 13.11 LCD RAM Map when Using Segment External Expansi
(SGX = "1", SGS3 to SGS0 = "0000" 1/2 Duty)

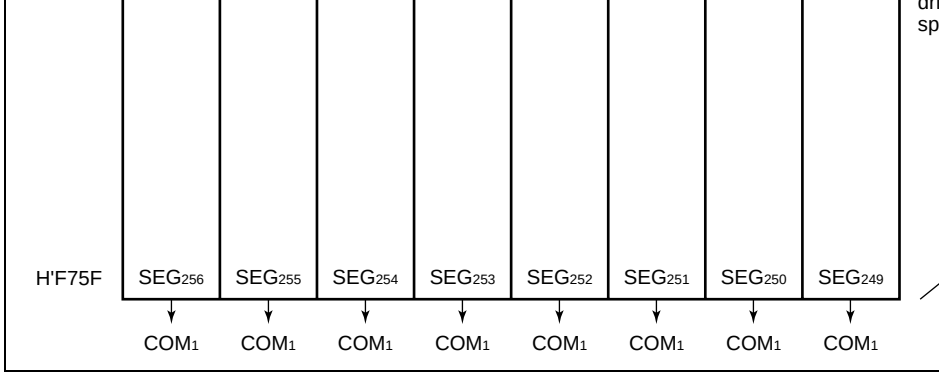


Figure 13.12 LCD RAM Map when Using Segment External Expansion (SGX = "1", SGS3 to SGS0 = "0000" Static)

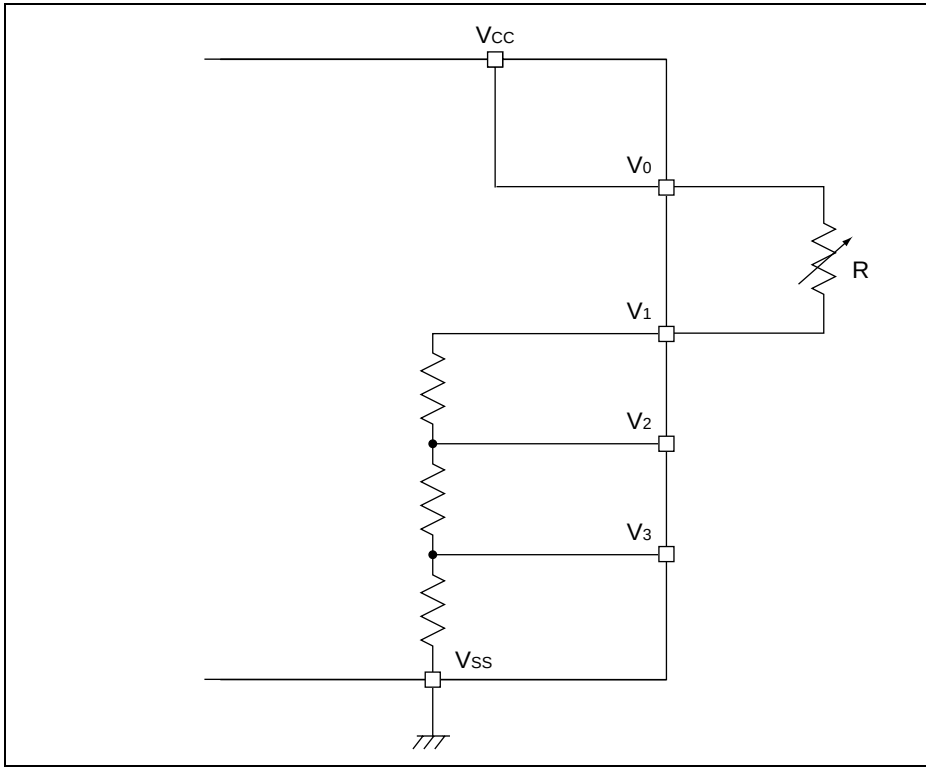


Figure 13.13 LCD Drive Power Supply Unit

power supply circuit for the LCD panel's current dissipation.

1. Principles

a. Capacitors are connected as external circuits to LCD power supply pins V1, V2, an shown in figure 13.14.

b. The capacitors connected to V1, V2, and V3 are repeatedly charged and discharged cycle shown in figure 13.14, maintaining the potentials.

c. At this time, the charged potential is a potential corresponding to the V1, V2, and V respectively. (For example, with 1/3 bias drive, the charge for V2 is 2/3 that of V1, V3 is 1/3 that of V1.)

d. Power is supplied to the LCD panel by means of the charges accumulated in these c

e. The capacitances and charging/discharging periods of these capacitors are therefore determined by the current dissipation of the LCD panel.

f. The charging and discharging periods can be selected by software.

2. Example of Operation (with 1/3 bias drive)

a. During charging period Tc in the figure, the potential is divided among pins V1, V2 by the built-in split-resistance (the potential of V2 being 2/3 that of V1, and that of 1/3 that of V1), as shown in figure 13.14, and external capacitors C1, C2, and C3 ar The LCD panel is continues to be driven during this time.

b. In the following discharging period, Tdc, charging is halted and the charge accumul each capacitor is discharged, driving the LCD panel.

c. At this time, a slight voltage drop occurs due to the discharging; optimum values m selected for the charging period and the capacitor capacitances to ensure that this do affect the driving of the LCD panel.

d. In this way, the capacitors connected to V1, V2, and V3 are repeatedly charged and discharged in the cycle shown in figure 13.14, maintaining the potentials and contin driving the LCD panel.

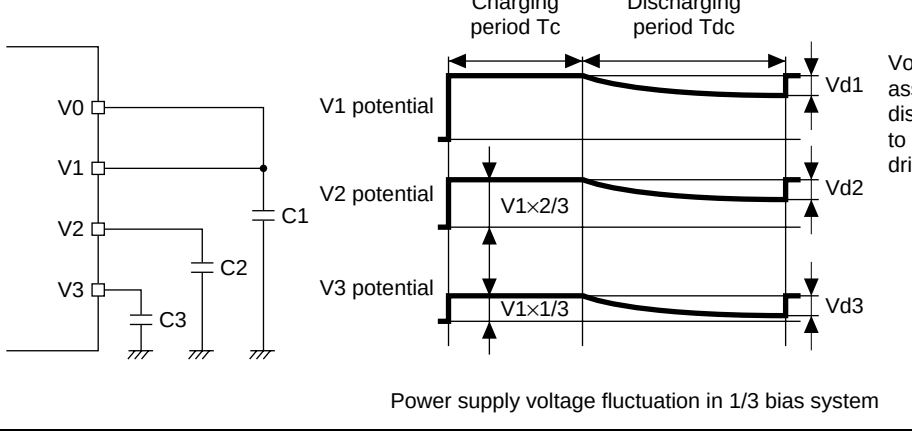


Figure 13.14 Example of Low-Power-Consumption LCD Drive Operat

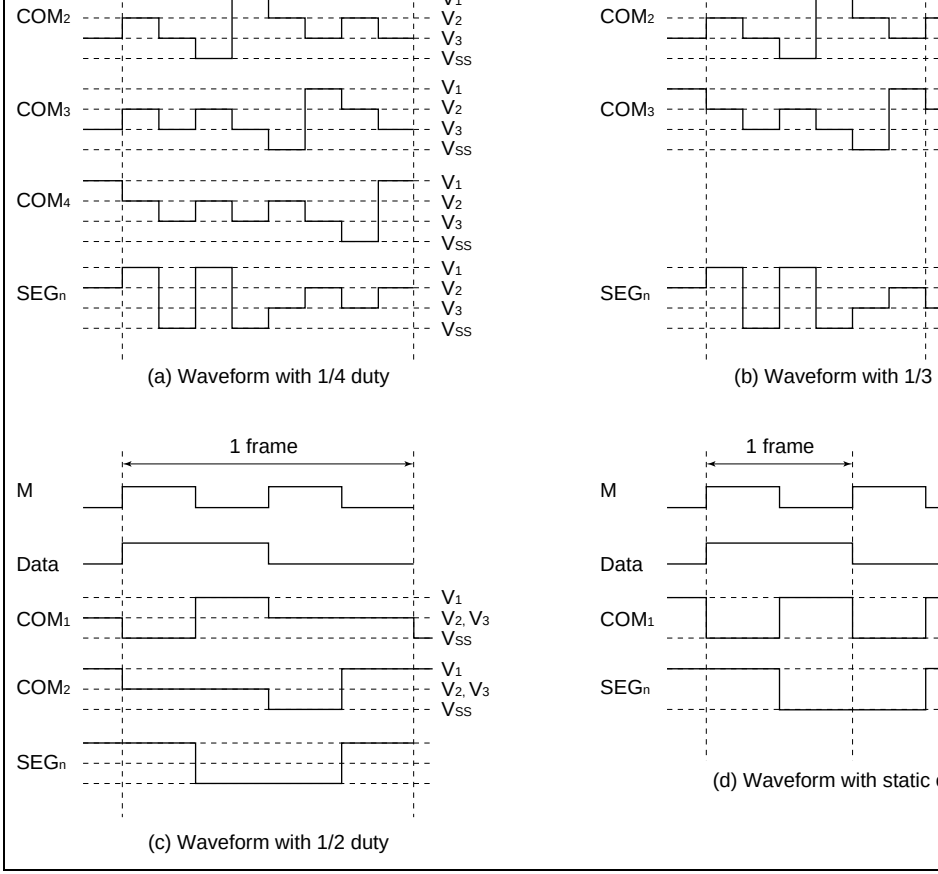
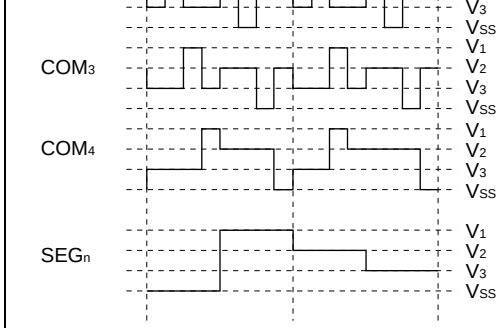
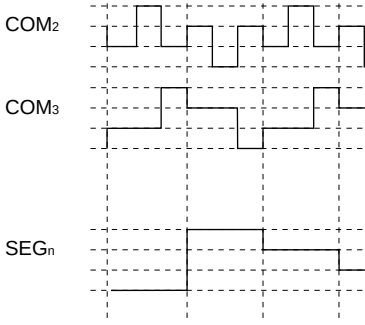


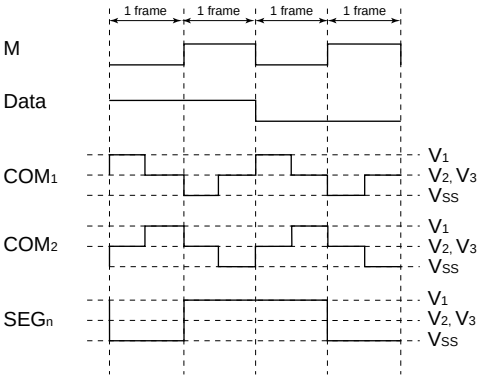
Figure 13.15 Output Waveforms for Each Duty Cycle (A Waveform)



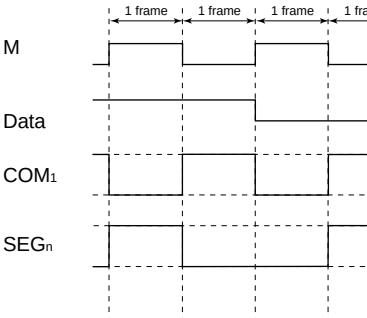
(a) Waveform with 1/4 duty



(b) Waveform with 1/3 du



(c) Waveform with 1/2 duty



(d) Waveform with static out

Figure 13.16 Output Waveforms for Each Duty Cycle (B Waveform

1/3 duty	Common output	V_3	V_2	V_1	V_S
	Segment output	V_2	V_3	V_{SS}	V_1
1/4 duty	Common output	V_3	V_2	V_1	V_S
	Segment output	V_2	V_3	V_{SS}	V_1

13.3.5 Operation in Power-Down Modes

In this LSI, the LCD controller/driver can be operated even in the power-down modes. operating state of the LCD controller/driver in the power-down modes is summarized i 13.4.

In subactive mode, watch mode, and subsleep mode, the system clock oscillator stops, therefore, unless ϕw, ϕw/2, or ϕw/4 has been selected by bits CKS3 to CKS0, the clock supplied and display will halt. Since there is a possibility that a direct current will be a the LCD panel in this case, it is essential to ensure that ϕw, ϕw/2, or ϕw/4 is selected. (medium-speed) mode, the system clock is switched, and therefore CKS3 to CKS0 mus modified to ensure that the frame frequency does not change.

ACT = "1"	Stops	Functions	Functions	Functions *3	Functions *3	Functions *3	Stops

Notes: 1. The subclock oscillator does not stop, but clock supply is halted.
2. The LCD drive power supply is turned off regardless of the setting of the PS
3. Display operation is performed only if φw, φw/2, or φw/4 is selected as the c clock.
4. The clock supplied to the LCD stops.

13.3.6 Boosting the LCD Drive Power Supply

When a large panel is driven, the on-chip power supply capacity may be insufficient. supply capacity is insufficient when V_{CC} is used as the power supply, the power suppl must be reduced. This can be done by connecting bypass capacitors of around 0.1 to 0 pins V_1 to V_3, as shown in figure 13.17, or by adding a split-resistance externally.

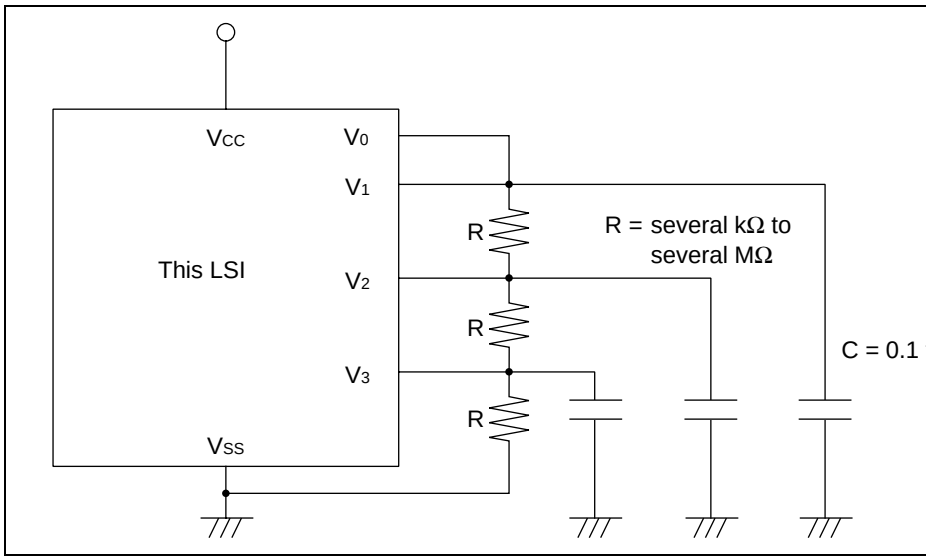
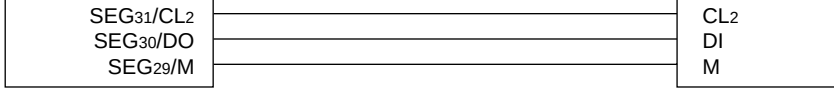


Figure 13.17 Connection of External Split-Resistance

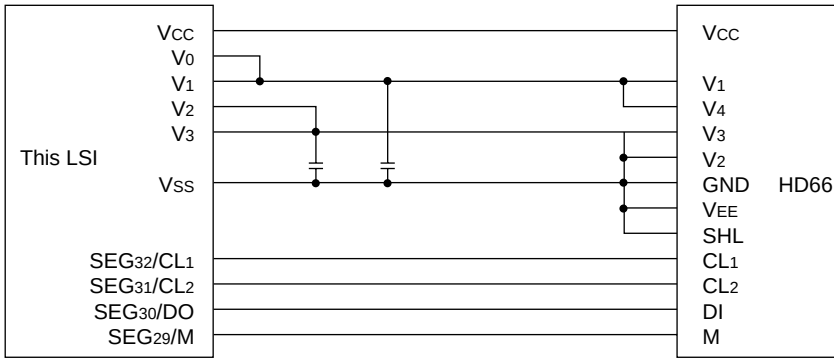
Figure 13.18 shows examples of connection to an HD66100. The output level is deterr
combination of the data and the M pin output, but these combinations differ from those
HD66100. Table 13.3 shows the output levels of the LCD drive power supply, and figu
and 13.16 show the common and segment waveforms for each duty cycle.

When ACT is cleared to 0, operation stops with $CL_2 = 0$, $CL_1 = 0$, M = 0, and DO at the
(1 or 0) being output at that instant. In standby mode, the expansion pins go to the high
impedance (floating) state.

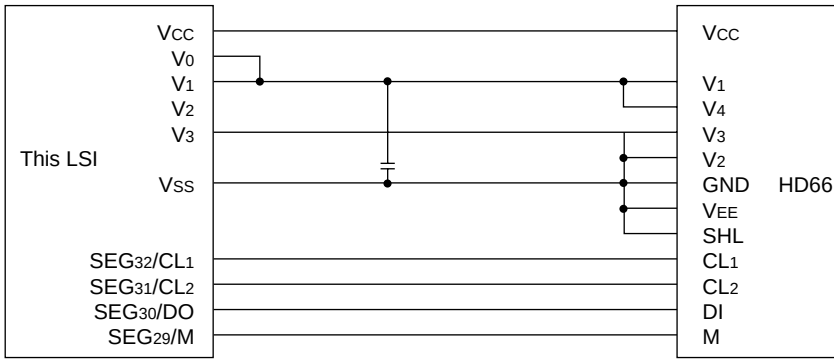
When external expansion is implemented, the load in the LCD panel increases and the
power supply may not provide sufficient current capacity. In this case, measures shoul
as described in section 13.3.6, Boosting the LCD Drive Power Supply.



(a) 1/3 bias, 1/4 or 1/3 duty



(b) 1/2 duty



(c) Static mode

Figure 13.18 Connection to HD66100

to the external V_{CC} pin. As a result, the current consumed when an external power supp
3.0 V or above can be held down to virtually the same low level as when used at appro
3.0 V. If the external power supply is 3.0 V or below, the internal voltage will be prac
same as the external voltage. It is, of course, also possible to use the same level of ext
supply voltage and internal power supply voltage without using the internal power sup
down circuit.

14.2 When Using Internal Power Supply Step-Down Circuit

Connect the external power supply to the V_{CC} pin, and connect a capacitance of approx
0.1 µF, in the case of the H8/3827R, or approximately 0.33 µF, in the case of the H8/3
H8/38427, between CV_{CC} and V_{SS}, as shown in figure 14.1. The internal step-down cir
effective simply by adding this external circuit. In the external circuit interface, the ex
supply voltage connected to V_{CC} and the GND potential connected to V_{SS} are the refere
For example, for port input/output levels, the V_{CC} level is the reference for the high lev
V_{SS} level is that for the low level. The LCD power supply and A/D converter analog p
are not affected by the internal step-down current.

In the H8/3827R Group the operating range differs depending on whether or not the in
down circuit is used. For details, see section 15, Electrical Characteristics.

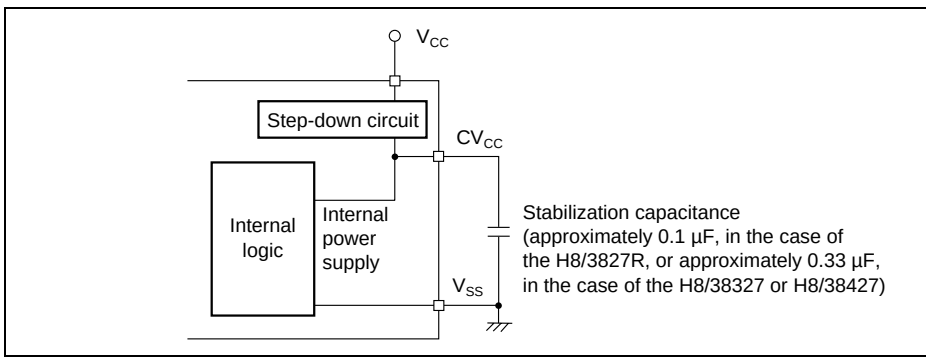


Figure 14.1 Power Supply Connection when Internal Step-Down Circuit i

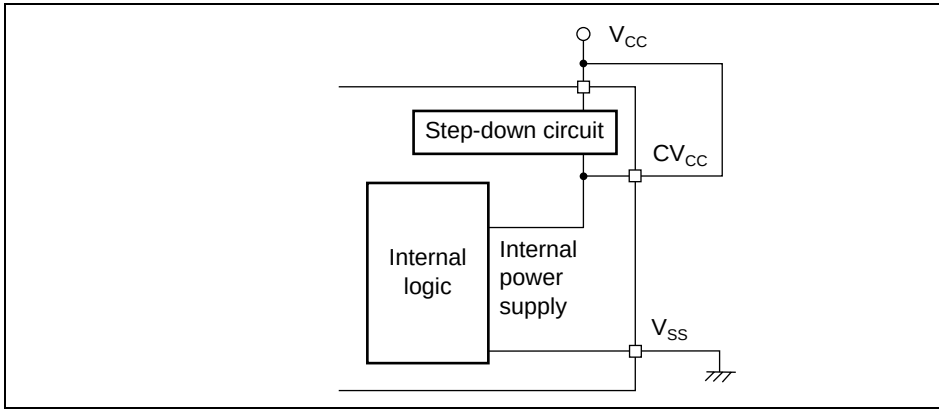


Figure 14.2 Power Supply Connection when Internal Step-Down Circuit is N

14.4 H8/3827S Group

The H8/3827S Group has two V_{CC} pins, which should be interconnected externally.

14.5 Notes on Switching from the H8/3827R to the H8/38327 or H

Examine the following with regard to the power supply circuit.

(1) If the internal power supply step-down circuit was used on the H8/3827R

The stabilization capacitance value differs between the products. It is necessary to c value from 0.1 µF (H8/3827R) to 0.33 µF (H8/38327 or H8/38427). Note that these rough guidelines and it is still necessary to confirm system operation.

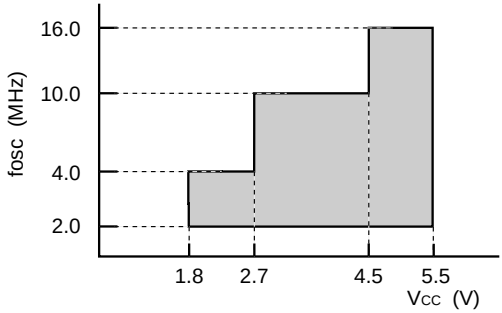
(2) If the internal power supply step-down circuit was not used on the H8/3827R

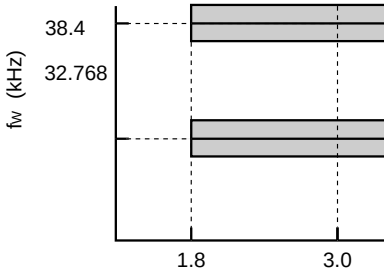
Use of the internal power supply step-down circuit of the H8/38327 or H8/38427 is recommended. Furthermore, operation at a V_{CC} of 3.6 V or greater is not guaranteed internal power supply step-down circuit is not used. It is therefore necessary to char CV_{CC} connection to use the internal power supply step-down circuit.

Table 13.1 Absolute Maximum Ratings

Item		Symbol	Value	Uni
Power supply voltage		V_{CC}, CV_{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 Port B	V_{in}	–0.3 to V_{CC} +0.3	V
	Port B	AV_{in}	–0.3 to AV_{CC} +0.3	V
Operating temperature		T_{opr}	–20 to +75*2	°C
Storage temperature		T_{stg}	–55 to +125	°C

Notes: 1. Permanent damage may occur to the chip if maximum ratings are exceede
operation should be under the conditions specified in Electrical Characteris
Exceeding these values can result in incorrect operation and reduced reliab
2. The operating temperature is the temperature range in which power (voltag
in "Electrical Characteristics") can be applied to the chip.

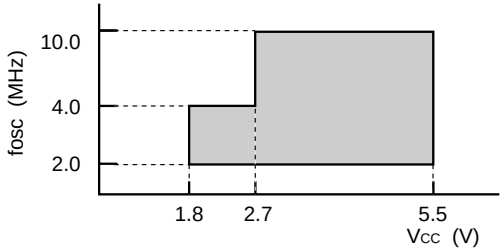




- Active (high-speed) mode
- Sleep (high-speed) mode
- Internal power supply step-down circuit not used

Note: fosc is the oscillator frequency. When external clocks are used, fosc=1MHz is the minimum.

- All operating modes

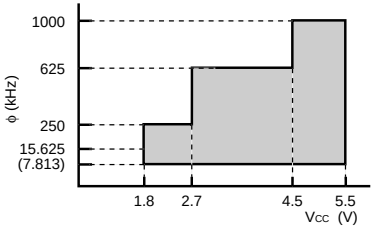


- Active (high-speed) mode
- Sleep (high-speed) mode
- Internal power supply step-down circuit used

Note: fosc is the oscillator frequency. When external clocks are used, fosc=1MHz is the minimum.

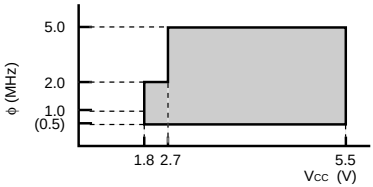
- Active (high-speed) mode
- Sleep (high-speed) mode (except CPU)
- Internal power supply step-down circuit not used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=1MHz.



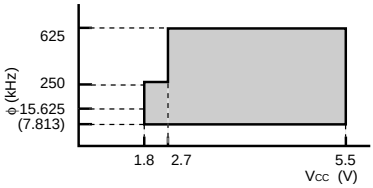
- Active (medium-speed) mode (except A/D converter)
- Sleep (medium-speed) mode (except A/D converter)
- Internal power supply step-down circuit not used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=15.625kHz.



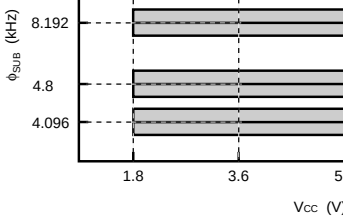
- Active (high-speed) mode
- Sleep (high-speed) mode (except CPU)
- Internal power supply step-down circuit used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=1MHz.



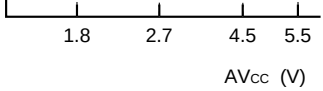
- Active (medium-speed) mode (except A/D converter)
- Sleep (medium-speed) mode (except A/D converter)
- Internal power supply step-down circuit used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=15.625kHz.

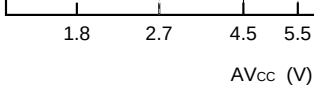


- Subactive mode
- Subsleep mode (except CPU)
- Watch mode (except CPU)

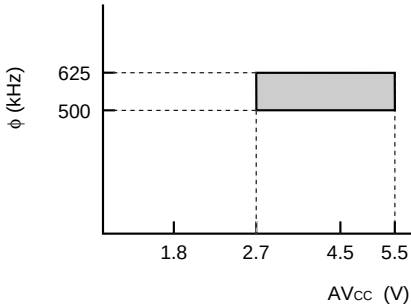
RENESAS



- Active (high-speed) mode
- Sleep (high-speed) mode
- Internal power supply step-down circuit not used and used



- Active (medium-speed) mode
- Sleep (medium-speed) mode
- Internal power supply step-down circu



- Active (medium-speed) mode
- Sleep (medium-speed) mode
- Internal power supply step-down circu

Item	Symbol	Applicable Pins	Values Min	Typ	Max	Unit	Test Condition
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, AEVL, AEVH, TMIC, TMIF, TMIG SCK_{31}, SCK_{32}, $\overline{ADTRG}$	0.8 V_{CC}	—	Vcc + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.9 V_{CC}	—	V_{CC} + 0.3		Except the above
		RXD_{31}, RXD_{32}, UD	0.7 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.8 V_{CC}	—	V_{CC} + 0.3		Except the above
		OSC_1	0.8 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.9 V_{CC}	—	V_{CC} + 0.3		Except the above
		X_1	0.9 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 1.8 V to 5.5
		$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	0.7 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.8 V_{CC}	—	V_{CC} + 0.3		Except the above
		PB_0 to PB_7	0.7 V_{CC}	—	AV_{CC} + 0.3		V_{CC} = 4.0 V to 5.5
			0.8 V_{CC}	—	AV_{CC} + 0.3		Except the above

Note: Connect the TEST pin to V_{SS}.

		$\overline{ADTRG}$				
		RXD_{31}, RXD_{32}, UD	–0.3	—	0.3 V_{CC}	V	V_{CC} = 4.0 V to 5.5 V
			–0.3	—	0.2 V_{CC}		Except the above
		OSC_1	–0.3	—	0.2		When internal step-down circuit is used.
			–0.3	—	0.2 V_{CC}	V	V_{CC} = 4.0 V to 5.5 V
			–0.3	—	0.1 V_{CC}		Except the above
		X_1	–0.3	—	0.1 V_{CC}	V	V_{CC} = 1.8 V to 5.5 V
		$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3, PB_0 to PB_7	–0.3	—	0.3 V_{CC}	V	V_{CC} = 4.0 V to 5.5 V
			–0.3	—	0.2 V_{CC}		Except the above
Output high voltage	V_{OH}	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	V_{CC} – 1.0	—	—	V	V_{CC} = 4.0 V to 5.5 V, $-I_{OH}$ = 1.0 mA
			V_{CC} – 0.5	—	—		V_{CC} = 4.0 V to 5.5 V, $-I_{OH}$ = 0.5 mA
			V_{CC} – 0.3	—	—		$-I_{OH}$ = 0.1 mA

		PA_0 to PA_3					
		$P3_0$ to $P3_7$	—	—	1.5		V_{CC} = 4.0 V to 5.5 I_{OL} = 10 mA
			—	—	0.6		V_{CC} = 4.0 V to 5.5 I_{OL} = 1.6 mA
			—	—	0.5		I_{OL} = 0.4 mA
Input/output leakage current	$\lvert I_{IL} \rvert$	$\overline{RES}$, $P4_3$	—	—	20.0	µA	V_{IN} = 0.5 V to V_{CC} – 0.5 V
			—	—	1.0		
		OSC_1, X_1, $P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	—	—	1.0	µA	V_{IN} = 0.5 V to V_{CC} – 0.5 V
		PB_0 to PB_7	—	—	1.0		V_{IN} = 0.5 V to AV_{CC} – 0.5 V
Pull-up MOS current	$-I_p$	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$	50.0	—	300.0	µA	V_{CC} = 5 V, V_{IN} = 0 V
			—	35.0	—		V_{CC} = 2.7 V, V_{IN} = 0 V

		$P4_3$	—	—	50.0		
			—	—	15.0		
		PB_0 to PB_7	—	—	15.0		
Active mode current dissipation	I_{OPE1}	V_{CC}	—	4.5	6.5	mA	Active (high-speed mode V_{CC} = 5 V, f_{OSC} = 10MHz
	I_{OPE2}	V_{CC}	—	1.3	2.0	mA	Active (medium-speed) mode V_{CC} = 5 V, f_{OSC} = 10MHz Divided by 128
Sleep mode current dissipation	I_{SLEEP}	V_{CC}	—	2.5	4.0	mA	V_{CC}=5 V, f_{OSC}=10MHz
Subactive mode current dissipation	I_{SUB}	V_{CC}	—	15	30	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator ($\phi_{SUB}=\phi_w/2$)
			—	8	—	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator ($\phi_{SUB}=\phi_w/8$)
Subsleep mode current dissipation	I_{SUBSP}	V_{CC}	—	7.5	16	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator ($\phi_{SUB}=\phi_w/2$)

dissipation							
RAM data retaining voltage	V_{RAM}	V_{CC}	1.5	—	—	V	
Allowable output low current (per pin)	I_{OL}	Output pins except port 3	—	—	2.0	mA	V_{CC} = 4.0 V to 5.5
		Port 3	—	—	10.0		V_{CC} = 4.0 V to 5.5
		All output pins	—	—	0.5		
Allowable output low current (total)	ΣI_{OL}	Output pins except port 3	—	—	40.0	mA	V_{CC} = 4.0 V to 5.5
		Port 3	—	—	80.0		V_{CC} = 4.0 V to 5.5
		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
			—	—	0.2		Except the above
Allowable output high current (total)	$\Sigma - I_{OH}$	All output pins	—	—	15.0	mA	V_{CC} = 4.0 V to 5.5
			—	—	10.0		Except the above

Notes: 1. Applies to the Mask ROM products.
2. Applies to the HD6473827R.

Subactive mode	V_{CC}	Only CPU operates	V_{CC}	Halted	System clo crystal
Subsleep mode	V_{CC}	Only timers operate, CPU stops	V_{CC}	Halted	Subclock o crystal
Watch mode	V_{CC}	Only time base operates, CPU stops	V_{CC}	Halted	
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Halted	System clo crystal Subclock o Pin X_1 = Gl

4. The guaranteed temperature as an electrical characteristic for Die products i
5. Excludes current in pull-up MOS transistors and output buffers.
6. When internal step-down circuit is used.

Item	Symbol	Applicable Pins	Values Min	Typ	Max	Unit	Test Condition
System clock oscillation frequency	f_{OSC}	OSC_1, OSC_2	2	—	16	MHz	V_{CC} = 4.5 V to 5.5
			2	—	10		V_{CC} = 2.7 V to 5.5
			2	—	4		V_{CC} = 1.8 V to 5.5
OSC clock (ϕ_{OSC}) cycle time	t_{OSC}	OSC_1, OSC_2	62.5	—	500 (1000)	ns	V_{CC} = 4.5 V to 5.5
			100	—	500 (1000)		V_{CC} = 2.7 V to 5.5
			250	—	500 (1000)		V_{CC} = 1.8 V to 5.5
System clock (ϕ) cycle time	t_{cyc}		2	—	128	t_{OSC}	
			—	—	244.1	µs	
Subclock oscillation frequency	f_W	X_1, X_2	—	32.768 or 38.4	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X_1, X_2	—	30.5 or 26.0	—	µs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}		2	—	8	t_W	
Instruction cycle time			2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time	t_{rc}	OSC_1, OSC_2	—	20	45	µs	Figure 15.9 V_{CC} = 2.2 V to 5.5
			—	0.1	8	ms	Figure 15.9 V_{CC} = 2.2 V to 5.5
			—	—	50	ms	Except the above

			100	—	—		V_{CC} = 1.8 V to 5.5 V
		X_1	—	15.26 or 13.02	—	µs	
External clock low width	t_{CPL}	OSC_1	25	—	—	ns	V_{CC} = 4.5 V to 5.5 V
			40	—	—		V_{CC} = 2.7 V to 5.5 V
			100	—	—		V_{CC} = 1.8 V to 5.5 V
		X_1	—	15.26 or 13.02	—	µs	
External clock rise time	t_{CPr}	OSC_1	—	—	6	ns	V_{CC} = 4.5 V to 5.5 V
			—	—	10		V_{CC} = 2.7 V to 5.5 V
			—	—	25		V_{CC} = 1.8 V to 5.5 V
		X_1	—	—	55.0	ns	
External clock fall time	t_{CPf}	OSC_1	—	—	6	ns	V_{CC} = 4.5 V to 5.5 V
			—	—	10		V_{CC} = 2.7 V to 5.5 V
			—	—	25		V_{CC} = 1.8 V to 5.5 V
		X_1	—	—	55.0	ns	
Pin $\overline{RES}$ low width	t_{REL}	$\overline{RES}$	10	—	—	t_{cyc}	

		$\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIC, TMIF, TMIG, AEVL, AEVH				t_{subcyc}
UD pin minimum modulation width	t_{UDH} t_{UDL}	UD	4	—	—	t_{cyc} t_{subcyc}

Notes: 1. Selected with SA1 and SA0 of system clock control register 2 (SYSCR2).
2. Internal power supply step-down circuit not used
3. Figures in parentheses are the maximum t_{OSC} rate with external clock input
4. The guaranteed temperature as an electrical characteristic for Die products

Table 15.4 Serial Interface (SCI3-1, SCI3-2) Timing

V_{CC} = 1.8 V to 5.5 V, AV_{CC} = 1.8 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –20°C to +75
(including subactive mode) unless otherwise indicated.

			Values				
Item		**Symbol**	**Min**	**Typ**	**Max**	**Unit**	**Test Conditions**
Input clock cycle	Asynchronous	t_{scyc}	4	—	—	t_{cyc} or t_{subcyc}	
	Synchronous		6	—	—		
Input clock pulse width		t_{SCKW}	0.4	—	0.6	t_{scyc}	
Transmit data delay time (synchronous)		t_{TXD}	—	—	1	t_{cyc} or t_{subcyc}	V_{CC} = 4.0 V to 5.5 V
			—	—	1		Except the above
Receive data setup time (synchronous)		t_{RXS}	200.0	—	—	ns	V_{CC} = 4.0 V to 5.5 V
			400.0	—	—		Except the above
Receive data hold time (synchronous)		t_{RXH}	200.0	—	—	ns	V_{CC} = 4.0 V to 5.5 V
			400.0	—	—		Except the above

Notes: 1. When internal step-down circuit is not used.
2. The guaranteed temperature as an electrical characteristic for Die products

Item	Symbol	Applicable Pins	Values			Unit	Test Condition
			Min	Typ	Max		
Analog power supply voltage	AV_{CC}	AV_{CC}	1.8	—	5.5	V	
Analog input voltage	AV_{IN}	AN_0 to AN_7	– 0.3	—	AV_{CC} + 0.3	V	
Analog power supply current	AI_{OPE}	AV_{CC}	—	—	1.5	mA	AV_{CC} = 5 V
	AI_{STOP1}	AV_{CC}	—	600	—	µA	
	AI_{STOP2}	AV_{CC}	—	—	5	µA	
Analog input capacitance	C_{AIN}	AN_0 to AN_7	—	—	15.0	pF	
Allowable signal source impedance	R_{AIN}		—	—	10.0	kΩ	
Resolution (data length)			—	—	10	bit	
Nonlinearity error			—	—	±2.5	LSB	AV_{CC} = 2.7 V to 5.5 V V_{CC} = 2.7 V to 5.5 V
			—	—	±5.5		AV_{CC} = 2.0 V to 5.5 V V_{CC} = 2.0 V to 5.5 V
			—	—	±7.5		Except the above
Quantization error			—	—	±0.5	LSB	

time					V_{CC} = 2.7 V to 5.5 V
		62	—	124	Except the above

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 i
3. AI_{STOP2} is the current at reset and in standby, watch, subactive, and subsle while the A/D converter is idle.
4. When internal step-down circuit is not used.
5. Conversion time 62 µs
6. The guaranteed temperature as an electrical characteristic for Die products

Item	Symbol	Applicable Pins	Test Conditions	Values Min	Typ	Max	Unit
Segment driver drop voltage	V_{DS}	SEG_1 to SEG_{32}	I_D = 2 µA V_1 = 2.7 to 5.5 V	—	—	0.6	V
Common driver drop voltage	V_{DC}	COM_1 to COM_4	I_D = 2 µA V_1 = 2.7 to 5.5 V	—	—	0.3	V
LCD power supply split-resistance	R_{LCD}		Between V_1 and V_{SS}	0.5	3.0	9.0	MΩ
Liquid crystal display voltage	V_{LCD}	V_1		2.2	—	5.5	V

Notes: 1. The voltage drop from power supply pins V_1, V_2, V_3, and V_{SS} to each segmen common pin.

2. When the liquid crystal display voltage is supplied from an external power so ensure that the following relationship is maintained: $V_1 \geq V_2 \geq V_3 \geq V_{SS}$.

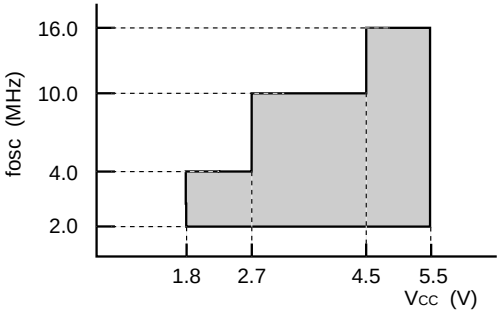
3. The guaranteed temperature as an electrical characteristic for Die products i

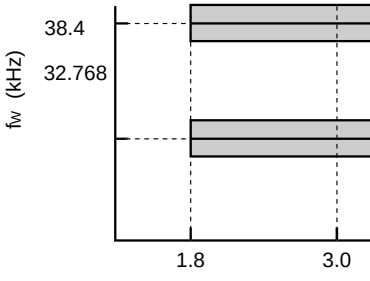
Clock low width	t_{CWL}	CL_2		800	—	—	ns
Clock setup time	t_{CSU}	CL_1, CL_2	*1	500	—	—	ns
Data setup time	t_{SU}	DO	*1	300	—	—	ns
Data hold time	t_{DH}	DO	*1	300	—	—	ns
M delay time	t_{DM}	M	*1	–1000	—	1000	ns
Clock rise and fall times	t_{CT}	CL_1, CL_2		—	—	170	ns

Notes: 1. Value when the frame frequency is set to between 30.5 Hz and 488 Hz.
2. The guaranteed temperature as an electrical characteristic for Die products

Power supply voltage		V_{CC}, CV_{CC}	–0.3 to +7.0
Analog power supply voltage		AV_{CC}	–0.3 to +7.0
Programming voltage		V_{PP}	–0.3 to +13.0
Input voltage	Ports other than Port B	V_{in}	–0.3 to V_{CC} +0.3
	Port B	AV_{in}	–0.3 to AV_{CC} +0.3
Operating temperature		T_{opr}	–40 to +85
Storage temperature		T_{stg}	–55 to +125

Note: Permanent damage may occur to the chip if maximum ratings are exceeded. No operation should be under the conditions specified in Electrical Characteristics. these values can result in incorrect operation and reduced reliability.

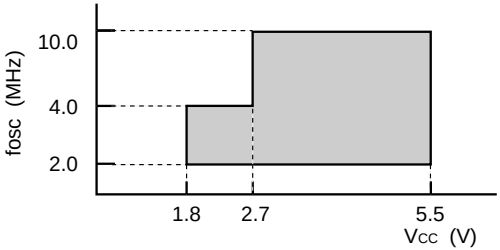




- Active (high-speed) mode
- Sleep (high-speed) mode
- Internal power supply step-down circuit not used

Note: fosc is the oscillator frequency. When external clocks are used, fosc=1MHz is the minimum.

- All operating modes

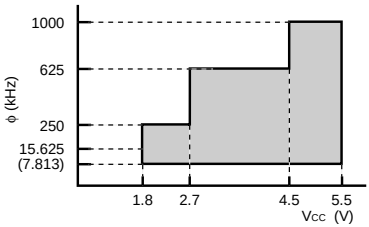


- Active (high-speed) mode
- Sleep (high-speed) mode
- Internal power supply step-down circuit used

Note: fosc is the oscillator frequency. When external clocks are used, fosc=1MHz is the minimum.

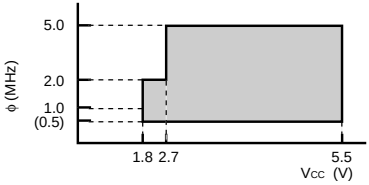
• Active (high-speed) mode
• Sleep (high-speed) mode (except CPU)
• Internal power supply step-down circuit not used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=1MHz.



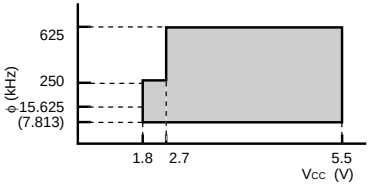
• Active (medium-speed) mode (except A/D converter)
• Sleep (medium-speed) mode (except A/D converter)
• Internal power supply step-down circuit not used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=15.625kHz.



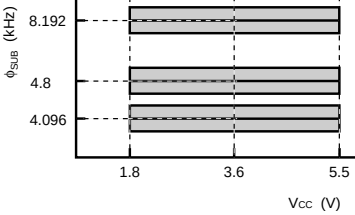
• Active (high-speed) mode
• Sleep (high-speed) mode (except CPU)
• Internal power supply step-down circuit used

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=1MHz.

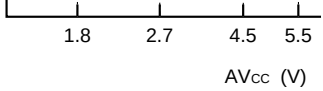


• Active (medium-speed) mode (except A/D converter)
• Sleep (medium-speed) mode (except A/D converter)
• Internal power supply step-down circuit used

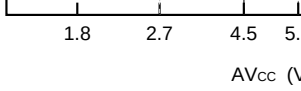
Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=15.625kHz.



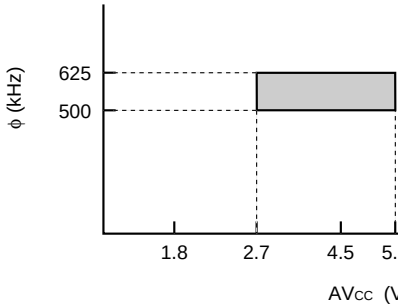
• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)



- Active (high-speed) mode
- Sleep (high-speed) mode
- Internal power supply step-down circuit not used and used



- Active (medium-speed) mode
- Sleep (medium-speed) mode
- Internal power supply step-down circ



- Active (medium-speed) mode
- Sleep (medium-speed) mode
- Internal power supply step-down circ

Item	Symbol	Applicable Pins	Values Min	Typ	Max	Unit	Test Condition
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{WKP}_0$ to $\overline{WKP}_7$, $\overline{IRQ}_0$ to $\overline{IRQ}_4$, AEVL, AEVH, TMIC, TMIF, TMIG SCK_{31}, SCK_{32}, $\overline{ADTRG}$	0.8 V_{CC}	—	Vcc + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.9 V_{CC}	—	V_{CC} + 0.3		Except the above
		RXD_{31}, RXD_{32}, UD	0.7 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.8 V_{CC}	—	V_{CC} + 0.3		Except the above
		OSC_1	0.8 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.9 V_{CC}	—	V_{CC} + 0.3		Except the above
		X_1	0.9 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 1.8 V to 5.5
		$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	0.7 V_{CC}	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5.5
			0.8 V_{CC}	—	V_{CC} + 0.3		Except the above
		PB_0 to PB_7	0.7 V_{CC}	—	AV_{CC} + 0.3		V_{CC} = 4.0 V to 5.5
			0.8 V_{CC}	—	AV_{CC} + 0.3		Except the above

Note: Connect the TEST pin to V_{SS}.

	$\overline{ADTRG}$					
	RXD_{31}, RXD_{32}, UD	–0.3	—	0.3 V_{CC}	V	V_{CC} = 4.0 V to 5.5
		–0.3	—	0.2 V_{CC}		Except the above
	OSC_1	–0.3	—	0.2		When internal ste down circuit is used.
		–0.3	—	0.2 V_{CC}	V	V_{CC} = 4.0 V to 5.5
		–0.3	—	0.1 V_{CC}		Except the above
	X_1	–0.3	—	0.1 V_{CC}	V	V_{CC} = 1.8 V to 5.5
	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3, PB_0 to PB_7	–0.3	—	0.3 V_{CC}	V	V_{CC} = 4.0 V to 5.5
		–0.3	—	0.2 V_{CC}		Except the above
Output high voltage	V_{OH} $P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	V_{CC} – 1.0	—	—	V	V_{CC} = 4.0 V to 5.5 $-I_{OH}$ = 1.0 mA
		V_{CC} – 0.5	—	—		V_{CC} = 4.0 V to 5.5 $-I_{OH}$ = 0.5 mA
		V_{CC} – 0.3	—	—		$-I_{OH}$ = 0.1 mA

		PA_0 to PA_3					
		$P3_0$ to $P3_7$	—	—	1.5		V_{CC} = 4.0 V to 5.5 V I_{OL} = 10 mA
			—	—	0.6		V_{CC} = 4.0 V to 5.5 V I_{OL} = 1.6 mA
			—	—	0.5		I_{OL} = 0.4 mA
Input/output leakage current	$\lvert I_{IL} \rvert$	$\overline{RES}$, $P4_3$	—	—	20.0	µA	V_{IN} = 0.5 V to V_{CC} – 0.5 V
			—	—	1.0		
		OSC_1, X_1, $P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	—	—	1.0	µA	V_{IN} = 0.5 V to V_{CC} – 0.5 V
		PB_0 to PB_7	—	—	1.0		V_{IN} = 0.5 V to AV_{CC} – 0.5 V
Pull-up MOS current	$-I_p$	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$	50.0	—	300.0	µA	V_{CC} = 5 V, V_{IN} = 0 V
			—	35.0	—		V_{CC} = 2.7 V, V_{IN} = 0 V

		$P4_3$	—	—	50.0		
			—	—	15.0		
		PB_0 to PB_7	—	—	15.0		
Active mode current dissipation	I_{OPE1}	V_{CC}	—	4.5	6.5	mA	Active (high-speed mode V_{CC} = 5 V, f_{OSC} = 10MHz
	I_{OPE2}	V_{CC}	—	1.3	2.0	mA	Active (medium-speed) mode V_{CC} = 5 V, f_{OSC} = 10MHz Divided by 128
Sleep mode current dissipation	I_{SLEEP}	V_{CC}	—	2.5	4.0	mA	V_{CC}=5 V, f_{OSC}=10MHz
Subactive mode current dissipation	I_{SUB}	V_{CC}	—	15	30	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator ($\phi_{SUB}=\phi_w/2$)
			—	8	—	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator ($\phi_{SUB}=\phi_w/8$)
Subsleep mode current dissipation	I_{SUBSP}	V_{CC}	—	7.5	16	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator ($\phi_{SUB}=\phi_w/2$)

dissipation							
RAM data retaining voltage	V_{RAM}	V_{CC}	1.5	—	—	V	
Allowable output low current (per pin)	I_{OL}	Output pins except port 3	—	—	2.0	mA	V_{CC} = 4.0 V to 5.5 V
		Port 3	—	—	10.0		V_{CC} = 4.0 V to 5.5 V
		All output pins	—	—	0.5		
Allowable output low current (total)	ΣI_{OL}	Output pins except port 3	—	—	40.0	mA	V_{CC} = 4.0 V to 5.5 V
		Port 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		Except the above
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		Except the above

Notes: 1. Applies to the Mask ROM products.
2. Applies to the HD6473827R.

Subactive mode	V_{CC}	Only CPU operates	V_{CC}	Halted	System cl crystal
Subsleep mode	V_{CC}	Only timers operate, CPU stops	V_{CC}	Halted	Subclock crystal
Watch mode	V_{CC}	Only time base operates, CPU stops	V_{CC}	Halted	
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Halted	System cl crystal Subclock Pin X_1 = C

4. Excludes current in pull-up MOS transistors and output buffers.
5. When internal step-down circuit is used.

Item	Symbol	Applicable Pins	Values Min	Typ	Max	Unit	Test Condition
System clock oscillation frequency	f_{OSC}	OSC_1, OSC_2	2	—	16	MHz	V_{CC} = 4.5 V to 5.5 V
			2	—	10		V_{CC} = 2.7 V to 5.5 V
			2	—	4		V_{CC} = 1.8 V to 5.5 V
OSC clock (ϕ_{OSC}) cycle time	t_{OSC}	OSC_1, OSC_2	62.5	—	500 (1000)	ns	V_{CC} = 4.5 V to 5.5 V
			100	—	500 (1000)		V_{CC} = 2.7 V to 5.5 V
			250	—	500 (1000)		V_{CC} = 1.8 V to 5.5 V
System clock (ϕ) cycle time	t_{cyc}		2	—	128	t_{OSC}	
			—	—	244.1	µs	
Subclock oscillation frequency	f_W	X_1, X_2	—	32.768 or 38.4	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X_1, X_2	—	30.5 or 26.0	—	µs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}		2	—	8	t_W	
Instruction cycle time			2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time	t_{rc}	OSC_1, OSC_2	—	20	45	µs	Figure 15.9 V_{CC} = 2.2 V to 5.5 V
			—	0.1	8	ms	Figure 15.9 V_{CC} = 2.2 V to 5.5 V
			—	—	50	ms	Except the above

			100	—	—		V_{CC} = 1.8 V to 5.5
		X_1	—	15.26 or 13.02	—	µs	
External clock low width	t_{CPL}	OSC_1	25	—	—	ns	V_{CC} = 4.5 V to 5.5
			40	—	—		V_{CC} = 2.7 V to 5.5
			100	—	—		V_{CC} = 1.8 V to 5.5
		X_1	—	15.26 or 13.02	—	µs	
External clock rise time	t_{CPr}	OSC_1	—	—	6	ns	V_{CC} = 4.5 V to 5.5
			—	—	10		V_{CC} = 2.7 V to 5.5
			—	—	25		V_{CC} = 1.8 V to 5.5
		X_1	—	—	55.0	ns	
External clock fall time	t_{CPf}	OSC_1	—	—	6	ns	V_{CC} = 4.5 V to 5.5
			—	—	10		V_{CC} = 2.7 V to 5.5
			—	—	25		V_{CC} = 1.8 V to 5.5
		X_1	—	—	55.0	ns	
Pin $\overline{RES}$ low width	t_{REL}	$\overline{RES}$	10	—	—	t_{cyc}	

			$\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIC, TMIF, TMIG, AEVL, AEVH				t_{subcyc}
UD pin minimum modulation width		t_{UDH} t_{UDL}	UD	4	—	—	t_{cyc} t_{subcyc}

Notes: 1. Selected with SA1 and SA0 of system clock control register 2 (SYSCR2).
2. Internal power supply step-down circuit not used
3. Figures in parentheses are the maximum t_{OSC} rate with external clock input.

Table 15.11 Serial Interface (SCI3-1, SCI3-2) Timing

V_{CC} = 1.8 V to 5.5 V, AV_{CC} = 1.8 V to 5.5 V, V_{SS} = AV_{SS} = 0.0 V, T_a = –40°C to +85°C otherwise indicated.

Item		Symbol	Values Min	Typ	Max	Unit	Test Conditions	R Fi
Input clock cycle	Asynchronous	t_{scyc}	4	—	—	t_{cyc} or t_{subcyc}		Fi
	Synchronous		6	—	—			
Input clock pulse width		t_{SCKW}	0.4	—	0.6	t_{scyc}		Fi
Transmit data delay time (synchronous)		t_{TXD}	—	—	1	t_{cyc} or t_{subcyc}	V_{CC} = 4.0 V to 5.5 V	Fi
			—	—	1		Except the above	
Receive data setup time (synchronous)		t_{RXS}	200.0	—	—	ns	V_{CC} = 4.0 V to 5.5 V	Fi
			400.0	—	—		Except the above	Fi
Receive data hold time (synchronous)		t_{RXH}	200.0	—	—	ns	V_{CC} = 4.0 V to 5.5 V	Fi
			400.0	—	—		Except the above	Fi

Note: * When internal step-down circuit is not used.

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition
Analog power supply voltage	AV_{CC}	AV_{CC}	1.8	—	5.5	V	
Analog input voltage	AV_{IN}	AN_0 to AN_7	– 0.3	—	AV_{CC} + 0.3	V	
Analog power supply current	AI_{OPE}	AV_{CC}	—	—	1.5	mA	AV_{CC} = 5 V
	AI_{STOP1}	AV_{CC}	—	600	—	µA	
	AI_{STOP2}	AV_{CC}	—	—	5	µA	
Analog input capacitance	C_{AIN}	AN_0 to AN_7	—	—	15.0	pF	
Allowable signal source impedance	R_{AIN}		—	—	10.0	kΩ	
Resolution (data length)			—	—	10	bit	
Nonlinearity error			—	—	±2.5	LSB	AV_{CC} = 2.7 V to 5.5 V V_{CC} = 2.7 V to 5.5 V
			—	—	±5.5		AV_{CC} = 2.0 V to 5.5 V V_{CC} = 2.0 V to 5.5 V
			—	—	±7.5		Except the above
Quantization error			—	—	±0.5	LSB	

time					V_{CC} = 2.7 V to 5.5 V
		62	—	124	Except the above

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 id
3. AI_{STOP2} is the current at reset and in standby, watch, subactive, and subslee while the A/D converter is idle.
4. When internal step-down circuit is not used.
5. Conversion time 62 µs

Item	Symbol	Applicable Pins	Test Conditions	Values			Unit
				Min	Typ	Max	
Segment driver drop voltage	V_{DS}	SEG_1 to SEG_{32}	I_D = 2 µA V_1 = 2.7 to 5.5 V	—	—	0.6	V
Common driver drop voltage	V_{DC}	COM_1 to COM_4	I_D = 2 µA V_1 = 2.7 to 5.5 V	—	—	0.3	V
LCD power supply split-resistance	R_{LCD}		Between V_1 and V_{SS}	0.5	3.0	9.0	MΩ
Liquid crystal display voltage	V_{LCD}	V_1		2.2	—	5.5	V

Notes: 1. The voltage drop from power supply pins V_1, V_2, V_3, and V_{SS} to each segme common pin.

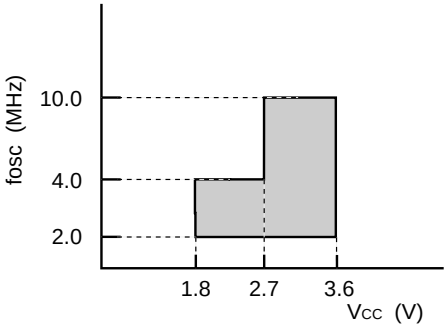
2. When the liquid crystal display voltage is supplied from an external power s ensure that the following relationship is maintained: $V_1 \geq V_2 \geq V_3 \geq V_{SS}$.

Clock low width	t_{CWL}	CL_2	*	800	—	—	ns
Clock setup time	t_{CSU}	CL_1, CL_2	*	500	—	—	ns
Data setup time	t_{SU}	DO	*	300	—	—	ns
Data hold time	t_{DH}	DO	*	300	—	—	ns
M delay time	t_{DM}	M	*	–1000	—	1000	ns
Clock rise and fall times	t_{CT}	CL_1, CL_2		—	—	170	ns

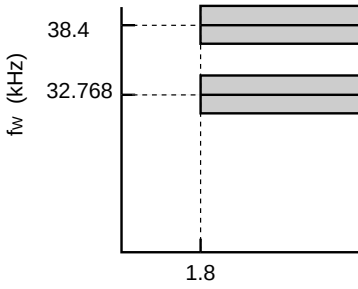
Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

Analog power supply voltage		AV_{CC}	–0.3 to +4.3	V
Input voltage	Ports other than Port B	V_{in}	–0.3 to V_{CC} +0.3	V
	Port B	AV_{in}	–0.3 to AV_{CC} +0.3	V
Operating temperature		T_{opr}	–20 to +75 (Regular specifications)	°C
			–40 to +85 (wide-range specifications)	
			+75 (products shipped as chips)*2	
Storage temperature		T_{stg}	–55 to +125	°C

Notes: 1. Permanent damage may occur to the chip if maximum ratings are exceede
operation should be under the conditions specified in Electrical Characteris
Exceeding these values can result in incorrect operation and reduced reliab
2. Power may be applied when the temperature is between –20 and –75°C.



• Active (high-speed) mode
• Sleep (high-speed) mode

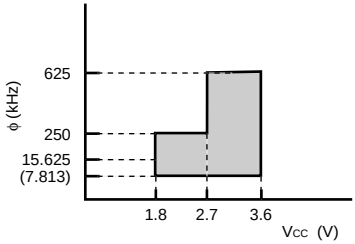


• All operating modes

Note: fosc is the oscillator frequency. When external clocks are used, fosc=1MHz is the minimum.

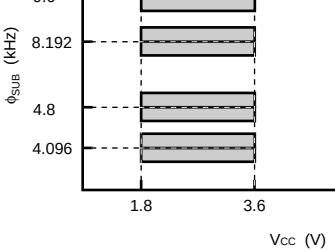
• Active (high-speed) mode
• Sleep (high-speed) mode (except CPU)

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=1MHz.



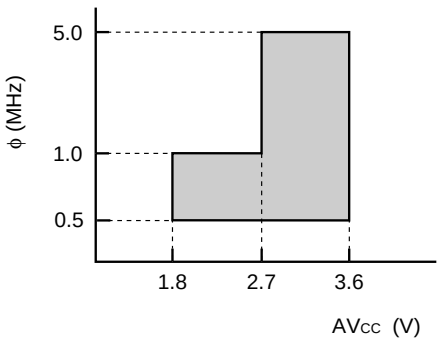
• Active (medium-speed) mode (except A/D converter)
• Sleep (medium-speed) mode (except A/D converter)

Note: Figures in parentheses are the minimum operating frequency of a case external clocks are used. When using an oscillator, the minimum operating frequency is ϕ=15.625kHz.

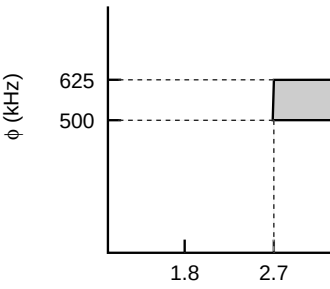


• Subactive mode
• Subsleep mode (except CPU)
• Watch mode (except CPU)

3. Analog Power Supply Voltage and A/D Converter Operating Range



• Active (high-speed) mode
• Sleep (high-speed) mode



• Active (medium-speed) m
• Sleep (medium-speed) mo

Input high voltage	V_{IH}	$\overline{RES}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, AEVL, AEVH, TMIC, TMIF, TMIG SCK_{31}, SCK_{32}, $\overline{ADTRG}$	0.9 V_{CC}	—	V_{CC} + 0.3	V	
		RXD_{31}, RXD_{32}, UD	0.8 V_{CC}	—	V_{CC} + 0.3	V	
		OSC_1	0.9 V_{CC}	—	V_{CC} + 0.3	V	
		X_1	0.9 V_{CC}	—	V_{CC} + 0.3	V	
		$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	0.8 V_{CC}	—	V_{CC} + 0.3	V	
		PB_0 to PB_7	0.8 V_{CC}	—	AV_{CC} + 0.3		

Note: Connect the TEST pin to V_{SS}.

	$\overline{ADTRG}$					
	RXD_{31}, RXD_{32}, UD	–0.3	—	0.2 V_{CC}	V	
	OSC_1	–0.3	—	0.1 V_{CC}	V	
	X_1	–0.3	—	0.1 V_{CC}	V	
	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3, PB_0 to PB_7	–0.3	—	0.2 V_{CC}	V	
Output high voltage	V_{OH} $P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	V_{CC} – 0.3	—	—	V	$-I_{OH}$ = 0.1 mA

		$P3_0$ to $P3_7$	—	—	0.5		I_{OL} = 0.4 mA
Input/output leakage current	$\lvert I_{IL} \rvert$	$\overline{RES}$, OSC_1, X_1, $P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	—	—	1.0	µA	V_{IN} = 0.5 V to V_{CC} – 0.5 V
		PB_0 to PB_7	—	—	1.0		V_{IN} = 0.5 V to AV_{CC} – 0.5 V
Pull-up MOS current	$-I_p$	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$	10.0	—	300.0	µA	V_{CC} = 3 V, V_{IN} = 0 V

			—	1.4	*3		Active (high-speed) mode V_{CC} = 3 V, f_{OSC} = 4 MHz
			—	3.5	5.5		Active (high-speed) mode V_{CC} = 3 V, f_{OSC} = 10 MHz
	I_{OPE2}	V_{CC}	—	0.1	*3		Active (medium-speed) mode V_{CC} = 1.8 V, f_{OSC} = 2 MHz ϕ_{OSC}/128
			—	0.3	*3		Active (medium-speed) mode V_{CC} = 3 V, f_{OSC} = 4 MHz ϕ_{OSC}/128
			—	0.7	1.6		Active (medium-speed) mode V_{CC} = 3 V, f_{OSC} = 10 MHz ϕ_{OSC}/128
Sleep mode current dissipation	I_{SLEEP}	V_{CC}	—	0.2	*3	mA	V_{CC} = 1.8 V, f_{OSC} = 2 MHz
			—	0.6	*3		V_{CC} = 3 V, f_{OSC} = 4 MHz
			—	1.4	2.9		V_{CC} = 3 V, f_{OSC} = 10 MHz

							(ϕ_{SUB} = ϕ_W/8)
			—	14	*3		V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator (ϕ_{SUB} = ϕ_W/2)
Subsleep mode current dissipation	I_{SUBSP}	V_{CC}	—	5.0	12	µA	V_{CC} = 2.7 V, LCD on 32 kHz crystal oscillator (ϕ_{SUB}=ϕ_W/2)
Watch mode current dissipation	I_{WATCH}	V_{CC}	—	1.4	*3	µA	V_{CC} = 1.8 V, Ta = 25°C 32 kHz crystal oscillator LCD not used
			—	2.2	*3		V_{CC} = 2.7 V, Ta = 25°C 32 kHz crystal oscillator LCD not used
			—	2.8	6		V_{CC} = 2.7 V, 32 kHz crystal oscillator LCD not used
Standby mode current dissipation	I_{STBY}	V_{CC}	—	0.3	*3	µA	32 kHz crystal oscillator not used V_{CC} = 1.8 V, Ta = 25°C
			—	0.5	*3		32 kHz crystal oscillator not used V_{CC} = 2.7 V, Ta = 25°C
			—	1	5		Except the above
RAM data retaining voltage	V_{RAM}	V_{CC}	1.5	—	—	V	

(total)							
Allowable output high current (per pin)	$-I_{OH}$	All output pins	—	—	0.2	mA	
Allowable output high current (total)	$\Sigma - I_{OH}$	All output pins	—	—	10.0	mA	

Notes: 1. Pin states during current measurement.

Mode	$\overline{RES}$ Pin	Internal State	Other Pins	LCD Power Supply	Oscillato
Active (high-speed) mode (I_{OPE1})	V_{CC}	Only CPU operates	V_{CC}	Halted	System cl crystal
Active (medium-speed) mode (I_{OPE2})					Subclock Pin X_1 = C
Sleep mode	V_{CC}	Only timers operate	V_{CC}	Halted	
Subactive mode	V_{CC}	Only CPU operates	V_{CC}	Halted	System cl crystal
Subsleep mode	V_{CC}	Only timers operate, CPU stops	V_{CC}	Halted	Subclock crystal
Watch mode	V_{CC}	Only time base operates, CPU stops	V_{CC}	Halted	
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Halted	System cl crystal
					Subclock Pin X_1 = C

2. Excludes current in pull-up MOS transistors and output buffers.
3. The maximum current consumption value (standard) is 1.1 × typ.

System clock oscillation frequency	f_{OSC}	OSC_1, OSC_2	2	—	10	MHz	V_{CC} = 2.7 V to 3.6 V
			2	—	4		V_{CC} = 1.8 V to 3.6 V
OSC clock (ϕ_{OSC}) cycle time	t_{OSC}	OSC_1, OSC_2	100	—	500 (1000)	ns	V_{CC} = 2.7 V to 3.6 V
			250	—	500 (1000)		V_{CC} = 1.8 V to 3.6 V
System clock (ϕ) cycle time	t_{cyc}		2	—	128	t_{OSC}	
			—	—	128	µs	
Subclock oscillation frequency	f_W	X_1, X_2	—	32.768 or 38.4	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X_1, X_2	—	30.5 or 26.0	—	µs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}		2	—	8	t_W	
Instruction cycle time			2	—	—	t_{cyc} t_{subcyc}	

				8.0	2	ms	Crystal Oscillator Parameters V_{CC} = 2.7 V to 3.6
			—	1.2	3		Crystal Oscillator Parameters V_{CC} = 2.2 V to 3.6
			—	4.0	—		Crystal Oscillator Parameters Except the above
			—	—	50		Except the above
		X1, X2	—	—	2	s	V_{CC} = 2.2 V to 3.6
			—	4	—		Except the above
External clock high width	t_{CPH}	OSC_1	40	—	—	ns	V_{CC} = 2.7 V to 3.6
			100	—	—		V_{CC} = 1.8 V to 3.6
		X_1	—	15.26 or 13.02	—	µs	
External clock low width	t_{CPL}	OSC_1	40	—	—	ns	V_{CC} = 2.7 V to 3.6
			100	—	—		V_{CC} = 1.8 V to 3.6
		X_1	—	15.26 or 13.02	—	µs	
External clock rise time	t_{CPr}	OSC_1	—	—	10	ns	V_{CC} = 2.7 V to 3.6
			—	—	25		V_{CC} = 1.8 V to 3.6
		X_1	—	—	55.0		
External clock fall time	t_{CPf}	OSC_1	—	—	10	ns	V_{CC} = 2.7 V to 3.6
			—	—	25		V_{CC} = 1.8 V to 3.6
		X_1	—	—	55.0		
Pin $\overline{RES}$ low width	t_{REL}	$\overline{RES}$	10	—	—	t_{cyc}	

Input pin low width	t_{IL}	$\overline{IRQ_0}$ to $\overline{IRQ_4}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIC, TMIF, TMIG, AEVL, AEVH	2	—	—	t_{cyc} t_{subcyc}
UD pin minimum modulation width	t_{UDH} t_{UDL}	UD	4	—	—	t_{cyc} t_{subcyc}

Notes: 1. Selected with SA1 and SA0 of system clock control register 2 (SYSCR2).
2. Figures in parentheses are the maximum t_{OSC} rate with external clock input.

Table 15.18 Serial Interface (SCI3-1, SCI3-2) Timing

Item		Symbol	Values Min	Typ	Max	Unit	Test Conditions	R Fi
Input clock cycle	Asynchronous	t_{scyc}	4	—	—	t_{cyc} or t_{subcyc}		Fi
	Synchronous		6	—	—			
Input clock pulse width		t_{SCKW}	0.4	—	0.6	t_{scyc}		Fi
Transmit data delay time (synchronous)		t_{TXD}	—	—	1	t_{cyc} or t_{subcyc}		Fi
Receive data setup time (synchronous)		t_{RXS}	400.0	—	—	ns		Fi
Receive data hold time (synchronous)		t_{RXH}	400.0	—	—	ns		Fi

Analog power supply voltage	AV_{CC}	AV_{CC}	1.8	—	3.6	V	
Analog input voltage	AV_{IN}	AN_0 to AN_7	– 0.3	—	AV_{CC} + 0.3	V	
Analog power supply current	AI_{OPE}	AV_{CC}	—	—	1.2	mA	AV_{CC} = 3.0 V
	AI_{STOP1}	AV_{CC}	—	600	—	µA	
	AI_{STOP2}	AV_{CC}	—	—	5	µA	
Analog input capacitance	C_{AIN}	AN_0 to AN_7	—	—	15.0	pF	
Allowable signal source impedance	R_{AIN}		—	—	10.0	kΩ	
Resolution (data length)			—	—	10	bit	
Nonlinearity error			—	—	±3.5	LSB	AV_{CC} = 2.7 V to 3.6 V V_{CC} = 2.7 V to 3.6 V
			—	—	±5.5		AV_{CC} = 2.0 V to 3.6 V V_{CC} = 2.0 V to 3.6 V
			—	—	±7.5		Except the above
Quantization error			—	—	±0.5	LSB	

time					V_{CC} = 2.7 V to 3.6 V
	62	—	124		Except the above

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 id
3. AI_{STOP2} is the current at reset and in standby, watch, subactive, and subslee while the A/D converter is idle.
4. Conversion time 62 µs

Segment driver drop voltage	V_{DS}	SEG_1 to SEG_{32}	I_D = 2 µA V_1 = 2.7 to 3.6 V	—	—	0.6	V
Common driver drop voltage	V_{DC}	COM_1 to COM_4	I_D = 2 µA V_1 = 2.7 to 3.6 V	—	—	0.3	V
LCD power supply split-resistance	R_{LCD}		Between V_1 and V_{SS}	1.5	3.5	7	MΩ
Liquid crystal display voltage	V_{LCD}	V_1		2.2	—	3.6	V

Notes: 1. The voltage drop from power supply pins V_1, V_2, V_3, and V_{SS} to each segme common pin.

2. When the liquid crystal display voltage is supplied from an external power s ensure that the following relationship is maintained: $V_1 \geq V_2 \geq V_3 \geq V_{SS}$.

Table 15.21 AC Characteristics for External Segment Expansion

Item	Symbol	Applicable Pins	Test Conditions	Values			Unit
				Min	Typ	Max	
Clock high width	t_{CWH}	CL_1, CL_2	*	800.0	—	—	ns
Clock low width	t_{CWL}	CL_2	*	800.0	—	—	ns
Clock setup time	t_{CSU}	CL_1, CL_2	*	500.0	—	—	ns
Data setup time	t_{SU}	DO	*	300.0	—	—	ns
Data hold time	t_{DH}	DO	*	300.0	—	—	ns
M delay time	t_{DM}	M		–1000.0	—	1000.0	ns
Clock rise and fall times	t_{CT}	CL_1, CL_2		—	—	170.0	ns

Note: * Value when the frame frequency is set to between 30.5 Hz and 488 Hz.

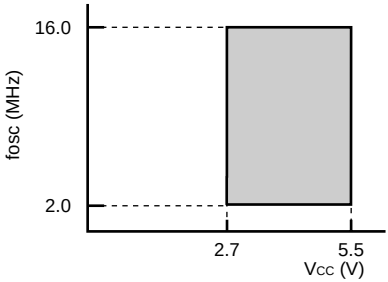
		CV_{CC}	–0.3 to +4.3	V
Analog power supply voltage		AV_{CC}	–0.3 to +7.0	V
Input voltage	Other than port B	V_{in}	–0.3 to V_{CC} +0.3	V
	Port B	AV_{in}	–0.3 to AV_{CC} +0.3	V
Operating temperature		T_{opr}	–20 to +75[*2] (regular specifications)	°C
			–40 to +85[*2] (wide-range temperature specifications)	
			+75[*3] (chip shipment specifications)	
Storage temperature		T_{stg}	–55 to +125	°C

Notes: 1. Permanent damage may result if maximum ratings are exceeded. Normal op
should be under the conditions specified in Electrical Characteristics. Exceed
values can result in incorrect operation and reduced reliability.

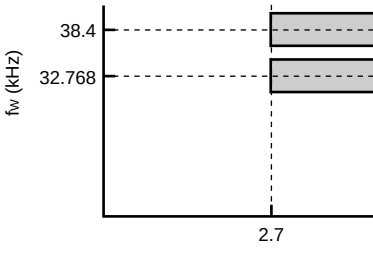
2. The operating temperature ranges from –20°C to +75°C when programming
the flash memory.

3. The temperature range in which power may be applied to the device is –20°

• H8/38327 Group

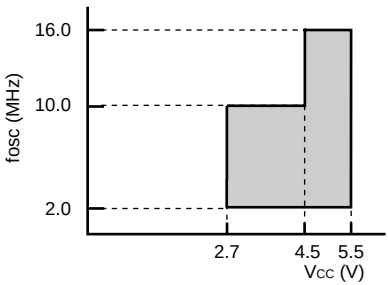


• Active (high-speed) mode
• Sleep (high-speed) mode

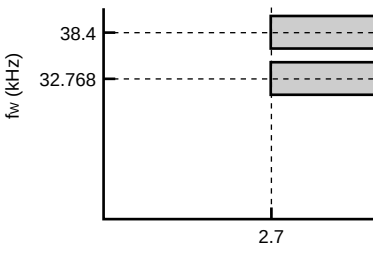


• All operating modes

• H8/38427 Group

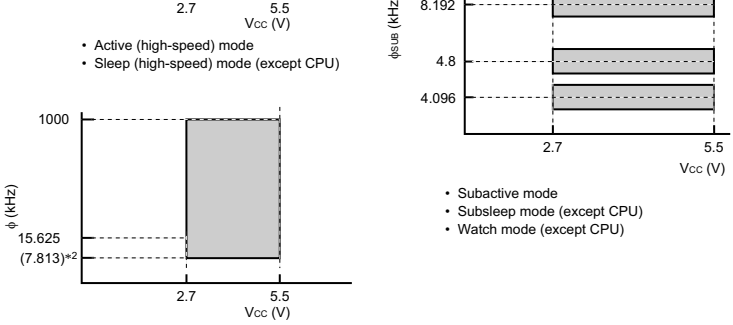


• Active (high-speed) mode
• Sleep (high-speed) mode



• All operating modes

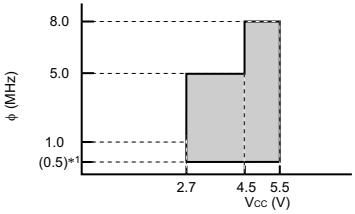
Note: fosc is the oscillator frequency. When an external clock is used 1 MHz is the m
fosc value.



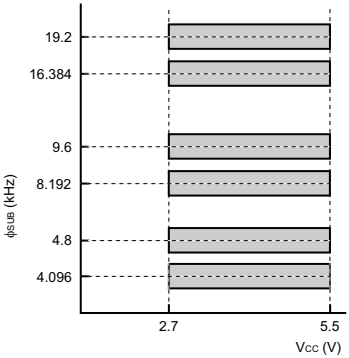
- Subactive mode
- Subsleep mode (except CPU)
- Watch mode (except CPU)

- Active (medium-speed) mode
- Sleep (medium-speed) mode (except A/D converter)

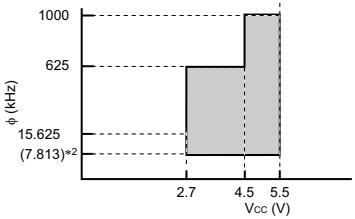
- H8/38427 Group



- Active (high-speed) mode
- Sleep (high-speed) mode (except CPU)

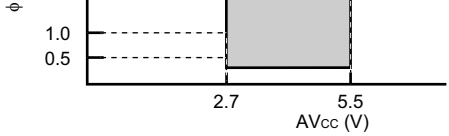


- Subactive mode
- Subsleep mode (except CPU)
- Watch mode (except CPU)

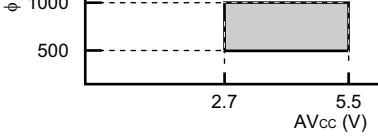


- Active (medium-speed) mode
- Sleep (medium-speed) mode (except A/D converter)

Notes 1. The figure in parentheses ( ) indicates the minimum operating frequency when an external clock is used. When the resonator is used the minimum operating frequency (φ) is 1 MHz.
2. The figure in parentheses ( ) indicates the minimum operating frequency when an external clock is used. When the resonator is used the minimum operating frequency (φ) is 15.625 kHz.

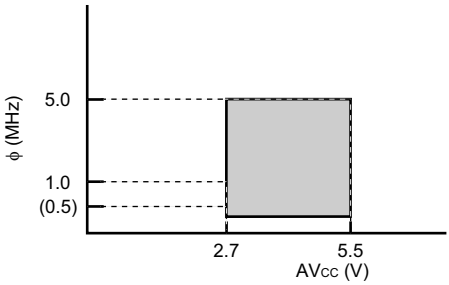


- Active (high-speed) mode
- Sleep (high-speed) mode

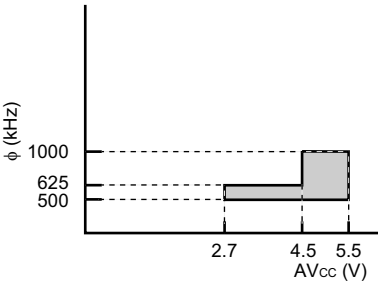


- Active (medium-speed) mode
- Sleep (medium-speed) mode

- H8/38427 Group



- Active (high-speed) mode
- Sleep (high-speed) mode



- Active (medium-speed) mode
- Sleep (medium-speed) mode

Item	Symbol	Applicable Pins	Min	Typ	Max	Unit	Test Condition
Input high voltage	V_{IH}	$\overline{RES}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{IRQ_0}$ to $\overline{IRQ_4}$, AEVL, AEVH, TMIC, TMIF, TMIG, ADTRG, SCK_{31}, SCK_{32}	$V_{CC} \times 0.8$	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5
			$V_{CC} \times 0.9$	—	V_{CC} + 0.3		Other than abo
		RXD_{31}, RXD_{32}, UD	$V_{CC} \times 0.7$	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5
			$V_{CC} \times 0.8$	—	V_{CC} + 0.3		Other than abo
		OSC_1	$V_{CC} \times 0.8$	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5
			$V_{CC} \times 0.9$	—	V_{CC} + 0.3		Other than abo
		$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	$V_{CC} \times 0.7$	—	V_{CC} + 0.3	V	V_{CC} = 4.0 V to 5
			$V_{CC} \times 0.8$	—	V_{CC} + 0.3		Other than abo
		PB_0 to PB_7	$V_{CC} \times 0.7$	—	AV_{CC} + 0.3	V	V_{CC} = 4.0 V to 5
			$V_{CC} \times 0.8$	—	AV_{CC} + 0.3		Other than abo
		EXCL	$V_{CC} \times 0.9$	—	V_{CC} + 0.3	V	

Note: Connect the TEST pin to V_{SS}.

		RXD_{31}, RXD_{32}, UD	– 0.3	—	$V_{CC} \times 0.3$	V	V_{CC} = 4.0 V to
			– 0.3	—	$V_{CC} \times 0.2$		Other than ab
		OSC_1	– 0.3	—	$V_{CC} \times 0.2$	V	V_{CC} = 4.0 V to
			– 0.3	—	$V_{CC} \times 0.1$		Other than ab
		EXCL	– 0.3	—	0.1 V_{CC}	V	
		$P1_0$ to $P1_7$ $P3_0$ to $P3_7$, $P4_0$ to $P4_3$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3, PB_0 to PB_7	– 0.3	—	$V_{CC} \times 0.3$	V	V_{CC} = 4.0 V to
			– 0.3	—	$V_{CC} \times 0.2$		Other than ab
Output high voltage	V_{OH}	$P1_0$ to $P1_7$ $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	V_{CC} – 1.0	—	—	V	V_{CC} = 4.0 V to $-I_{OH}$ = 1.0 mA
			V_{CC} – 0.5	—	—		V_{CC} = 4.0 V to $-I_{OH}$ = 0.5 mA
			V_{CC} – 0.3	—	—		$-I_{OH}$ = 0.1 mA

		$P3_0$ to $P3_7$	—	—	1.0		V_{CC} = 4.0 V to 5
							I_{OL} = 10 mA
			—	—	0.6		V_{CC} = 4.0 V to 5
							I_{OL} = 1.6 mA
			—	—	0.5		I_{OL} = 0.4 mA
Input/ output leakage current	| I_{IL} |	$\overline{RES}$, $P4_3$, $P1_0$ to $P1_7$, OSC_1, X_1, $P3_0$ to $P3_7$, $P4_0$ to $P4_2$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, PA_0 to PA_3	—	—	1.0	μA	V_{IN} = 0.5 V to V 0.5 V
		PB_0 to PB_7	—	—	1.0		V_{IN} = 0.5 V to A – 0.5 V
Pull-up MOS current	$-I_p$	$P1_0$ to $P1_7$, $P3_0$ to $P3_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$	20	—	200	μA	V_{CC} = 5.0 V, V_{IN} = 0.0 V
			—	40	—		V_{CC} = 2.7 V, V_{IN} = 0.0 V
Input capaci-tance	C_{in}	All input pins except power supply pin	—	—	15.0	pF	f = 1 MHz, V_{IN} = 0.0 V, T_a = 25°C

—	1.0	—	Active (high-speed mode V_{CC} = 5 V, f_{OSC} = 2 MHz
—	1.5	—	
—	2.0	—	Active (high-speed mode V_{CC} = 5 V, f_{OSC} = 4 MHz
—	2.4	—	
—	4.0	7.0	Active (high-speed mode V_{CC} = 5 V, f_{OSC} = 10 MHz
—	4.9	7.0	

—	0.5	—	Active (medium-speed) mode V_{CC} = 5 V, f_{OSC} = 2 MHz, ϕ_{OSC}/128
—	1.0	—	
—	0.8	—	Active (medium-speed) mode V_{CC} = 5 V, f_{OSC} = 4 MHz, ϕ_{OSC}/128
—	1.2	—	
—	1.2	3.0	Active (medium-speed) mode V_{CC} = 5 V, f_{OSC} = 10 MHz, ϕ_{OSC}/128
—	1.7	3.0	

			—	0.7	—		V_{CC} = 5 V, f_{OSC} = 2 MHz
			—	1.2	—		
			—	1.1	—		V_{CC} = 5 V, f_{OSC} = 4 MHz
			—	1.6	—		
			—	1.9	5.0		V_{CC} = 5 V, f_{OSC} = 10 MHz
			—	2.6	5.0		
Subactive mode current consump-tion	I_{SUB}	V_{CC}	—	12	—	µA	V_{CC} = 2.7 V, LCD on, 32-kHz crystal resonator used ($\phi_{SUB} = \phi_W/8$)
			—	15	—		
			—	18	50		V_{CC} = 2.7 V, LCD on, 32-kHz crystal resonator used ($\phi_{SUB} = \phi_W/2$)
			—	30	50		

consumption							32-kHz crystal resonator used, LCD not used
			—	1.8	—		
			—	3.0	6.0		V_{CC} = 2.7 V, 32-kHz crystal resonator used, LCD not used
Standby mode current consumption	I_{STBY}	V_{CC}	—	0.3	—	µA	V_{CC} = 2.7 V, T_a = 25°C, 32-kHz crystal resonator not used
			—	0.3	—		V_{CC} = 2.7 V, T_a = 25°C, 32-kHz crystal resonator not used
			—	0.4	—		V_{CC} = 5.0 V, T_a = 25°C, 32-kHz crystal resonator not used
			—	0.5	—		
			—	1.0	5.0		32-kHz crystal resonator not used
RAM data retaining voltage	V_{RAM}	V_{CC}	2.0	—	—	V	
Allowable output low current (per pin)	I_{OL}	Output pins except port 3	—	—	2.0	mA	V_{CC} = 4.0 V to 5.5 V
		Port 3	—	—	10.0		V_{CC} = 4.0 V to 5.5 V
		All output pins	—	—	0.5		
Allowable output low current (total)	$\sum I_{OL}$	Output pins except port 3	—	—	40.0	mA	V_{CC} = 4.0 V to 5.5 V
		Port 3	—	—	80.0		V_{CC} = 4.0 V to 5.5 V
		All output pins	—	—	20.0		

high current (total)					

Notes: Connect the TEST pin to V_{SS}.

1. Applies to the mask-ROM version.
2. Applies to the F-ZTAT version.
3. Pin states when current consumption is measured

Mode	$\overline{RES}$ Pin	Internal State	Other Pins	LCD Power Supply	Oscillat
Active (high-speed) mode (I_{OPE1})	V_{CC}	Only CPU operates	V_{CC}	Stops	System crystal r
Active (medium-speed) mode (I_{OPE2})					Subcloc Pin X_1 =
Sleep mode	V_{CC}	Only all on-chip timers operate	V_{CC}	Stops	
Subactive mode	V_{CC}	Only CPU operates	V_{CC}	Stops	System crystal r
Subsleep mode	V_{CC}	Only all on-chip timers operate CPU stops	V_{CC}	Stops	Subcloc crystal r
Watch mode	V_{CC}	Only clock time base operates CPU stops	V_{CC}	Stops	
Standby mode	V_{CC}	CPU and timers both stop	V_{CC}	Stops	System crystal r
					Subcloc Pin X_1 =

4. Except current which flows to the pull-up MOS or output buffer
5. Voltage maintained in standby mode

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition
System clock oscillation frequency	f_{OSC}	OSC_1, OSC_2	2.0	—	16.0	MHz	
			2.0	—	16.0		V_{CC} = 4.5 to 5.5 V
			2.0	—	10.0		V_{CC} = 2.7 to 5.5 V
OSC clock (ϕ_{OSC}) cycle time	t_{OSC}	OSC_1, OSC_2	62.5	—	500 (1000)	ns	
			62.5	—	500 (1000)		V_{CC} = 4.5 to 5.5 V
			100	—	500 (1000)		V_{CC} = 2.7 to 5.5 V
System clock (ϕ) cycle time	t_{cyc}		2	—	128	t_{OSC}	
			—	—	128	µs	
Subclock oscillation frequency	f_W	X_1, X_2, EXCL	—	32.768 or 38.4	—	kHz	
Watch clock (ϕ_W) cycle time	t_W	X_1, X_2, EXCL	—	30.5 or 26.0	—	µs	
Subclock (ϕ_{SUB}) cycle time	t_{subcyc}		2	—	4	t_W	
Instruction cycle time			2	—	—	t_{cyc} t_{subcyc}	
Oscillation stabilization time	t_{rc}	OSC_1, OSC_2	—	20	45	µs	Ceramic resonator (V_{CC} = 3.0 to 5.5 V)
			—	80	—		Ceramic resonator other than above
			—	0.8	2	ms	Crystal resonator
			—	—	50		Other than above
	t_{rc}	X_1, X_2	—	—	2.0	s	

External clock low width	t_{CPL}	OSC_1	25	—	—	ns	
			25	—	—		V_{CC} = 4.5 to 5.5 V
			40	—	—		V_{CC} = 2.7 to 5.5 V
		EXCL	—	15.26 or 13.02	—	µs	
External clock rise time	t_{CPr}	OSC_1	—	—	6	ns	
			—	—	6		V_{CC} = 4.5 to 5.5 V
			—	—	10		V_{CC} = 2.7 to 5.5 V
		EXCL	—	—	55.0		
External clock fall time	t_{CPf}	OSC_1	—	—	6	ns	
			—	—	6		V_{CC} = 4.5 to 5.5 V
			—	—	10		V_{CC} = 2.7 to 5.5 V
		EXCL	—	—	55.0		
$\overline{RES}$ pin low width	t_{REL}	$\overline{RES}$	10	—	—	t_{cyc}	
Input pin high width	t_{IH}	$\overline{IRQ_0}$ to $\overline{IRQ_4}$, $\overline{WKP_0}$ to $\overline{WKP_7}$, $\overline{ADTRG}$, TMIC, TMIF, TMIG	2	—	—	t_{cyc} t_{subcyc}	
		AEVL, AEVH	32	—	—	ns	

UD pin minimum transition width	t_{UDH} t_{UDL}	UD	4	—	—	t_{cyc} t_{subcyc}	

Notes: 1. Determined by the SA1 and SA0 bits in the system control register 2 (SYSCR
2. The figure in parentheses ( ) indicates the maximum fosc value when an ext is used.
3. Also applies to H8/38327 Group.
4. Also applies to H8/38427 Group.

Table 15.25 Serial Interface (SCI3) 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, unless otherwise spe

Item		Symbol	Values Min	Typ	Max	Unit	Test Condition	Ref Figu
Input clock cycle	Asynchronous	t_{scyc}	4	—	—	t_{cyc} or t_{subcyc}		Figu
	Clocked synchronous		6	—	—			
Input clock pulse width		t_{SCKW}	0.4	—	0.6	t_{scyc}		Figu
Transmit data delay time (clocked synchronous)		t_{TXD}	—	—	1	t_{cyc} or t_{subcyc}		Figu
Receive data setup time (clocked synchronous)		t_{RXS}	200	—	—	ns		Figu
Receive data hold time (clocked synchronous)		t_{RXH}	200	—	—	ns		Figu

Item	Symbol	Applicable Pins	Min	Typ	Max	Unit	Test Condition
Analog power supply voltage	AV_{CC}	AV_{CC}	2.7	—	5.5	V	
Analog input voltage	AV_{IN}	AN_0 to AN_3	– 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}	—	600	—	µA	
	AI_{STOP2}	AV_{CC}	—	—	5.0	µA	
Analog input capacitance	C_{AIN}	AN_0 to AN_7	—	—	15.0	pF	
Allowable signal source impedance	R_{AIN}		—	—	10.0	kΩ	
Resolution (data length)			—	—	10	bit	
Nonlinearity error			—	—	±3.5	LSB	AV_{CC} = 4.0 V to 5.5 V
			—	—	±7.5		AV_{CC} = 2.7 V to 5.5 V
Quantization error			—	—	±0.5	LSB	
Absolute accuracy			—	±2.0	±4.0	LSB	AV_{CC} = 4.0 V to 5.5 V
			—	±2.0	±8.0		AV_{CC} = 2.7 V to 5.5 V
Conversion time			7.8	—	124	µs	
			12.4	—	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 i
3. AI_{STOP2} is the current at reset and in standby, watch, subactive, and subsle while the A/D converter is idle.
4. Also applies to H8/38327 Group.
5. Also applies to H8/38427 Group.

Item	Symbol	Pins	Min	Typ	Max	Unit	Test Condition
Segment driver step-down voltage	V_{DS}	SEG_1 to SEG_{32}	—	—	0.6	V	I_D = 2 µA V1 = 2.7 V to 5.5 V
Common driver step-down voltage	V_{DC}	COM_1 to COM_4	—	—	0.3	V	I_D = 2 µA V1 = 2.7 V to 5.5 V
LCD power supply split-resistance	R_{LCD}		1.5	3.0	7.0	MΩ	Between V1 and V_{SS}
Liquid crystal display voltage	V_{LCD}	V_1	2.7	—	5.5	V	

Notes: 1. The voltage step-down from power supply pins V1, V2, V3, and V_{SS} to each pin or common pin.

2. When the liquid crystal display voltage is supplied from an external power su ensure that the following relationship is maintained: $V1 \geq V2 \geq V3 \geq V_{SS}$.

Item		Symbol	Values			Unit	T C
			Min	Typ	Max		
Programming time*1*2*4		t_P	—	7	200	ms/128 bytes	
Erase time*1*3*5		t_E	—	100	1200	ms/block	
Reprogramming count		N_{WEC}	1000*8	10000*9	—	times	
Data retain period		t_{DRP}	10*10	—	—	year	
Programming	Wait time after SWE-bit setting*1	x	1	—	—	µs	
	Wait time after PSU-bit setting*1	y	50	—	—	µs	
	Wait time after P-bit setting*1*4	z1	28	30	32	µs	1
		z2	198	200	202	µs	7
		z3	8	10	12	µs	A pu
	Wait time after P-bit clear*1	α	5	—	—	µs	
	Wait time after PSU-bit clear*1	β	5	—	—	µs	
	Wait time after PV-bit setting*1	γ	4	—	—	µs	
	Wait time after dummy write*1	ε	2	—	—	µs	
	Wait time after PV-bit clear*1	η	2	—	—	µs	
	Wait time after SWE-bit clear*1	θ	100	—	—	µs	
	Maximum programming count*1*4*5	N	—	—	1000	times	

Wait time after E-bit clear*1	α	10	—	—	µs
Wait time after ESU-bit clear*1	β	10	—	—	µs
Wait time after EV-bit setting*1	γ	20	—	—	µs
Wait time after dummy write*1	ε	2	—	—	µs
Wait time after EV-bit clear*1	η	4	—	—	µs
Wait time after SWE-bit clear*1	θ	100	—	—	µs
Maximum erase count*1*6*7	N	—	—	120	times

Notes:
1. Set the times according to the program/erase algorithms.
2. Programming time per 128 bytes (Shows the total period for which the P bit in FLMCR does not include the programming verification time.)
3. Block erase time (Shows the total period for which the E bit in FLMCR1 is set. It does the erase verification time.)
4. Maximum programming time (t_P (max))
   t_P (max) = Wait time after P-bit setting (z) × maximum number of writes (N)
5. The maximum number of writes (N) should be set according to the actual set value of z3 to allow programming within the maximum programming time (t_P (max)).
   The wait time after P-bit setting (z1 and z2) should be alternated according to the num (n) as follows:
   $1 \leq n \leq 6$ z1 = 30 µs
   $7 \leq n \leq 1000$ z2 = 200 µs
6. Maximum erase time (t_E (max))
   t_E (max) = Wait time after E-bit setting (z) × maximum erase count (N)
7. The maximum number of erases (N) should be set according to the actual set value o erasing within the maximum erase time (t_E (max)).
8. This minimum value guarantees all characteristics after reprogramming (the guarante from 1 to the minimum value).
9. Reference value when the temperature is 25°C (normally reprogramming will be perfo count).
10. This is a data retain characteristic when reprogramming is performed within the specif including this minimum value.

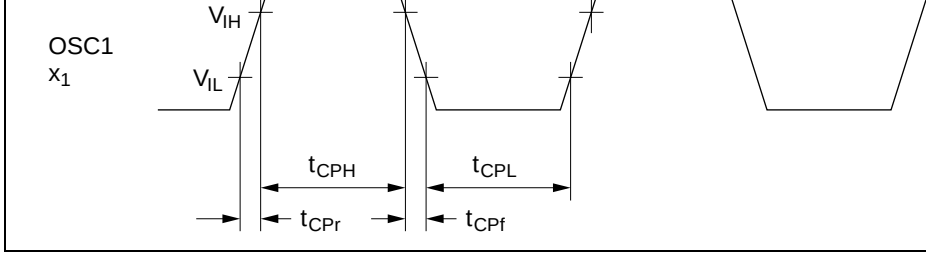


Figure 15.1 Clock Input Timing

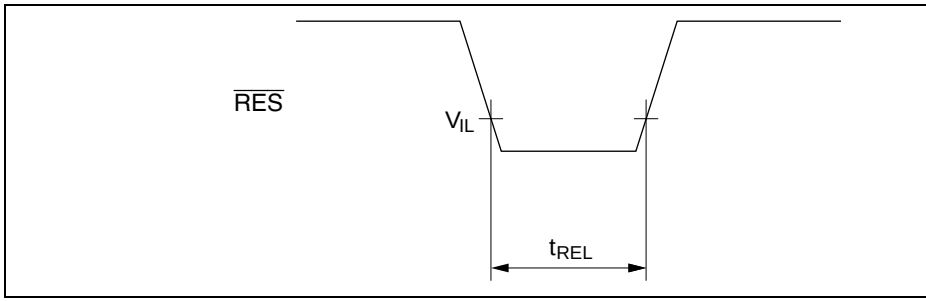


Figure 15.2 $\overline{\text{RES}}$ Low Width

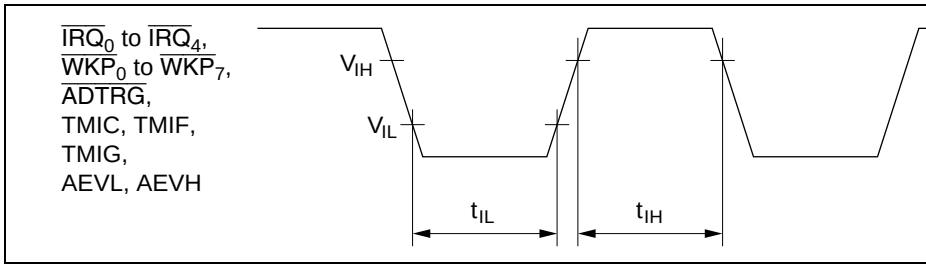


Figure 15.3 Input Timing

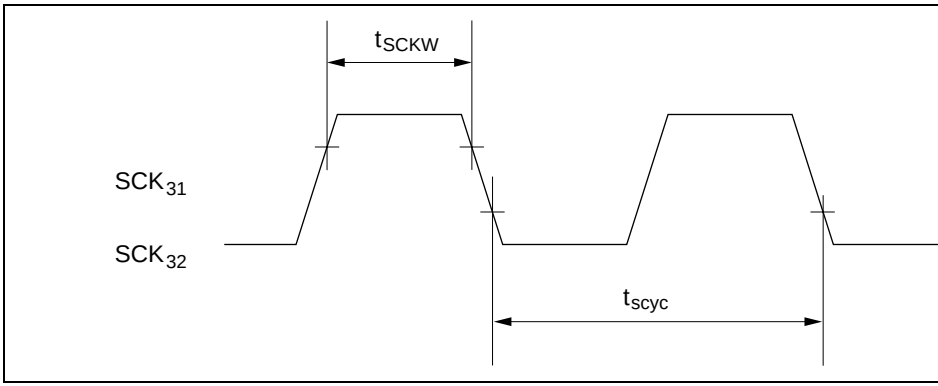
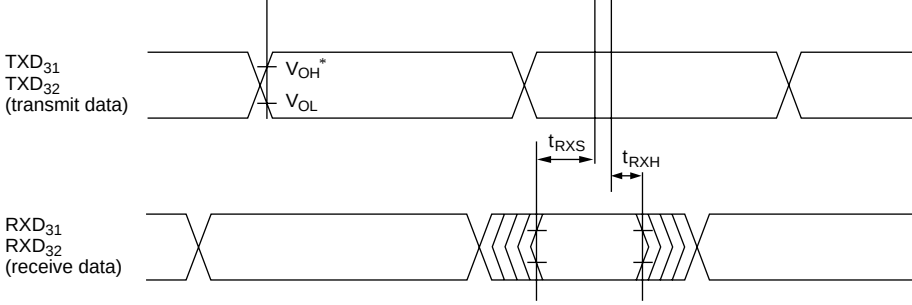


Figure 15.5 SCK3 Input Clock Timing



Note: * Output timing reference levels

Output high	V_{OH} = 1/2Vcc + 0.2 V
Output low	V_{OL} = 0.8 V

Load conditions are shown in figure 15.8.

Figure 15.6 SCI3 Synchronous Mode Input/Output Timing

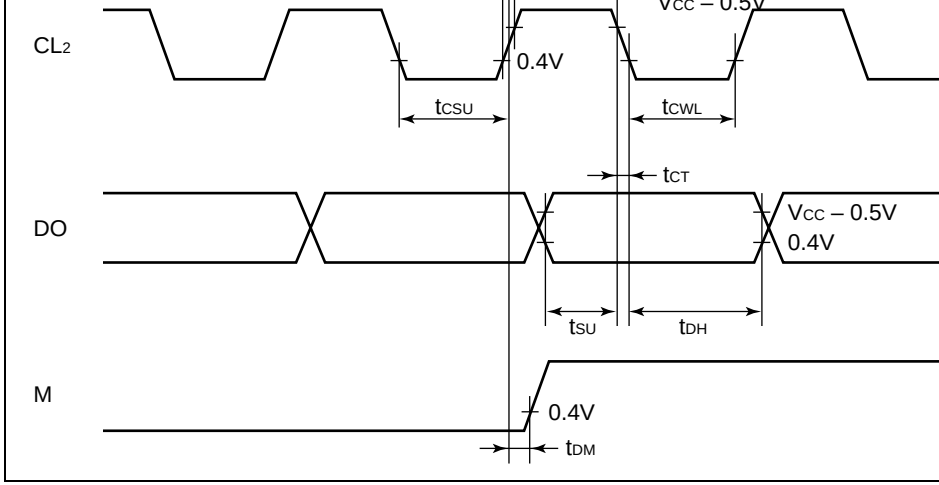


Figure 15.7 Segment Expansion Signal Timing

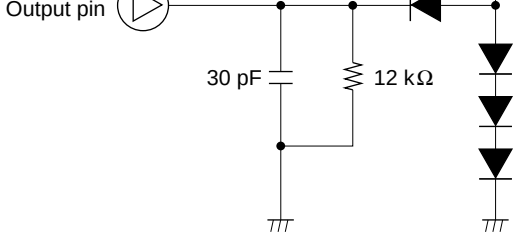
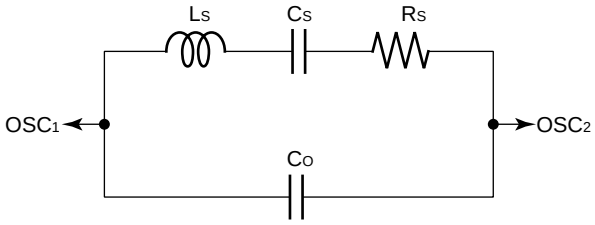


Figure 15.8 Output Load Condition

15.11 Resonator



Ceramic Oscillator Parameters

Frequency	4 MHz	Manufacturer	Products Nar
R_S	Manufacturer's Publicly Released Values Max. 8.8	MURATA	CSTLS 4M00G 53/56
C_O	Max. 36 pF		

Crystal Oscillator Parameters

Frequency	4.193 MHz	Manufacturer	Products Nar
R_S	Manufacturer's Publicly Released Values Max. 100	Nihon Denpa Kogyo	NR-18
C_O	Max. 16 pF		

Figure 15.9 Resonator Equivalent Circuit

Ceramic resonator

Resonating Frequency	Manufacturer	Model	C_1, C_2
2 MHz	MURATA	CSTCC2M00G53-B0	15pF ± 20%
		CSTCC2M00G56-B0	47pF ± 20%
4 MHz		CSTLS4M00G53-B0	15pF ± 20%
		CSTLS4M00G56-B0	47pF ± 20%
10 MHz		CSTLS10M0G53-B0	15pF ± 20%
		CSTLS10M0G56-B0	47pF ± 20%

Figure 15.10 Resonator Equivalent Circuit

15.12 Usage Note

Each of the products covered in this manual satisfy the electrical characteristics indicat
However, the actual electrical characteristics, operating margin and noise margin may c
the indicated values due to differences in the manufacturing process, built-in ROM, lay
and other factors.

If a system evaluation test is conducted with the ZTAT or F-ZTAT version, when switc
mask ROM version, perform the same evaluation test with the mask ROM version.

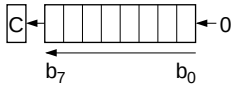
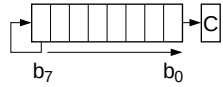
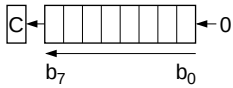
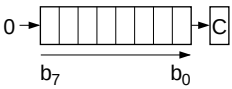
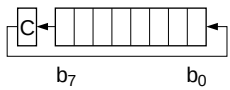
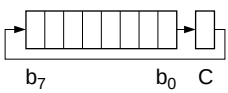
Rs8/16	General register (source) (8 or 16 bits)
Rn8/16	General register (8 or 16 bits)
CCR	Condition code register
N	N (negative) flag in CCR
Z	Z (zero) flag in CCR
V	V (overflow) flag in CCR
C	C (carry) flag in CCR
PC	Program counter
SP	Stack pointer
#xx: 3/8/16	Immediate data (3, 8, or 16 bits)
d: 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

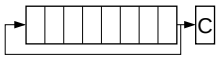
Condition Code Notation

Symbol	
↕	Modified according to the instruction result
*	Not fixed (value not guaranteed)
0	Always cleared to 0
—	Not affected by the instruction execution result

Mnemonic	Operation	#xx:8	Rn	@Rn	@(d:16, Rn)	@–Rn/@Rn+	@aa:8/@aa:16	@(d:16, PC)	@@aa:8	Implied	Condition Code I	H	N
MOV.B #xx:8, Rd	#xx:8 → Rd8	2									—	—	↕
MOV.B Rs, Rd	Rs8 → Rd8		2								—	—	↕
MOV.B @Rs, Rd	@Rs16 → Rd8			2							—	—	↕
MOV.B @(d:16, Rs), Rd	@(d:16, Rs16)→ Rd8				4						—	—	↕
MOV.B @Rs+, Rd	@Rs16 → Rd8 Rs16+1 → Rs16					2					—	—	↕
MOV.B @aa:8, Rd	@aa:8 → Rd8						2				—	—	↕
MOV.B @aa:16, Rd	@aa:16 → Rd8						4				—	—	↕
MOV.B Rs, @Rd	Rs8 → @Rd16			2							—	—	↕
MOV.B Rs, @(d:16, Rd)	Rs8 → @(d:16, Rd16)				4						—	—	↕
MOV.B Rs, @–Rd	Rd16–1 → Rd16 Rs8 → @Rd16					2					—	—	↕
MOV.B Rs, @aa:8	Rs8 → @aa:8						2				—	—	↕
MOV.B Rs, @aa:16	Rs8 → @aa:16						4				—	—	↕
MOV.W #xx:16, Rd	#xx:16 → Rd	4									—	—	↕
MOV.W Rs, Rd	Rs16 → Rd16		2								—	—	↕
MOV.W @Rs, Rd	@Rs16 → Rd16			2							—	—	↕
MOV.W @(d:16, Rs), Rd	@(d:16, Rs16) → Rd16				4						—	—	↕
MOV.W @Rs+, Rd	@Rs16 → Rd16 Rs16+2 → Rs16					2					—	—	↕
MOV.W @aa:16, Rd	@aa:16 → Rd16						4				—	—	↕
MOV.W Rs, @Rd	Rs16 → @Rd16			2							—	—	↕
MOV.W Rs, @(d:16, Rd)	Rs16 → @(d:16, Rd16)				4						—	—	↕
MOV.W Rs, @–Rd	Rd16–2 → Rd16 Rs16 → @Rd16					2					—	—	↕
MOV.W Rs, @aa:16	Rs16 → @aa:16						4				—	—	↕
POP Rd	@SP → Rd16 SP+2 → SP					2					—	—	↕
PUSH Rs	SP–2 → SP Rs16 → @SP					2					—	—	↕

ADD.W Rs, Rd	W	Rd16+Rs16 → Rd16		2							—	(1)	↕
ADDX.B #xx:8, Rd	B	Rd8+#xx:8 +C → Rd8	2								—	↕	↕
ADDX.B Rs, Rd	B	Rd8+Rs8 +C → Rd8		2							—	↕	↕
ADDS.W #1, Rd	W	Rd16+1 → Rd16		2							—	—	—
ADDS.W #2, Rd	W	Rd16+2 → Rd16		2							—	—	—
INC.B Rd	B	Rd8+1 → Rd8		2							—	—	↕
DAA.B Rd	B	Rd8 decimal adjust → Rd8		2							—	*	↕
SUB.B Rs, Rd	B	Rd8–Rs8 → Rd8		2							—	↕	↕
SUB.W Rs, Rd	W	Rd16–Rs16 → Rd16		2							—	(1)	↕
SUBX.B #xx:8, Rd	B	Rd8–#xx:8 –C → Rd8	2								—	↕	↕
SUBX.B Rs, Rd	B	Rd8–Rs8 –C → Rd8		2							—	↕	↕
SUBS.W #1, Rd	W	Rd16–1 → Rd16		2							—	—	—
SUBS.W #2, Rd	W	Rd16–2 → Rd16		2							—	—	—
DEC.B Rd	B	Rd8–1 → Rd8		2							—	—	↕
DAS.B Rd	B	Rd8 decimal adjust → Rd8		2							—	*	↕
NEG.B Rd	B	0–Rd → Rd		2							—	↕	↕
CMP.B #xx:8, Rd	B	Rd8–#xx:8	2								—	↕	↕
CMP.B Rs, Rd	B	Rd8–Rs8		2							—	↕	↕
CMP.W Rs, Rd	W	Rd16–Rs16		2							—	(1)	↕

		(RdH: remainder, RdL: quotient)												
AND.B #xx:8, Rd	B	Rd8∧#xx:8 → Rd8	2								—	—	↕
AND.B Rs, Rd	B	Rd8∧Rs8 → Rd8		2							—	—	↕
OR.B #xx:8, Rd	B	Rd8∨#xx:8 → Rd8	2								—	—	↕
OR.B Rs, Rd	B	Rd8∨Rs8 → Rd8		2							—	—	↕
XOR.B #xx:8, Rd	B	Rd8⊕#xx:8 → Rd8	2								—	—	↕
XOR.B Rs, Rd	B	Rd8⊕Rs8 → Rd8		2							—	—	↕
NOT.B Rd	B	$\overline{\text{Rd}} \rightarrow$ Rd		2							—	—	↕
SHAL.B Rd	B			2							—	—	↕
SHAR.B Rd	B			2							—	—	↕
SHLL.B Rd	B			2							—	—	↕
SHLR.B Rd	B			2							—	—	0
ROTXL.B Rd	B			2							—	—	↕
ROTXR.B Rd	B			2							—	—	↕

		b_7 b_0												
ROTR.B Rd	B	 b_7 b_0		2								—	—	↕
BSET #xx:3, Rd	B	(#xx:3 of Rd8) ← 1		2								—	—	—
BSET #xx:3, @Rd	B	(#xx:3 of @Rd16) ← 1			4							—	—	—
BSET #xx:3, @aa:8	B	(#xx:3 of @aa:8) ← 1						4				—	—	—
BSET Rn, Rd	B	(Rn8 of Rd8) ← 1		2								—	—	—
BSET Rn, @Rd	B	(Rn8 of @Rd16) ← 1			4							—	—	—
BSET Rn, @aa:8	B	(Rn8 of @aa:8) ← 1						4				—	—	—
BCLR #xx:3, Rd	B	(#xx:3 of Rd8) ← 0		2								—	—	—
BCLR #xx:3, @Rd	B	(#xx:3 of @Rd16) ← 0			4							—	—	—
BCLR #xx:3, @aa:8	B	(#xx:3 of @aa:8) ← 0						4				—	—	—
BCLR Rn, Rd	B	(Rn8 of Rd8) ← 0		2								—	—	—
BCLR Rn, @Rd	B	(Rn8 of @Rd16) ← 0			4							—	—	—
BCLR Rn, @aa:8	B	(Rn8 of @aa:8) ← 0						4				—	—	—
BNOT #xx:3, Rd	B	(#xx:3 of Rd8) ← ($\overline{\text{#xx:3}}$ of $\overline{\text{Rd8}}$)		2								—	—	—
BNOT #xx:3, @Rd	B	(#xx:3 of @Rd16) ← ($\overline{\text{#xx:3}}$ of $\overline{\text{@Rd16}}$)			4							—	—	—
BNOT #xx:3, @aa:8	B	(#xx:3 of @aa:8) ← ($\overline{\text{#xx:3}}$ of $\overline{\text{@aa:8}}$)						4				—	—	—
BNOT Rn, Rd	B	(Rn8 of Rd8) ← ($\overline{\text{Rn8}}$ of $\overline{\text{Rd8}}$)		2								—	—	—
BNOT Rn, @Rd	B	(Rn8 of @Rd16) ← ($\overline{\text{Rn8}}$ of $\overline{\text{@Rd16}}$)			4							—	—	—
BNOT Rn, @aa:8	B	(Rn8 of @aa:8) ← ($\overline{\text{Rn8}}$ of $\overline{\text{@aa:8}}$)						4				—	—	—

BTST #xx:3, @aa:8	B	(#xx:3 of @aa:8) → Z						4				—	—	—
BTST Rn, Rd	B	(Rn8 of Rd8) → Z		2								—	—	—
BTST Rn, @Rd	B	(Rn8 of @Rd16) → Z			4							—	—	—
BTST Rn, @aa:8	B	(Rn8 of @aa:8) → Z						4				—	—	—
BLD #xx:3, Rd	B	(#xx:3 of Rd8) → C		2								—	—	—
BLD #xx:3, @Rd	B	(#xx:3 of @Rd16) → C			4							—	—	—
BLD #xx:3, @aa:8	B	(#xx:3 of @aa:8) → C						4				—	—	—
BILD #xx:3, Rd	B	(#xx:3 of Rd8) → C		2								—	—	—
BILD #xx:3, @Rd	B	(#xx:3 of @Rd16) → C			4							—	—	—
BILD #xx:3, @aa:8	B	(#xx:3 of @aa:8) → C						4				—	—	—
BST #xx:3, Rd	B	C → (#xx:3 of Rd8)		2								—	—	—
BST #xx:3, @Rd	B	C → (#xx:3 of @Rd16)			4							—	—	—
BST #xx:3, @aa:8	B	C → (#xx:3 of @aa:8)						4				—	—	—
BIST #xx:3, Rd	B	$\overline{C}$ → (#xx:3 of Rd8)		2								—	—	—
BIST #xx:3, @Rd	B	$\overline{C}$ → (#xx:3 of @Rd16)			4							—	—	—
BIST #xx:3, @aa:8	B	$\overline{C}$ → (#xx:3 of @aa:8)						4				—	—	—
BAND #xx:3, Rd	B	C∧(#xx:3 of Rd8) → C		2								—	—	—
BAND #xx:3, @Rd	B	C∧(#xx:3 of @Rd16) → C			4							—	—	—
BAND #xx:3, @aa:8	B	C∧(#xx:3 of @aa:8) → C						4				—	—	—
BIAND #xx:3, Rd	B	C∧(#xx:3 of Rd8) → C		2								—	—	—
BIAND #xx:3, @Rd	B	C∧(#xx:3 of @Rd16) → C			4							—	—	—
BIAND #xx:3, @aa:8	B	C∧(#xx:3 of @aa:8) → C						4				—	—	—
BOR #xx:3, Rd	B	C∨(#xx:3 of Rd8) → C		2								—	—	—
BOR #xx:3, @Rd	B	C∨(#xx:3 of @Rd16) → C			4							—	—	—
BOR #xx:3, @aa:8	B	C∨(#xx:3 of @aa:8) → C						4				—	—	—
BIOR #xx:3, Rd	B	C∨(#xx:3 of Rd8) → C		2								—	—	—
BIOR #xx:3, @Rd	B	C∨(#xx:3 of @Rd16) → C			4							—	—	—

BXOR #xx:3, @Rd	B	C⊕(#xx:3 of @Rd16) → C					4						—	—	—
BXOR #xx:3, @aa:8	B	C⊕(#xx:3 of @aa:8) → C								4			—	—	—
BIXOR #xx:3, Rd	B	C⊕($\overline{\#xx:3}$ of $\overline{Rd8}$) → C			2								—	—	—
BIXOR #xx:3, @Rd	B	C⊕($\overline{\#xx:3}$ of $\overline{@Rd16}$) → C					4						—	—	—
BIXOR #xx:3, @aa:8	B	C⊕($\overline{\#xx:3}$ of $\overline{@aa:8}$) → C								4			—	—	—
BRA d:8 (BT d:8)	—	PC ← PC+d:8									2		—	—	—
BRN d:8 (BF d:8)	—	PC ← PC+2									2		—	—	—
BHI d:8	—	If condition is true then PC ← PC+d:8 else next;	C ∨ Z = 0								2		—	—	—
BLS d:8	—		C ∨ Z = 1								2		—	—	—
BCC d:8 (BHS d:8)	—		C = 0								2		—	—	—
BCS d:8 (BLO d:8)	—		C = 1								2		—	—	—
BNE d:8	—		Z = 0								2		—	—	—
BEQ d:8	—		Z = 1								2		—	—	—
BVC d:8	—		V = 0								2		—	—	—
BVS d:8	—		V = 1								2		—	—	—
BPL d:8	—		N = 0								2		—	—	—
BMI d:8	—		N = 1								2		—	—	—
BGE d:8	—		N⊕V = 0								2		—	—	—
BLT d:8	—		N⊕V = 1								2		—	—	—
BGT d:8	—		Z ∨ (N⊕V) = 0								2		—	—	—
BLE d:8	—		Z ∨ (N⊕V) = 1								2		—	—	—
JMP @Rn	—	PC ← Rn16					2						—	—	—
JMP @aa:16	—	PC ← aa:16								4			—	—	—
JMP @@aa:8	—	PC ← @aa:8									2		—	—	—
BSR d:8	—	SP–2 → SP PC → @SP PC ← PC+d:8									2		—	—	—

JSR @aa:16	—	SP–2 → SP PC → @SP PC ← aa:16						4				—	—	—	-
JSR @@aa:8		SP–2 → SP PC → @SP PC ← @aa:8								2		—	—	—	-
RTS	—	PC ← @SP SP+2 → SP									2	—	—	—	-
RTE	—	CCR ← @SP SP+2 → SP PC ← @SP SP+2 → SP									2	↕	↕	↕	↕
SLEEP	—	Transit to sleep mode.									2	—	—	—	-
LDC #xx:8, CCR	B	#xx:8 → CCR	2									↕	↕	↕	↕
LDC Rs, CCR	B	Rs8 → CCR		2								↕	↕	↕	↕
STC CCR, Rd	B	CCR → Rd8		2								—	—	—	-
ANDC #xx:8, CCR	B	CCR∧#xx:8 → CCR	2									↕	↕	↕	↕
ORC #xx:8, CCR	B	CCR∨#xx:8 → CCR	2									↕	↕	↕	↕
XORC #xx:8, CCR	B	CCR⊕#xx:8 → CCR	2									↕	↕	↕	↕
NOP	—	PC ← PC+2									2	—	—	—	-
EEPMOV	—	if R4L≠0 Repeat @R5 → @R6 R5+1 → R5 R6+1 → R6 R4L–1 → R4L Until R4L=0 else next;									4	—	—	—	-

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 cleare

(3) Set to 1 if decimal adjustment produces a carry; otherwise retains value prior to arithme

(4) The number of states required for execution is 4n + 9 in the H8/3827R Group and 4n + H8/3827S Group, H8/38327 Group and H8/38427 Group (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.

High \ Low	0	1	2	3	4	5	6	7	8	9	A	B	C
0	NOP	SLEEP	STC	LDC	ORC	XORC	ANDC	LDC	ADD		INC	ADDS	MOV
1	SHLL / SHAL	SHLR / SHAR	ROTXL / ROTL	ROTXR / ROTR	OR	XOR	AND	NOT / NEG	SUB		DEC	SUBS	CMP
2	MOV												
3													
4	BRA	BRN	BHI	BLS	BCC	BCS	BNE	BEQ	BVC	BVS	BPL	BMI	BGE
5	MULXU	DIVXU			RTS	BSR	RTE				JMP		
6	BSET	BNOT	BCLR	BTST	BOR / BIOR	BXOR / BIXOR	BAND / BIAND	BST / BILD	MOV*				
7								BIST / BILD		MOV		EEPMOV	Bit-m
8	ADD												
9	ADDX												
A	CMP												
B	SUBX												
C	OR												
D	XOR												
E	AND												
F	MOV												

Examples: When instruction is fetched from on-chip ROM, and an on-chip RAM is a

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 R
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 M
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 section 2.9.1, Notes o
Access for details.

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		
	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

	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
	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

DAA	DAA.B Rd	1					
DAS	DAS.B Rd	1					
DEC	DEC.B Rd	1					
DIVXU	DIVXU.B Rs, Rd	1					
EEPMOV	EEPMOV	2				2n+2*1	
INC	INC.B Rd	1					
JMP	JMP @Rn	2					
	JMP @aa:16	2					
	JMP @@aa:8	2	1				
JSR	JSR @Rn	2		1			
	JSR @aa:16	2		1			
	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					
	MOV.B @Rs, Rd	1			1		
	MOV.B @(d:16, Rs), Rd	2			1		
	MOV.B @Rs+, Rd	1			1		
	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		
	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
	MOV.W @aa:16, Rd	2					1

NOT	NOT.B Rd	1		
OR	OR.B #xx:8, Rd	1		
	OR.B Rs, Rd	1		
ORC	ORC #xx:8, CCR	1		
ROTL	ROTL.B Rd	1		
ROTR	ROTR.B Rd	1		
ROTXL	ROTXL.B Rd	1		
ROTXR	ROTXR.B Rd	1		
RTE	RTE	2	2	
RTS	RTS	2	1	
SHAL	SHAL.B Rd	1		
SHAR	SHAR.B Rd	1		
SHLL	SHLL.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		
POP	POP Rd	1	1	
PUSH	PUSH Rs	1	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		

Notes: 1. n: Initial value in R4L. The source and destination operands are accessed n + 1 times each.
2. 1 in the H8/3827R Group and 0 in the H8/3827S Group, H8/38327 Group, and H8/38427 Group.

H'21	FLMCR2	FLER	—	—	—	—	—	—	—
H'22	FLPWCR	PDWND	—	—	—	—	—	—	—
H'23	EBR	EB7	EB6	EB5	EB4	EB3	EB2	EB1	EB0
H'24									
H'25									
H'26									
H'27									
H'28									
H'29									
H'2A									
H'2B	FENR	FLSHE	—	—	—	—	—	—	—
H'2C									
H'2D									
H'2E									
H'2F									

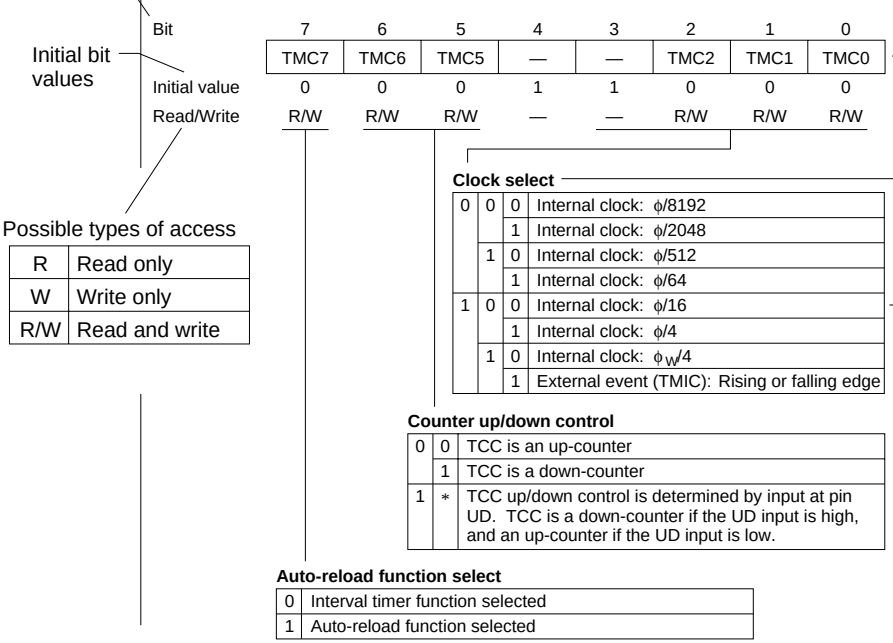
H'95	ECCSR	OVH	OVL	—	CH2	CUEH	CUEL	CRCH	CRCL
H'96	ECH	ECH7	ECH6	ECH5	ECH4	ECH3	ECH2	ECH1	ECH0
H'97	ECL	ECL7	ECL6	ECL5	ECL4	ECL3	ECL2	ECL1	ECL0
H'98	SMR31	COM31	CHR31	PE31	PM31	STOP31	MP31	CKS311	CKS31
H'99	BRR31	BRR317	BRR316	BRR315	BRR314	BRR313	BRR312	BRR311	BRR31
H'9A	SCR31	TIE31	RIE31	TE31	RE31	MPIE31	TEIE31	CKE31	CKE31
H'9B	TDR31	TDR317	TDR316	TDR315	TDR314	TDR313	TDR312	TDR311	TDR31
H'9C	SSR31	TDRE31	RDRF31	OER31	FER31	PER31	TEND31	MPBR31	MPBT3
H'9D	RDR31	RDR317	RDR316	RDR315	RDR314	RDR313	RDR312	RDR311	RDR31
H'9E									
H'9F									
H'A0									
H'A1									
H'A2									
H'A3									
H'A4									
H'A5									
H'A6									
H'A7									
H'A8	SMR32	COM32	CHR32	PE32	PM32	STOP32	MP32	CKS321	CKS32
H'A9	BRR32	BRR327	BRR326	BRR325	BRR324	BR323	BRR322	BRR321	BRR32
H'AA	SCR32	TIE32	RIE32	TE32	RE32	MPIE32	TEIE32	CKE321	CKE32
H'AB	TDR32	TDR327	TDR326	TDR325	TDR324	TDR323	TDR322	TDR321	TDR32
H'AC	SSR32	TDRE32	RDRF32	OER32	FER32	PER32	TEND32	MPBR32	MPBT3
H'AD	RDR32	RDR327	RDR326	RDR325	RDR324	RDR323	RDR322	RDR321	RDR32
H'AE									
H'AF									
H'B0	TMA	TMA7	TMA6	TMA5	—	TMA3	TMA2	TMA1	TMA0

H'B6	TCRF	TOLH	CKSH2	CKSH1	CKSH0	TOLL	CKSL2	CKSL1	CKSL0
H'B7	TCSRF	OVFH	CMFH	OVIEH	CCLRH	OVFL	CMFL	OVIEL	CCLRL
H'B8	TCFH	TCFH7	TCFH6	TCFH5	TCFH4	TCFH3	TCFH2	TCFH1	TCFH0
H'B9	TCFL	TCFL7	TCFL6	TCFL5	TCFL4	TCFL3	TCFL2	TCFL1	TCFL0
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
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
H'C1	LCR	—	PSW	ACT	DISP	CKS3	CKS2	CKS1	CKS0
H'C2	LCR2	LCDAB	—	—	—	CDS3	CDS2	CDS1	CDS0
H'C3									
H'C4	ADRRH	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2
H'C5	ADRRL	ADR1	ADR0	—	—	—	—	—	—
H'C6	AMR	CKS	TRGE	—	—	CH3	CH2	CH1	CH0
H'C7	ADSR	ADSF	—	—	—	—	—	—	—
H'C8	PMR1	IRQ3	IRQ2	IRQ1	IRQ4	TMIG	TMOFH	TMOFL	TMOW
H'C9	PMR2	EXCL	—	—	—	—	—	—	—
H'CA	PMR3	AEVL	AEVH	WDCKS	NCS	IRQ0	RESO	UD	PWM
H'CB									
H'CC	PMR5	WKP7	WKP6	WKP5	WKP4	WKP3	WKP2	WKP1	WKP0
H'CD									
H'CE									
H'CF									
H'D0	PWCR	—	—	—	—	—	—	PWCR1	PWCR0
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	P17	P16	P15	P14	P13	P12	P11	P10

H'DB	PDR8	P87	P86	P85	P84	P83	P82	P81	P80
H'DC									
H'DD	PDRA	—	—	—	—	PA3	PA2	PA1	PA0
H'DE	PDRB	PB7	PB6	PB5	PB4	PB3	PB2	PB1	PB0
H'DF									
H'E0	PUCR1	PUCR17	PUCR16	PUCR15	PUCR14	PUCR13	PUCR12	PUCR11	PUCR
H'E1	PUCR3	PUCR37	PUCR36	PUCR35	PUCR34	PUCR33	PUCR32	PUCR31	PUCR
H'E2	PUCR5	PUCR57	PUCR56	PUCR55	PUCR54	PUCR53	PUCR52	PUCR51	PUCR
H'E3	PUCR6	PUCR67	PUCR66	PUCR65	PUCR64	PUCR63	PUCR62	PUCR61	PUCR
H'E4	PCR1	PCR17	PCR16	PCR15	PCR14	PCR13	PCR12	PCR11	PCR10
H'E5									
H'E6	PCR3	PCR37	PCR36	PCR35	PCR34	PCR33	PCR32	PCR31	PCR30
H'E7	PCR4	—	—	—	—	—	PCR42	PCR41	PCR40
H'E8	PCR5	PCR57	PCR56	PCR55	PCR54	PCR53	PCR52	PCR51	PCR50
H'E9	PCR6	PCR67	PCR66	PCR65	PCR64	PCR63	PCR62	PCR61	PCR60
H'EA	PCR7	PCR77	PCR76	PCR75	PCR74	PCR73	PCR72	PCR71	PCR70
H'EB	PCR8	PCR87	PCR86	PCR85	PCR84	PCR83	PCR82	PCR81	PCR80
H'EC									
H'ED	PCRA	—	—	—	—	PCRA3	PCRA2	PCRA1	PCRA0
H'EE									
H'EF									
H'F0	SYSCR1	SSBY	STS2	STS1	STS0	LSON	—	MA1	MA0
H'F1	SYSCR2	—	—	—	NESEL	DTON	MSON	SA1	SA0
H'F2	IEGR	—	—	—	IEG4	IEG3	IEG2	IEG1	IEG0
H'F3	IENR1	IENTA	—	IENWP	IEN4	IEN3	IEN2	IEN1	IEN0
H'F4	IENR2	IENDT	IENAD	—	IENTG	IENTFH	IENTFL	IENTC	IENEC
H'F5									
H'F6	IRR1	IRRTA	—	—	IRRI4	IRRI3	IRRI2	IRRI1	IRRI0
H'F7	IRRI2	IRRDT	IRRAD	—	IRRTG	IRRTFH	IRRTFL	IRRTC	IRREC
H'F8									

H'FE	
H'FF	

Legend:
SCI: Serial Communication Interface


Bit
7 6 5 4 3 2 1 0
Initial bit values
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
Possible types of access
R Read only
W Write only
R/W Read and write
Clock select
0 0 0 Internal clock: ϕ/8192
0 1 Internal clock: ϕ/2048
1 0 Internal clock: ϕ/512
1 Internal clock: ϕ/64
1 0 0 Internal clock: ϕ/16
1 Internal clock: ϕ/4
1 0 Internal clock: ϕW/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 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.
Auto-reload function select
0 Interval timer function selected
1 Auto-reload function selected
*: Don't care

Program

0	Program mode cleared (in
1	Transition to program mod [Setting condition] When SWE = 1 and PSU =

Erase

0	Erase mode cleared (initial value
1	Transition to erase mode [Setting condition] When SWE = 1 and ESU = 1

Program-Verify

0	Program-verify mode cleared (initial va
1	Transition to program-verify mode [Setting condition] When SWE = 1

Erase-Verify

0	Erase-verify mode cleared (initial value)
1	Transition to erase-verify mode [Setting condition] When SWE = 1

Program-Setup

0	Program-setup cleared (initial value)
1	Program setup [Setting condition] When SWE = 1

Erase-Setup

0	Erase-setup cleared (initial value)
1	Erase setup [Setting condition] When SWE = 1

Software write enable bit

0	Writing/erasing disabled (initial value)
1	Writing/erasing enabled

Flash memory error

Note: A write to FLMCR2 is prohibited.

FLPWCR—Flash Memory Power Control Register H'F022 Fl

Bit	7	6	5	4	3	2	1
	PDWND	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	—	—	—	—	—	—

Power-down Disable

0	When the system transits to sub-active mode, the flash memory changes to low-power mode
1	When the system transits to sub-active mode, the flash memory changes to normal mode

Blocks 7 to 0

0	When a block of EB7 to EB0 is not selected (init
1	When a block of EB7 to EB0 is selected

Note: Set the bit of EBR to H'00 when erasing.

FENR Flash Memory Enable Register **H'F02B** **Fla**

Bit	7	6	5	4	3	2	1	
	FLSHE	—	—	—	—	—	—	
Initial value	0	0	0	0	0	0	0	
Read/Write	R/W	—	—	—	—	—	—	

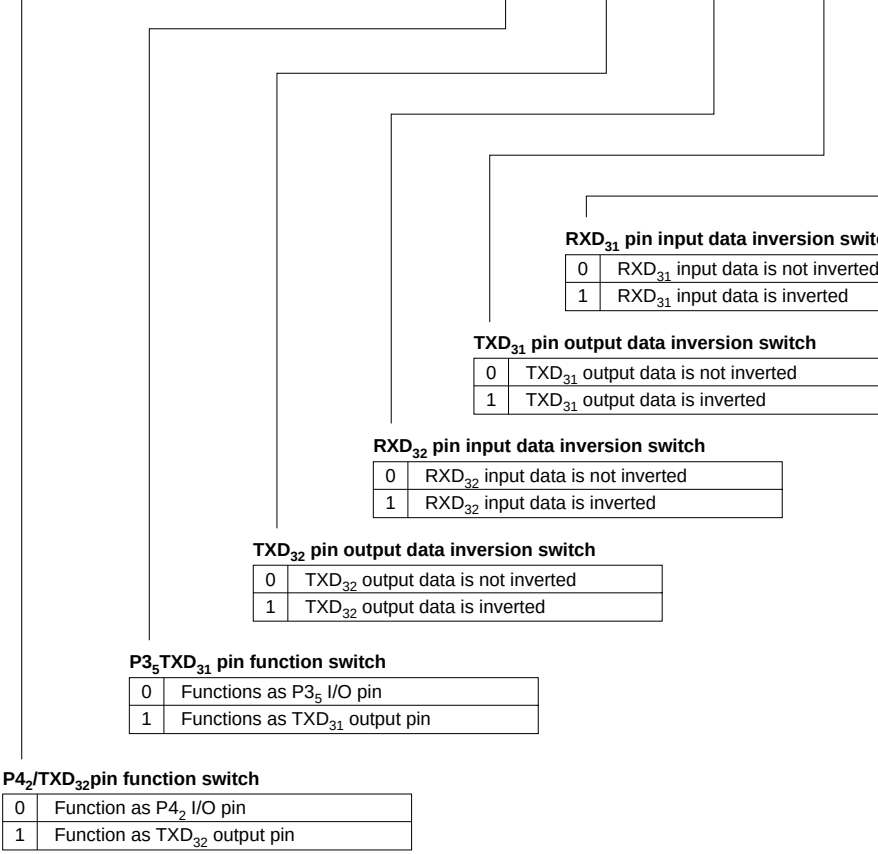
Flash Memory Control Register Enable

0	The flash memory control register cannot be accessed
1	The flash memory control register can be accessed

$\overline{\text{WKP}}$n edge selected

0	$\overline{\text{WKP}}$n pin falling edge detected
1	$\overline{\text{WKP}}$n pin rising edge detected

(n = 0 to 7)



RXD_{31} pin input data inversion swit

0	RXD_{31} input data is not inverted
1	RXD_{31} input data is inverted

TXD_{31} pin output data inversion switch

0	TXD_{31} output data is not inverted
1	TXD_{31} output data is inverted

RXD_{32} pin input data inversion switch

0	RXD_{32} input data is not inverted
1	RXD_{32} input data is inverted

TXD_{32} pin output data inversion switch

0	TXD_{32} output data is not inverted
1	TXD_{32} output data is inverted

$P3_5/TXD_{31}$ pin function switch

0	Functions as $P3_5$ I/O pin
1	Functions as TXD_{31} output pin

$P4_2/TXD_{32}$pin function switch

0	Function as $P4_2$ I/O pin
1	Function as TXD_{32} output pin

TMOW pin clock select

0	Clock output from TMA is outp
1	ϕ_W is output

Counter reset co

0	ECL is reset
1	ECL reset is c and count-up is enabled

Counter reset control H

0	ECH is reset
1	ECH reset is cleared a count-up function is e

Count-up enable L

0	ECL event clock input is d ECL value is held
1	ECL event clock input is e

Count-up enable H

0	ECH event clock input is disa ECH value is held
1	ECH event clock input is enal

Channel select

0	ECH and ECL are used together as a channel 16-bit event counter
1	ECH and ECL are used as two indepe 8-bit event counter channels

Counter overflow L

0	ECL has not overflowed
1	ECL has overflowed

Counter overflow H

0	ECH has not overflowed
1	ECH has overflowed

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

Count value

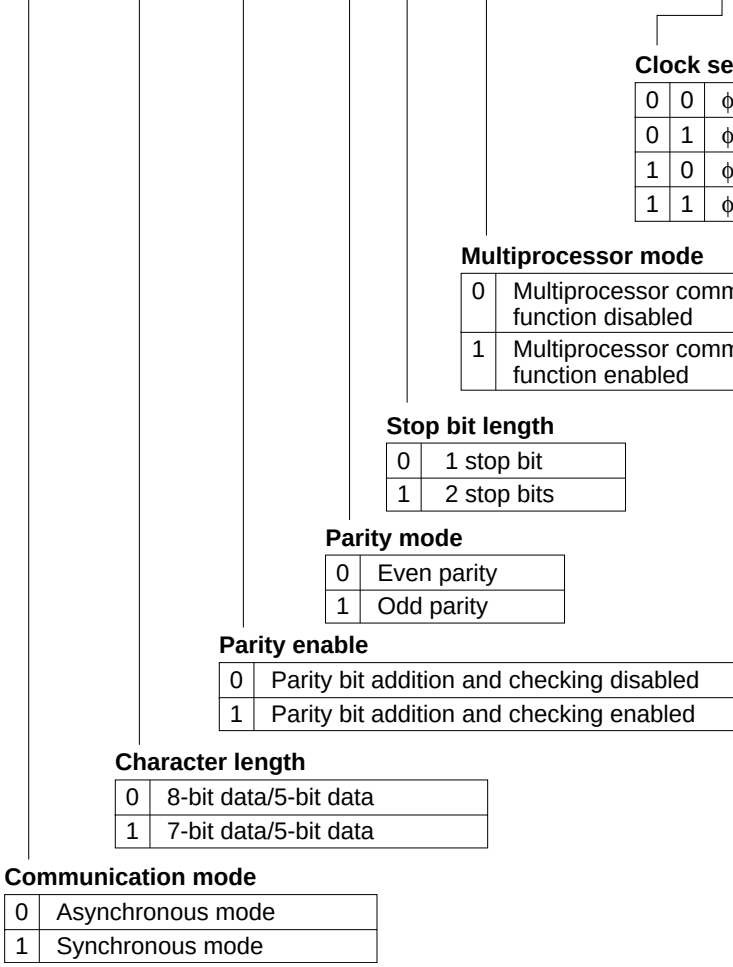
Note: ECH and ECL can also be used as the upper and lower halves, respectively, event counter (EC).

ECL—Event Counter L **H'97**

Bit	7	6	5	4	3	2	1
	ECL7	ECL6	ECL5	ECL4	ECL3	ECL2	ECL1
Initial value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

Count value

Note: ECH and ECL can also be used as the upper and lower halves, respectively, event counter (EC).


Clock se
0 0 ϕ
0 1 ϕ
1 0 ϕ
1 1 ϕ
Multiprocessor mode
0 Multiprocessor comm function disabled
1 Multiprocessor comm 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 addition and checking disabled
1 Parity bit addition and checking enabled
Character length
0 8-bit data/5-bit data
1 7-bit data/5-bit data
Communication mode
0 Asynchronous mode
1 Synchronous mode

Serial transmit/receive bit rate

Clock enable

Bit 1	Bit 0	Description		
CKE311	CKE310	Communication Mode	Clock Source	SCK_3 Pin Function
0	0	Asynchronous	Internal clock	I/O port
		Synchronous	Internal clock	Serial clock output
0	1	Asynchronous	Internal clock	Clock output
		Synchronous	Reserved (Do not specify this combinatio	
1	0	Asynchronous	External clock	Clock input
		Synchronous	External clock	Serial clock input
1	1	Asynchronous	Reserved (Do not specify this combinatio	
		Synchronous	Reserved (Do not specify this combinatio	

Transmit end interrupt enable

0	Transmit end interrupt request (TEI) disabled
1	Transmit end interrupt request (TEI) enabled

Multiprocessor interrupt enable

0	Multiprocessor interrupt request disabled (normal receive operation) [Clearing condition] When data is received in which the multiprocessor bit is set to 1
1	Multiprocessor interrupt request enabled The receive interrupt request (RXI), receive error interrupt request (ERI), and setting of th RDRF, FER, and OER flags in the serial status register (SSR), are disabled until data wit the multiprocessor bit set to 1 is received.

Receive enable

0	Receive operation disabled (RXD pin is I/O port)
1	Receive operation enabled (RXD pin is receive data pin)

Transmit enable

0	Transmit operation disabled (TXD pin is transmit data pin)
1	Transmit operation enabled (TXD pin is 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

Multiprocessor bit transfer

0	A 0 multiprocessor bit is transmitted
1	A 1 multiprocessor bit is transmitted

Multiprocessor bit receive

0	Data in which the multiprocessor bit is 0 has been received
1	Data in which the multiprocessor bit is 1 has been received

Transmit end

0	Transmission in progress [Clearing conditions] • After reading TDRE31 = 1, cleared by writing 0 to TDRE • When data is written to TDR31 by an instruction
1	Transmission ended [Setting conditions] • When bit TE in serial control register 31 (SCR31) is cleared to 0 • When bit TDRE31 is set to 1 when the last bit of a transmit character is sent

Parity error

0	Reception in progress or completed normally [Clearing condition] After reading PER31 = 1, cleared by writing 0 to PER31
1	A parity error has occurred during reception [Setting condition] When the number of 1 bits in the receive data plus parity bit does not match the parity designated by the parity mode bit (PM31) in the serial mode register (SMR31)

Framing error

0	Reception in progress or completed normally [Clearing condition] After reading FER31 = 1, cleared by writing 0 to FER31
1	A framing error has occurred during reception [Setting condition] When the stop bit at the end of the receive data is checked for a value of 1 at completion reception, and the stop bit is 0

Overrun error

0	Reception in progress or completed [Clearing condition] After reading OER31 = 1, cleared by writing 0 to OER31
1	An overrun error has occurred during reception [Setting condition] When the next serial reception is completed with RDRF31 set to 1

Receive data register full

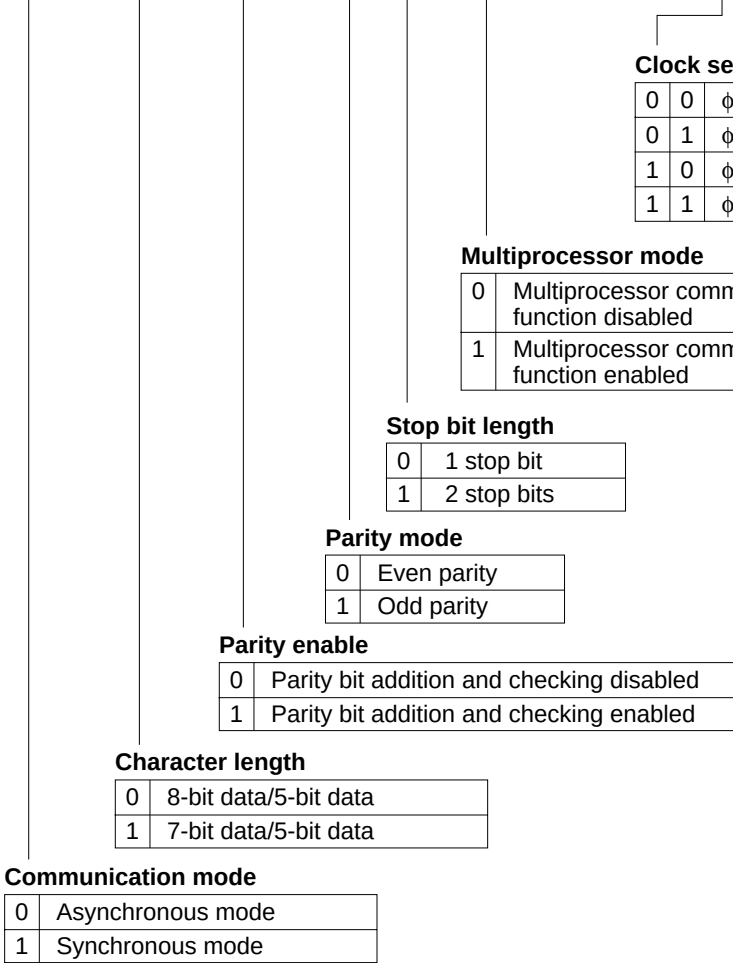
0	There is no receive data in RDR31 [Clearing conditions] • After reading RDRF31 = 1, cleared by writing 0 to RDRF31 • When RDR31 data is read by an instruction
1	There is receive data in RDR31 [Setting condition] When reception ends normally and receive data is transferred from RSR31 to RDR31

Transmit data register empty

0	Transmit data written in TDR31 has not been transferred to TSR31 [Clearing conditions] • After reading TDRE31 = 1, cleared by writing 0 to TDRE31 • When data is written to TDR31 by an instruction
1	Transmit data has not been written to TDR31, or transmit data written in TDR31 has been transferred to TSR31 [Setting conditions] • When bit TE in serial control register 31 (SCR31) is cleared to 0 • When data is transferred from TDR31 to TSR31

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

Serial receiving data are stored


Clock se
0 0 ϕ
0 1 ϕ
1 0 ϕ
1 1 ϕ
Multiprocessor mode
0 Multiprocessor comm function disabled
1 Multiprocessor comm 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 addition and checking disabled
1 Parity bit addition and checking enabled
Character length
0 8-bit data/5-bit data
1 7-bit data/5-bit data
Communication mode
0 Asynchronous mode
1 Synchronous mode

Serial transmit/receive bit rate

Clock enable

Bit 1	Bit 0	Description		
CKE321	CKE320	Communication Mode	Clock Source	SCK_3 Pin Function
0	0	Asynchronous	Internal clock	I/O port
		Synchronous	Internal clock	Serial clock output
0	1	Asynchronous	Internal clock	Clock output
		Synchronous	Reserved (Do not specify this combinatio	
1	0	Asynchronous	External clock	Clock input
		Synchronous	External clock	Serial clock input
1	1	Asynchronous	Reserved (Do not specify this combinatio	
		Synchronous	Reserved (Do not specify this combinatio	

Transmit end interrupt enable

0	Transmit end interrupt request (TEI) disabled
1	Transmit end interrupt request (TEI) enabled

Multiprocessor interrupt enable

0	Multiprocessor interrupt request disabled (normal receive operation) [Clearing condition] When data is received in which the multiprocessor bit is set to 1
1	Multiprocessor interrupt request enabled The receive interrupt request (RXI), receive error interrupt request (ERI), and setting of th RDRF, FER, and OER flags in the serial status register (SSR), are disabled until data wit the multiprocessor bit set to 1 is received.

Receive enable

0	Receive operation disabled (RXD pin is I/O port)
1	Receive operation enabled (RXD pin is receive data pin)

Transmit enable

0	Transmit operation disabled (TXD pin is transmit data pin)
1	Transmit operation enabled (TXD pin is 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

Multiprocessor bit transfer

0	A 0 multiprocessor bit is transmitted
1	A 1 multiprocessor bit is transmitted

Multiprocessor bit receive

0	Data in which the multiprocessor bit is 0 has been received
1	Data in which the multiprocessor bit is 1 has been received

Transmit end

0	Transmission in progress [Clearing conditions] • After reading TDRE32 = 1, cleared by writing 0 to TDRE32 • When data is written to TDR32 by an instruction
1	Transmission ended [Setting conditions] • When bit TE in serial control register 32 (SCR32) is cleared to 0 • When bit TDRE32 is set to 1 when the last bit of a transmit character is sent

Parity error

0	Reception in progress or completed normally [Clearing condition] After reading PER32 = 1, cleared by writing 0 to PER32
1	A parity error has occurred during reception [Setting condition] When the number of 1 bits in the receive data plus parity bit does not match the parity designated by the parity mode bit (PM32) in the serial mode register (SMR32)

Framing error

0	Reception in progress or completed normally [Clearing condition] After reading FER32 = 1, cleared by writing 0 to FER32
1	A framing error has occurred during reception [Setting condition] When the stop bit at the end of the receive data is checked for a value of 1 at completion reception, and the stop bit is 0

Overrun error

0	Reception in progress or completed [Clearing condition] After reading OER32 = 1, cleared by writing 0 to OER32
1	An overrun error has occurred during reception [Setting condition] When the next serial reception is completed with RDRF32 set to 1

Receive data register full

0	There is no receive data in RDR32 [Clearing conditions] • After reading RDRF32 = 1, cleared by writing 0 to RDRF32 • When RDR32 data is read by an instruction
1	There is receive data in RDR32 [Setting condition] When reception ends normally and receive data is transferred from RSR32 to RDR32

Transmit data register empty

0	Transmit data written in TDR32 has not been transferred to TSR32 [Clearing conditions] • After reading TDRE32 = 1, cleared by writing 0 to TDRE32 • When data is written to TDR32 by an instruction
1	Transmit data has not been written to TDR32, or transmit data written in TDR32 has been transferred to TSR32 [Setting conditions] • When bit TE32 in serial control register 32 (SCR32) is cleared to 0 • When data is transferred from TDR32 to TSR32

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

Serial receiving data are stored

Internal clock select

TMA3	TMA2	TMA1	TMA0	Prescaler and Divider Ratio or Overflow Period		F
0	0	0	0	PSS	ϕ/8192	I
			1	PSS	ϕ/4096	t
		1	0	PSS	ϕ/2048	
			1	PSS	ϕ/512	
	1	0	0	PSS	ϕ/256	
			1	PSS	ϕ/128	
		1	0	PSS	ϕ/32	
			1	PSS	ϕ/8	
1	0	0	0	PSW	1 s	T
			1	PSW	0.5 s	b
		1	0	PSW	0.25 s	(v
			1	PSW	0.03125 s	u
	1	0	0	PSW and TCA are reset		3
			1			
		1	0			
			1			

Clock output select*

0	0	0	ϕ/32
		1	ϕ/16
	1	0	ϕ/8
		1	ϕ/4
1	0	0	ϕ_W/32
		1	ϕ_W/16
	1	0	ϕ_W/8
		1	ϕ_W/4

Note: * Values when the CWOS bit in CWOSR is cleared to 0. When the CWOS bit is set to 1, ϕ_W is output regardless of the value of bits TMA7 to TMA5.

Count value

Watchdog timer reset

0	[Clearing conditions] • Reset by $\overline{\text{RES}}$ pin • When TCSRWE = 1, and 0 is written in both B0WI and W
1	[Setting condition] When TCW overflows and a reset signal is generated

Bit 0 write inhibit

0	Bit 0 is write-enabled
1	Bit 0 is write-protected

Watchdog timer on

0	Watchdog timer operation is disabled
1	Watchdog timer operation is enabled

Bit 2 write inhibit

0	Bit 2 is write-enabled
1	Bit 2 is write-protected

Timer control/status register W write enable

0	Data cannot be written to bits 2 and 0
1	Data can be written to bits 2 and 0

Bit 4 write inhibit

0	Bit 4 is write-enabled
1	Bit 4 is write-protected

Timer counter W write enable

0	Data cannot be written to TCW
1	Data can be written to TCW

Bit 6 write inhibit

0	Bit 6 is write-enabled
1	Bit 6 is write-protected

Note: * Write is permitted only under certain conditions.

Count value

TMC—Timer Mode Register C H'B4

Bit	7	6	5	4	3	2	1
	TMC7	TMC6	TMC5	—	—	TMC2	TMC1
Initial value	0	0	0	1	1	0	0
Read/Write	R/W	R/W	R/W	—	—	R/W	R/W

Clock select

0	0	0	Internal clock: ϕ/8192
0	0	1	Internal clock: ϕ/2048
0	1	0	Internal clock: ϕ/512
0	1	1	Internal clock: ϕ/64
1	0	0	Internal clock: ϕ/16
1	0	1	Internal clock: ϕ/4
1	1	0	Internal clock: ϕ_W/4
1	1	1	External event (TMIC): C on rising or falling edge

Counter up/down control

0	0	TCC is an up-counter
0	1	TCC is a down-counter
1	*	Hardware control of TCC up/down operation by UD pin UD pin input high: Down-counter UD pin input low: Up-counter

Auto-reload function select

0	Interval timer function selected
1	Auto-reload function selected

*

Count value

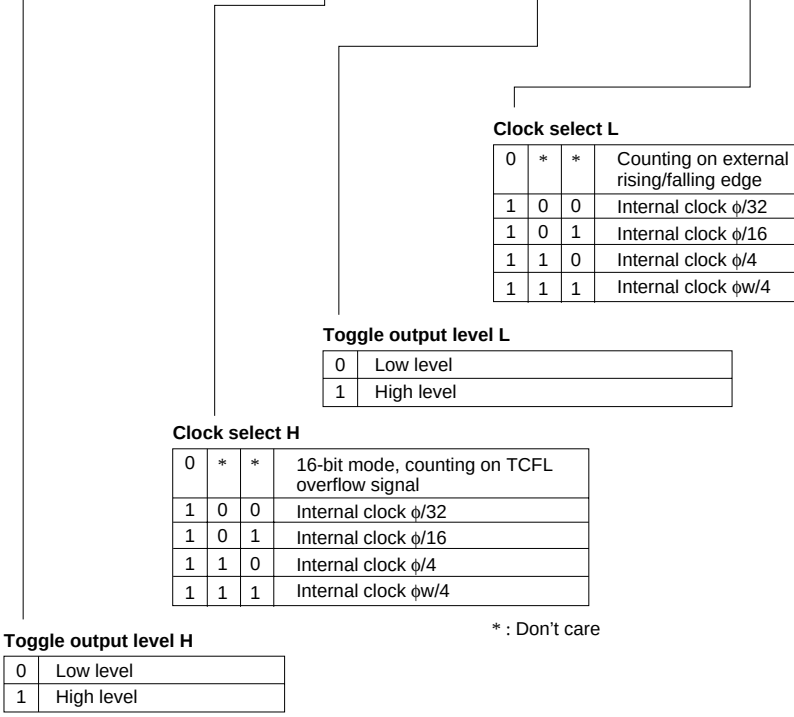
Note: TCC is assigned to the same address as TLC. In a read, the TCC value is read

TLC—Timer Load Register C **H'B5**

Bit	7	6	5	4	3	2	1
	TLC7	TLC6	TLC5	TLC4	TLC3	TLC2	TLC1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Reload value

Note: TLC is assigned to the same address as TCC. In a write, the TLC value is written



Clock select L

0	*	*	Counting on external rising/falling edge
1	0	0	Internal clock ϕ/32
1	0	1	Internal clock ϕ/16
1	1	0	Internal clock ϕ/4
1	1	1	Internal clock ϕw/4

Toggle output level L

0	Low level
1	High level

Clock select H

0	*	*	16-bit mode, counting on TCFL overflow signal
1	0	0	Internal clock ϕ/32
1	0	1	Internal clock ϕ/16
1	1	0	Internal clock ϕ/4
1	1	1	Internal clock ϕw/4

* : Don't care

Toggle output level H

0	Low level
1	High level

Counter clear L

0	TCFL clearing by compare match is disabled
1	TCFL clearing by compare match is enabled

Timer overflow interrupt enable L

0	TCFL overflow interrupt request is disabled
1	TCFL overflow interrupt request is enabled

Compare match flag L

0	[Clearing condition] After reading CMFL = 1, cleared by writing 0 to CMFL
1	[Setting condition] Set when the TCFL value matches the OCRFL value

Timer overflow flag L

0	[Clearing condition] After reading OVFL = 1, cleared by writing 0 to OVFL
1	[Setting condition] Set when TCFL overflows from H'FF to H'00

Counter clear H

0	16-bit mode: TCF clearing by compare match is disabled 8-bit mode: TCFH clearing by compare match is disabled
1	16-bit mode: TCF clearing by compare match is enabled 8-bit mode: TCFH clearing by compare match is enabled

Timer overflow interrupt enable H

0	TCFH overflow interrupt request is disabled
1	TCFH overflow interrupt request is enabled

Compare match flag H

0	[Clearing condition] After reading CMFH = 1, cleared by writing 0 to CMFH
1	[Setting condition] Set when the TCFH value matches the OCRFH value

Timer overflow flag H

0	[Clearing condition] After reading OVFH = 1, cleared by writing 0 to OVFH
1	[Setting condition] Set when TCFH overflows from H'FF to H'00

Note: * Bits 7, 6, 3, and 2 can only be written with 0, for flag clearing.

Count value

Note: TCFH and TCFL can also be used as the upper and lower halves, respectivel of a 16-bit event counter (TCF).

TCFL—8-Bit Timer Counter FL **H'B9**

Bit	7	6	5	4	3	2	1
	TCFL7	TCFL6	TCFL5	TCFL4	TCFL3	TCFL2	TCFL1
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Count value

Note: TCFH and TCFL can also be used as the upper and lower halves, respective of a 16-bit event counter (TCF).

output compare register (OCRF).

OCRFL—Output Compare Register FL **H'BB**

Bit	7	6	5	4	3	2	1
	OCRFL7	OCRFL6	OCRFL5	OCRFL4	OCRFL3	OCRFL2	OCRFL1
Initial value	1	1	1	1	1	1	1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: ECH and ECL can also be used as the upper and lower halves, respectively, o
output compare register (OCRF).

0	0	Internal clock: cour
0	1	Internal clock: cour
1	0	Internal clock: cour
1	1	Internal clock: cour

Counter clear

0	0	TCG clearing is disabled
0	1	TCG cleared by falling edge of input capture input s
1	0	TCG cleared by rising edge of input capture input s
1	1	TCG cleared by both edges of input capture input s

Input capture interrupt edge select

0	Interrupt generated on rising edge of input capture input signal
1	Interrupt generated on falling edge of input capture input signal

Timer overflow interrupt enable

0	TCG overflow interrupt request is disabled
1	TCG overflow interrupt request is enabled

Timer overflow flag L

0	[Clearing condition] After reading OVFL = 1, cleared by writing 0 to OVFL
1	[Setting condition] Set when TCG overflows from H'FF to H'00

Timer overflow flag H

0	[Clearing condition] After reading OVFH = 1, cleared by writing 0 to OVFH
1	[Setting condition] Set when TCG overflows from H'FF to H'00

Note: * Bits 7 and 6 can only be written with 0, for flag clearing.

Stores TCG value at falling edge of input capture signal

ICRGR—Input Capture Register GR **H'BE**

Bit	7	6	5	4	3	2	1
	ICRGR7	ICRGR6	ICRGR5	ICRGR4	ICRGR3	ICRGR2	ICRGR1
Initial value	0	0	0	0	0	0	0
Read/Write	R	R	R	R	R	R	R

Stores TCG value at rising edge of input capture signal

Clock enable

Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Function of Pins SEG_{32} to SEG_1							
SGX	SGS3	SGS2	SGS1	SGS0	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	SE to S
0	0	0	0	0	Port	Port	Port	Port	Port	Port	Port	P
	0	0	0	1	Port	Port	Port	Port	Port	Port	Port	P
	0	0	1	*	SEG	SEG	Port	Port	Port	Port	Port	P
	0	1	0	*	SEG	SEG	SEG	SEG	Port	Port	Port	P
	0	1	1	*	SEG	SEG	SEG	SEG	SEG	SEG	Port	P
	1	*	*	*	SEG	SEG	SEG	SEG	SEG	SEG	SEG	SI
1	0	0	0	0	Port*	Port	Port	Port	Port	Port	Port	P
	*	*	*	*	Use prohibited							

Note: * SEG32 to SEG29 are external expansion pins.

Bit 4	Description
SGX	
0	Pins $SEG_{32}$to SEG_{29}* (Initial value)
1	Pins CL_1, CL_2, DO, M

Note: * These pins function as ports when the setting of SGS3 to SGS0 is 0000 or 0001.
In the case of the H8/38327 Group and H8/38427 Group the initial values of these bits must not be changed.

Duty select, common function select

Bit 7	Bit 6	Bit 5	Duty Cycle	Common Drivers	Notes
DTS1	DTS0	CMX			
0	0	0	Static	COM_1	
		1		COM_4 to COM_1	COM_4 to COM_2 output the same waveform as COM_1
0	1	0	1/2 duty	COM_2 to COM_1	
		1		COM_4 to COM_1	COM_4 outputs the same waveform as COM_3 and COM_2 outputs the same waveform as COM
1	0	0	1/3 duty	COM_3 to COM_1	
		1		COM_4 to COM_1	COM_4 outputs a non-selected waveform
1	1	0	1/4 duty	COM_4 to COM_1	—
		1			

Frame frequency select

Bit 3	Bit 2	Bit 1	Bit 0	Oper
CKS3	CKS2	CKS1	CKS0	
0	*	0	0	
0	*	0	1	
0	*	1	*	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

*

Display data control

0	Blank data is displayed
1	LCD RAM data is displayed

Display function activate

0	LCD controller/driver operation halted
1	LCD controller/driver operates

LCD drive power supply on/off control

0	LCD drive power supply off
1	LCD drive power supply on

Charge/discharge pulse duty cy

Bit 3	Bit 2	Bit 1	Bit 0	Du
CDS3	CDS2	CDS1	CDS0	
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	*	*	
1	1	*	*	

A waveform/B waveform switching control

0	Drive using A waveform
1	Drive using B waveform

Read/Write	R	R	R	R	R	R	R

A/D conversion result

ADRRL

Bit	7	6	5	4	3	2	1
	ADR1	ADR0	—	—	—	—	—
Initial value	Not fixed	Not fixed	—	—	—	—	—
Read/Write	R	R	—	—	—	—	—

A/D conversion result

Channel select

Bit 3	Bit 2	Bit 1	Bit 0	
CH3	CH2	CH1	CH0	Analog Input
0	0	*	*	No channel s
	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	1	*	*	Setting prohi

* : I

External trigger select

0	Disables start of A/D conversion by external trigger
1	Enables start of A/D conversion by rising or falling ed of external trigger at pin $\overline{\text{ADTRG}}$

Clock select

Bit 7		Conversion Time	
CKS	Conversion Period	ϕ = 1 MHz	ϕ = 5 MHz
0	62/ϕ	62 µs	12.4 µs
1	31/ϕ	31 µs	—

0	Read	Indicates completion of A/D conversion
	Write	Stops A/D conversion
1	Read	Indicates A/D conversion in progress
	Write	Starts A/D conversion

$P1_7/\overline{IRQ_3}/TMIF$ pin function switch

0	Functions as $P1_7$ I/O pin
1	Functions as $\overline{IRQ_3}$/TMIF input pin

$P1_6/\overline{IRQ_2}$ pin function switch

0	Functions as $P1_6$ I/O pin
1	Functions as $\overline{IRQ_2}$ input pin

$P1_5/\overline{IRQ_1}/TMIC$ pin function switch

0	Functions as $P1_5$ I/O pin
1	Functions as $\overline{IRQ_1}$/TMIC input pin

$P1_4/\overline{IRQ_4}/\overline{ADTRG}$ pin function switch

0	Functions as $P1_4$ I/O pin
1	Functions as $\overline{IRQ_4}/\overline{ADTRG}$ input pin

$P1_3$/TMIG pin function switch

0	Functions as $P1_3$ I/O pin
1	Functions as TMIG input pin

$P1_2$/TMOFH pin function switch

0	Functions as $P1_2$ I/O pin
1	Functions as TMOFH output pin

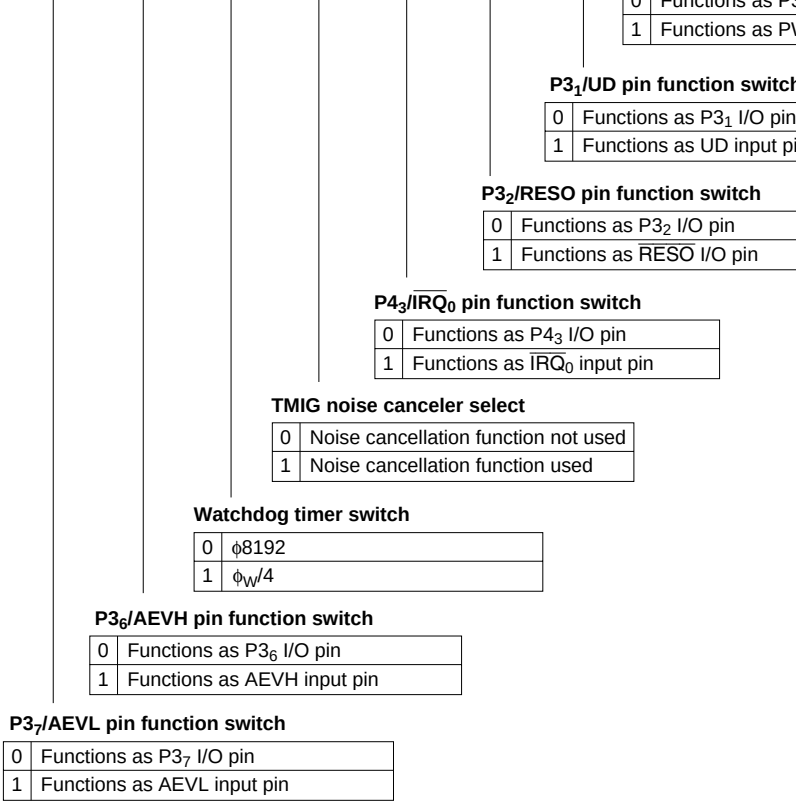
$P1_1$/TMOFL pin function switc

0	Functions as $P1_1$ I/O pin
1	Functions as TMOFL output

1	Functions as TMO

0	Functions as $P3_1$/UD I/O pin
1	Functions as EXCL input pin

Note: The information on this register applies to the H8/38327 Group and H8/38427 G



Note: * In the H8/38327 Group and H8/38427 Group this bit is reserved and cannot be written to.

$P5_n$/WKP_n/SEG_n+1 pin functio

0	Functions as $P5_n$ I/O pin
1	Functions as $\overline{WKP}_n$ input pin

(n = 7 to

PWCR—PWM Control Register **H'D0** **1**

Bit	7	6	5	4	3	2	1	
	—	—	—	—	—	—	PWCR1	
Initial value	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/ϕ, with a minimum modulation width of
	The input clock is $\phi/4$ ($t\phi^* = 4/\phi$) The conversion period is 32,768/ϕ, with a minimum modulation width of
1	The input clock is $\phi/8$ ($t\phi^* = 8/\phi$) The conversion period is 65,536/ϕ, with a minimum modulation width of
	The input clock is $\phi/16$ ($t\phi^* = 16/\phi$) The conversion period is 131,072/ϕ, with a minimum modulation width o

Note: * $t\phi$: Period of PWM input clock

PWDRL—PWM Data Register L **H'D2**

Bit	7	6	5	4	3	2	1
	PWDRL7	PWDRL6	PWDRL5	PWDRL4	PWDRL3	PWDRL2	PWDRL1
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

Lower 8 bits of data for generating PWM waveform

PDR1—Port Data Register 1 **H'D4**

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

Data for port 1 pins

PDR3—Port Data Register 3 **H'D6**

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

Data for port 3 pins

Pin $P4_3$ state is read

PDR5—Port Data Register 5 **H'D8**

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

Data for port 5 pins

PDR6—Port Data Register 6 **H'D9**

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

Data for port 6 pins

PDR7—Port Data Register 7 **H'DA**

Bit	7	6	5	4	3	2	1
	$P7_7$	$P7_6$	$P7_5$	$P7_4$	$P7_3$	$P7_2$	$P7_1$
Initial value	0	0	0	0	0	0	0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Data for port 7 pins

PDRA—Port Data Register A **H'DD**

Bit	7	6	5	4	3	2	1
	—	—	—	—	PA_3	PA_2	PA_1
Initial value	1	1	1	1	0	0	0
Read/Write	—	—	—	—	R/W	R/W	R/W

Data for port A pin

PDRB—Port Data Register B **H'DE**

Bit	7	6	5	4	3	2	1
	PB_7	PB_6	PB_5	PB_4	PB_3	PB_2	PB_1
Read/Write	R	R	R	R	R	R	R

Data for port B pins

Port 1 input pull-up MOS control

0	Input pull-up MOS is off
1	Input pull-up MOS is on

Note: When the PCR1 specification is 0. (Input port spe

PUCR3—Port Pull-Up Control Register 3 H'E1

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

Port 3 input pull-up MOS control

0	Input pull-up MOS is off
1	Input pull-up MOS is on

Note: When the PCR3 specification is 0. (Input port spe

PUCR5—Port Pull-Up Control Register 5 H'E2

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

Port 5 input pull-up MOS control

0	Input pull-up MOS is off
1	Input pull-up MOS is on

Note: When the PCR5 specification is 0. (Input port spe

Port 6 input pull-up MOS control

0	Input pull-up MOS is off
1	Input pull-up MOS is on

Note: When the PCR6 specification is 0. (Input port sp

PCR1—Port Control Register 1 H'E4

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

Port 1 input/output select

0	Input pin
1	Output pin

PCR3—Port Control Register 3 H'E6

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

Port 3 input/output select

0	Input pin
1	Output pin

0	Input pin
1	Output pin

PCR5—Port Control Register 5 H'E8

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

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

Port 6 input/output select

0	Input pin
1	Output pin

0	Input pin
1	Output pin

PCR8—Port Control Register 8 H'EB

Bit	7	6	5	4	3	2	1
	$PCR8_7$	$PCR8_6$	$PCR8_5$	$PCR8_4$	$PCR8_3$	$PCR8_2$	$PCR8_1$
Initial value	0	0	0	0	0	0	0
Read/Write	W	W	W	W	W	W	W

Port 8 input/output select

0	Input pin
1	Output pin

PCRA—Port Control Register A H'ED

Bit	7	6	5	4	3	2	1
	—	—	—	—	$PCRA_3$	$PCRA_2$	$PCRA_1$
Initial value	0	0	0	0	0	0	0
Read/Write	—	—	—	—	W	W	W

Port A input/output sele

0	Input pin
1	Output pin

Active (medium
mode clock sele

0	0	$\phi_{OSC}/16$
	1	$\phi_{OSC}/32$
1	0	$\phi_{OSC}/64$
	1	$\phi_{OSC}/128$

Low speed on flag

0	The CPU operates on the system
1	The CPU operates on the subcloc

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	0	0	Wait time = 131,072 states
		1	Wait time = 2 states
	1	0	Wait time = 8 states
		1	Wait time = 16 states

Software standby

0	• When a SLEEP instruction is executed in active mode, a trans made to sleep mode • When a SLEEP instruction is executed in subactive mode, a t is made to subsleep mode
1	• When a SLEEP instruction is executed in active mode, a trans made to standby mode or watch mode • When a SLEEP instruction is executed in subactive mode, a t is made to watch mode

Subactive mode clock select

0	0	$\phi_W/8$
	1	$\phi_W/4$
1	*	$\phi_W/2$

*: Don't

Medium speed on flag

0	Operates in active (high-speed) mode
1	Operates in active (medium-speed) mode

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 • 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 di transition is made to active (medium-speed) mode if SSBY = 0, MSON = LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = 1 • When a SLEEP instruction is executed in active (medium-speed) mode, transition is made to active (high-speed) mode if SSBY = 0, MSON = 0, a LSON = 0, or to subactive mode if SSBY = 1, TMA3 = 1, and LSON = 1 • When a SLEEP instruction is executed in subactive mode, a direct transition is made to active (high-speed) mode if SSBY = 1, TMA3 = 1, L and MSON = 0, or to active (medium-speed) mode if SSBY = 1, TMA3 = LSON = 0, and MSON = 1

Noise elimination sampling frequency select

0	Sampling rate is $\phi_{OSC}/16$
1	Sampling rate is $\phi_{OSC}/4$

$\overline{IRQ}_0$ edge select

0	Falling edge of $\overline{IRQ}_0$ pin input is d
1	Rising edge of $\overline{IRQ}_0$ pin input is de

$\overline{IRQ}_1$ edge select

0	Falling edge of $\overline{IRQ}_1$, TMIC pin input is detec
1	Rising edge of $\overline{IRQ}_1$, TMIC pin input is detec

$\overline{IRQ}_2$ edge select

0	Falling edge of $\overline{IRQ}_2$ pin input is detected
1	Rising edge of $\overline{IRQ}_2$ 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 $\overline{IRQ}_4$ pin and $\overline{ADTRG}$ pin is detected
1	Rising edge of $\overline{IRQ}_4$ pin and $\overline{ADTRG}$ pin is detected

$\overline{IRQ}_4$ to $\overline{IRQ}_0$ interrupt enable

0	Disables $\overline{IRQ}_4$ to $\overline{IRQ}_0$ interrupt req
1	Enables $\overline{IRQ}_4$ to $\overline{IRQ}_0$ interrupt requ

Wakeup interrupt enable

0	Disables $\overline{WKP}_7$ to $\overline{WKP}_0$ interrupt requests
1	Enables $\overline{WKP}_7$ to $\overline{WKP}_0$ interrupt requests

Timer A interrupt enable

0	Disables timer A interrupt requests
1	Enables timer A interrupt requests

Asynchronous event counter interrupt e

0	Disables asynchronous event counter interrupt requests
1	Enables asynchronous event counter interrupt requests

Timer C interrupt enable

0	Disables timer C interrupt requests
1	Enables timer C interrupt requests

Timer FL interrupt enable

0	Disables timer FL interrupt requests
1	Enables timer FL interrupt requests

Timer FH interrupt enable

0	Disables timer FH interrupt requests
1	Enables timer FH interrupt requests

Timer G interrupt enable

0	Disables timer G interrupt requests
1	Enables timer G interrupt requests

A/D converter interrupt enable

0	Disables A/D converter interrupt requests
1	Enables A/D converter interrupt requests

Direct transition interrupt enable

0	Disables direct transition interrupt requests
1	Enables direct transition interrupt requests

IRQ4 to IRQ0 interrupt request flags

0	[Clearing condition] When IRRIn = 1, it is cleared by writin
1	[Setting condition] When pin IRQn is designated for inter input and the designated signal edge

Timer A interrupt request flag

0	[Clearing condition] When IRRTA = 1, it is cleared by writing 0
1	[Setting condition] When the timer A counter value overflows (from H'FF to H'00)

Note: * Bits 7 and 4 to 0 can only be written with 0, for flag clearing.

Asynchronous event counter interrupt request flag

0	[Clearing condition] When IRREC = 1, it is cleared by writing 0
1	[Setting condition] When the asynchronous event counter value overflo

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 value 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 and output compare register FL match in 8-bit timer mode

Timer FH interrupt request flag

0	[Clearing condition] When IRRTFH = 1, it is cleared by writing 0
1	[Setting condition] When counter FH and output compare register FH match in 8-bit timer mode, or when 16-bit counters FL and FH and output compare registers FL and FH match in 16-bit timer mode

Timer G interrupt request flag

0	[Clearing condition] When IRRTG = 1, it is cleared by writing 0
1	[Setting condition] When the TMIG pin is designated for TMIG input and the designated signal edge is input

A/D converter interrupt request flag

0	[Clearing condition] When IRRAD = 1, it is cleared by writing 0
1	[Setting condition] When the A/D converter completes conversion and ADSF is reset

Direct transition interrupt request flag

0	[Clearing condition] When IRRDT = 1, it is cleared by writing 0
1	[Setting condition] When a SLEEP instruction is executed while DTON is set to 1, and a direct transition is made

Note: * Bits 7, 6 and 4 to 0 can only be written with 0, for flag clearing.

Wakeup interrupt request register

0	[Clearing condition] When IWPFn = 1, it is cleared by writing 0
1	[Setting condition] When pin $\overline{\text{WKP}}$n is designated for wakeup input a falling edge is input at that pin

(

Note: * All bits can only be written with 0, for flag clearing.

Timer A module standby mode co

0	Timer A is set to module standby
1	Timer A module standby mode is

Timer C module standby mode control

0	Timer C is set to module standby mode
1	Timer C module standby mode is cleare

Timer F module standby mode control

0	Timer F is set to module standby mode
1	Timer F module standby mode is cleared

Timer G interrupt enable

0	Timer G is set to module standby mode
1	Timer G module standby mode is cleared

A/D converter module standby mode control

0	A/D converter is set to module standby mode
1	A/D converter module standby mode is cleared

SCI3-2 module standby mode control

0	SCI3-2 is set to module standby mode
1	SCI3-2 module standby mode is cleared

SCI3-1 module standby mode control

0	SCI3-1 is set to module standby mode
1	SCI3-1 module standby mode is cleared

LCD module standby mode cont

0	LCD is set to module standby
1	LCD module standby mode is

PWM module standby mode control

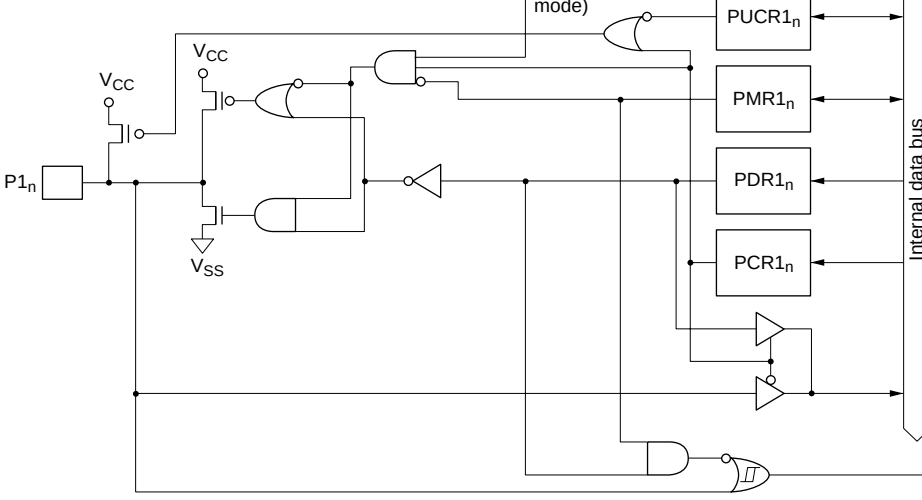
0	PWM is set to module standby mode
1	PWM module standby mode is cleared

WDT module standby mode control

0	WDT is set to module standby mode
1	WDT module standby mode is cleared

Asynchronous event counter module standby mode contr

0	Asynchronous event counter is set to module standby mo
1	Asynchronous event counter module standby mode is clea



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

*: n = 7 to 5 → n–4
n = 4 → n

Figure C.1 (a) Port 1 Block Diagram (Pins $P1_7$ to $P1_4$)

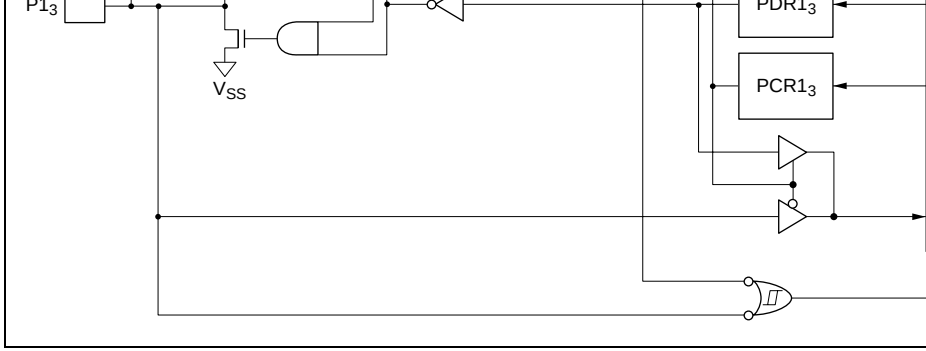
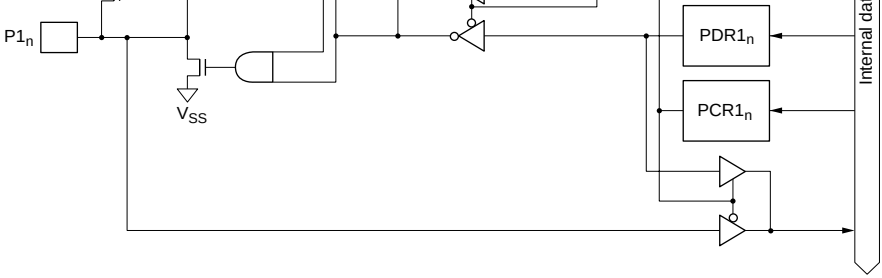


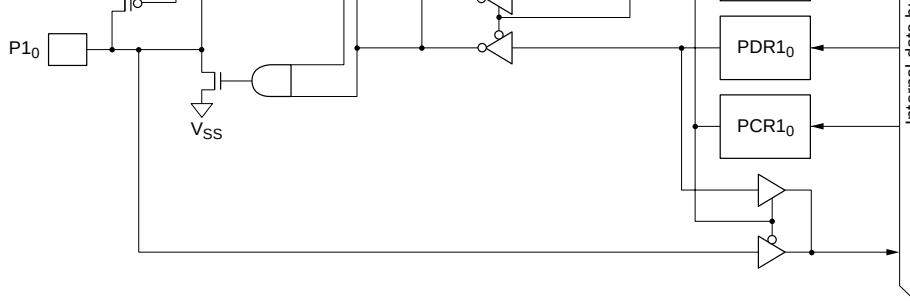
Figure C.1 (b) Port 1 Block Diagram (Pin P1$_3$)



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

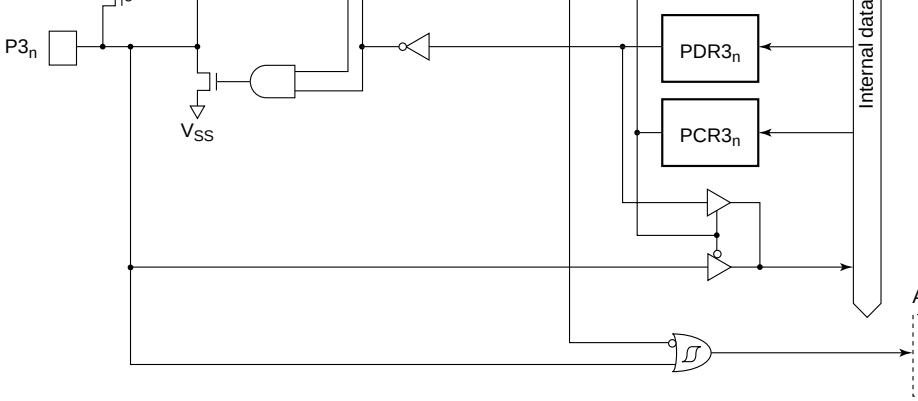
n= 2, 1

Figure C.1 (c) Port 1 Block Diagram (Pin $P1_2$, $P1_1$)



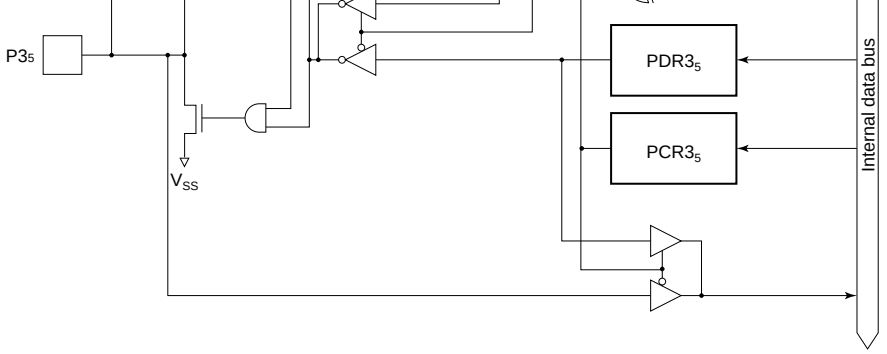
PDR1: Port data register 1
PCR1: Port control register 1
PMR1: Port mode register 1
PUCR1: Port pull-up control register 1

Figure C.1 (d) Port 1 Block Diagram (Pin $P1_0$)



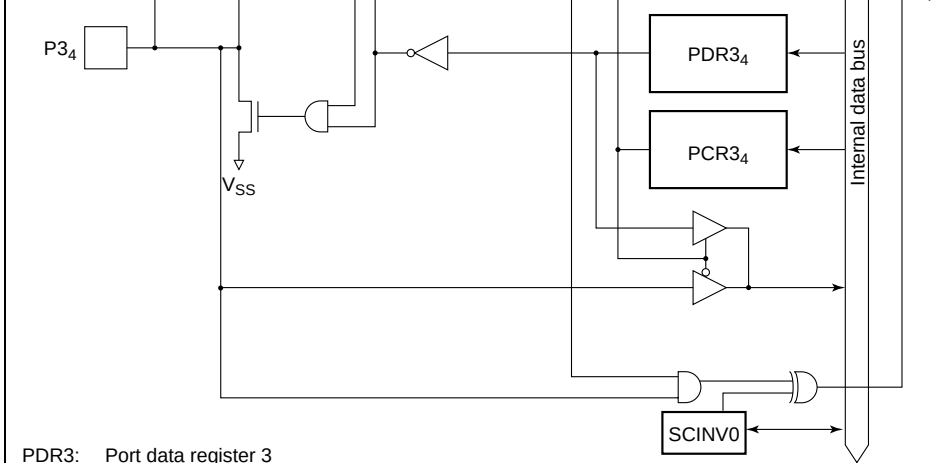
PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3
n=7 to 6

Figure C.2 (a) Port 3 Block Diagram (Pin $P3_7$ to $P3_6$)



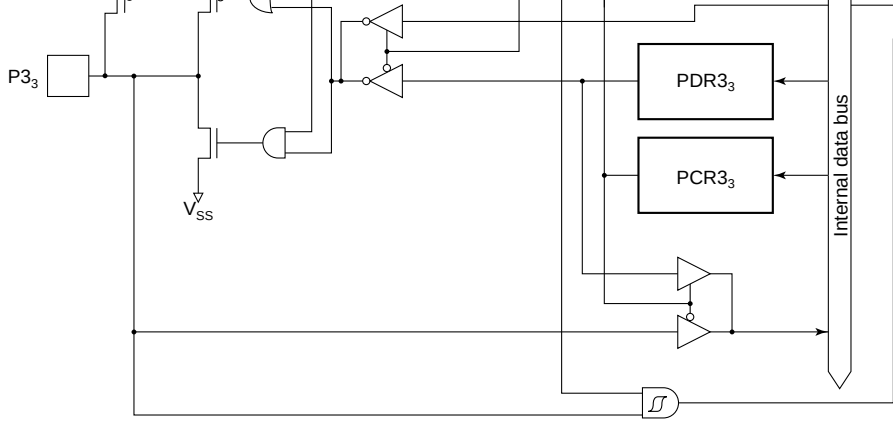
PDR3: Port data register 3
PCR3: Port control register 3
PUCR3: Port pull-up control register 3
SCINV1: Bit 1 of serial port control register (SPCR)
SPC31: Bit 4 of serial port control register (SPCR)

Figure C.2 (b) Port 3 Block Diagram (Pin $P3_5$)



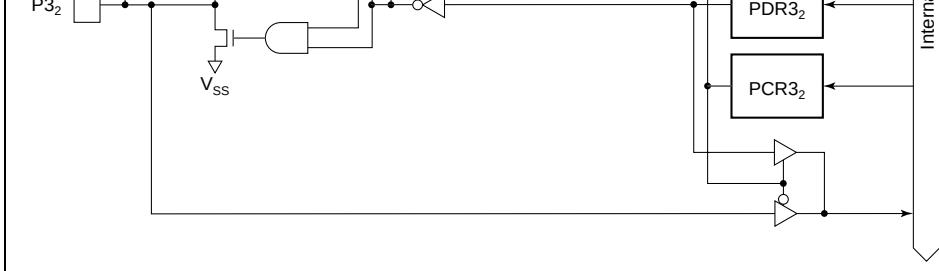
PDR3: Port data register 3
PCR3: Port control register 3
PUCR3: Port pull-up control register 3
SCINV0: Bit 0 of serial port control register (SPCR)

Figure C.2 (c) Port 3 Block Diagram (Pin P3$_4$)



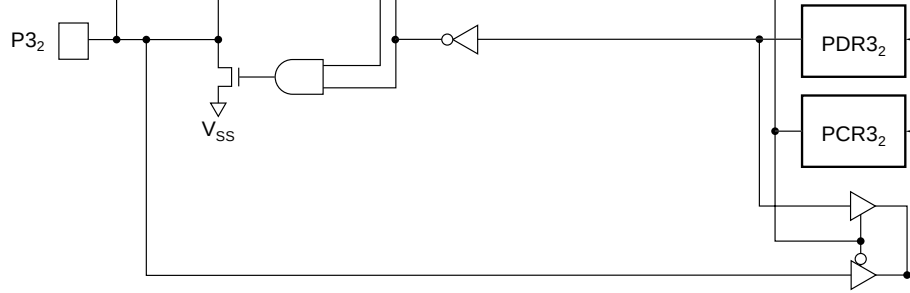
PDR3: Port data register 3
PCR3: Port control register 3
PUCR3: Port pull-up control register 3

Figure C.2 (d) Port 3 Block Diagram (Pin P3$_3$)



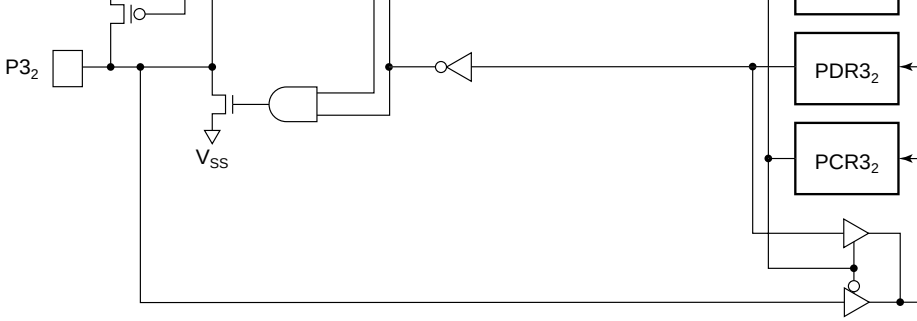
PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3

Figure C.2 (e-1) Port 3 Block Diagram (Pin $P3_2$, H8/3827R Group and H8/3827



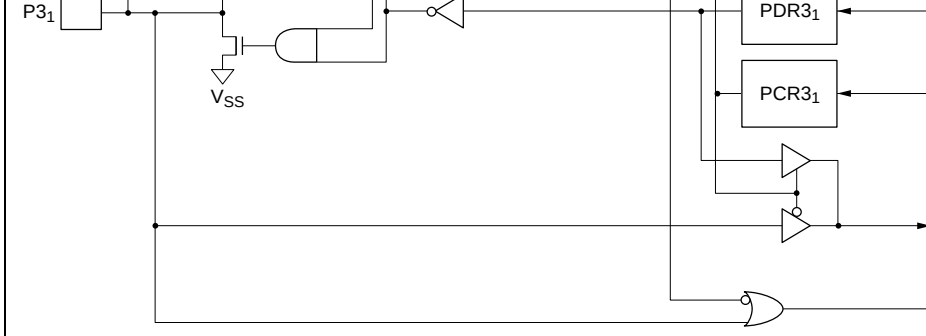
PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3

Figure C.2 (e-2) Port 3 Block Diagram (Pin P3$_2$ in the Mask ROM Version of th Group and H8/38427 Group)



PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3

Figure C.2 (e-3) Port 3 Block Diagram (Pin P3$_2$ in the F-ZTAT Version of the H
Group and H8/38427 Group)



PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3

Figure C.2 (f-1) Port 3 Block Diagram (Pin P3$_1$, H8/3827R Group and H8/382

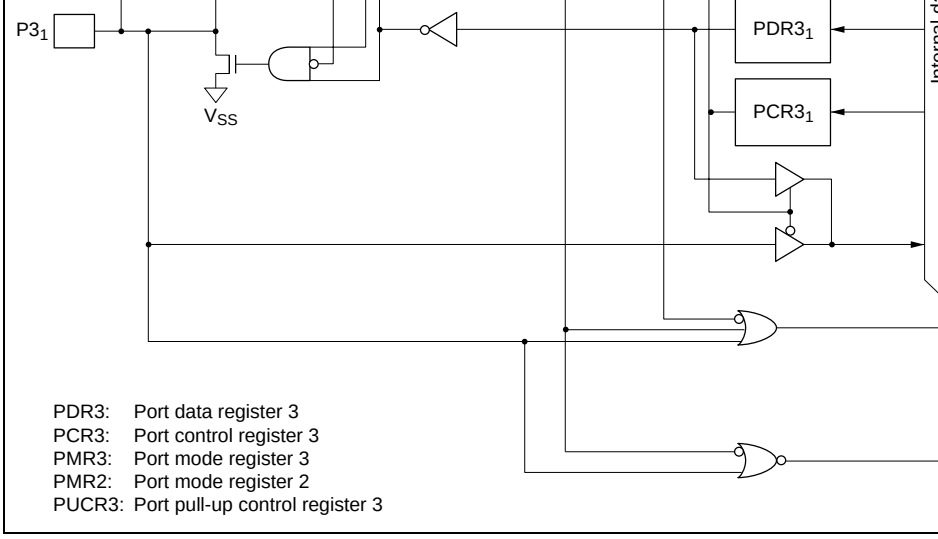
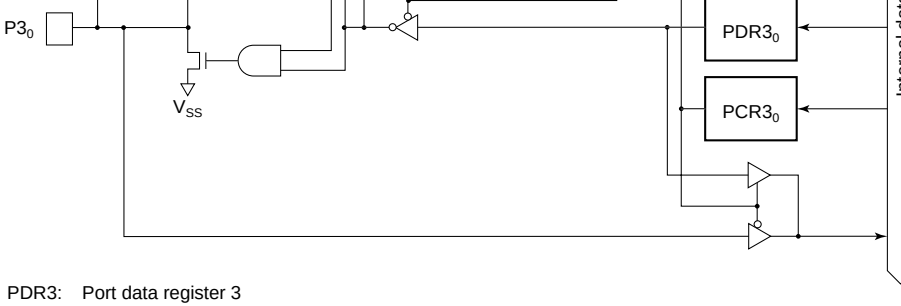


Figure C.2 (f-2) Port 3 Block Diagram (Pin P3$_1$, H8/38327 Group and H8/3842



PDR3: Port data register 3
PCR3: Port control register 3
PMR3: Port mode register 3
PUCR3: Port pull-up control register 3

Figure C.2 (g) Port 3 Block Diagram (Pin P3$_0$)

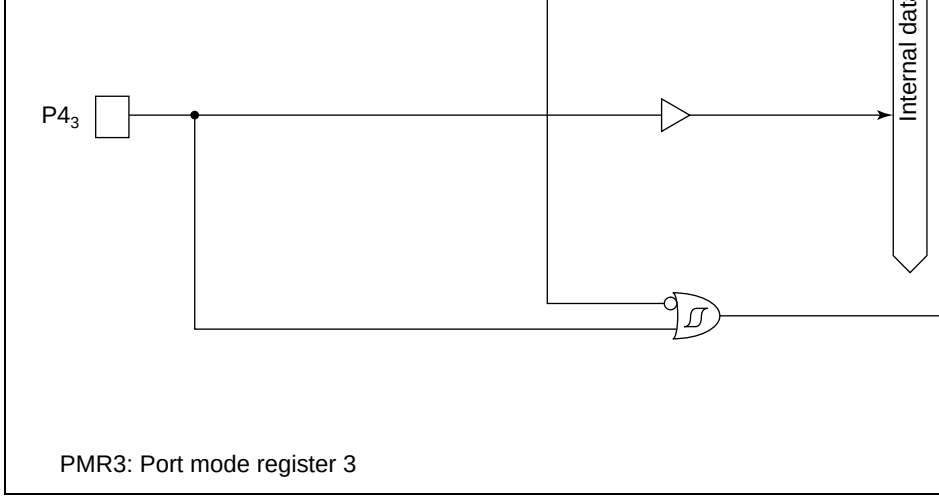
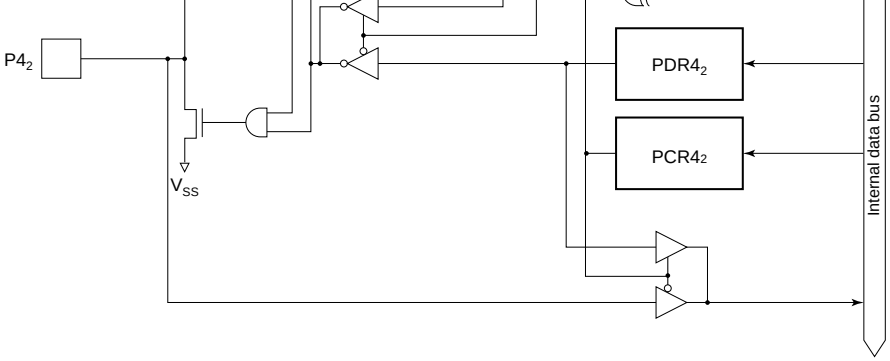
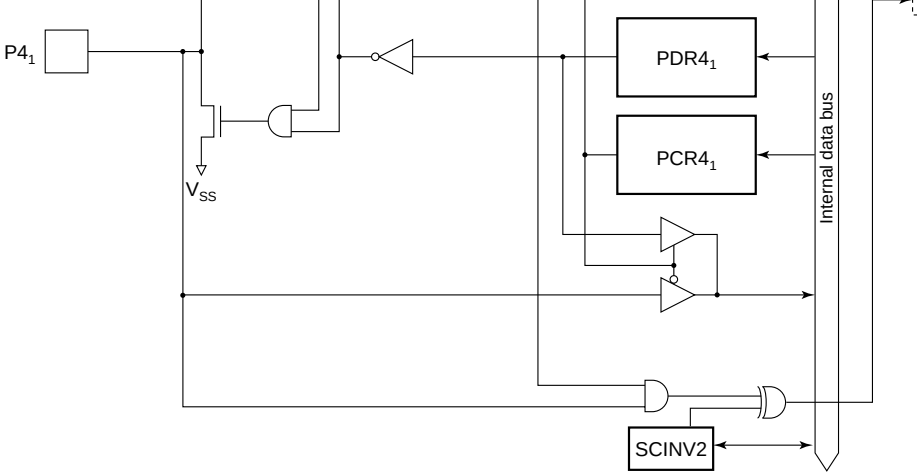


Figure C.3 (a) Port 4 Block Diagram (Pin P4$_3$)



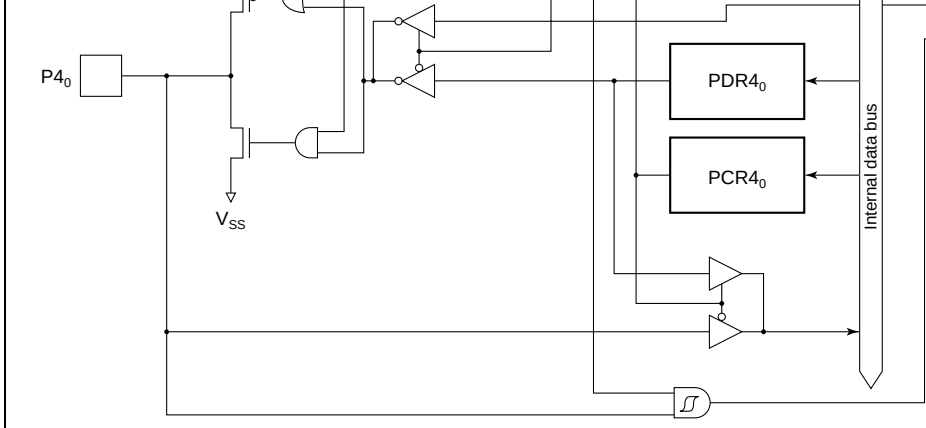
PDR4: Port data register 4
PCR4: Port control register 4
SCINV3: Bit 3 of serial port control register (SPCR)
SPC32: Bit 5 of serial port control register (SPCR)

Figure C.3 (b) Port 4 Block Diagram (Pin $P4_2$)



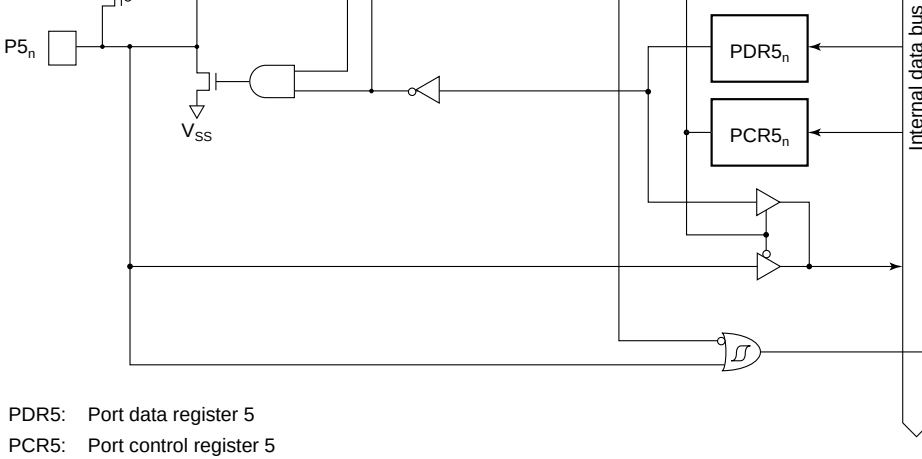
PDR4: Port data register 4
PCR4: Port control register 4
SCINV2: Bit 2 of serial port control register (SPCR)

Figure C.3 (c) Port 4 Block Diagram (Pin P4$_1$)



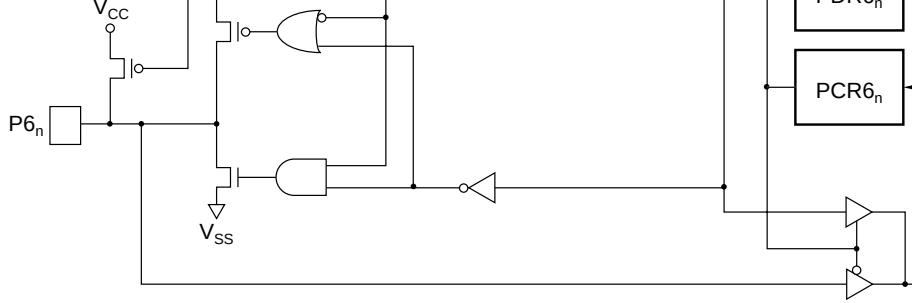
PDR4: Port data register 4
PCR4: Port control register 4

Figure C.3 (d) Port 4 Block Diagram (Pin P4$_0$)



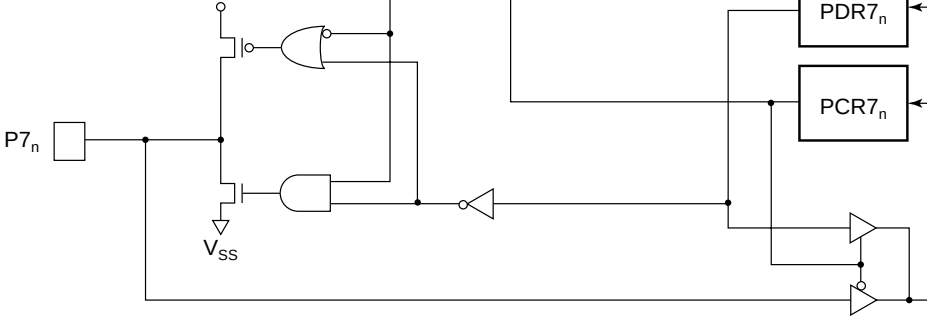
PDR5: Port data register 5
PCR5: Port control register 5
PMR5: Port mode register 5
PUCR5: Port pull-up control register 5
n = 7 to 0

Figure C.4 Port 5 Block Diagram



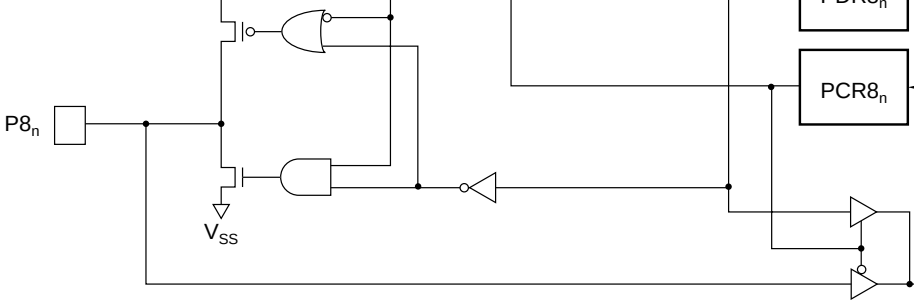
PDR6: Port data register 6
PCR6: Port control register 6
PUCR6: Port pull-up control register 6
n = 7 to 0

Figure C.5 Port 6 Block Diagram



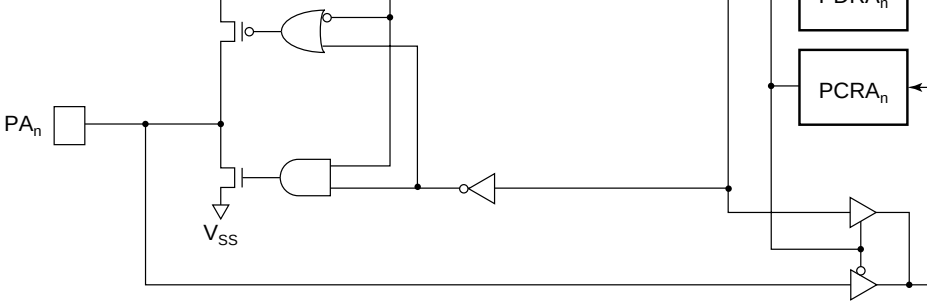
PDR7: Port data register 7
PCR7: Port control register 7
n = 7 to 0

Figure C.6 Port 7 Block Diagram



PDR8: Port data register 8
PCR8: Port control register 8
n= 7 to 0

Figure C.7 Port 8 Block Diagram



PDRA: Port data register A
PCRA: Port control register A
n = 3 to 0

Figure C.8 Port A Block Diagram

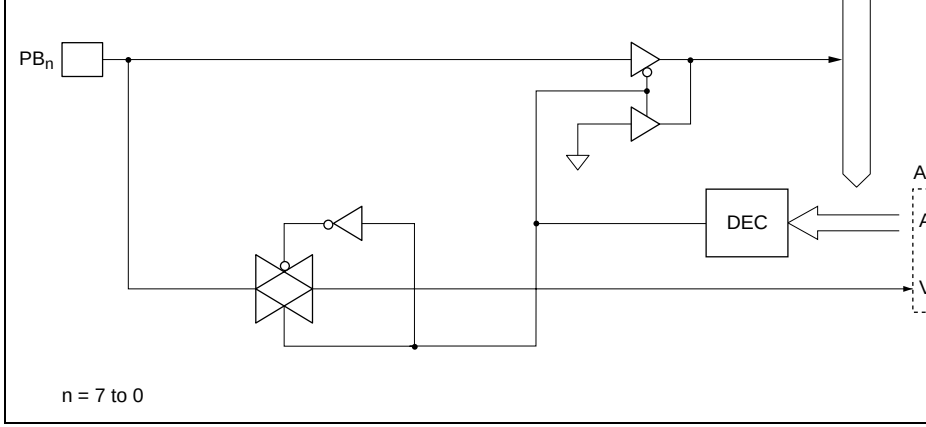


Figure C.9 Port B Block Diagram

$P3_0$	impedance[*2]			impedance[*1]		
$P4_3$ to $P4_0$	High impedance	Retained	Retained	High impedance	Retained	Functions
$P5_7$ to $P5_0$	High impedance	Retained	Retained	High impedance[*1]	Retained	Functions
$P6_7$ to $P6_0$	High impedance	Retained	Retained	High impedance	Retained	Functions
$P7_7$ to $P7_0$	High impedance	Retained	Retained	High impedance	Retained	Functions
$P8_7$ to $P8_0$	High impedance	Retained	Retained	High impedance	Retained	Functions
PA_3 to PA_0	High impedance	Retained	Retained	High impedance	Retained	Functions
PB_7 to PB_0	High impedance	High impedance	High impedance	High impedance	High impedance	High impedance

Notes: 1. High level output when MOS pull-up is in on state.

2. Reset output from $P3_2$ pin only (H8/3827R Group and H8/3827S Group).
On-chip pull-up MOS turns on for pin $P3_2$ only (F-ZTAT Version of the H8/38 and H8/38427 Group).

			HD6433822RW	HD6433822R(***)W	80-pin T
			HCD6433822R	—	Die
		Wide-range specification products	HD6433822RD	HD6433822R(***)H	80-pin Q
			HD6433822RE	HD6433822R(***)F	80-pin Q
			HD6433822RWI	HD6433822R(***)W	80-pin T
H8/3823R	Mask ROM versions	Regular products	HD6433823RH	HD6433823R(***)H	80-pin Q
			HD6433823RF	HD6433823R(***)F	80-pin Q
			HD6433823RW	HD6433823R(***)W	80-pin T
			HCD6433823R	—	Die
		Wide-range specification products	HD6433823RD	HD6433823R(***)H	80-pin Q
			HD6433823RE	HD6433823R(***)F	80-pin Q
			HD6433823RWI	HD6433823R(***)W	80-pin T
H8/3824R	Mask ROM versions	Regular products	HD6433824RH	HD6433824R(***)H	80-pin Q
			HD6433824RF	HD6433824R(***)F	80-pin Q
			HD6433824RW	HD6433824R(***)W	80-pin T
			HCD6433824R	—	Die
		Wide-range specification products	HD6433824RD	HD6433824R(***)H	80-pin Q
			HD6433824RE	HD6433824R(***)F	80-pin Q
			HD6433824RWI	HD6433824R(***)W	80-pin T
H8/3825R	Mask ROM versions	Regular products	HD6433825RH	HD6433825R(***)H	80-pin Q
			HD6433825RF	HD6433825R(***)F	80-pin Q
			HD6433825RW	HD6433825R(***)W	80-pin T
			HCD6433825R	—	Die
		Wide-range specification products	HD6433825RD	HD6433825R(***)H	80-pin Q
			HD6433825RE	HD6433825R(***)F	80-pin Q
			HD6433825RWI	HD6433825R(***)W	80-pin T

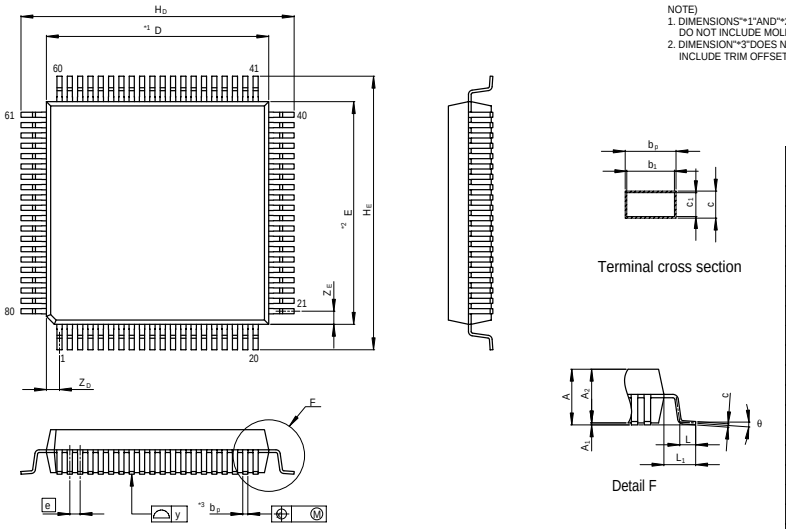
				HD6433826RWI	HD6433826R(***)W	80-pin TQ
	H8/3827R	Mask ROM versions	Regular products	HD6433827RH	HD6433827R(***)H	80-pin QF
				HD6433827RF	HD6433827R(***)F	80-pin QF
				HD6433827RW	HD6433827R(***)W	80-pin TQ
				HCD6433827R	—	Die
			Wide-range specification products	HD6433827RD	HD6433827R(***)H	80-pin QF
				HD6433827RE	HD6433827R(***)F	80-pin QF
				HD6433827RWI	HD6433827R(***)W	80-pin TQ
		ZTAT versions	Regular products	HD6473827RH	HD6473827RH	80-pin QF
				HD6473827RF	HD6473827RF	80-pin QF
				HD6473827RW	HD6473827RW	80-pin TQ
			Wide-range specification products	HD6473827RD	HD6473827RH	80-pin QF
				HD6473827RE	HD6473827RF	80-pin QF
				HD6473827RWI	HD6473827RW	80-pin TQ
H8/3827S Group	H8/3824S	Mask ROM versions	Regular products	HD6433824SH	HD6433824S(***)H	80-pin QF
				HD6433824SW	HD6433824S(***)W	80-pin TQ
				HCD6433824S	—	Die
			Wide-range specification products	HD6433824SD	HD6433824S(***)H	80-pin QF
				HD6433824SWI	HD6433824S(***)W	80-pin TQ
	H8/3825S	Mask ROM versions	Regular products	HD6433825SH	HD6433825S(***)H	80-pin QF
				HD6433825SW	HD6433825S(***)W	80-pin TQ
				HCD6433825S	—	Die
			Wide-range specification products	HD6433825SD	HD6433825S(***)H	80-pin QF
				HD6433825SWI	HD6433825S(***)W	80-pin TQ

		ROM versions	products	HD6433827SW	HD6433827S(***)W	80-pin T
				HCD6433827S	—	Die
			Wide-range specification products	HD6433827SD	HD6433827S(***)H	80-pin Q
				HD6433827SWI	HD6433827S(***)W	80-pin T
H8/38327 Group	H8/38322	Mask ROM versions	Regular products	HD64338322H	38322H	80-pin Q
				HD64338322W	38322W	80-pin T
				HCD64338322	—	Die
			Wide-range specification products	HD64338322HW	38322H	80-pin Q
				HD64338322WW	38322W	80-pin T
	H8/38323	Mask ROM versions	Regular products	HD64338323H	38323H	80-pin Q
				HD64338323W	38323W	80-pin T
				HCD64338323	—	Die
			Wide-range specification products	HD64338323HW	38323H	80-pin Q
				HD64338323WW	38323W	80-pin T
	H8/38324	Mask ROM versions	Regular products	HD64338324H	38324H	80-pin Q
				HD64338324W	38324W	80-pin T
				HCD64338324	—	Die
			Wide-range specification products	HD64338324HW	38324H	80-pin Q
				HD64338324WW	38324W	80-pin T
		F-ZTAT versions	Regular products	HD64F38324H	F38324H	80-pin Q
				HD64F38324W	F38324W	80-pin T
			Wide-range specification products	HD64F38324HW	F38324H	80-pin Q
				HD64F38324WW	F38324W	80-pin T

		ROM versions	products	HD64338326W	38326W	80-pin TQ
				HCD64338326	—	Die
			Wide-range specification products	HD64338326HW	38326H	80-pin QF
				HD64338326WW	38326W	80-pin TQ
	H8/38327	Mask ROM versions	Regular products	HD64338327H	38327H	80-pin QF
				HD64338327W	38327W	80-pin TQ
				HCD64338327	—	Die
			Wide-range specification products	HD64338327HW	38327H	80-pin QF
				HD64338327WW	38327W	80-pin TQ
		F-ZTAT versions	Regular products	HD64F38327H	F38327H	80-pin QF
				HD64F38327W	F38327W	80-pin TQ
				HCD64F38327	—	Die
			Wide-range specification products	HD64F38327HW	F38327H	80-pin QF
				HD64F38327WW	F38327W	80-pin TQ
H8/38427 Group	H8/38422	Mask ROM versions	Regular products	HD64338422H	38422H	80-pin QF
				HD64338422W	38422W	80-pin TQ
				HCD64338422	—	Die
			Wide-range specification products	HD64338422HW	38422H	80-pin QF
				HD64338422WW	38422W	80-pin TQ
	H8/38423	Mask ROM versions	Regular products	HD64338423H	38423H	80-pin QF
				HD64338423W	38423W	80-pin TQ
				HCD64338423	—	Die
			Wide-range specification products	HD64338423HW	38423H	80-pin QF
				HD64338423WW	38423W	80-pin TQ

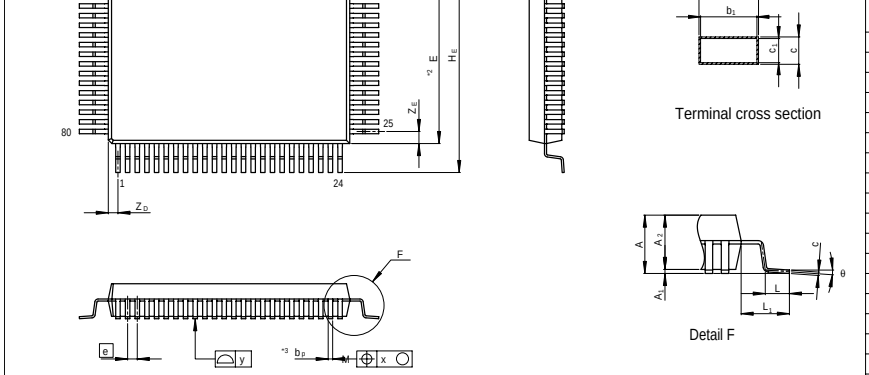
			HD64F38424W	F38424W	80-pin T
		Wide-range specification products	HD64F38424HW	F38424H	80-pin Q
			HD64F38424WW	F38424W	80-pin T
H8/38425	Mask ROM versions	Regular products	HD64338425H	38425H	80-pin Q
			HD64338425W	38425W	80-pin T
			HCD64338425	—	Die
		Wide-range specification products	HD64338425HW	38425H	80-pin Q
			HD64338425WW	38425W	80-pin T
H8/38426	Mask ROM versions	Regular products	HD64338426H	38426H	80-pin Q
			HD64338426W	38426W	80-pin T
			HCD64338426	—	Die
		Wide-range specification products	HD64338426HW	38426H	80-pin Q
			HD64338426WW	38426W	80-pin T
H8/38427	Mask ROM versions	Regular products	HD64338427H	38427H	80-pin Q
			HD64338427W	38427W	80-pin T
			HCD64338427	—	Die
		Wide-range specification products	HD64338427HW	38427H	80-pin Q
			HD64338427WW	38427W	80-pin T
	F-ZTAT versions	Regular products	HD64F38427H	F38427H	80-pin Q
			HD64F38427W	F38427W	80-pin T
			HCD64F38427	—	Die
		Wide-range specification products	HD64F38427HW	F38427H	80-pin Q
			HD64F38427WW	F38427W	80-pin T

Note: For mask ROM versions, (***) is the ROM code.



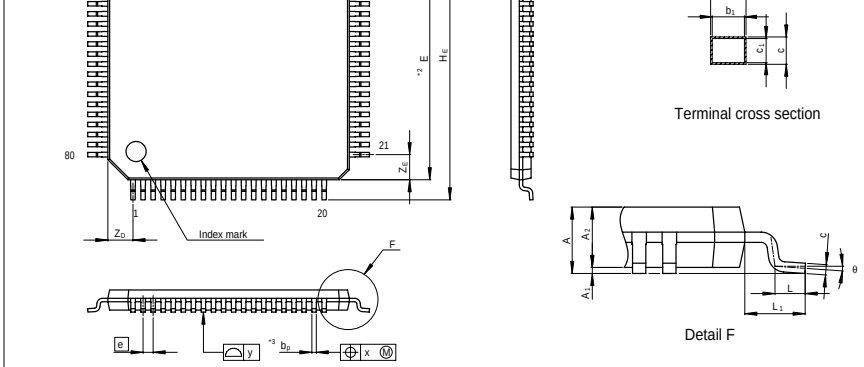
Reference Symbol	Dim
	Min
D	—
E	—
A_2	—
H_D	16.9
H_E	16.9
A	—
A_1	0.00
b_p	0.24
b_1	—
c	0.12
c_1	—
θ	0°
e	—
x	—
y	—
Z_D	—
Z_E	—
L	0.5
L_1	—

Figure F.1 FP-80A Package Dimensions



Reference Symbol
D
E
A_2
H_D
H_E
A
A_1
b_p
b_1
c
c_1
θ
e
x
y
Z_D
Z_E
L
L_1

Figure F.2 FP-80B Package Dimensions



Reference Symbol	M
D	—
E	—
A_2	—
H_D	1
H_E	1
A	—
A_1	0
b_p	0
b_1	—
c	0
c_1	—
θ	
e	—
x	—
y	—
Z_D	—
Z_E	—
L	0
L_1	—

Figure F.3 TFP-80C Package Dimensions

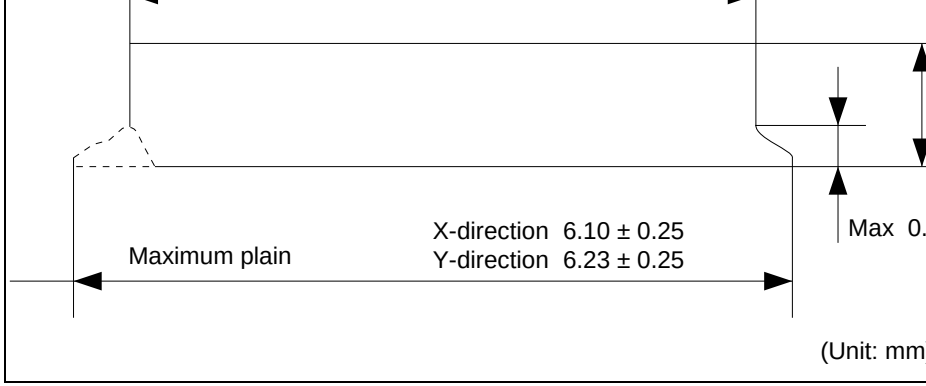


Figure G.1 Chip Sectional Figure

The specifications of the chip form of the HCD6433827S, HCD6433826S, HCD6433
HCD6433824S are shown in figure G.2.

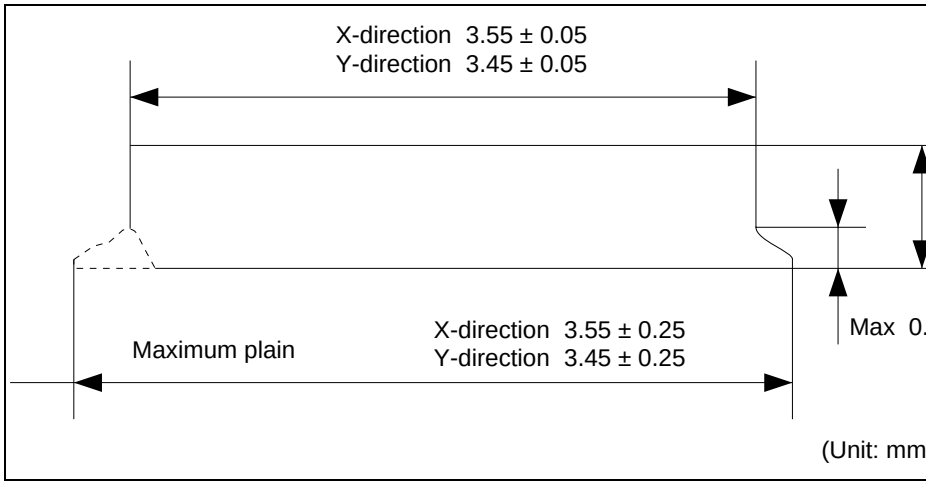


Figure G.2 Chip Sectional Figure

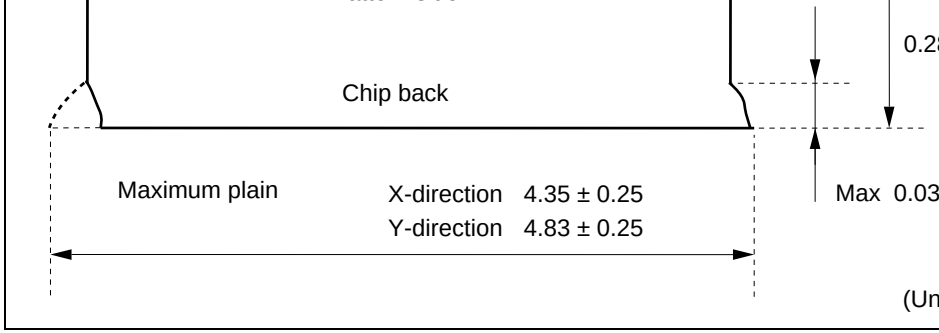


Figure G.3 Chip Sectional Figure

The specifications of the chip form of the H8/38327 Group (mask ROM version) and H
Group (mask ROM version) are shown in figure G.4.

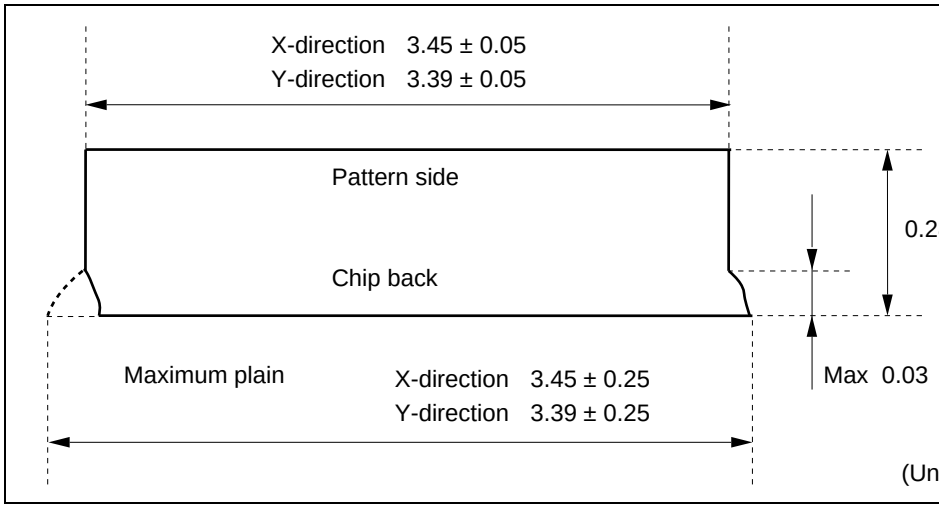


Figure G.4 Chip Sectional Figure

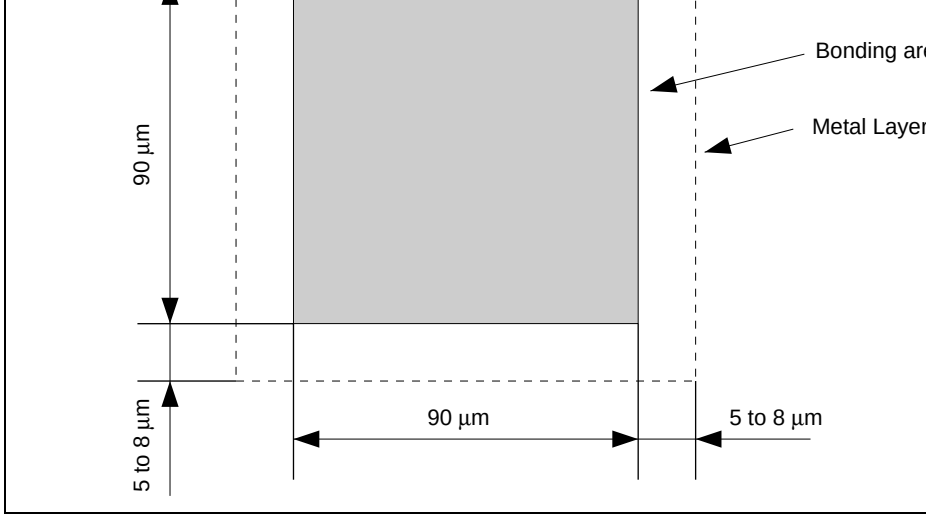


Figure H.1 Bonding Pad Form

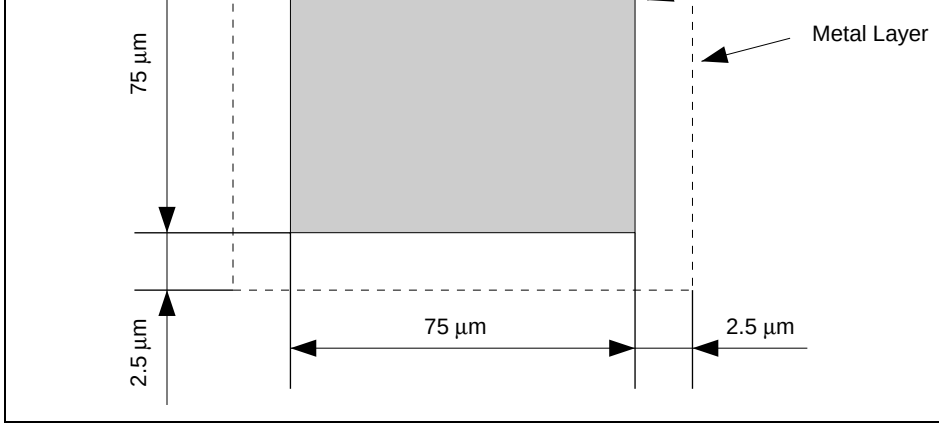


Figure H.2 Bonding Pad Form

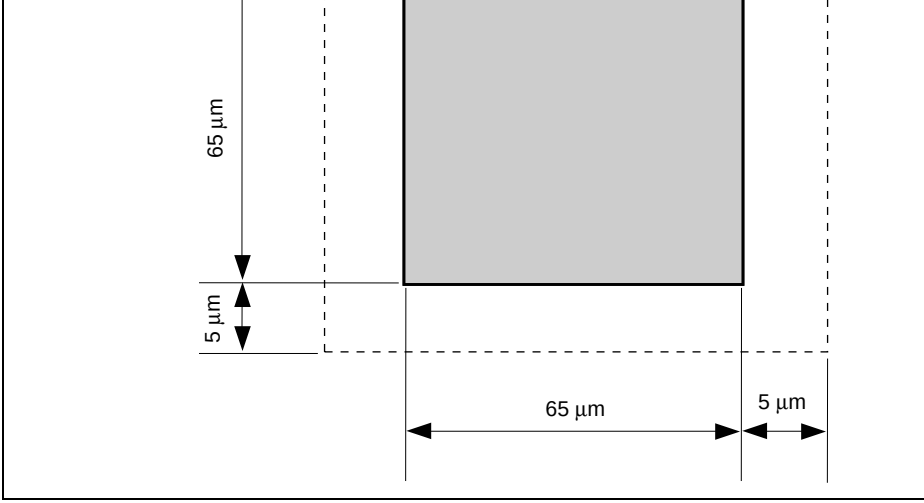


Figure H.3 Bonding Pad Form

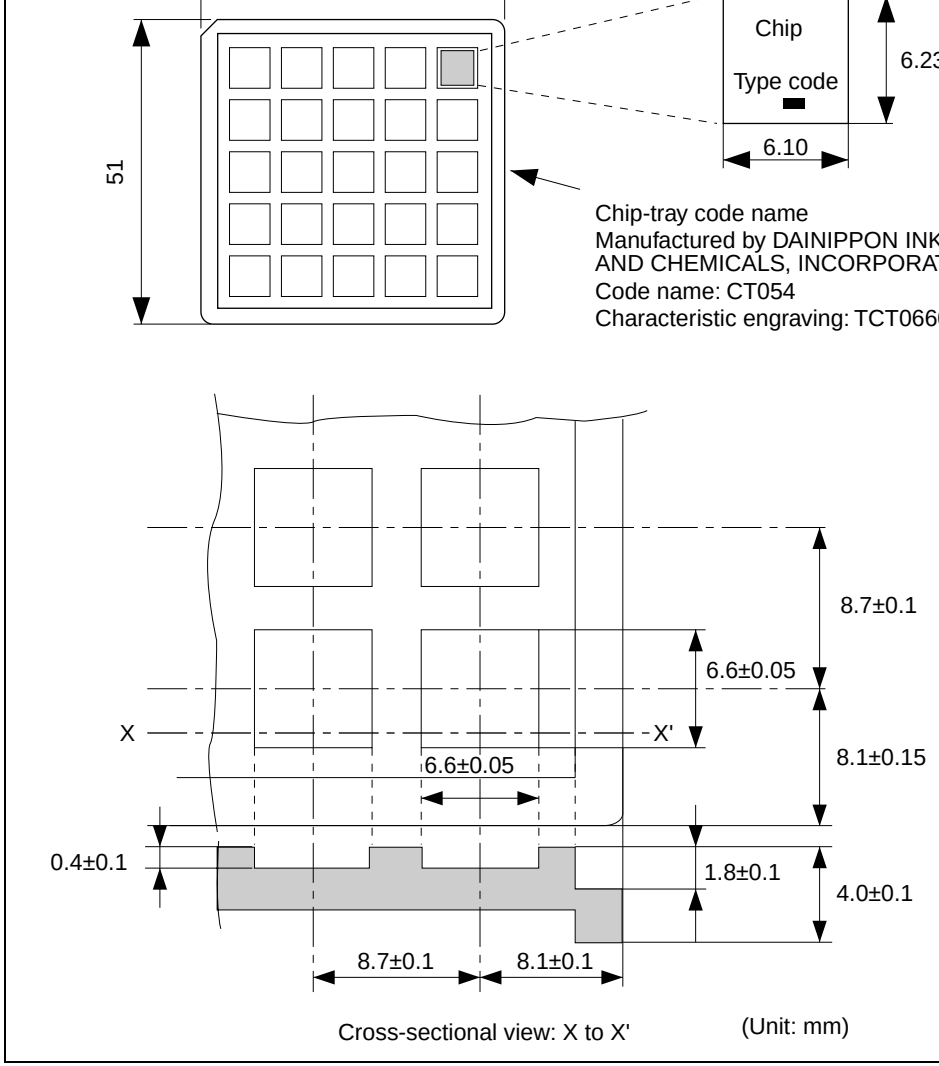


Figure I.1 Specifications of Chip Tray

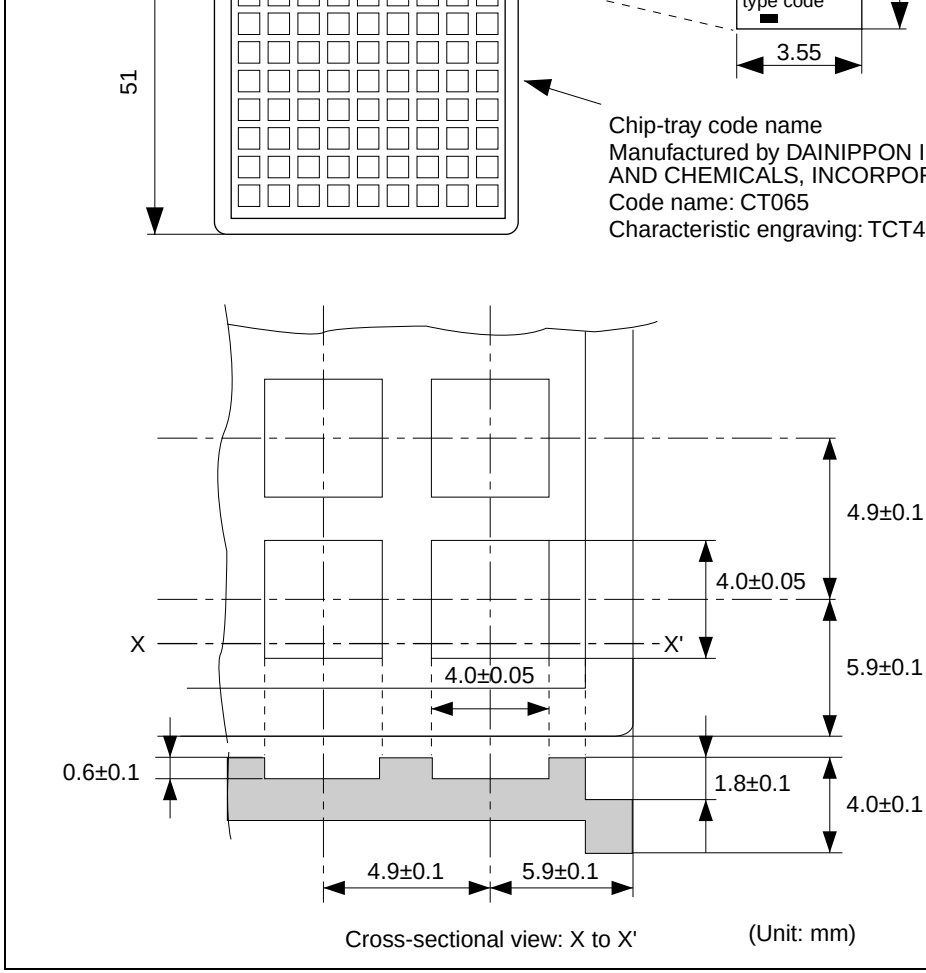
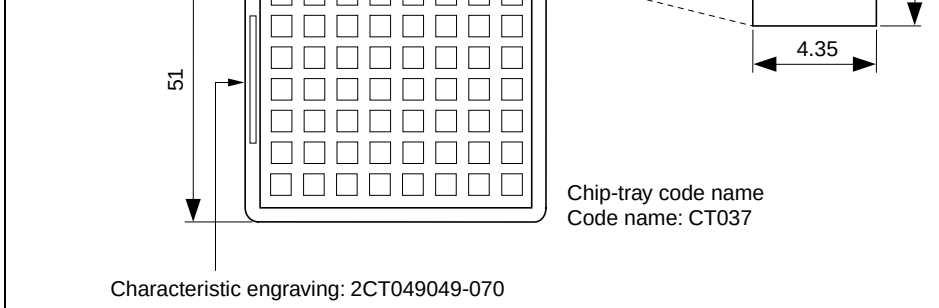
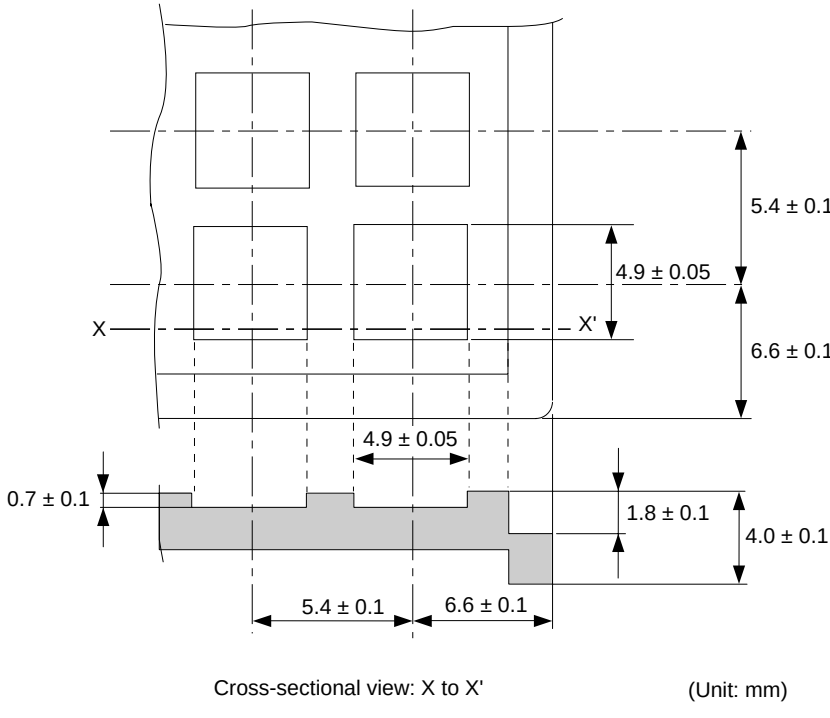


Figure I.2 Specifications of Chip Tray





Cross-sectional view: X to X'

(Unit: mm)

Figure I.3 Specifications of Chip Tray

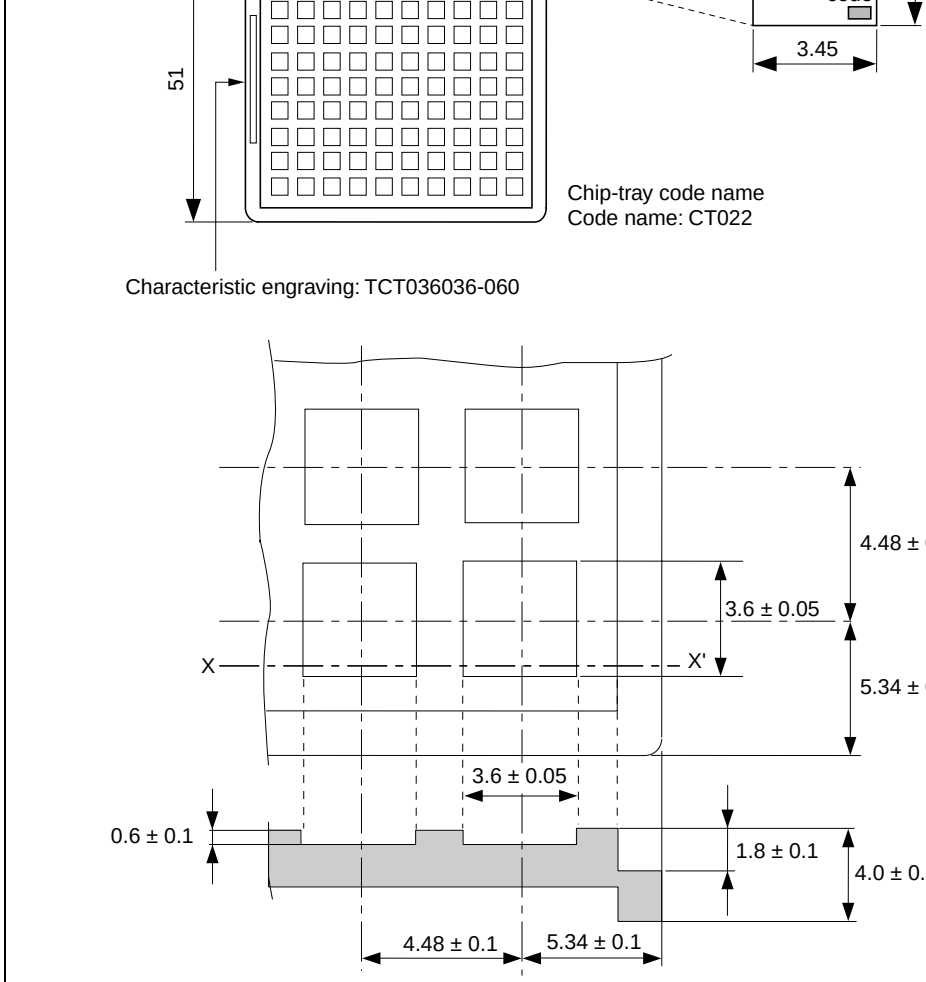


Figure I.4 Specifications of Chip Tray

Renesas 8-Bit Single-Chip Microcomputer
Hardware Manual
H8/3827R Group, H8/3827S Group, H8/38327 Group, H8/38427 Group

Publication Date: 1st Edition, September, 1999
Rev.6.00, August 04, 2006
Published by: Sales Strategic Planning Div.
Renesas Technology Corp.
Edited by: Customer Support Department
Global Strategic Communication Div.
Renesas Solutions Corp.

©2006. Renesas Technology Corp., All rights reserved. Printed in Japan.

RENESAS SALES OFFICES

Refer to "**http://www.renesas.com/en/network**" for the latest and detailed information.

Renesas Technology America, Inc.
450 Holger Way, San Jose, CA 95134-1368, U.S.A
Tel: <1> (408) 382-7500, Fax: <1> (408) 382-7501

Renesas Technology Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K.
Tel: <44> (1628) 585-100, Fax: <44> (1628) 585-900

Renesas Technology (Shanghai) Co., Ltd.
Unit 204, 205, AZIACenter, No.1233 Lujiazui Ring Rd, Pudong District, Shanghai, China 200120
Tel: <86> (21) 5877-1818, Fax: <86> (21) 6887-7898

Renesas Technology Hong Kong Ltd.
7th Floor, North Tower, World Finance Centre, Harbour City, 1 Canton Road, Tsimshatsui, Kowloon, Hong Kong
Tel: <852> 2265-6688, Fax: <852> 2730-6071

Renesas Technology Taiwan Co., Ltd.
10th Floor, No.99, Fushing North Road, Taipei, Taiwan
Tel: <886> (2) 2715-2888, Fax: <886> (2) 2713-2999

Renesas Technology Singapore Pte. Ltd.
1 Harbour Front Avenue, #06-10, Keppel Bay Tower, Singapore 098632
Tel: <65> 6213-0200, Fax: <65> 6278-8001

Renesas Technology Korea Co., Ltd.
Kukje Center Bldg. 18th Fl., 191, 2-ka, Hangang-ro, Yongsan-ku, Seoul 140-702, Korea
Tel: <82> (2) 796-3115, Fax: <82> (2) 796-2145

Renesas Technology Malaysia Sdn. Bhd
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No.18, Jalan Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsa
Tel: <603> 7955-9390, Fax: <603> 7955-9510

H8/3827R Group, H8/3827S Group, H8/38327 Group, H8/38427 Group

Hardware Manual

Renesas Electronics Corporation
1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan

REJ09B0144-0600

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for Microprocessors - MPU ***category:***

Click to view products by Renesas ***manufacturer:***

Other Similar products are found below :

MC68302EH20C MC7457RX1000LC MC7457RX1267LC MC7457VG1267LC A2C00010998 A A2C52004004 R5F117BCGNA#20 R5F52106BDLA#U0 R5S72690W266BG#U0 ADJ3400IAA5DOE MPC8245TVV266D MPC8245TZU300D MPC8260ACVVMHBB MPC8323ECVRAFDCA MPC8323VRADDCA MPC8536ECVJAVLA BOXNUC5PGYH0AJ 20-668-0024 P1010NSN5DFB P2010NSN2MHC P2020NXE2HHC P5020NSE7QMB P5020NSE7TNB P5020NSE7VNB LS1020ASN7KQB LS1020AXN7HNB LS1020AXN7KQB A2C00010729 A A2C00039344 T1022NSE7MQB T1022NXN7PQB T1023NSE7MQA T1024NXE7PQA T1042NSE7MQB T1042NSN7MQB T1042NXN7WQB T2080NSE8TTB T2080NSN8PTB T2080NXE8TTB T2081NXN8TTB R5F101AFASP#V0 MC68302CEH20C TS68040MF33A MPC8260ACVVMIBB MPC8280CZUUPEA MPC8313ECVRAFFC MPC8313ECVRAGDC MPC8313EVRADDC MPC8313EVRAFFC MPC8313VRADDC